Chapter 1

Probability

Uncertainty is a crucial aspect of many business problems. Probability theory is the mathematical way to model uncertainty. In this chapter we give a refresher of probability theory and we introduce some more advanced topics.

1.1 Random variables

Consider an experiment that can have multiple possible outcomes. For such an experiment we call the set of possible outcomes the *sample space*, usually denoted with Ω. A probability measure $\mathbb{P}$ assigns probabilities to subsets of Ω. These subsets are called *events*. The practical interpretation of probability is that if an experiment is repeated many times, then the probability of an event signifies the long-run fraction of times (or *relative frequency*) that the event occurs. This is called the frequentist interpretation of probability. It shows also how probabilities can be estimated, by observing replications of an experiment and then taking the frequency. Indeed, frequentism is strongly related to statistical estimation methods. (Alternatively, probability can also be seen as the quantification of a belief: this is the Bayesian interpretation.)

Example 1.1.1 The daily number of arrivals to a certain service center can be any natural number. In this case $\Omega = \{0, 1, \ldots\} = \mathbb{N}_0$. Repetition of this experiment leads to a row of natural numbers, e.g., 1,1,3,1,0,5,3,3,2,2,0,2. Then, for example, the fraction of 0's, 2/12 for the current realizations, can be used as an approximation for $\mathbb{P}(\{0\})$.

The sample space can be any set, but usually $\Omega \subset \mathbb{R}$. If this is not the case, then there is often a one-to-one relation between the elements of Ω and (a subset of) the real numbers. A random experiment taking values in the real numbers is called a *random variable*.

Example 1.1.2 Consider a machine, working at the beginning of a day, that may go down during that day or not. Then we can take $\Omega = \{\text{on}, \text{off}\}$, with $\mathbb{P}(\{\text{on}\})$ the probability that the machine does not go down. Now define the random variable X by taking $X(\text{off}) = 0$ and $X(\text{on}) = 1$. Then $\{X = 1\}$ corresponds to the event that the machine is on.

From now on we consider random variables taking values in the real numbers. There is no need to consider the (possibly different) underlying sample space.

There are several ways to characterize random variables. One is through the (cumulative) distribution function, usually denoted with F_X or F. The distribution function F_X of a random variable X denotes the following: $F_X(t) = \mathbb{P}(X \leq t) = \mathbb{P}(X \in (-\infty, t])$. Indeed, any (reasonable) set is composed of sets of the form $(-\infty, t])$. For example,

$$\mathbb{P}(X \in (s,t]) = \mathbb{P}(X \in (-\infty,t]) - \mathbb{P}(X \in (-\infty,s]) = F_X(t) - F_X(s),$$

for $s < t$. Therefore F_X fully specifies the distribution of X.

Note that F_X is always increasing. The set of all distributions can be roughly divided into two groups: those for which F_X is piecewise constant with at most a countable number of jumps, the discrete distributions, and those for which F is continuous and differentiable, the continuous distributions. (Also mixed versions of both types are possible, but they are less relevant.)

Discrete distributions are characterized by the probability mass on the points where F is discontinuous, i.e., where F makes a jump. Continuous distributions are characterized by $dF(t)/dt$, the *density* of the distribution, usually denoted with f. Note that

$$\int_u^v f(t)dt = \int_u^v F'(t)dt = F(v) - F(u),$$

thus f completely determines F. More generally, $\mathbb{P}(X \in A) = \int_A f(t)dt$ for $A \subset \mathbb{R}$. Instead of $\mathbb{P}(X \in A)$ we sometimes write $\mathbb{P}_X(A)$ or even $\mathbb{P}(A)$ when it is clear which random variable is meant.

Sometimes we are interested in the point t such that $F(t) \geq p$ for some $p \in (0,1)$. This point t is given by $t = F^{-1}(p)$ as long as F^{-1} is well defined. However, this is not always the case. For example, a discrete distribution is piecewise constant and F^{-1} is nowhere defined. To solve this we introduce the *quantile function*

$$F^{-1}(p) = \min_t \{F(t) \geq p\}. \tag{1.1}$$

Note that when F is strictly increasing then the quantile function coincides with the regular inverse of F.

The final characterization of a distribution is through its *hazard rate* function. This is the subject of Section 1.5. First however we need to introduce some other basic concepts of probability.

1.2 Expectations and moments

Quite often we are not interested in the outcome of an experiment, but in some function of the outcome. In the case of a random experiment, from a practical point of view, we are interested in the (long-run) average value of repetitions of the experiment. In mathematical terms, for a discrete distribution we define the expectation of $g(X)$, written as $\mathbb{E}g(X)$, as

$$\mathbb{E}g(X) = \sum_{x\in\mathbb{R}:\mathbb{P}(X=x)>0} g(x)\mathbb{P}(X = x). \tag{1.2}$$

If X has a continuous distribution with density f then we have

$$\mathbb{E}g(X) = \int_{-\infty}^{\infty} g(x)f(x)dx. \tag{1.3}$$

Note that any probability can be written as an expectation: $\mathbb{P}(X \in A) = \mathbb{E}\mathbb{I}\{X \in A\}$ with $\mathbb{I}$ the indicator function, i.e., $\mathbb{I}\{\cdot\} = 1$ if the argument is true, 0 otherwise.

The definition of $\mathbb{E}g(X)$ given in Equation (1.3) can be written as $\int g(x)dF(x)$. This notation is also used for the discrete case of Equation (1.2).

Note that, in general, $\mathbb{E}g(X) \neq g(\mathbb{E}X)$. In practice however, people often ignore variability when taking decisions. This is called the *Flaw of Averages*: Plans based on averages are wrong on average.

Another reason to study expectations is the following. A random variable is completely specified by its distribution function. But unless it has a known distribution depending on a few parameters, it cannot be easily characterized. This is certainly the case with measured data. Instead one often gives (estimators of) its first few *moments*. The kth moment of a r.v. X is defined as $\mathbb{E}X^k$. The first moment is of course simply the expectation. The kth *central* moment is defined by $\mathbb{E}(X - \mathbb{E}X)^k$. The second central moment is known as the variance, usually denoted with $\sigma^2(X)$, its root is called the standard deviation. The variance has the following properties:

$$\sigma^2(X) = \mathbb{E}(X - \mathbb{E}X)^2 = \mathbb{E}X^2 - (\mathbb{E}X)^2,\ \sigma^2(aX) = a^2\sigma^2(X), \text{ and } \sigma^2(a) = 0.$$

A dimensionless number that characterizes the variation is the *squared coefficient of variation* of a random variable X. It is defined by $c^2(X) = \sigma^2(X)/(\mathbb{E}X)^2$. It has the following properties: $c^2(aX) = c^2(X)$ and $c^2(a) = 0$. For examples of computations, see Section 1.6.

1.3 Multiple random variables and independence

Consider two random variables, X and Y (defined on the same probability space). We are interested in answering questions such as: what is $\mathbb{E}(X + Y)$? and $\mathbb{E}XY$?

The r.v. (X, Y) can be considered as a single two-dimensional random variable. If (X, Y) is discrete, then its distribution is defined by probabilities $\mathbb{P}((X, Y) = (x, y)) \geq 0$ with $\sum_{x,y}\mathbb{P}((X, Y) = (x, y)) = 1$, where the summation ranges over all x, y such that $\mathbb{P}((X, Y) = (x, y)) > 0$.

X itself is a random variable, with $\mathbb{P}(X = x) = \sum_y \mathbb{P}((X, Y) = (x, y))$. This is called the *marginal distribution*. In the same way the marginal distribution of Y can be found.

It holds that

$$\mathbb{E}(X+Y) = \sum_{x,y}(x+y)\mathbb{P}((X,Y)=(x,y)) =$$

$$\sum_x x \sum_y \mathbb{P}((X,Y)=(x,y)) + \sum_y y \sum_x \mathbb{P}((X,Y)=(x,y)) = \mathbb{E}X + \mathbb{E}Y.$$

Thus, independent of the *simultaneous* distribution of (X,Y), $\mathbb{E}(X+Y)$ depends only on the marginal distributions of X and Y.

On the other hand, it does not always hold that $\mathbb{E}XY = \mathbb{E}X\mathbb{E}Y$, as the following example shows: take $\mathbb{P}((X,Y)=(0,1)) = \mathbb{P}((X,Y)=(1,0)) = 1/2$. Then $\mathbb{E}XY = 0 \neq 1/4 = \mathbb{E}X\mathbb{E}Y$. For $\mathbb{E}XY = \mathbb{E}X\mathbb{E}Y$ we need an additional condition, which we introduce next: *independence*.

The events A and B are independent if

$$\mathbb{P}(A \cap B) = \mathbb{P}(A)\mathbb{P}(B).$$

Two random variables X and Y are called independent if

$$\mathbb{P}(X \leq x, Y \leq y) = \mathbb{P}(X \leq x)\mathbb{P}(Y \leq y)$$

for all x, y. In that case the joint 2-dimensional distribution function is the product of the marginal 1-dimensional distribution functions. The same holds for the density of a continuous distribution.

In the case of a discrete distribution we now have

$$\mathbb{E}XY = \sum_{x,y} xy\mathbb{P}((X,Y)=(x,y)) = \sum_{x,y} xy\mathbb{P}(X=x)\mathbb{P}(Y=y) =$$

$$\sum_x x\mathbb{P}(X=x) \sum_y y\mathbb{P}(Y=y) = \mathbb{E}X\mathbb{E}Y,$$

using independence in the second step.

From this follows, by some calculations, the following important formula for independent random variables X and Y:

$$\sigma^2(X+Y) = \sigma^2(X) + \sigma^2(Y).$$

We give some examples on how to compute expectations of functions of multiple independent random variables. Because of its relevance for the Pollaczek-Khintchine formula (see Section 5.3) we concentrate on the calculation of second moments.

An important class of compound distributions are random mixtures of the form $S = ZX_1 + (1-Z)X_2$ with $Z \in \{0,1\}$ and all variables independent. Then $\mathbb{E}S = p\mathbb{E}X_1 + (1-p)\mathbb{E}X_2$ with $p = \mathbb{P}(Z=1)$. Now

$$\mathbb{E}S^2 = \mathbb{E}[ZX_1 + (1-Z)X_2]^2 = \mathbb{E}(ZX_1)^2 + \mathbb{E}((1-Z)X_2)^2 + 2\mathbb{E}Z(1-Z)X_1X_2.$$

Using $Z(1-Z)=0$, $\mathbb{E}Z^2=\mathbb{E}Z$, and $\mathbb{E}(1-Z)^2=\mathbb{E}(1-Z)$, all because $Z\in\{0,1\}$, and independence, we find

$$\mathbb{E}S^2 = p\mathbb{E}X_1^2 + (1-p)\mathbb{E}X_2^2.$$

It is interesting to compare this with distributions of the form $\hat{S}=pX_1+(1-p)X_2$, a convex combination of X_1 and X_2. Then $\mathbb{E}S=\mathbb{E}\hat{S}$, but

$$\mathbb{E}\hat{S}^2 = p^2\mathbb{E}X^2 + (1-p)^2\mathbb{E}X_2^2 + 2p(1-p)\mathbb{E}X_1\mathbb{E}X_2.$$

An important example of a random mixture is the hyperexponential distribution, in which case the X_i are exponentially distributed.

1.4 Conditional probability

There is another important concept related to probability measures that will be used often throughout these lecture notes: conditional probability. Consider two events $A,B\subset\Omega$. Then the conditional probability that A occurs given B, written as $\mathbb{P}(A|B)$, is defined by

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(A\cap B)}{\mathbb{P}(B)}, \tag{1.4}$$

which is only defined if $\mathbb{P}(B)>0$. We often write $\mathbb{P}(AB)$ instead of $\mathbb{P}(A\cap B)$.

For a random variable X we are sometimes not just interested in $\mathbb{P}(X\in A|X\in B)=\mathbb{P}(A|B)$, but in $\mathbb{P}(X\in A|X\in B)$ for all possible A. The resulting random variable, taking values in B, is denoted with $X|X\in B$. Now the expectation $\mathbb{E}[g(X)|X\in B]$ can be defined in the obvious way. For example, if X is discrete, we get:

$$\mathbb{E}[g(X)|X\in B] = \sum_x g(x)\mathbb{P}(X=x|X\in B).$$

We use Equation (1.4) to obtain:

$$\mathbb{P}(A) = \mathbb{P}(AB)+\mathbb{P}(AB^c) = \mathbb{P}(A|B)\mathbb{P}(B)+\mathbb{P}(A|B^c)\mathbb{P}(B^c),$$

where B^c denotes the complement of B. This is called the *law of total probability*. It can be generalized as follows: let $B_1,B_2,\dots$ be events such that $B_i\cap B_j=\emptyset$, and $\cup_{k=1}^\infty B_k\supset A$. Then

$$\mathbb{P}(A) = \sum_{k=1}^\infty \mathbb{P}(A|B_k)\mathbb{P}(B_k). \tag{1.5}$$

Using (1.4) we find

$$\mathbb{P}(A|C) = \frac{\mathbb{P}(AC)}{\mathbb{P}(C)} = \sum_{k=1}^\infty \frac{\mathbb{P}(AB_kC)\mathbb{P}(B_kC)}{\mathbb{P}(B_kC)\mathbb{P}(C)} = \sum_{k=1}^\infty \mathbb{P}(A|B_kC)\mathbb{P}(B_k|C), \tag{1.6}$$

a useful generalization of (1.5).

Let X be some r.v. on the same probability space. By integrating or summing Equation (1.5) over the probability space we find another useful formula:

$$\mathbb{E}X = \sum_{k=1}^{\infty} \mathbb{E}(X|B_k)\mathbb{P}(B_k). \tag{1.7}$$

1.5 Hazard rates

The idea of the hazard rate comes from the maintenance of systems, where it is crucial to know the remaining lifetime of a component given that it is currently functioning. It is also important to insurances.

Let X be a positive, continuous random variable with density f. Then, with $\overline{F}(t) = 1 - F(t)$:

$$\mathbb{P}(X \leq t+h|X > t) = \frac{\mathbb{P}(t < X \leq t+h)}{\mathbb{P}(X > t)} = \frac{\int_t^{t+h} f(s)ds}{\overline{F}(t)} \approx \frac{f(t)h}{\overline{F}(t)}. \tag{1.8}$$

This approximation gets more accurate and eventually becomes an equality as $h \to 0$. To be able to write this in a mathematically correct way we introduce the following concept.

Definition 1.5.1 *A function $f(h)$ is* of small order $g(h)$, *notated as $o(g(h))$, if*

$$\lim_{h\to 0} \frac{f(h)}{g(h)} = 0.$$

Example 1.5.2 $h^2 = o(h)$, because $h^2/h \to 0$.

Using Definition 1.5.1 we can make Equation (1.8) more precise:

$$\mathbb{P}(X \leq t+h|X > t) = \frac{f(t)h}{\overline{F}(t)} + o(h). \tag{1.9}$$

This motivates the definition of the *hazard rate* $\lambda(t)$.

Definition 1.5.3 *The hazard rate $\lambda(t)$ of a random variable with density f is given by*

$$\lambda(t) = \frac{f(t)}{\overline{F}(t)}.$$

Thus $\lambda(t)h$ is approximately the probability that X fails in the first h time units after t. Instead of hazard rate one also uses the term *failure rate*; this terminology comes evidently from its use in the study of systems that are prone to failure. We use the more neutral term hazard rate.

Example 1.5.4 An exponential distribution with parameter γ has $F(t) = 1 - \exp(-\gamma t)$, $f(t) = \gamma \exp(-\gamma t)$, and thus $\lambda(t) = \gamma$ for all t.

The hazard rate completely characterizes a distribution. To see this, define $\Lambda(t) = \int_0^t \lambda(s)ds$, the *hazard function*. We see that

$$\Lambda(t) = \int_0^t \frac{f(s)}{\overline{F}(s)}ds = -\log \overline{F}(t), \tag{1.10}$$

and therefore $\overline{F}(t) = \exp(-\Lambda(t))$. Thus the distribution function F is completely determined by λ (and by Λ as well).

1.5.1 Minima and sums of random variables

A useful property of random variables with hazard rates is the following. Consider independent X and Y with hazard rates λ_X and λ_Y. Then

$$\mathbb{P}(\max\{X,Y\} \le t+h|X,Y > t) = \mathbb{P}(X,Y \le t+h|X,Y > t) =$$
$$\mathbb{P}(X \le t+h|X > t)\mathbb{P}(Y \le t+h|Y > t) =$$
$$[\lambda_X(t)h + o(h)][\lambda_Y(t)h + o(h)] = o(h),$$

because $h^2 = o(h)$. This can also be interpreted as follows: if events are happening at a certain rate in parallel, then the probability of more than one event happening in an interval of length h is $o(h)$.

A similar argument can be used to determine the hazard rate of minima of random variables:

$$\mathbb{P}(\min\{X,Y\} \le t+h|X,Y > t) =$$
$$\mathbb{P}(X \le t+h, Y > t+h|X,Y > t) + \mathbb{P}(X > t+h, Y \le t+h|X,Y > t)+$$
$$\mathbb{P}(X,Y \le t+h|X,Y > t) =$$
$$[\lambda_X(t)h + o(h)][1 - \lambda_Y(t)h + o(h)] + [1 - \lambda_X(t)h + o(h)][\lambda_Y(t)h + o(h)] + o(h) =$$
$$\lambda_X(t)h + \lambda_Y(t)h + o(h).$$

Thus the hazard rate of a minimum is the sum of the hazard rates.

Instead of maxima we can also look at sums:

$$\mathbb{P}(X+Y \le s+t+h|X > s, Y > t) \le \mathbb{P}(X \le s+h, Y \le t+h|X > s, Y > t) =$$
$$\mathbb{P}(X \le s+h|X > s)\mathbb{P}(Y \le t+h|Y > t) = \tag{1.11}$$
$$[\lambda_X(s)h + o(h)][\lambda_Y(t)h + o(h)] = o(h).$$

Again, the probability of two events happening in h time is $o(h)$.

1.6 Some important distributions

In this section we present some distributions that play a role in these lecture notes. They are all implemented in mathematical packages such as Maple and Matlab, and also in spreadsheet packages such as Excel.

1.6.1 The alternative or Bernoulli distribution

A random variable Z on $\{0, 1\}$ has an alternative or Bernoulli distribution with parameter $p \in (0, 1)$ if $\mathbb{P}(Z = 1) = 1 - \mathbb{P}(Z = 0) = p$. It represents the outcome of flipping a coin: if $p = 0.5$ then the coin is called *unbiased*. We find $\mathbb{E}Z = \mathbb{E}Z^2 = p$, and thus $\sigma^2(Z) = p(1-p)$.

1.6.2 The geometric distribution

A random variable N on $\mathbb{N}$ has a geometric distribution with parameter p if the following holds for $n \in \mathbb{N}$:

$$\mathbb{P}(N = n) = (1 - p)^{n-1}p.$$

Some important properties of the geometric distribution are:

$$\mathbb{E}N = 1/p, \quad \mathbb{P}(N \leq n) = 1 - (1 - p)^n.$$

A geometric distribution can be constructed as follows. Consider a number of independent alternative experiments with parameter p. If 1 occurs then we stop. The total number of experiments then has a geometric distribution with parameter p.

A special property of the geometric distribution is the fact that it is *memoryless*:

$$\mathbb{P}(N = m + n|N > n) = \frac{\mathbb{P}(N = m + n)}{\mathbb{P}(N > n)} = \frac{(1 - p)^{m+n-1}p}{(1 - p)^n} = \mathbb{P}(N = m),$$

for all $m > 0$ and $n \geq 0$. We conclude that the distribution of $N - n|N > n$ is independent of n. Thus, in terms of life times, the remaining life time distribution is independent of the current age. This is why N is called memoryless.

1.6.3 The binomial distribution

A binomial random variable N with parameters K and p has the following distribution:

$$\mathbb{P}(N = n) = \binom{K}{n} p^n (1 - p)^{K-n},$$

for $n \in \{0, \dots, K\}$. Remember that $\binom{K}{n} = \frac{K!}{n!(K-n)!}$. A binomial random variable can be interpreted as the sum of K alternative experiments with parameter p. Thus N denotes the number of 1's. From this it follows that $\mathbb{E}N = pK$ and $\sigma^2(N) = p(1 - p)K$. From this interpretation it is also intuitively clear that sums of binomial random variables with the same probabilities are again binomial random variables.

1.6.4 The Poisson distribution

The Poisson distribution is a discrete distribution on $\mathbb{N}_0$. For a Poisson distribution N with parameter λ holds:

$$\mathbb{P}(N = n) = \frac{\lambda^n}{n!} e^{-\lambda}.$$

We have

$$\mathbb{E}N = \lambda \text{ and } \mathbb{E}N^2 = \lambda(1 + \lambda)$$

and thus

$$\sigma^2(N) = \lambda \text{ and } c^2(N) = \frac{\sigma^2(N)}{(\mathbb{E}N)^2} = \frac{1}{\lambda}.$$

An important property of the Poisson distribution is the fact that sums of independent Poisson distributed random variables have again Poisson distributions. From this and the Central Limit Theorem (see Section 1.7) it follows that the Poisson distribution $Poisson(\lambda)$ with λ big is well approximated by the normal distribution $N(\lambda, \lambda)$. (See Subsection 1.6.8 for more information on the normal distribution.)

A Poisson distribution can be interpreted as the limit of a number of binomial distributions; see Section 2.1.

Another property is as follows. Construct from the Poisson distribution N two new distributions N_1 and N_2: each point in N is assigned independently to N_1 according to an alternative distribution with success probability p, otherwise it is assigned to N_2. Then N_1 and N_2 have independent Poisson distributions.

1.6.5 The exponential distribution

The exponential distribution plays a crucial role in many parts of these lecture notes. Recall that for X exponentially distributed with parameter $\mu \in \mathbb{R}_{>0}$ holds:

$$\begin{gathered}
F_X(t) = \mathbb{P}(X \leq t) = 1 - e^{-\mu t}, \quad f_X(t) = F'_X(t) = \mu e^{-\mu t}, \quad t \geq 0, \\
\mathbb{E}X = \int_0^\infty t\mu e^{-\mu t} dt = \frac{1}{\mu}, \quad \mathbb{E}X^2 = \int_0^\infty t^2 \mu e^{-\mu t} dt = \frac{2}{\mu^2}, \qquad (1.12) \\
\sigma^2(X) = \mathbb{E}X^2 - (\mathbb{E}X)^2 = \frac{1}{\mu^2}, \text{ and } c^2(X) = \frac{\sigma^2(X)}{(\mathbb{E}X)^2} = 1.
\end{gathered}$$

For the hazard rate we find

$$\lambda(t) = \frac{f(t)}{\overline{F}(t)} = \frac{\mu e^{-\mu t}}{e^{-\mu t}} = \mu.$$

An extremely important property of the exponential distribution is the fact that it is *memoryless*:

$$\mathbb{P}(X \leq t + s | X > t) = \frac{\mathbb{P}(X \leq t + s, X > t)}{\mathbb{P}(X > t)} = \frac{\mathbb{P}(X \leq t + s) - \mathbb{P}(X \leq t)}{e^{-\lambda t}} =$$

$$\frac{e^{-\lambda t} - e^{-\lambda(t+s)}}{e^{-\lambda t}} = 1 - e^{-\lambda s} = \mathbb{P}(X \leq s).$$

We continue with some properties of $\min\{X, Y\}$ if both X and Y are exponentially distributed (with parameters λ and μ, respectively) and independent:

$$\mathbb{P}(\min\{X, Y\} \leq t) = 1 - \mathbb{P}(\min\{X, Y\} > t) = 1 - \mathbb{P}(X > t, Y > t) =$$

$$1 - \mathbb{P}(X > t)\mathbb{P}(Y > t) = 1 - e^{-\lambda t}e^{-\mu t} = 1 - e^{-(\lambda+\mu)t}.$$

Thus $\min\{X, Y\}$ is again exponentially distributed with as rate the sum of the individual rates. Repeating this argument shows that the minimum of any number of exponentially distributed random variables has again an exponential distribution. We also have:

$$\mathbb{P}(X \leq Y | \min\{X, Y\} \geq t) = \frac{\mathbb{P}(X \leq Y, \min\{X, Y\} \geq t)}{\mathbb{P}(\min\{X, Y\} \geq t)} =$$

$$\frac{\mathbb{P}(X \leq Y, X \geq t, Y \geq t)}{\mathbb{P}(X \geq t, Y \geq t)} = \frac{\mathbb{P}(X \leq Y, X \geq t)}{\mathbb{P}(X \geq t)\mathbb{P}(Y \geq t)} = \frac{\int_t^\infty \int_x^\infty \lambda e^{-\lambda x} \mu e^{-\mu y} dy dx}{e^{-\lambda t}e^{-\mu t}} =$$

$$\frac{\int_t^\infty \lambda e^{-\lambda x} e^{-\mu x} dx}{e^{-\lambda t}e^{-\mu t}} = \frac{\frac{\lambda}{\lambda+\mu} e^{-\lambda t}e^{-\mu t}}{e^{-\lambda t}e^{-\mu t}} = \frac{\lambda}{\lambda + \mu}.$$

This means that the probability that the minimum is attained by X in $\min\{X, Y\}$ is proportional to the rate of X, independent of the value of $\min\{X, Y\}$.

Finally, consider aX with a a constant and X exponentially distributed with parameter μ. Then

$$\mathbb{P}(aX \leq t) = \int_0^{\frac{t}{a}} \mu e^{-\mu x} dx = \frac{1}{a} \int_0^t \mu e^{-\mu \frac{x}{a}} dx = \mathbb{P}(Y \leq t)$$

with Y exponentially distributed with parameter μ/a. Thus the parameter of the exponential distribution is a *scale* parameter: changing it does not change the shape, only the scale. A consequence is that the coefficient of variation does not depend on the parameter.

1.6.6 The gamma distribution

The sum of k independent exponentially distributed random variables with parameter $\mu \in \mathbb{R}_{>0}$ has a gamma distribution with parameters $k \in \mathbb{N}$ and μ. For obvious reasons k is called the shape parameter and μ is called the scale parameter. We have for $X \sim \text{gamma}(k, \mu)$:

$$\mathbb{E}X = \frac{k}{\mu}, \ \sigma^2(X) = \frac{k}{\mu^2}, \text{ and } c^2(X) = \frac{1}{k}. \tag{1.13}$$

The density f_X and distribution function F_X are as follows, for $t \geq 0$:

$$f_X(t) = \frac{\mu e^{-\mu t}(\mu t)^{k-1}}{(k-1)!}, \quad F_X(t) = 1 - \sum_{n=0}^{k-1} \frac{(\mu t)^n}{n!} e^{-\mu t}.$$

The gamma distribution can also be defined for $k \in \mathbb{R}_{>0}$, using the gamma function. For integer k the gamma distribution is also known under the name Erlang distribution.

For N a Poisson distribution with parameter μt the following interesting relation exists: $F_X(t) = \mathbb{P}(N \geq k)$. The intuition behind this relation is explained in Chapter 2.

1.6.7 The uniform distribution

The uniform distribution on $[0,1]$ has a density 1 on $[0,1]$ and 0 elsewhere. Its distribution function F is given by $F(t) = t$ on $[0,1]$, $F(t) = 0$ for $t \leq 0$ and $F(t) = 1$ for $t \geq 1$. Let X be uniform on $[0,1]$. Its properties are:

$$\mathbb{E}X = \frac{1}{2},\ \sigma^2(X) = \frac{1}{12},\ \lambda(t) = \frac{1}{1-t}.$$

This distribution can simply be generalized to the domain $[a,b]$ by taking $f = (b-a)^{-1}$ on it and 0 elsewhere.

1.6.8 The normal distribution

The normal distribution arises naturally when averaging i.i.d. (independent and identically distributed) random variables, see the Central Limit Theorem (Section 1.7). A normally distributed r.v. X with parameters μ and σ (denoted as $X \sim N(\mu, \sigma^2)$) has a density f given by:

$$f(t) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)^2}.$$

This density is symmetric around μ, and has the well-known bell shape. The distribution is also known as the Gaussian distribution.

A closed-form expression for the distribution function does not exist. The first two central moments are given by:

$$\mathbb{E}X = \mu, \quad \sigma^2(X) = \sigma^2.$$

The standard normal distribution is denoted with $N(0,1)$, and $(X-\mu)/\sigma \sim N(0,1)$. The distribution function of the standard normal distribution is usually denoted with Φ (and its density with ϕ). Thus, with $X \sim N(0,1)$, $\mathbb{P}(X \leq x) = \Phi(x)$. Traditionally, books on probability or statistics contained a table with $\Phi(x)$ for different values of x. They are nowadays replaced by software, for example, the Excel function NORMDIST.

To have a general idea of the meaning of the standard deviation of the normal distribution it is convenient to remember:

$$\mathbb{P}(\mu - \sigma \leq X \leq \mu + \sigma) \approx 0.68 \text{ and } \mathbb{P}(\mu - 2\sigma \leq X \leq \mu + 2\sigma) \approx 0.95.$$

An important property of the normal distribution is that independent sums of normal distributions have again normal distributions.

1.6.9 The lognormal distribution

When Y has a normal distribution, then the positive random variable $X = e^Y$ has a lognormal distribution.

The normal distribution arises when we encounter sums of random variables; the lognormal show up in case of multiple multiplicative effects. There is evidence that this is the case with the length of conversations, and indeed the lognormal distribution fits well historical data on call durations.

If $Y \sim N(\mu, \sigma^2)$, then

$$\mathbb{E}X = e^{\mu+\sigma^2/2}, \quad \sigma^2(X) = (e^{\sigma^2} - 1)e^{2\mu+\sigma^2}.$$

1.7 Limit theorems

In the beginning of this chapter it was noted that the practical interpretation of the probability of an event is that of the long-run frequency that the event occurs. To make probability theory practically relevant it is therefore crucial that within the mathematical framework frequencies of events converge to the corresponding probabilities. For this reason limit theorems, and especially the *law of large numbers*, are at the heart of probability.

Consider some r.v. X for which we like to approximate $\mathbb{E}g(X)$. We do n i.i.d. experiments $X_1, \dots, X_n$ with $X_i \sim X$. Then the Law of Large Numbers tells us that

$$\frac{g(X_1) + \cdots + g(X_n)}{n} \to \mathbb{E}g(X) \tag{1.14}$$

with probability 1. If we take $g(x) = \mathbb{I}\{x \in A\}$ for some event A, then we find:

$$\frac{\mathbb{I}\{X_1 \in A\} + \cdots + \mathbb{I}\{X_n \in A\}}{n} \to \mathbb{P}(X \in A),$$

which means that the frequency of an event converges to its probability, exactly as we want it to be.

Equation (1.14) is intuitively clear, because

$$\sigma^2\Big(\frac{g(X_1) + \cdots + g(X_n)}{n}\Big) = \frac{n}{n^2}\sigma^2(g(X)) \to 0,$$

the variance of the average of n i.i.d. random variables converges to 0.

How quickly does the variance of the average converge to 0? To answer this question, we make the following assumption to simplify notation: $g(x) = x$. This is not a restriction, as $g(X)$ is also a random variable. We use the following notation: $\hat{X}_n = (X_1 + \cdots + X_n)/n$. Then $\sigma^2(\hat{X}_n) = \sigma^2(X)/n$. Thus $\sqrt{n}(\hat{X}_n - \mathbb{E}X)$ has expectation 0 and variance $\sigma^2(X)$, independent of n. In addition, the distribution of $\sqrt{n}(\hat{X}_n - \mathbb{E}X)$ tends in the limit to a normal distribution:

$$\frac{\sqrt{n}(\hat{X}_n - \mathbb{E}X)}{\sigma(X)} = \frac{X_1 + \cdots + X_n - n\mathbb{E}X}{\sqrt{n}\sigma(X)} \to N(0,1),$$

with $N(0,1)$ the standard normal distribution. This result is known as the *Central Limit Theorem*. It is often used in statistics and in simulations, by assuming that it already holds for moderate values of n. In the following sections we will use it exactly for these reasons: first for parameter estimation, and then for Monte Carlo simulation.

However, in these situations the variance $\sigma^2(X)$ is usually not known, it has to be estimated as well. For this reason we introduce the *sample variance* $S_n^2(X)$ by

$$S_n^2(X) = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \hat{X}_n)^2.$$

It can be shown that $\mathbb{E}S_n^2(X) = \sigma^2(X)$, i.e., $S_n^2(X)$ is an unbiased estimator of $\sigma^2(X)$. Again, the Central Limit Theorem tells us that

$$\frac{\sqrt{n}(\hat{X}_n - \mathbb{E}X)}{S_n} \to N(0,1),$$

with S_n the square root of S_n^2. This gives us the possibility to derive, for n sufficiently large, confidence intervals for $\mathbb{E}X$: with Φ the distribution function of $N(0,1)$, the $1-2\alpha$ confidence interval is given by

$$\Big[\hat{X}_n - \Phi^{-1}(1-\alpha)\frac{S_n}{\sqrt{n}}, \hat{X}_n + \Phi^{-1}(1-\alpha)\frac{S_n}{\sqrt{n}}\Big]. \tag{1.15}$$

It is interesting to note that in order to reduce the length of the confidence interval by a factor 2, approximately 4 times as many observations are needed.

Also quantiles and other performance measures can be analyzed in the same way, the former by writing expressions of the form $\mathbb{P}(X \geq t)$ as $\mathbb{E}\mathbb{I}\{X \geq t\}$.

The average and the sample variance are implemented in Excel under the functions AVERAGE and VAR.

1.8 Parameter estimation

Most of the models discussed in Part III need one or more parameters as input. Estimating these parameters is the subject of this section. Sometimes the form of a distribution needs to be estimated as well, but this happens rarely: often we have certain reasons to assume that a random variable has a certain distribution (for example the Poisson distribution as a model for customer arrivals; see Section 2.1), or the form of the distribution is of little or no relevance to the model (as is the case for the higher moments of the service time distribution in the Erlang B model of Theorem 5.4.3).

Many distributions (such as the exponential and Poisson distributions) are determined by a single parameter, the expectation. According to the Law of Large Numbers it suffices in these cases to average the outcomes to obtain an estimate of the parameter value.

Example 1.8.1 In call centers, most performance models use exponential service times. The parameter is estimated by averaging over a large number of realizations. Another parameter that needs to be approximated is the *patience* of customers: some callers abandon before they get connected. In these cases their patience was shorter than their waiting time. In cases where the waiting time is shorter than the patience, then the latter is not observed. Thus we deal with so-called *censored data*. If the patience X is exponentially distributed, and Y is the waiting time distribution, then for $Z = \min\{X, Y\}$ it follows (see Exercise 1.12) that $\mathbb{E}X = \mathbb{E}Z/\mathbb{P}(X < Y)$. By applying the Law of Large Numbers to both $\mathbb{E}Z$ and $\mathbb{P}(X < Y)$ we find as estimator for $\mathbb{E}X$ the sum over all waiting times divided by the number of abandoned customers. This is a special case of the *Kaplan-Meier estimator*.

Evidently, the average over a finite number of realization is rarely exactly equal to the expectation. Therefore we should also take the variance of the parameter estimation into account, using the Central Limit Theorem.

For a random variable X with a single parameter the estimator of $\mathbb{E}X$ can be used to estimate $\sigma^2(X)$. For a random variable with more parameters a separate estimator of the variance should be used: the sample variance $S_n^2(X)$ (see Section 1.7).

Example 1.8.2 We repeat an experiment involving a biased coin: $X \in \{0, 1\}$. We are interested in $\mathbb{P}(X = 1)$. For this reason we take $g(X) = \mathbb{I}\{X = 1\} = X$. Note that the last equality holds because $X \in \{0, 1\}$. Thus $\hat{X}_n = (X_1 + \cdots + X_n)/n \to \mathbb{P}(X = 1)$. Now we should realize that $\sigma^2(X) = \mathbb{P}(X = 1)(1 - \mathbb{P}(X = 1)) \leq 1/4$. To obtain a 95% precision interval of width 0.02 for our estimate $\hat{X}_n$ (± 0.01) we need that $\sigma(\hat{X}_n) \leq 0.005$ because of the normal approximation (see Section 1.6.8). We have $\sigma(\hat{X}_n) = \sigma(X)/\sqrt{n} \leq 1/(2\sqrt{n})$, Thus to obtain $\sigma(\hat{X}_n) \leq 0.005$ we need $n \geq 10000$: we need not less than 10000 repetitions to obtain the required precision!

Example 1.8.3 The number of arrivals N to a service system is counted for 12 identical periods: 2, 2, 2, 4, 2, 5, 1, 2, 1, 1, 0, 5. If N has a Poisson distribution then it suffices to compute $\hat{N}_n = 2.25$. This gives immediately an estimator $\hat{\sigma}^2(N)$ of the variance as $\mathbb{E}N = \sigma^2(N)$: $\hat{\sigma}^2(N) = 2.25$. Thus $[2.25 - \Phi^{-1}(0.975)\sqrt{2.25}/\sqrt{12}, 2.25 + \Phi^{-1}(0.975)\sqrt{2.25}/\sqrt{12}] = [2.25 - 2\sqrt{2.25}/\sqrt{12}, 2.25 + 2\sqrt{2.25}/\sqrt{12}] = [1.38, 3.12]$ is a 95% confidence interval for the parameter of the Poisson distribution. When the distribution of N is unknown we have to compute the sample variance: $S_n^2(N) = 2.57$. In this case the confidence interval for the expected number of arrivals becomes even wider.

The last example showed realizations of the number of arrivals to a service system, from identical periods. This is often an unrealistic assumption. In the general situation we have to make predictions of the future on the basis of historical data. This process, commonly known as *forecasting*, is discussed in Section 2.5.

1.9 Monte Carlo simulation

In Section 1.2 we saw how to compute $\mathbb{E}g(X)$ when the distribution of X is completely specified. However, obtaining a closed-form expression is not always possible. As an alternative, we could use the Law of Large Numbers: we make the computer generate a number of experiments $x_1, \dots, x_n$ according to the distribution X and then we use $(g(x_1) + \cdots + g(x_n))/n$ as an estimate for $\mathbb{E}g(X)$. Then what remains is an estimation problem as in Section 1.8. This method is called (computer) *simulation*. Note that we usually have no idea about the distribution of the value we measure, otherwise we would probably be able to compute its value directly. Therefore we need the sample variance to compute confidence intervals for our observations.

There are two types of simulation models. The first type is the one described above, with X often multi-dimensional. This type is usually called *Monte Carlo simulation*. The other type consists of models in which we are interested in the long-run behavior of systems evolving over time (so called *dynamic systems*). This form of simulation is called *discrete-event simulation*. Discrete-event simulation is dealt with in Chapter 3. Here we consider Monte Carlo simulation.

Example 1.9.1 The estimation of next year's income statement of any company depends on many unknowns. Instead of working with point estimates for all entries, financial managers can work with distributions. Now, not only the expected income can be calculated, but also the probability of loss, and so forth.

There are several tools available for Monte Carlo simulation. Most used are those that are add-ins to the spreadsheet Excel, such as Crystal Ball. A disadvantage is that they are less focused on a mathematical analysis of the output. However, most tools do calculate the sample variance. In Crystal Ball it can be found in the report under "variance".

We finish this discussion of Monte Carlo simulation with discussing the way "random" numbers can be generated. This is crucial to any simulation program.

1.9.1 Pseudo-random numbers

The basic functionality of any simulation program is the fact that it can generate *pseudo-random numbers*. Pseudo-random numbers are not really random (they are generated by some deterministic algorithm), but a sequence of pseudo-random numbers resembles, for most practical purposes, sufficiently well to real random numbers. Pseudo-random numbers are usually integers between 0 and some very large number (say N). From that we can construct realizations of other probability distributions. For example, by dividing by N we get numbers that are uniformly distributed on $[0, 1]$. Any random variable X with a known inverse distribution function F^{-1} can now be sampled as follows: if U is uniformly distributed on $[0, 1]$, then $F^{-1}(U)$ has the same distribution as X:

$$\mathbb{P}(F^{-1}(U) \leq x) = \mathbb{P}(U \leq F(x)) = F(x) = \mathbb{P}(X \leq x).$$

For some distribution functions the inverse has a simple closed-form expression. This is for example the case for the exponential distribution. For other distribution functions the inverse is harder or even impossible to give. Distributions of this kind that we often use are the gamma distribution and the normal distribution. The gamma distribution can be seen as a sum of exponential distribution, and is therefore as easy to simulate as the exponential distribution. For the normal distribution efficient numerical methods exist.

The user is often not aware of the mathematical details discussed above. For example, the spreadsheet program Excel comes with a pseudo-random number generator RAND(), that can be used in conjunction with for example GAMMAINV() to generate random numbers from a number of different distributions. Additional functionality can be obtained by using add-ins, for example Crystal Ball. For debugging purposes it is essential that the program can be instructed to use the same random sequence every time, by setting a 'seed' value (the initial value of the random number generator).

1.10 Further reading

A book (and a website) dedicated to the Flaw of Averages is Savage [80].

Hazard rates and their properties are discussed in most books on reliability, such as Barlow & Proschan [14], Chapter 9 of Ross [75] (text book level) and Aven & Jensen [11] (advanced).

Most introductions to probability theory derive the Law of Large Numbers and the Central Limit Theorem. See, e.g., Ross [76].

1.11 Exercises

Exercise 1.1 Prove that $\mathbb{E}\max\{X,Y\} \geq \max\{\mathbb{E}X, \mathbb{E}Y\}$ for X and Y independent. Hint: Prove first that $\mathbb{E}\max\{a,X\} \geq \max\{a, \mathbb{E}X\}$.

Exercise 1.2 Prove Equation (1.9).

Exercise 1.3 Show that the sum of two independent Poisson distributions has again a Poisson distribution.

Exercise 1.4 A shop has a certain inventory of a certain product. The demand is exponentially distributed. Demand in excess of the inventory is lost. Give a formula for the expected sales and simplify this as much as possible.

Exercise 1.5 Consider a positive continuous random variable X. A graphical representation of it is the *Lorenz curve* L:

$$L(p) = \frac{\int_0^{F^{-1}(p)} tf(t)dt}{\mathbb{E}X}, \quad p \in [0,1].$$

a. Give an interpretation of the Lorenz curve: what does $L(p)$ mean?
b. Give expressions of the Lorenz curve for X uniformly and exponentially distributed.

Exercise 1.6 From the Poisson distribution N two new distributions N_1 and N_2 are formed: each point in N is assigned independently to N_1 according to an alternative distribution with success probability p, otherwise it is assigned to N_2. Show that N_1 and N_2 have independent Poisson distributions.

Exercise 1.7 Let X and Y be independent exponentially distributed, with parameters λ and μ. Show that $\mathbb{P}(X < Y) = \lambda/(\lambda + \mu)$.

Exercise 1.8 a. Show (1.12).
b. Show (1.13).
c. Determine the hazard rate of a gamma distribution with 2 phases. Is it increasing or decreasing? Explain intuitively your answer.

Exercise 1.9 Let $X \sim \text{Uniform}[0, 1]$ and $Y \sim \text{Uniform}[a, b]$ with $a < b$.
a. Express Y as a function of X.
b. Compute $\mathbb{E}Y$ and $\sigma^2(Y)$ by using the answer to a and the expressions for X given in Section 1.6.7, and by calculating them directly.

Exercise 1.10 Consider a hospital with two parallel ORs, each available for 8 hours. We have 14 operations to plan, all with expectation 1 hour. 8 of these have standard deviation 0, the 6 others 15 minutes.
a. We plan the operations with no variance on one OR, the other operations on the other OR. Estimate the expected number of operating rooms that exceed the planned finish time.
b. We plan 4 operations with no variance on each OR, and 3 operations with variance. Estimate the expected number of operating rooms that exceed the planned finish time.
c. Interpret the difference in findings between a and b.
For the total durations normal distributions can be used.

Exercise 1.11 Let $N_\lambda \sim \text{Poisson}(\lambda)$, and $X_\lambda \sim N(\lambda, \lambda)$.
a. Motivate, using the Central Limit Theorem, that X_λ is a good approximation for N_λ when λ gets big.
b. Determine, using some appropriate software tool, the minimal λ for which $\max_k |\mathbb{P}(N_\lambda \leq k) - \mathbb{P}(X_\lambda \leq k)| \leq 0.01$.

Exercise 1.12 This exercise is related to Example 1.8.1.
a. For $Z = \min\{X, a\}$ and X exponential, show that $\mathbb{E}Z = \mathbb{E}X\mathbb{P}(X < a)$.
b. Use the answer to a. to show that $\mathbb{E}Z = \mathbb{E}X\mathbb{P}(X < Y)$ for $Z = \min\{X, Y\}$ and X exponential.

Exercise 1.13 In a call center the number of calls that arrive during a day has a Poisson distribution. At the end of a certain day there have been 5327 calls. Give a 95% confidence interval for the parameter of the distribution.

Exercise 1.14 During a day 4 operations are planned in a certain operation room in a hospital, each having a $N(1.5, 0.25)$ distributed duration. The operation room is reserved for 7 hours. There is no lost time between operations and at the beginning of the day.
a. Use a simulation tool to estimate the probability that the operations take longer than the reserved time.
b. Verify this using a calculation.
c. Use a simulation tool to estimate the average time that operations exceed the planned time (finishing early is counted as 0).

Exercise 1.15 A bank employee processes mortgage requests involving a number of steps. Arrivals occur during working hours according to a Poisson process with rate 0.5. There are three processing steps, all having a uniformly distributed duration. Steps 1 and 3 are executed for every request, step 2 only for 30% of them. The upper and lower bound of the distribution are, in minutes: 30/40, 20/40, 40/60, respectively. Denote with S the time the employee works on an arbitrary request.
a. Compute $\mathbb{E}S$.
b. Estimate $\mathbb{E}S^2$ using simulation.
c. Estimate the expected time between arrival of a request and the time it has been processed using Theorem 5.3.2.

Exercise 1.16 The numbers of arrivals to a service center are noted during 100 days. The number at day i is given by x_i. We have $\sum_{i=1}^{100} x_i = 1763$ and $\sum_{i=1}^{100}(x_i - 17.63)^2 = 2351$.
a. Give a 95% confidence interval for the expected number of arrivals on a day.
b. Somebody thinks that the number of arrivals per day has a Poisson distribution. Is this likely to be the case? Motivate your answer.

Exercise 1.17 Consider n numbers $x_1, \dots, x_n$.
a. Show that $\alpha = \sum_i x_i/n$ minimizes $\sum_i (x_i - \alpha)^2$.
b. Which number minimizes $\sum_i |x_i - \alpha|$?
c. And which number minimizes $\sum_i [p(x_i - \alpha)^+ + (1-p)(\alpha - x_i)^+]$, for $p \in (0, 1)$?
Assume that $x_1, \dots, x_n$ are i.i.d. realizations of some r.v. X.
d. Rephrase the results in terms of unbiased estimators of functions of X.

Exercise 1.18 Let 0.13, 0.47 and 0.67 be 3 realizations of a uniform distribution on [0,1]. On the basis of these numbers, calculate 3 realizations of an exponential distribution with rate 2 using the method described in Section 1.9.1.

Exercise 1.19 Consider an operating room where 3 operations are planned, with expected durations 2, 3, and 2 hours, and with standard deviations 30, 60, and 15 minutes. The time reserved for the operations is 8 hours. The durations are assumed to be independent, and the operations are scheduled in the given order. We use a normal approximation for the total operation length. Give answers to the following questions, using simulations.
a. Give the probability that the total operating time exceeds 8 hours.
b. Give the expected tardiness, i.e., the time that the operations run late, not counting

the time that the operations end early. In mathematical notation: if X is the duration, then the tardiness is equal to $(X-8)^+$.
c. Give the expected fraction of cancelled operations, assuming that operations are cancelled when they are expected to finish late.
d. Repeat the previous questions, now using calculations based on ϕ and Φ.

Exercise 1.20 Consider a random variable X with hazard rate $\lambda(t)$, with the property that there is a $\lambda > 0$ such that $\lambda(t) \leq \lambda$ for all t. Now construct a random variable Y as follows. Sample according to an exponential distribution with rate λ. Let the result be t. Now we draw according to a Bernoulli distribution with parameter $\lambda(t)/\lambda$. In the case of success $Y = t$. Otherwise, we sample again according to an exponential distribution with rate λ. For the result s, we draw according to a Bernoulli distribution with parameter $\lambda(t+s)/\lambda$. In the case of success $Y = t+s$. Otherwise, we draw again from the exponential distribution, etc.
Show that X and Y have the same distribution. (Hint: show that Y has hazard rate $\lambda(t)$.)
Note that this procedure gives a way to simulate distributions which have a bounded, known hazard rate.

Chapter 2

The Poisson Process

In this chapter we study arrival processes, especially the Poisson process. We give an informal introduction, based on an intuitive understanding of the Poisson process as modeling arrivals coming from a large population.

2.1 Motivation

Suppose we have a population of size K, where each individual has probability p_K of generating a request for service. Then the number of requests N_K has a binomial distribution:

$$\mathbb{P}(N_K = n) = \binom{K}{n} p_K^n (1-p_K)^{K-n}.$$

It is intuitively clear that $\mathbb{E}N_K = \sum_{n=1}^{K} n\mathbb{P}(N_K = n) = p_K K$. Now increase K, while keeping the expected number of requests constant, i.e., $\mathbb{E}N_K = p_K K = \lambda$. Thus $p_K = \lambda/K$. Then

$$\lim_{K\to\infty} \mathbb{P}(N_K = n) = \lim_{K\to\infty} \binom{K}{n} p_K^n (1-p_K)^{K-n} = \lim_{K\to\infty} \binom{K}{n} (\frac{\lambda}{K})^n (1-\frac{\lambda}{K})^{K-n} =$$

$$\frac{\lambda^n}{n!} \lim_{K\to\infty} (1-\frac{\lambda}{K})^K \frac{K!}{(K-n)!(K-\lambda)^n} = \frac{\lambda^n}{n!} e^{-\lambda} = \mathbb{P}(N = n)$$

for N having a Poisson distribution with parameter λ. Thus the Poisson distribution can be used to model the number of service requests coming from a large group of potential users of the service. Its parameter λ represents the expected number of requests.

Now we consider also the *time* at which arrivals occur. Assume that they can occur in the interval $[0,T]$. For every request its arrival time is generated according to a probability distribution on $[0,T]$, independent from any other request. Now split $[0,T]$ in two intervals $[0,t]$ and $[t,T]$. Now let each request determine its arrival time according to a uniform distribution on $[0,T]$, thus every instant in $[0,T]$ is equally likely. Then an arrival occurs in $[0,t]$ ($[t,T]$) with probability t/T ($(T-t)/T$). (This is the homogeneous case, in Section

2.4 we discuss the general case.) Denote with $N(t,t')$ (or simply $N(t')$ if $t=0$) the number of arrivals in $[t,t']$, for $0 \leq t < t' \leq T$. Assume that $\mathbb{E}N(T) = \lambda T$, thus on average λ arrivals occur per time unit. Then, according to Exercise 1.6, $N(t)$ and $N(t,T)$ have independent Poisson distributions with parameters λt and $\lambda(T-t)$.

Both facts, derived from practically logical customer behavior, are used as a definition of the Poisson process.

2.2 The homogeneous Poisson process

Consider events, typically arrivals to some service center, that occur at random moments in $[0,\infty)$. Let $N(t)$, for every $t \in [0,\infty)$, be a random variable that counts the number of events in $[0,t]$. Then we call $N(t)$ a *counting process*. We also define $N(s,t) = N(t) - N(s)$, the number of arrivals in $(s,t]$, for $0 \leq s < t$.

Definition 2.2.1 *The counting process $N(t)$ on $[0,\infty)$ is called a (homogeneous) Poisson process with rate λ if:*
- $N(s,t)$ has a Poisson distribution with expectation $\lambda(t-s)$ for all $0 \leq s < t$;
- $N(s,t)$ and $N(s',t')$ are stochastically independent for all $0 \leq s < t \leq s' < t'$.

Now we consider interarrival times. Let X_1 be the time until the first arrival of a request. Let $0 \leq t \leq T$. Then

$$\mathbb{P}(X_1 > t) = P(N(t) = 0) = e^{-t\lambda}.$$

Thus X_1 has an exponential distribution. The same holds for X_2, the time between the first and second arrival:

$$\mathbb{P}(X_2 > t | X_1 = s) = P(N(s,s+t) = 0) = e^{-t\lambda}.$$

Note that $\mathbb{P}(X_2 > t | X_1 = s)$ does not depend on s, and thus X_1 and X_2 are also independent. This argument can be repeated for all other interarrival times. Thus all interarrival times of a homogeneous Poisson process have independent exponentially distributed interarrival times with the same parameter. This is the most often used definition of the Poisson process.

Definition 2.2.2 *The counting process $N(t)$ on $[0,\infty)$ is called a (homogeneous) Poisson process with rate λ and interarrival times $X_1, X_2, \ldots$ if all X_i are independent and identically exponentially distributed with parameter λ.*

At the end of this section we will show that both definitions are equivalent.

Definition 2.2.2 is very useful when we try to simulate a Poisson process: we simple generate realizations of the exponential sojourn times using the method described in Section 1.9.1.

There is a third definition of the Poisson process based on the concept of *rates*. In Section 1.5 we saw that the exponential distribution is defined by the fact that the hazard rate is constant. Thus we can see the Poisson process as a stream of points generated at a constant rate. In the next definition we use the notion "small order of", see Definition 1.5.1.

Definition 2.2.3 *The counting process $N(t)$ on $[0, \infty)$ is called a (homogeneous) Poisson process with rate λ if:*
- $N(s, t)$ and $N(s', t')$ are stochastically independent for all $0 \leq s < t \leq s' < t'$;
- $\mathbb{P}(N(t, t + h) = 1) = \lambda h + o(h)$, $\mathbb{P}(N(t, t + h) > 1) = o(h)$ for all $t \geq 0$ and $h > 0$.

Theorem 2.2.4 *Definitions 2.2.1, 2.2.2, and 2.2.3 are equivalent.*

Proof It has already been argued that Definition 2.2.2 follows from Definition 2.2.1. Vice versa, the time until the kth arrival from s on has a gamma or Erlang distribution. Using its distribution function it can be seen that $N(s, t)$ has a Poisson distribution. The independence follows from the memoryless property of the exponential distribution.

Definition 2.2.3 suggests yet another interpretation of the Poisson process. Every h time units a Bernoulli experiment is executed with success probability λh. If successful, an arrival occurs, otherwise nothing happens. In the limit, as $h \to 0$, this also gives a Poisson process (see Exercise 2.3).

The first part of Definition 2.2.3 follows from Definition 2.2.1, the second part from Definition 2.2.2. For the reverse we refer to Theorem 5.1 of Ross [75]. □

2.3 Merging and splitting

In this section we consider merging and splitting of Poisson processes. We start with merging: what can we say about the sum N of two independent Poisson processes N_1 and N_2? To show that N is again a Poisson process we check Definition 2.2.1. The first point of the definition is satisfied because of Exercise 1.3. The second point follows directly from the independence.

Now we consider splitting. With splitting we mean that from a Poisson process N two new processes N_1 and N_2 are formed: each point in N is assigned independently to N_1 according to an alternative distribution with success probability p, otherwise it is assigned to N_2. We claim that N_1 and N_2 are independent Poisson processes. This can be shown by checking the conditions of Definition 2.2.1, see Exercise 1.6.

2.4 The inhomogeneous Poisson process

As in Section 2.1, consider two intervals $[0, t]$ and $[t, T]$. For each arrival occuring in $[0, T]$ it was assumed that the arrival moment was determined according to a uniform distribution on $[0, T]$. Here we abandon this assumption, instead we assume that the arrival time is determined according to a distribution with a piece-wise continuous density f on $[0, T]$.

Define $\gamma = \mathbb{E}N(T)$. Then $N(s,t)$ has a Poisson distribution with parameter $\gamma \int_s^t f(u)du$, and arrivals in disjunct intervals are again independent.

Define $\lambda(t) = f(t)\gamma$. The function $\lambda(t)$ is called the rate function, and it has the following interpretation: $\mathbb{E}N(s,t) = \int_s^t \lambda(u)du$, and $\frac{dN(t)}{dt} = \lambda(t)$ for all t (for which it exists). These observations lead to the following definition.

Definition 2.4.1 *The counting process $N(t)$ on $[0,\infty)$ is called an inhomogeneous Poisson process with rate function $\lambda(t)$ if:*
- $N(s,t)$ has a Poisson distribution with expectation $\int_s^t \lambda(u)du$ for all $0 \le s < t$;
- $N(s,t)$ and $N(s',t')$ are stochastically independent for all $0 \le s < t \le s' < t'$.

Note that if $\lambda(t)$ is constant then we have a homogeneous Poisson process.

Let us now study the interarrival times. The time until the next arrival after a fixed point in time is characterized by the rate function $\lambda(t)$, which should now be interpreted as the hazard rate (see Section 1.5, in which we used the same notation). Indeed, with X_1 the time until the first arrival,

$$\mathbb{P}(X_1 > t) = \mathbb{P}(N(t) = 0) = e^{-\int_0^t \lambda(s)ds},$$

which is, according to Equation (1.10), equivalent to saying that $\lambda(t)$ is the hazard rate of X_1. Thus X_1 can have any distribution, depending on the rate function $\lambda(t)$.

Let us now consider the second interarrival time X_2. We have

$$\mathbb{P}(X_2 > t|X_1 = s) = P(N(s,s+t) = 0) = e^{-\int_s^{s+t} \lambda(u)du}.$$

This clearly depends on s, and thus X_1 and X_2 are dependent in general. For this reason we cannot formulate a definition equivalent to Definition 2.2.2 for the inhomogeneous Poisson process.

A definition using rates, equivalent to Definition 2.2.3, is more easily formulated.

Definition 2.4.2 *The counting process $N(t)$ on $[0,\infty)$ is called an inhomogeneous Poisson process with rate $\lambda(t)$ if:*
- $N(s,t)$ and $N(s',t')$ are stochastically independent for all $0 \le s < t \le s' < t'$;
- $\mathbb{P}(N(t,t+h) = 1) = \lambda(t)h + o(h)$, $\mathbb{P}(N(t,t+h) > 1) = o(h)$ for all $t \ge 0$ and $h > 0$.

Definition 2.4.2 can be used when we want to simulate inhomogeneous Poisson processes with bounded rates. This works as follows. Let the rate of the Poisson process be bounded by the number λ, that is, $\lambda(t) \le \lambda$. Now we simulate a Poisson process with rate λ, and we take a point t of this simulation as a point in the inhomogeneous Poisson process with probability $\lambda(t)/\lambda$. This gives a process with rate $\lambda(t)$ (see also exercise 1.20).

Theorem 2.4.3 *Definitions 2.4.1 and 2.4.2 are equivalent.*

Proof Definition 2.4.2 follows from Definition 2.4.1 using properties of the Poisson distribution. The reverse follows from Theorem 1.3.1 of Tijms [93]. □

Definition 2.4.2 is equivalent to the results in Section 1.5.1 in the sense that for both cases, events governed by rates running in parallel and in series, the probability of two events in time h is $o(h)$.

2.5 Parameter estimation and forecasting

Suppose we observe a homogeneous Poisson process on $[0,\infty)$ with an (unknown) rate λ. Then $N(t)/t$ is an unbiased estimator of λ. We also have $N(t) = \sum_{s=1}^{t} N(s-1,s)$, with all $N(s-1,s)$ i.i.d. (independent and identically distributed). Thus the law of large numbers applies, and $N(t)/t \to \lambda$.

Consider that we have realizations $x_1,\dots,x_n$ of the numbers of arrivals of consecutive time periods. To check whether they are likely to come from a homogeneous Poisson process we can calculate the sample variance

$$s_n^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \hat{x}_n)^2.$$

If $s_n^2 \approx \hat{x}_n$ then it is well possible that the numbers come from Poisson distributions with the same rate because for a Poisson distribution expectation and variance are equal. However, it often occurs that $s_n^2 > \hat{x}_n$. We call this *overdispersion* with respect to the Poisson process. There are two possible reasons for this: either the numbers of arrivals are not Poisson, or the arrivals are Poisson but with different parameters.

Example 2.5.1 Traffic accidents occur according to a Poisson process, but they regularly involve more than one person. These *batch arrivals*, as seen by the ambulances and hospital emergency departments, cause overdispersion. At the same time, the parameter of the Poisson process depends on the day of the week, weather conditions, and so forth. This also causes overdispersion.

Many customer arrival processes in practice are inhomogeneous Poisson processes. They are usually modeled with piecewise-constant rate functions, because the expected numbers of arrivals are estimated for intervals of a fixed length in which the arrival rate is assumed to be constant. It is observed that these arrival processes often have weekly and daily patterns that change little over time. Forecasting arrival processes then amounts to determining the weekly and daily patterns once, and then estimating the weekly totals through some time series method (moving averages or exponential smoothing are often used). The advantage of this is that the weekly totals have a relatively low variance (that is, a low coefficient of variation, because the expectation is big), and thus there is relatively little noise in the observations. This is the way it is often done in for example call centers (discussed in Chapter 13). The disadvantage of moving averages and exponential smoothing is that trends are not followed well: the forecast is a weighted average of previous points.

Methods like Holt-Winters and linear regression do capture trends: they can be seen as generalizations of exponential smoothing and moving averages, respectively.

Additional variables determining estimations are related to exceptional events which can both internal or external to the company (e.g., a marketing campaign, or the holiday calendar). Combining all these factors into reasonable estimations is a difficult exercise which requires more than applying the right mathematical formula.

In practice these point estimates for parameter values are often used. But, as Silver & Peterson [83] state it: "Only one thing is certain—the forecast will be in error." Therefore we should do more than looking at point estimates, but for example look at confidence intervals. A way to implement this is *quantile regression*. For points (x_i, y_i) linear regression minimizes $\sum_i (y_i - a - bx_i)^2$. It can be seen (see Exercise 1.17) that $\sum_i [(1-\tau)(a+bx_i - y_i)^+ + \tau(y_i - a - bx_i)^+]$ is minimized by a τ-quantile. The quantile can be found using linear programming (see Exercise 2.7).

2.6 Other arrival processes

Many arrivals streams in practice can well be modeled by the (inhomogeneous) Poisson process. However, there are a number of generalizations worth mentioning. We already encountered the Poisson process where each point is actually a batch of arrivals.

Another generalization starts from Definition 2.2.2. The interarrival times are again i.i.d., but not exponentially distributed anymore. Such a process is called a *renewal process*. Although mathematically interesting, there are few applications of this type of process. One exception is the process where the interarrival times are deterministic. This is typically the situation in which arrivals to a system are planned, such as production orders or patients in a clinic. This last example is particularly interesting, because patients often do not arrive exactly on time, but often a few minutes early and sometimes a little late. This adds a random shift to each arrival moment, making the analysis of systems with this type of processes particularly difficult. *Discrete-event simulation* is about the only useful solution technique. It is the subject of the next chapter.

2.7 Further reading

Almost every book on probability or stochastic models introduces the Poisson process. Chapter 1 of Tijms [93] gives an excellent introduction to the many properties of the Poisson process.

Both Hopp & Spearman [48] and Silver & Peterson [83] discuss moving averages and exponential smoothing with trends and seasonal patterns (both in the context of supply chains). Especially Section 13.3 of [48] is very readable; Chapter 4 of [83] goes into more details. Koenker & Hallock [59] introduce quantile regression in a very accessible way.

Andrews & Cunningham [6] is a short paper showing the impact of good forecasting (using ARIMA models) in call centers. An excellent text book on forecasting is Diebold

[32].

Renewal processes are extensively discussed in Ross [75] and Tijms [93].

2.8 Exercises

Exercise 2.1 A department of a bank processes mortgage applications. There are 15 employees, each one is supposed to be able to handle 3 applications per day. On average 42 requests arrive at the beginning of each day. Applications have to be dealt with the day they arrive, when necessary in overtime.
a. Do you think that the Poisson distribution is a good choice for modeling the number of applications per day? Motivate your answer.
b. Give a formula for the probability that overtime is necessary and the expected number of applications that are processed during overtime.
c. Calculate these numbers using some appropriate software tool.

Exercise 2.2 Consider a process N on $[0, \infty)$ with the following properties:
- $N(s,t)$ has a Poisson distribution with expectation $\lambda(t-s)$ for all $0 \leq s < t$;
- given $N(s,t) = k$, the arrivals in $[s,t]$ are distributed according to k i.i.d. uniform distributions on $[s,t]$.
Show that this is an alternative definition of a Poisson process.

Exercise 2.3 Consider counting processes N_h, $h > 0$, where at each point kh, $k \in \mathbb{N}$, an arrival occurs with probability λh. Show that N_h converges, as $h \to 0$, to a Poisson process with rate λ.

Exercise 2.4 Consider a Poisson distribution X with parameter 10. Give its expectation and bounds l and u such that $\mathbb{P}(l \leq X \leq u) \approx 0.9$.
a. Do the same for Poisson distributions with parameters 100, 1000, and 10000.
b. Use the law of large numbers to explain your findings.
c. What are the implications when predicting future arrival count to a service facility?

Exercise 2.5 Let X have a Poisson distribution with parameter λ. Let Y have a normal distribition with $\mathbb{E}Y = \mathbb{E}X$ and $\sigma^2(Y) = \sigma^2(X)$.
a. What is $\sigma^2(Y)$?
b. Make an Excel sheet with the values of $\mathbb{P}(Y \leq n) - \mathbb{P}(X \leq n)$ for all relevant values of $n \in \mathbb{N}_0$.
c. Vary Y and determine values of λ for which Y is a good approximation of X.
d. Design and implement an Excel function POISSONINV() that generates random outcome of the Poisson distribution.

Exercise 2.6 The arrival process to the "First Cardiac Aid" department (FCA) of a hospital is modeled as an inhomogeneous Poisson process with the following rate: from 8.00 to 22.00 it is 1.5 per hour, from 22.00 to 8.00 0.5 per hour.

a. What is the expected number of patients per day (from midnight to midnight)?
b. Patients stay exactly 6 hours. Give for each point in time the expected number of patients.
c. What is the distribution of the number of patients at a certain point in time? Motivate your answer.
d. 50% of the patients stay only 4 hours, the other 50% stay 8 hours. What is the distribution and its parameter(s) of the number of patients at 13.00?

Exercise 2.7 Formulate an LP model that minimizes $\sum_i[(1-\tau)(a+bx_i-y_i)^+ + \tau(y_i - a - bx_i)^+]$.

Chapter 3

Regenerative Processes

In this chapter we discuss stochastic processes, regenerative processes and discrete-event simulation.

3.1 Stochastic processes

A stochastic process is a collection of random variables X_t, $t \geq 0$. A realization or trajectory of the process is a function from $[0, \infty)$ to $\mathbb{R}$ (or $\mathbb{R}^m$). X_t is often called the *state* of the process at time t. It is convenient to introduce the notation $\pi_t(A) = \mathbb{P}(X_t \in A)$ for some set A.

Example 3.1.1 Consider some service system with arrivals and departures. As X_t we can take the number of customers in the system at time t. An alternative would be to take the total workload in the system.

We are interested in calculating performance measures such as $T^{-1}\mathbb{E}\int_0^T f(X_s)ds$ and $\mathbb{E}f(X_T)$. A number of different techniques exist depending on the structure of X_t. Special cases are for example Markov chains, for which special methods exist, and sometimes even closed-form expressions can be derived. A widely used technique that can be used for a very general class of models is *discrete-event simulation.* As can be expected from the name this technique works for models that have discrete events, usually called discrete-event systems.

3.2 Discrete-event simulation

In this section we assume $X_t \subset \mathbb{N}_0^m$. Thus every trajectory is a piece-wise constant function, the process makes discrete jumps. For this reason such a process is called a discrete-event system. Discrete-event simulation is about generating trajectories of discrete-event systems. This is done by starting at time 0 and then constructing a trajectory by sampling one by one events in the system. Usually there is an obvious way to do this.

Example 3.2.1 Consider a service system with a single server, Poisson arrivals and i.i.d. service times. At time 0 the system is empty. In computer memory 4 numbers are stored: the number of customers in the system x, the time of the next arrival a, the time of the next departure b (if $x > 0$), and the current time t. By sampling from the interarrival and service time distributions (using the method of Section 1.9.1) we generate future events. Then the time t is augmented to $\min\{a, b\}$, x is increased of decreased by 1 according to the type of event that attained the minimum, and an interarrival time or service time is sampled (unless a departure leaves an empty queue behind). Now time is increased again, and so forth. This way we simulate a whole trajectory, a realization of the stochastic process.

If the performance measure is of the form $\mathbb{E}f(X_T)$ or $T^{-1}\mathbb{E}\int_0^T f(X_s)ds$ then it suffices to simulate the stochastic process up to T. In this case the ideas from Section 1.9 can be used to derive a confidence interval for the measure we try to estimate. Evidently, we need multiple runs from 0 to T to perform such an analysis. Many simulation packages exist for executing this type of discrete-event simulations.

When we are interested in calculating $T^{-1}\mathbb{E}\int_0^T f(X_s)ds$ for T big then we often see that the trajectories all give approximately the same value: the bigger T, the smaller the variation of the outcome. To explain this phenomenon we need renewal theory, the subject of the next section.

3.3 Renewal theory

Consider some stochastic process X_t, $t \geq 0$. We are interested in calculating

$$\lim_{t\to\infty} \frac{1}{t}\int_0^t f(X_s)ds, \tag{3.1}$$

some long-run average performance measure of this process. Under what conditions does this number exist? Is it a number or a random variable? How to calculate it efficiently? These are the questions that we answer in this section.

In full generality it is impossible to answer the questions we posed above. For this reason we assume the following framework in which the bigger part of the models that we are interested in fit. Assume that there are random variables T_i, with $0 = T_0 \leq T_1 \leq T_2 \leq \cdots$, such that $\{X_t, T_i \leq t \leq T_{i+1}\}$ i.i.d. for all $i \geq 0$. Then the T_i are called renewal points, X_t is a regenerative process and the following theorem holds.

Theorem 3.3.1 *If* $0 < \mathbb{E}(T_1 - T_0) < \infty$ *and* $\mathbb{E}\int_{T_0}^{T_1} |f(X_s)|ds < \infty$ *then*

$$\lim_{t\to\infty} \frac{1}{t}\int_0^t f(X_s)ds = \frac{\mathbb{E}\int_{T_0}^{T_1} f(X_s)ds}{\mathbb{E}(T_1 - T_0)}.$$

Proof Take $f = f^+ - f^-$, $f^+, f^- \geq 0$, and consider $N_t = \max_n\{T_n \leq t\}$. Now

$$\frac{1}{t}\int_0^t f^+(X_s)ds \geq \frac{1}{t}\int_0^{T_{N_t}} f^+(X_s)ds = \frac{N_t}{t}\frac{\sum_{k=0}^{N_t-1}\int_{T_k}^{T_{k+1}} f^+(X_s)ds}{N_t}. \tag{3.2}$$

If $t \to \infty$, then so does N_t. Then, using the Law of Large Numbers, it follows that

$$\sum_{k=0}^{N_t-1}\int_{T_k}^{T_{k+1}} f^+(X_s)ds/N_t \to \mathbb{E}\int_{T_0}^{T_1} f^+(X_s)ds.$$

From the same law it also follows that

$$\frac{T_{N_t}}{N_t} = \frac{\sum_{k=0}^{N_t-1}(T_{k+1}-T_k)}{N_t} \to \mathbb{E}(T_1 - T_0).$$

Similarly we can show that $T_{N_t+1}/N_t \to \mathbb{E}(T_1 - T_0)$. Because

$$\frac{N_t}{T_{N_t+1}} \leq \frac{N_t}{t} \leq \frac{N_t}{T_{N_t}}$$

we find $N_t/t \to 1/\mathbb{E}(T_1 - T_0)$. Taking the limit for $t \to \infty$ in (3.2) leads to

$$\lim_{t\to\infty}\frac{1}{t}\int_0^t f^+(X_s)ds \geq \frac{\mathbb{E}\int_{T_0}^{T_1} f^+(X_s)ds}{\mathbb{E}(T_1 - T_0)}.$$

To obtain the other inequality we use

$$\frac{1}{t}\int_0^{T_{N_t+1}} f^+(X_s)ds \geq \frac{1}{t}\int_0^t f^+(X_s)ds.$$

This gives the result for f^+. The same arguments can be applied to f^-. □

It is important to note that

$$\frac{\mathbb{E}\int_{T_0}^{T_1} f(X_s)ds}{\mathbb{E}(T_1 - T_0)} \neq \mathbb{E}\frac{\int_{T_0}^{T_1} f(X_s)ds}{(T_1 - T_0)}.$$

Example 3.3.2 Consider a process that alternates between the states 0 and 1. The times it stays in 0 are i.i.d., the first length is A. The times the process stays in 1 are also i.i.d., with r.v. S. This is a suitable model for the repair process of a component (see Section 11.7), and when A is exponentially distributed then it is equivalent to the $M|G|1|1$ queue (see Section 5.1 for this notation). Assume that $0 < \mathbb{E}A + \mathbb{E}S < \infty$. The long-run fraction of time that the system is in state 1 is given by $\mathbb{E}S/(\mathbb{E}A + \mathbb{E}S)$. This follows from Theorem 3.3.1, with $f(x) = \mathbb{I}\{x = 1\}$.

It should also be observed that $\lim_{t\to\infty}\frac{1}{t}\int_0^t f(X_s)ds$ is a number, not a random variable. This explains the convergence of long replications to the same outcome.

3.4 Simulating long-run averages

Now suppose we want to calculate (3.1) for a regenerative process for which we have no closed-form expression for $\mathbb{E}\int_{T_0}^{T_1} f(X_s)ds$ and/or $\mathbb{E}(T_1-T_0)$. Renewal theory suggests simulating repeatedly busy periods and estimating the performance from that. Unfortunately, most simulation software tools are not well fit to do this. On the other hand, running a system indefinitely is also impossible: it would require an infinite running time. The solution is as follows. Instead of simulating X_t from 0 to ∞ and measuring $\mathbb{E}\int_0^\infty f(X_s)ds$ we simulate X_t from 0 to t_1 and we measure $\mathbb{E}\int_{t_0}^{t_1} f(X_s)ds$ for well-chosen constants t_0 and t_1, $0 < t_0 < t_1$. The choice of t_0 is of particular importance. Due to the start-up of the simulation the stochastic process does not show long-run behavior for small t. These *transient* effects disappear in the long-run average. However, when we simulate up to a constant t_1, this transient effect plays a role. By choosing t_0 large enough this should be avoided as much as possible, without taking t_0 too large to avoid spending too much of our computing time on simulating the warming-up period. The samples of $\mathbb{E}\int_{t_0}^{t_1} f(X_s)ds$ can be analyzed in the usual way.

Example 3.4.1 We simulated the system of Example 3.2.1 with arrival rate 1 and exponential service times with mean 0.8. The system starts initially empty. When simulating 2000 runs from 0 to 100 the average number of customers in the system was for our particular run 3.33. Simulating 1000 runs from 0 to 200 gives 3.64. Simulating 1000 runs from 50 to 250 finally gives 3.89. Thus we see an increase in average value as we increase the simulation length and as we introduce a warm-up period. Queueing theory (Theorem 5.3.1) tells us that the long-run average number of customers is exactly 4.

3.5 The long-run average and limiting distributions

Taking $f(X_t) = \mathbb{I}\{X_t \in A\}$ for some set A is an interesting special case of Theorem 3.3.1, because

$$\bar{\pi}_\infty(A) = \lim_{t\to\infty} \frac{1}{t}\int_0^t \mathbb{I}\{X_s \in A\}ds$$

can be interpreted as the long-run fraction of time that the system is in the set A. Evidently

$$0 \le \mathbb{E}\int_{T_0}^{T_1} \mathbb{I}\{X_s \in A\}ds \le \mathbb{E}\int_{T_0}^{T_1} ds = \mathbb{E}(T_1 - T_0),$$

thus the long-run average distribution exists and is unique if $0 < \mathbb{E}(T_1 - T_0) < \infty$.

Sometimes we are not interested in the long-run average distribution, but in the limiting distribution

$$\pi_\infty(A) = \lim_{t\to\infty} \pi_t(A) = \lim_{t\to\infty} \mathbb{P}(X_t \in A) = \lim_{t\to\infty} \mathbb{E}\mathbb{I}\{X_t \in A\}.$$

Unfortunately, this distribution does not always exist. For example, take a process that alternates between state 0 and 1 and stays in each state exactly 1 time unit. Then

$\lim_{t\to\infty}\frac{1}{t}\int_0^t \mathbb{I}\{X_s = 1\}ds = 0.5$ (see Example 3.3.2), but the limiting probability $\pi_\infty(1)$ does not exist, because $\pi_t(1)$ alternates between 0 and 1 and does not converge.

We need a condition on the cycle length distribution $T_1 - T_0$ to make π_t converge. We assume that $T_1 - T_0$ is *nonlattice*, which means that $T_1 - T_0$ is not concentrated on a set of the form $\{\delta, 2\delta, \ldots\}$.

Theorem 3.5.1 *Consider a regenerative process X_t with $0 < \mathbb{E}(T_1 - T_0) < \infty$ and $T_1 - T_0$ nonlattice. Then $\pi_\infty(A)$ exists for all A and is given by*

$$\pi_\infty(A) = \lim_{t\to\infty}\mathbb{P}(X_t \in A) = \frac{\mathbb{E}\int_{T_0}^{T_1}\mathbb{I}\{X_s \in A\}ds}{\mathbb{E}(T_1 - T_0)}.$$

Thus under the nonlattice condition the time-average distribution and the limiting distribution are equal. For a proof of Theorem 3.5.1 we refer to Asmussen [10, Theorem VI.1.2].

3.6 Poisson arrivals see time averages

In Part III we often consider systems in which customers arrive according to a Poisson process. For this reason it is important to ask ourselves the question: how do arriving customers perceive the system? Consider a regenerative process X_t and a Poisson process $N(t)$. X_t models some process for which $N(t)$ is the arrival process. For this reason, $X(s)$, $s \le t$, is a function of the points of $N(s)$ with $s \le t$. The points of N after t do not influence X before t. Thanks to the second property of Definition 2.2.1 $N(t)$ is independent of $N(t, \infty)$. Combining all gives that $\{X_s, s \le t\}$ is independent of $N(t, \infty)$.

Let Y_t be equal to X_{t^-}, conditioned on the fact that an arrival occurs at t (with t^- we mean the moment just before the arrival at t):

$$\mathbb{P}(Y_t \in A) = \lim_{h\to 0}\mathbb{P}(X_{t-h} \in A | N(t-h, t) = 1).$$

Define $\bar{\pi}_t$ by

$$\bar{\pi}_t(A) = \frac{1}{t}\int_0^t \mathbb{P}(X_s \in A)ds.$$

Note that for $\bar{\pi}_\infty$, as defined in the previous section, holds:

$$\bar{\pi}_\infty(A) = \lim_{t\to\infty}\bar{\pi}_t(A),$$

We also define $\bar{\alpha}_t$ and $\bar{\alpha}_\infty$ by

$$\bar{\alpha}_t(A) = \frac{1}{t}\int_0^t \mathbb{P}(Y_s \in A)ds \text{ and } \bar{\alpha}_\infty(A) = \lim_{t\to\infty}\bar{\alpha}_t(A).$$

The limiting average distribution $\bar{\pi}_\infty$ exists under the conditions of Theorem 3.3.1. The values of $\bar{\alpha}_t$ and $\bar{\alpha}_\infty$ follow from the following theorem.

Theorem 3.6.1 *In the case of Poisson arrivals $\bar{\alpha}_t = \bar{\pi}_t$ and $\bar{\alpha}_\infty = \bar{\pi}_\infty$, thus Poisson arrivals see time averages (generally known as the* PASTA *property).*

Proof The random variables $N(t-h)$ and $N(t-h,t)$ are independent, and therefore so are X_{t-h} and $N(t-h,t)$. Thus $\mathbb{P}(Y_t \in A) = \lim_{h\to 0} \mathbb{P}(X_{t-h} \in A | N(t-h,t) = 1) = \lim_{h\to 0} \mathbb{P}(X_{t-h} \in A) = \mathbb{P}(X_t \in A)$. Integrating, averaging, and taking the limit as $t \to \infty$ gives the required results. □

This result is very convenient when analyzing all kinds of systems with Poisson arrivals. When analyzing certain properties of customers (such as the waiting time upon arrival) we can base ourselves on the long-run average distribution: PASTA tells us that the time-averages are equal to the customer-averages. Note that (under the nonlattice condition) Theorem 3.5.1 tells us that the time-average distribution is equal to the limiting distribution. Thus an arbitrary arrival also sees the limiting distribution.

Example 3.6.2 Consider a service facility with a single server, Poisson arrivals, and customers leave when the server is occupied. The occupancy of the server can be analyzed using the model of Example 3.3.2, with $A \sim \exp(\lambda)$. We have

$$\mathbb{P}(\text{arbitrary arriving customer blocked}) = \bar{\alpha}_\infty(1) = \bar{\pi}_\infty(1) = \frac{\mathbb{E}S}{\lambda^{-1} + \mathbb{E}S}.$$

To show that Poisson arrivals are crucial to obtain $\bar{\alpha}_\infty = \bar{\pi}_\infty$, we consider the same model but with fixed interarrival times b and service times s, with $b > s > 0$. Then

$$\mathbb{P}(\text{arbitrary arriving customer blocked}) = 0 = \bar{\alpha}_\infty(1) \neq \bar{\pi}_\infty(1) = \frac{s}{b}.$$

3.7 The waiting-time paradox

Consider a regenerative process X_t. We are interested in the expected time until the next renewal point for an arbitrary outside observer. For a regenerative process on $[0,t]$, we define an outside observer as someone who observes the state of the system at an arbitrary instant ξ in $[0,t]$, that is, $\xi \sim \text{Uniform}[0,t]$. But arrivals in a Poisson process are, if they are not ordered, distributed as independent uniform distributions, according to Exercise 2.2. Thus the outside observer behaves as a Poisson arrival, and thus Theorem 3.6.1 applies also to the outside observer, who sees as a result time-average behavior.

In fact, an outside observer is a special case of the situation of Section 3.6: X_t and $N(s)$ are independent for all t and s. Thus, the expected time until the next renewal point for our outside observer is equal to the long-run average time until the next renewal.

Naively one could think that this number is equal to $\mathbb{E}S/2$ with $S \sim T_1 - T_0$, but this is only true for S deterministic. Let T_i be the time of the ith renewal. Define the process X_t as the time until the next renewal. Then we have $X_t = T_i - t$ for $T_{i-1} < t \le T_i$.

Now X_t is indeed a regenerative process, with T_i the ith moment that X_t gets 0. A typical realization of X_t is given in Figure 3.1. The upward jumps are distributed according to S.

Using Theorem 3.3.1 we find that

$$\lim_{t\to\infty} \frac{1}{t}\int_0^t X_s ds = \frac{\mathbb{E}\int_{T_0}^{T_1} X_s ds}{\mathbb{E}(T_1 - T_0)} = \frac{\mathbb{E}\int_0^S s\,ds}{\mathbb{E}S} = \frac{\int \int_0^t s\,ds\,dF_S(t)}{\mathbb{E}S} = \frac{\mathbb{E}S^2}{2\mathbb{E}S}.$$

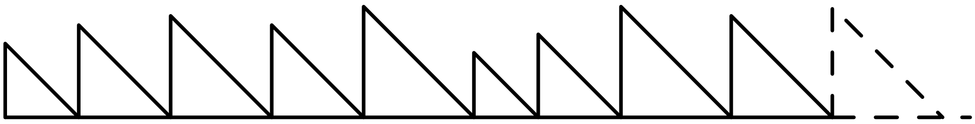

Figure 3.1: A typical realization of X_t.

The fact that $\mathbb{E}\int_{T_0}^{T_1} X_s ds = \mathbb{E}S^2/2$ can also be seen directly: it is the surface of a right-angled triangle with legs S.

Indeed, we see that the waiting time $\mathbb{E}S^2/(2\mathbb{E}S) = \mathbb{E}S/2$ if and only if S is deterministic. For example, if S is exponential, then $\mathbb{E}S^2/(2\mathbb{E}S) = \mathbb{E}S$.

This result is often used to explain the fact that when we go at an arbitrary moment to the bus stop we experience an expected waiting time longer than half the expected interarrival time. For this reason this result is known under the name *waiting-time paradox*. Note that it is assumed that the interarrival times are independent, which is a somewhat unrealistic assumption.

Note also that for the current model

$$\mathbb{E}\frac{\int_{T_0}^{T_1} f(X_s)ds}{(T_1 - T_0)} = \mathbb{E}S.$$

This nicely illustrates what happens if we calculate the expectation of the quotient instead of the quotient of the expectations.

This waiting-time paradox is also known under the name *inspection paradox*.

3.8 Cost equations and Little's Law

The previous sections dealt with the limiting state distribution perceived by an arriving customer and an outside observer such as the manager of the system. There is another set of interesting relations linking the views of customers and system managers, called *cost equations*.

The idea is as follows. Consider a process X_t and customers that arrive during a period $[0, T]$ according to some process $N(t)$, and leave according to another process $M(t)$. Note that $N(t)$ is not necessarily a Poisson process, and that the system is empty at t if $N(t) = M(t)$. Let A_k be the arrival time of the kth arriving customer, and D_k its departure time. Of course $0 \leq A_1 \leq A_2 \leq \cdots$, but the D_k need not be ordered. Evidently $A_k \leq D_k$. Each customer incurs costs while in the system, $C_k(t)$ for customer k at t. We assume $C_k(t) = 0$ if $t \leq A_k$ or $t \geq D_k$. Now there are two ways to calculate the expected costs: we can look at the expected amount the system receives, or at the expected number of arrivals times the average cost per customer. These must be equal. To formalize this, define

$$\lambda = \lim_{T\to\infty} N(T)/T, \; H(t) = \sum_{k=1}^{\infty} C_k(t), \text{ and } G_k = \int_0^{\infty} C_k(t)dt.$$

In the next theorem we will assume that arrivals and departures are functions of X_t, that is, from the evolution of X_t the processes $N(t)$ and $M(t)$ can be constructed. Then the renewals of X_t are also renewals of $N(t)$ and $M(t)$.

Theorem 3.8.1 *For every realization of X_t for which $N(T) = M(T)$ we have*

$$\int_0^T H(t)dt = \sum_{k=1}^{N(T)} G_k;$$

if X_t is a regenerative process with $0 < \mathbb{E}(T_1 - T_0) < \infty$ and $N(T_1) = M(T_1)$, then

$$\lim_{T\to\infty} \frac{\int_0^T H(t)dt}{T} = \lambda \lim_{n\to\infty} \frac{\sum_{k=1}^n G_k}{n}.$$

Proof The first statement follows from the fact that the system is empty at T, and thus $C_k(t) = 0$ for $t > T$ and $k \leq N(T)$:

$$\int_0^T \sum_{k=1}^{\infty} C_k(t)dt = \int_0^T \sum_{k=1}^{N(T)} C_k(t)dt = \sum_{k=1}^{N(T)} \int_0^T C_k(t)dt = \sum_{k=1}^{N(T)} \int_0^{\infty} C_k(t)dt.$$

The proof of the second statement is similar to that of Theorem 3.3.1. Consider $C_k^+(t) = \max\{C_k(t), 0\}$, and define $H^+(t)$ and G_k^+ accordingly. Then it is readily seen that for all T

$$\sum_{k=1}^{M(T)} G_k^+ \leq \int_0^T H^+(t)dt \leq \sum_{k=1}^{N(T)} G_k^+.$$

This is equal to

$$\frac{M(T)}{T} \frac{1}{M(T)} \sum_{k=1}^{M(T)} G_k^+ \leq \frac{1}{T} \int_0^T H^+(t)dt \leq \frac{N(T)}{T} \frac{1}{N(T)} \sum_{k=1}^{N(T)} G_k^+.$$

From the fact that X_t is regenerative it follows that $\lim_{T\to\infty} M(T)/T = \lim_{T\to\infty} N(T)/T = \lambda$ and that $N(T)$ and $M(T) \to \infty$ as $T \to \infty$. Hence

$$\lim_{T\to\infty} \frac{\int_0^T H^+(t)dt}{T} = \lim_{T\to\infty} \frac{N(T)}{T} \frac{\sum_{k=1}^{N(T)} G_k^+}{N(T)} = \lambda \lim_{n\to\infty} \frac{\sum_{k=1}^n G_k^+}{n},$$

for positive costs. A similar argument applies to $C_k^-(t) = \max\{-C_k(t), 0\}$. Summing gives the required result. □

The best known cost equation is Little's Law. Consider a customer who pays one unit for every time unit it stays in a system, thus $C_k(t) = 1$ for $A_k \leq t \leq D_k$. Then we have, with w the average time a customer spends in a system and l the long-run average number of customers in the system:

$$l = \lambda w. \tag{3.3}$$

Another useful inequality is that of a single-server system where every customer pays 1 unit when in service. Then, with S the service-time distribution:

$$\text{long-run average fraction of time server busy} = \lambda \mathbb{E} S. \tag{3.4}$$

3.9 Further reading

Çinlar [22] is the standard reference for stochastic processes. Renewal theory is extensively discussed in Ross [75] and Tijms [93]. A more technical reference is Asmussen [10].

There are many excellent books dedicated to simulation. We name Ross [74], Rubinstein [77], Kleijnen [55], Law & Kelton [64], and Kelton et al. [53]. The latter is at the same time an introduction to the simulation package Arena.

El-Taha & Stidham [33] is completely dedicated to cost equations; the approach in Section 3.8 is inspired by [33, Section 6.4], especially Theorem 6.8.

3.10 Exercises

Exercise 3.1 Consider Example 3.6.2 with Poisson arrivals and general i.i.d. service times.
a. Give renewal points of this process.
Now assume that the interarrival times are not exponentially distributed anymore, but i.i.d.
b. Give again renewal points.

Exercise 3.2 A person takes a bus each morning to go to work. If she catches the bus within t minutes after arriving at the bus stop she gets to work on time, otherwise she is late. Busses arrive according to a renewal process (see Section 2.6) with interarrival distribution S.
a. Give an expression for the probability that she arrives on time.
b. Calculate this for $t = 10$ and S exponentially distributed with average 8 minutes.

Exercise 3.3 Consider a model with Poisson arrivals, s servers, constant service times, and arrivals that find all servers busy are lost (the $M|G|s|s$ system).
a. Simulate this model using Arena for some well-chosen parameters and give an estimate for the blocking probability.
b. Motivate your choice of the number of replications and the lengths of the simulation and the warming-up period.
c. Give a confidence interval for the blocking probability.

Exercise 3.4 An emergency department in a hospital has capacity for 2 trauma patients. When the capacity is reached new patients are brought to another hospital. Arrivals occur according to a Poisson process, on average 4.2 per day are accepted. 16% of the time both beds are occupied.
a. What was the demand? Which percentage is sent elsewhere?
In reality there is a daily pattern in the arrivals and sometimes arrivals occur in small groups (trauma patients are often the result of traffic accidents).
b. What do you think that the influence of these facts will be on the outcomes?

Chapter 4

Markov Chains

Markov chains are the most often used class of stochastic processes. They are also fundamental to the study of queueing models.

There are several types of Markov chains between which we have to distinguish. A first distinction is between continuous and discrete time. We concentrate on continuous-time Markov chains, because most of the applications we consider evolve in continuous time. However, discrete-time Markov chains are conceptually simpler, therefore we pay attention to them first.

4.1 Discrete-time Markov chains

A discrete-time Markov chain is a special type of stochastic process X_t. This process takes values in some finite or countable state space $\mathcal{X}$. Often we take $\mathcal{X} = \{0, \dots, n\}$ or $\{0, 1, \dots\}$. A discrete-time Markov chain changes state only at the time instants $\{1, 2, \dots\}$, according to the following rule: when at $t \in \mathbb{N}$ the state $X_t = x$, then $X_{t+1} = y$ with probability $p(x, y)$, independent of the states visited before t. Thus $\mathbb{P}(X_{t+1} = y | X_t = x) = p(x, y)$. These probabilities are called the *one-step transition probabilities.*

This system is easy to simulate. We simply keep in memory the current state x, and generate the next state according to the distribution $p(x, \cdot)$. This way we can generate trajectories of the process X_t. But we can do better if $|\mathcal{X}| < \infty$: in this case we can calculate the distribution of X_t for each $t \in \mathbb{N}$. As in Section 3.5, we write $\pi_t(x) = \mathbb{P}(X_t = x)$. Then

$$\pi_{t+1}(y) = \sum_{x \in \mathcal{X}} \pi_t(x) p(x, y),$$

or, in matrix notation, $\pi'_{t+1} = \pi'_t P$ (with u' the transpose of u and the matrix P with entries $p(x, y)$). Recursively, this amounts to $\pi'_t = \pi'_0 P^t$. Thus computing π_t only requires a number of matrix multiplications. From these distributions we can compute performance measures such as $\mathbb{E} f(X_T)$ and $T^{-1} \mathbb{E} \sum_{t=1}^{T} f(X_t)$, the discrete-time equivalent of (3.1).

Next we study the behavior for the case $T \to \infty$. Define, in parallel with the definitions in Section 3.5,

$$\pi_\infty(x) = \lim_{t\to\infty} \pi_t(x)$$

and

$$\bar\pi_\infty(x) = \lim_{t\to\infty} \bar\pi_t(x) = \lim_{t\to\infty} \frac{1}{t}\mathbb{E}\sum_{s=1}^{t} \mathbb{I}(X_s = x).$$

We will apply Theorem 3.3.1. The renewal moments in our Markov chain are the instants at which we reach a certain state x^*. A necessary condition for $\mathbb{E}(T_1 - T_0) < \infty$ is that from x^* we can always go back to x^* in a finite number of transitions. An even stronger assumption, but which is usually satisfied in practical applications, is the following.

Assumption 4.1.1 *Every state can be reached from every other state, that is, for arbitrary $x, y \in \mathcal{X}$ there is a path with positive probability, which means that there are $x_1, \dots, x_k$ with $x = x_0$ and $y = x_k$ such that $p(x_0, x_1)\cdots p(x_{k-1}, x_k) > 0$.*

Thus every state can now serve as renewal point. Let T_x be the time, starting in x, to return to x. It follows immediately from Theorem 3.3.1 that $\bar\pi_\infty(x) = [\mathbb{E}T_x]^{-1}$ if $\mathbb{E}T_x < \infty$. It can be shown that if Assumption 4.1.1 holds and $|\mathcal{X}| < \infty$ then $\mathbb{E}T_x < \infty$. If $|\mathcal{X}| = \infty$ then it can also occur that $\mathbb{E}T_x = \infty$. Note however that also in this situation $\bar\pi_\infty(x) = [\mathbb{E}T_x]^{-1}$ holds.

For the computation of the actual value of $\bar\pi_\infty$ we cannot use renewal theory. We derive directly an equation for the value of $\bar\pi_\infty$. The expected number of times up to t that the chain moves from x to y is given by $t\bar\pi_t(x)p(x,y)$. Consider some set $\mathcal{Y} \subset \mathcal{X}$. The difference between the number of times that the chain goes out of $\mathcal{Y}$ or into $\mathcal{Y}$ is at most 1. Equating the flow in and out of $\mathcal{Y}$, dividing by t and taking $t \to \infty$ leads to the following theorem.

Theorem 4.1.2 *The numbers $\bar\pi_\infty(x)$ must satisfy*

$$\sum_{x\in\mathcal{Y}} \bar\pi_\infty(x) \sum_{y\in\mathcal{Y}^c} p(x,y) = \sum_{x\in\mathcal{Y}^c} \bar\pi_\infty(x) \sum_{y\in\mathcal{Y}} p(x,y) \tag{4.1}$$

for all $\mathcal{Y} \subset \mathcal{X}$ and

$$\sum_{x\in\mathcal{X}} \bar\pi_\infty(x) = 1. \tag{4.2}$$

If we take $|\mathcal{Y}| = 1$, then we get a system of equations which is known as the *equilibrium equations* (take $\mathcal{Y} = x$ and add $\bar\pi_\infty(x)p(x,x)$ to both sides):

$$\bar\pi_\infty(x) = \sum_{y\in\mathcal{X}} \bar\pi_\infty(y)p(y,x). \tag{4.3}$$

In certain situations however other choices of $\mathcal{Y}$ can be useful. Note also that if we find a solution of (4.3), then it is a solution for all possible $\mathcal{Y}$, by summing (4.3) over all $x \in \mathcal{Y}$.

Markov chain theory tells us that, under Assumption 4.1.1, Equations (4.1)-(4.2) have a unique solution if and only if $\mathbb{E}T_x < \infty$ for all x. In fact, either $\mathbb{E}T_x < \infty$ for all x or $\mathbb{E}T_x = \infty$ for all x. The latter case can only occur if $|\mathcal{X}| = \infty$.

To obtain the existence of π_∞ we need an extra condition, the equivalent of the non-lattice condition that was formulated in Section 3.5. This condition is simply that the greatest common divisor of all paths from x^* to x^* should be 1.

In Markov chains, the distribution $\bar{\pi}_\infty$ has another interesting interpretation. Assume that $\pi_0 = \bar{\pi}_\infty$. Then, according to Equation (4.3), $\pi_1 = \bar{\pi}_\infty$, and recursively, $\pi_t = \bar{\pi}_\infty$ for all t. For this reason we call $\bar{\pi}_\infty$ also the *stationary distribution*.

4.2 Continuous-time Markov chains

Let us define continuous-time Markov chains. They are again defined on some finite or countable set $\mathcal{X}$, the state space. The time that this process stays in a state is exponentially distributed, with parameter $\Lambda(x)$ in state x. When this time expires a transition is made to a new state y with probability $p(x, y)$. Once in y this process starts again.

Define $\lambda(x, y) = p(x, y)\Lambda(x)$. We can see $\lambda(x, y)$ as the rate at which the system moves from x to y. The state changes according to the first transition to occur. Then the time until the first transition is indeed exponentially distributed with parameter $\Lambda(x) = \sum_{y\in\mathcal{X}} \lambda(x, y)$, because the minimum of a number of exponential random variables is again exponential, and the probability of a transition to y is equal to $p(x, y) = \lambda(x, y)/\Lambda(x)$ (see Section 1.6.5).

A continuous-time Markov chain is easy to simulate: in x, one samples from an exponentially distributed random variable to determine the next transition epoch, and the next state is determined according to the distributed $p(x, \cdot)$. An alternative method is sampling from an exponential distribution for each of the possible transitions, using the rates $\lambda(x, y)$. The latter choice is sometimes more intuitive.

Example 4.2.1 Consider a queueing system with a single server, Poisson arrivals with rate λ and exponential service time distributions, rate μ. Then $\Lambda(x) = \lambda + \mu$ if $x > 0$, $\Lambda(0) = \lambda$. Also $p(0, 1) = 1$, and for $x > 0$ $p(x, x + 1) = \lambda/(\lambda + \mu)$ and $p(x, x - 1) = \mu/(\lambda + \mu)$. Thus we can sample first the time in each state and then the transition. It is more intuitive, with the actual system in mind, to sample the interarrival and service time distributions, using $\lambda(x, x + 1) = \lambda$ and $\lambda(x, x - 1) = \mu$ (the latter for $x > 0$).

The *distribution* of X_t is harder to determine than in the deterministic case; we will discuss this in Section 4.6, in the context of time-inhomogeneous chains. Here we are interested in the behavior of the Markov chain as the time goes to ∞. To be able to apply renewal theory we make the following assumption.

Assumption 4.2.2 *Every state can be reached from every other state, that is, for arbitrary $x, y \in \mathcal{X}$ there are $x_1, \dots, x_k$ with $x = x_0$ and $y = x_k$ such that $\lambda(x_0, x_1) \cdots \lambda(x_{k-1}, x_k) > 0$.*

Define $\pi_t(x)$, $\pi_\infty(x)$, $\bar\pi_t(x)$, and $\bar\pi_\infty(x)$ as in Section 3.5. The nonlattice condition is satisfied because exponential distributions are nonlattice. Therefore Theorem 3.5.1 can be applied and we find $\pi_\infty(x) = \bar\pi_\infty(x)$.

Let us for the moment delay questions about the existence of this limit. The expected number of times up to t that the chain moves from x to y is given by $\bar\pi_t(x)\lambda(x,y)$. Then, equivalent to Theorem 4.1.2, we get:

Theorem 4.2.3 *The numbers $\pi_\infty(x)$ must satisfy*

$$\sum_{x\in\mathcal{Y}} \pi_\infty(x) \sum_{y\in\mathcal{Y}^c} \lambda(x,y) = \sum_{x\in\mathcal{Y}^c} \pi_\infty(x) \sum_{y\in\mathcal{Y}} \lambda(x,y) \tag{4.4}$$

for all $\mathcal{Y} \subset \mathcal{X}$ and

$$\sum_{x\in\mathcal{X}} \pi_\infty(x) = 1. \tag{4.5}$$

Under Assumption 4.2.2 the following holds.

Theorem 4.2.4 *If $|\mathcal{X}| < \infty$, then a unique solution of (4.4)-(4.5) exists; if $|\mathcal{X}| = \infty$, then either a unique solution of (4.4)-(4.5) exists, or $\pi(x) = 0$ for all $x \in \mathcal{X}$ is the unique solution to (4.4) with $|\sum_{x\in\mathcal{X}} \pi(x)| < \infty$.*

Consider X_{t+h} for h small. Then we find, using the interpretation of $\lambda(x,y)$ as the hazard rate of going from x to y (using Equation (1.11)):

$$\pi_{t+h}(x) = \sum_{y\in\mathcal{X}} \pi_t(y)[\lambda(y,x)h + o(h)] + \pi_t(x)[1 - \Lambda(x)h + o(h)]. \tag{4.6}$$

Now assume that $\pi_t = \pi_\infty$ with π_∞ a solution of Equation (4.4). Plugging this in gives $\pi_{t+h}(x) = \pi_t(x) + o(h)$, and therefore $\frac{d}{dt}\pi_t(x) = 0$. Thus if $X_t \sim \pi_\infty$ for some t, then $X_s \sim \pi_\infty$ for all $s > t$. Thus π_∞ is also the *stationary distribution*, equivalent to what we found for the discrete-time case.

4.3 Birth-death processes

A special class of continuous-time Markov chains are those where $\mathcal{X} = \{0,\dots,n\}$ or $\{0,1,\dots\}$, and the only non-zero transition rates are $\lambda(x,x+1)$ for all $x < n$ (with n possibly ∞) and $\lambda(x,x-1)$ for all $x > 0$. In fact, we assume that $\lambda(x,x-1) > 0$ for all $x \in \mathcal{X}/\{0\}$, thus state 0 can be reached from any state. We also assume that $\lambda(x,x+1) > 0$ for all $x \in \mathcal{X}/\{n\}$ (or $x \in \mathcal{X}$ if $\mathcal{X}$ is countable), thus every two states are *communicating*, i.e., Assumption 4.2.2 holds.

Such a chain where in one step only neighboring states can be reached is called a *birth-death process*. Birth-death processes are easy to solve. To do so, define $\lambda_x = \lambda(x,x+1)$

and $\mu_x = \lambda(x, x-1)$. If we take, for $x > 0$, $Y = \{0, \dots, x-1\}$ in (4.4), then we find $\pi_\infty(x-1)\lambda_{x-1} = \pi_\infty(x)\mu_x$, and thus

$$\pi_\infty(x) = \frac{\lambda_{x-1}}{\mu_x}\pi_\infty(x-1) = \frac{\lambda_0 \cdots \lambda_{x-1}}{\mu_1 \cdots \mu_x}\pi_\infty(0). \tag{4.7}$$

All that remains is determining $\pi_\infty(0)$. This can be done using (4.5). Indeed,

$$\pi_\infty(0) = \left[1 + \sum_{x=1}^{n} \frac{\lambda_0 \cdots \lambda_{x-1}}{\mu_1 \cdots \mu_x}\right]^{-1}, \tag{4.8}$$

with $n = \infty$ if $|\mathcal{X}| = \infty$. Note that the convergence of the sum decides whether or not a stationary distribution exists. In many cases (often queueing models) there are explicit solutions for $\pi_\infty(x)$. We give one example.

Example 4.3.1 The system with $\lambda_x = \lambda$ and $\mu_x = \mu$, $|\mathcal{X}| = \infty$, is called the $M|M|1$ queue (for further details, see Chapter 5). Define $\rho = \lambda/\mu$. Then $\pi_\infty(x) = \rho^x \pi_\infty(0)$, by (4.7). The probability $\pi_\infty(0)$ is given by $\pi_\infty(0) = [\sum_{x=0}^{\infty} \rho^x]^{-1}$. The sum converges if and only if $\rho < 1$. Under this condition a stationary distribution exists with $\pi_\infty(x) = (1-\rho)\rho^x$; if $\rho \geq 1$ then the Markov chain has no stationary distribution. In queueing theory we say that the queue is *unstable* in this situation. Indeed, if we interpret λ as the arrival rate of customers, and μ as the service rate of the server, then $\lambda < \mu$ signifies that there are less arrivals than the server can handle, and thus the queue will always empty after some finite time. If $\lambda \geq \mu$ this is not the case. (In fact, if $\lambda = \mu$ then it takes on average an infinite amount of time to reach state 0; if $\lambda > \mu$ state 0 might not be reached again at all.) See also Exercise 4.2 for the discrete-time equivalent.

4.4 The Markov property

In Section 4.2 we defined a continuous-time Markov chain or Markov process by its state space $\mathcal{X}$ and its transition rates $\lambda(x, y)$. However, starting from a practical problem, it is not always clear how to choose $\mathcal{X}$ and λ. A characterization of Markov processes that helps us understand better what a Markov process actually is, is the so-called *Markov property*. We say that the process X_t satisfies the Markov property if for all $t_1 < \cdots < t_n$ the following holds:

$$\mathbb{P}(X_{t_n} = x_n | X_{t_1} = x_1, \dots, X_{t_{n-1}} = x_{n-1}) = \mathbb{P}(X_{t_n} = x_n | X_{t_{n-1}} = x_{n-1}).$$

In words: the evolution of a Markov process from some point in time t_{n-1} on does not depend on the history but only on the current state $X_{t_{n-1}}$. It can be seen as a memoryless property.

It is easily seen that the Markov property holds for a given Markov process, due to the transition machanism and the memoryless property of the exponential distribution (see Section 1.6.5). This helps us modeling a system as a Markov process: we should choose states and transition rates such that the future behavior depends only on the current state.

Example 4.4.1 Consider a service center with a single server and a queue for which we are interested in the use of the server. We could take $\mathcal{X} = \{0, 1\}$, corresponding to the states of the server, but this information is not enough to describe future behavior: we should also know the length of the queue. Thus $\mathcal{X} = \mathbb{N}_0$ is a good choice, representing the number of customers in the system.

4.5 Beyond PASTA

In Section 3.6 we showed that 'Poisson arrivals see time averages' (PASTA) in general renewal processes. This concept evidently holds also for the special case of continuous-time Markov chains. In the case of Markov chains we can go a step further: we can derive the distribution perceived by arrivals (or other events) that do not occur according to a Poisson process. For this, define (as in Section 3.6) the distribution $\alpha_t(x)$ of Y_t, the state of the Markov chain, conditioned on the fact that an arrival occured at t, just before the arrival:

$$\alpha_t(x) = \mathbb{P}(Y_t = x) = \lim_{h\to 0} \mathbb{P}(X_{t-h} = x | N(t-h, t) = 1).$$

We assume that there are two types of transition rates: $\lambda(x, y) = \lambda'(x, y) + \lambda''(x, y)$ with $\lambda'(x, y)$ representing the arrivals and $\lambda''(x, y)$ the other transitions. Define $\Lambda'(x) = \sum_y \lambda'(x, y)$, and let N now represent event of the λ'-type. $\alpha_t(x)$ can be seen as the fraction of arrivals that occurs in x. Thus $\alpha_t(x)$ is given by:

$$\alpha_t(x) = \lim_{h\to 0} \mathbb{P}(X_{t-h} = x | N(t-h, t] = 1) = \lim_{h\to 0} \frac{\mathbb{P}(X_{t-h} = x, N(t-h, t] = 1)}{\mathbb{P}(N(t-h, t] = 1)} =$$

$$\lim_{h\to 0} \frac{\pi_{t-h}(x)\Lambda'(x)h + o(h)}{\sum_{y\in\mathcal{X}} \pi_{t-h}(y)\Lambda'(y)h + o(h)} = \frac{\pi_t(x)\Lambda'(x)}{\sum_{y\in\mathcal{X}} \pi_t(y)\Lambda'(y)}.$$

If $\Lambda'(x)$ is constant then we find $\alpha_t(x) = \pi_t(x)$ as was expected from Theorem 3.6.1. Taking the limit for $t \to \infty$ gives $\alpha_\infty(x) = \pi_\infty(x)$.

4.6 Time-inhomogeneous chains

In this section we consider time-inhomogeneous Markov chains. For these chains we derive methods to compute the distribution of the process at T. A special case is the distribution of X_T for the time-homogeneous continuous-time Markov chain, which we delayed until this section.

Deriving results for discrete-time chains is easy. We make the chain time-inhomogeneous by letting P depend on t, thus P_t is the transition matrix at t. Now we get, following Section 4.1, $\pi'_{t+1} = \pi'_t P_{t+1}$, and recursively $\pi'_t = \pi'_0 P_1 \cdots P_t$.

Continuous-time chains are more challenging. We assume that the transition rates are piecewise constant. Now consider a time interval of length T for which the rates are constant. For simplicity we assume that the interval starts at 0. We will construct a method to approximate π_T starting from π_0. The only additional condition we need is that $\Lambda(x)$ is uniformly bounded. Note that this is always the case if $|\mathcal{X}| < \infty$.

Choose a constant Λ such that $\Lambda(x) \leq \Lambda$ for all $x \in \mathcal{X}$. Now add the possibility of "dummy" transitions from a state to itself. Take $\lambda(x,x) = \Lambda - \sum_{y \neq x} \lambda(x,y)$. Then the time between each two jumps is exponentially distributed with parameter Λ, independent of the current state. The total number of jumps between 0 and T has a Poisson distribution with rate ΛT. Conditioned on the number of jumps the transition process is a discrete-time Markov chain with transition probabilities $\hat{p}(x,y) = \lambda(x,y)/\Lambda$ for all y including x. Thus, if the number of jumps in $[0,T]$ is k, then the distribution at T will be $\pi_0'\hat{P}^k$. This leads to the following formula:

$$\pi_T' = \sum_{k=0}^{\infty} \frac{(\Lambda T)^k}{k!} \mathrm{e}^{-\Lambda T} \pi_0' \hat{P}^k. \tag{4.9}$$

If we want to use this in practice, we have to limit the summation to some upper bound K. This also limits the number of states that can be reached for any state x, which is helpful in case $|\mathcal{X}| = \infty$.

The process that we just described is called *uniformization*.

Example 4.6.1 Consider again the single-server queue of Example 4.3.1. Suppose that the system is initially empty. As an example, take $\lambda = 1$ for all t and $\mu = 1/2$ for $t \in [0,10]$, and $\mu = 2$ for $t > 10$, and let us see how $\pi_t(0)$ evolves over time. Computations show that $\pi_{10}(0) \approx 0.033$, $\pi_{15}(0) \approx 0.31$ and $\pi_{20}(0) \approx 0.44$. Note that until $t = 10$ the departures do not counterbalance the arrivals, explaining why $\pi_{10}(0)$ is that small. From time 10 on $\pi_t(0)$ increases steadily to the limit 0.5.

4.7 Further reading

Most books on probability theory, Operations Research, or queueing theory contain one or more chapters on Markov chains. Next to that there are many books entirely devoted to the subject. From this enormous literature we note the Chapters 4 and 7 of Ross [75] and Chapters 3 and 4 of Tijms [93].

Information on the inventor of Markov chains, A.A. Markov, can be found on Wikipedia.

4.8 Exercises

Exercise 4.1 Consider a finite-state discrete-time Markov chain.
a. Explain how to compute π_{64} with as few matrix multiplications as possible.
b. Do the same for π_{21}.

Exercise 4.2 Consider a Markov chain with $\mathcal{X} = \{0, 1, \ldots\}$. For $0 < q < 1$ and all $x \geq 0$ we take $p(x, x+1) = q$ and $p(x+1, x) = 1 - q$, $p(0,0) = 1 - q$.
a. Find all solutions of Equation (4.3).

b. Determine for which q there is a solution that also satisfies Equation (4.2).
c. Give an intuitive interpretation of your findings.

Exercise 4.3 Consider an arbitrary Markov chain with up to say 10 states.
a. Make an Excel sheet in which you can calculate $\pi_1, \pi_2, ..., \pi_{100}$.
b. Try different examples with a slow and fast convergence to stationarity.
c. Define the distance between π_k and π_{100} in some appropriate way and make a plot as a function of k.

Exercise 4.4 Consider a company with a central telephone switch that is connected to the public network by N outgoing lines. This means that there can be no more than N calls in parallel. Calls arrive according to a Poisson process of rate λ, each call has an exponential duration with parameter μ. We model the number of busy lines as a birth-death process.
a. Give the transition rates of this birth-death process.
b. Give a formula for the probability that all lines are occupied.
c. Calculate this number for $N = 3$, $\lambda = 1$ and $\mu = 0.5$.
d. What do you think that is unrealistic about this model?

Exercise 4.5 We model a hospital ward as follows. Patients arrive according to a Poisson process, and each has an exponentially distributed length of stay. We assume that there is enough capacity to handle all patients (thus, theoretically, an infinite number of beds). We model the occupancy as a birth-death process.
a. Give the transition rates of this birth-death process.
b. Give a formula for the long-run probability that x beds are occupied.
c. Give the coefficient of variation of the number of occupied beds.
d. Given the answer to c, what is your conclusion with respect to the size of hospital wards?

Exercise 4.6 Two machines are maintained by a single repairman. The repair time is exponential with rate μ, each machine fails (when functioning) with a rate λ. When both machines are down one is being repaired, the other is waiting for repair. We model the number of functioning machines as a continuous-time Markov chain.
a. Give the transition rates.
b. Give the expected number of functioning machines and the probability that both machines are not functioning.
c. What is the state distribution perceived by a machine going down?
d. What is the expected time between failure of a machine and the moment the repair is finished?

Exercise 4.7 Consider a system with 3 machines and 2 repairmen. Machines fail independently with rate λ. Each repairman repairs machines at rate μ. When one machine is down then only one repairman can work, when 2 or more machines are down both work.
a. Model this system as a birth-death process.

b. Calculate the stationary distribution and use this to derive the long-run expected number of machines that are functioning.
c. Derive the long-run distribution at moments that a machine fails.
d. Use this to derive the distribution of the long-run average time a machine waits before it is taken into service and calculate its expectation.

Exercise 4.8 Consider a Markov chain with $\mathcal{X} = \{0, 1\}$, $\lambda(0, 1) = \lambda(1, 0) = 1$, and $\pi_0(0) = 1$.
a. Show that $\pi_t(1) = (1 - \exp(-2t))/2$.
Now assume that $\lambda(0, 1)$ and $\lambda(1, 0)$ are different.
b. Derive, using Equation (4.6), an expression for $\frac{d}{dt}\pi_t(1)$.
c. Find an expression for $\pi_t(1)$.

Chapter 5

Queueing Models

In this chapter we study queueing models. Queueing models are characterized by the fact that customers or jobs compete for the same service(s). The focus in this chapter is on analytic solutions for time-stationary models. This is because there are few results for non-stationary models. Usually one has to rely on simulation or computational Markov chain methods in that case.

The first question for any queueing model is whether it is well-dimensioned: does the processing capacity exceed the expected load per unit of time? If this is the case then the model is called *stable*. However, most of the phenomena that we are interested in deal with consequences of randomness in arrival and service processes. E.g., stable call centers without fluctuations have zero waiting times and in hospitals without fluctuations the number of occupied beds is always the same. This is evidently not the case: randomness is predominant in queueing.

5.1 Classification

A rough classification of the most common models is as follows. A queueing model can have one or more nodes, one or more types of customers, and each node can have one or more servers. A model with multiple nodes is called a *queueing network*. In what follows we discuss first single-node single-type models, then single-node multi-type models, and finally queueing networks.

For the single-node single-type models we use the well-known Kendall notation, which is of the form $A|B|c|d$. Here A represents the arrival process, B the service time distribution, c the number of servers, and d the total number of places in the queueing system. A and B are usually either M ("Markovian", i.e., Poisson arrivals or exponential service times), D ("deterministic", i.e., constant interarrival or service times), or G ("general" interarrival or service times, sometimes denoted as GI to stress that the interarrivals or departures are mutually independent). Of course c and d are integers, with $c \leq d$. If $d = \infty$ it is often skipped.

Another aspect of queueing models is the *queueing discipline*. Usually we assume that customers at each node are served in the order of arrival, i.e., first-come-first-served (FCFS). In a single server system a customer that is served earlier than another customer leaves the system earlier as well; in this case FCFS is equivalent to first-in-first-out, FIFO. Other well-known disciplines are LIFO (last-in-first-out) and PS (processor sharing), which means that every customer gets an equal part of the service capacity.

5.2 Notation and queueing basics

Over the years a form of standard notation for queueing systems has evolved, which has the following ingredients:
- λ denotes the parameter of the Poisson arrival process;
- S is the service time distribution;
- μ is the parameter of the service time distribution in case it is exponential;
- $\beta = \mathbb{E}S$ is the expected service time (thus $\beta = 1/\mu$ in case of exponentiality);
- s is the number of servers.

In what follows next we use the following notation for waiting times and queue lengths:
- W_Q is the time that an arbitrary customer spends waiting before service, in a stationary situation;
- W is the time that an arbitrary customer spends in the system, while waiting and while being served;
- L_Q is the limiting number of customers in the queue;
- L is the limiting number of customers in the system;
- π is the stationary distribution of the number of customers in the system. In contrast to Chapter 4 we write π instead of π_∞.

Above we introduced notation for performance measures in a stationary situation. However, before we can study stationary behavior, we should determine whether or not the system eventually reaches equilibrium. In terms of discrete-time Markov chains (Section 4.1), this is equivalent to saying that $\mathbb{E}T_x < \infty$. In a queueing context, where all customers wait until they get serviced, stationarity is equivalent to requiring that, on average, per unti of time less work arrives than the server(s) can handle. This is the case if $\lambda < s/\beta$. Thus $\lambda\beta/s < 1$ is the stability condition. We usually write $\rho = \lambda\beta/s$. Note that when delayed customer leave, then there is no stability issue.

Example 5.2.1 The $M|M|1|\infty$ or $M|M|1$ queue is stable if and only if $\lambda < \mu$; see also Example 4.3.1. The $M|M|1|1$ queue (a two-state continuous-time Markov chain) is always stable.

In Section 3.8 a number of useful relations were derived using so-called *cost equations*. The best known cost equation is Little's law, which states that $l = \mathbb{E}L = \lambda\mathbb{E}W = \lambda w$ (Equation (3.3)) and $\mathbb{E}L_Q = \lambda\mathbb{E}W_Q$ for regenerative processes.

Another important property that we will regularly use is *PASTA*, which stands for "Poisson arrivals see time averages". See Section 3.6.

5.3 Single-server single-type queues

In this section we study the $M|M|1$ and the $M|G|1$ queues. The main results for the $M|M|1$ queue are:

Theorem 5.3.1 **($M|M|1$ queue)** *The following results hold for the $M|M|1$ queue with $\rho = \frac{\lambda}{\mu} < 1$: The stationary distribution π is geometric and given by*

$$\pi(j) = (1-\rho)\rho^j, \tag{5.1}$$

$$\mathbb{E}W_Q = \frac{\rho}{\mu(1-\rho)}, \quad \mathbb{E}L_Q = \frac{\rho^2}{1-\rho}, \tag{5.2}$$

$$\mathbb{E}W = \frac{1}{\mu(1-\rho)}, \quad \mathbb{E}L = \frac{\rho}{1-\rho}, \tag{5.3}$$

and

$$\mathbb{P}(W_Q > t) = \rho \mathrm{e}^{-(1-\rho)\mu t}.$$

Theorem 5.3.1 has some interesting consequences. We have $\pi(0) = 1-\rho$, thus the server is busy a fraction ρ of the time. Hence we would expect that $\mathbb{E}L - \mathbb{E}L_Q = \rho$, which is indeed the case. When the server is busy then arriving customers are delayed, and thus, using "Poisson arrivals see time averages" (PASTA): $\mathbb{P}(W_Q > 0) = \rho$. Note that

$$\mathbb{P}(W_Q > t | W_Q > 0) = \frac{\rho \mathrm{e}^{-(1-\rho)\mu t}}{\rho} = \mathrm{e}^{-(1-\rho)\mu t},$$

which is the tail of an exponential distribution. Thus given that you have to wait, your remaining waiting time is exponential. This means for example that the remaining expected waiting time never changes! (See Section 1.6.5 for properties of the exponential distribution.)

It is also interesting to note that $\mathbb{E}L$ and $\mathbb{E}L_Q$ are dimensionless, as they only depend on ρ. Thus if time is scaled, i.e., λ and μ are mutiplied by the same number, then the average queue length does not change.

Proof **of Theorem 5.3.1** The expression for $\pi(j)$ has been obtained in Example 4.3.1. The next four expression follow from $\mathbb{E}L = \sum_{j=0}^{\infty} j\pi(j)$, $\mathbb{E}L_Q = \sum_{j=1}^{\infty}(j-1)\pi(j)$, and Little's law, which states that $\mathbb{E}L_{(Q)} = \lambda \mathbb{E}W_{(Q)}$. For the expression of the waiting time distribution we refer to the discussion of the $M|M|s$ queue. □

Quite often arrival processes are (approximately) Poisson; rarely however service times are exponentially distributed. In what follows we derive the expected waiting time in the $M|G|1$ queue. This will be used to study the influence of randomness in the service times on the behavior of the queue. The arrival rate is again λ, service times are i.i.d., denoted with the r.v. S. In accordance with the exponential case we define $\rho = \lambda \mathbb{E}S$.

To be able to study the importance of randomness in what follows we will use the squared coefficient of variation of S. Recall from Section 1.2 that the squared coefficient of variation $c^2(X)$ of a distribution X is defined by $c^2(X) = \mathbb{E}(X - \mathbb{E}X)^2/(\mathbb{E}X)^2$.

Theorem 5.3.2 **($M|G|1$ queue)** *For the $M|G|1$ queue with $\rho = \lambda\mathbb{E}S < 1$ holds:*

$$\mathbb{E}W_Q = \frac{\lambda\mathbb{E}S^2}{2(1-\lambda\mathbb{E}S)} = \frac{\rho\mathbb{E}S(1+c^2(S))}{2(1-\rho)}, \quad \mathbb{E}L_Q = \frac{\lambda^2\mathbb{E}S^2}{2(1-\lambda\mathbb{E}S)} = \frac{\rho^2(1+c^2(S))}{2(1-\rho)}, \tag{5.4}$$

$$\mathbb{E}W = \mathbb{E}W_Q + \mathbb{E}S, \quad \mathbb{E}L = \mathbb{E}L_Q + \rho.$$

Formula 5.4 is the celebrated *Pollaczek-Khintchine formula*. We see that the expected waiting time depends only on the first two moments of S.

Proof of Theorem 5.3.2 First we calculate the total expected amount of work that an arbitrary customer has in queue over time, i.e., the contribution of a customer to the workload process. When looking at Figure 5.1, this corresponds to the surface between the arrival and the departure of the customer, the shaded area. We call this variable U. Then

$$\mathbb{E}U = \mathbb{E}\Big(SW_Q + \int_0^S (S-x)dx\Big) = \mathbb{E}S\mathbb{E}W_Q + \frac{\mathbb{E}S^2}{2}.$$

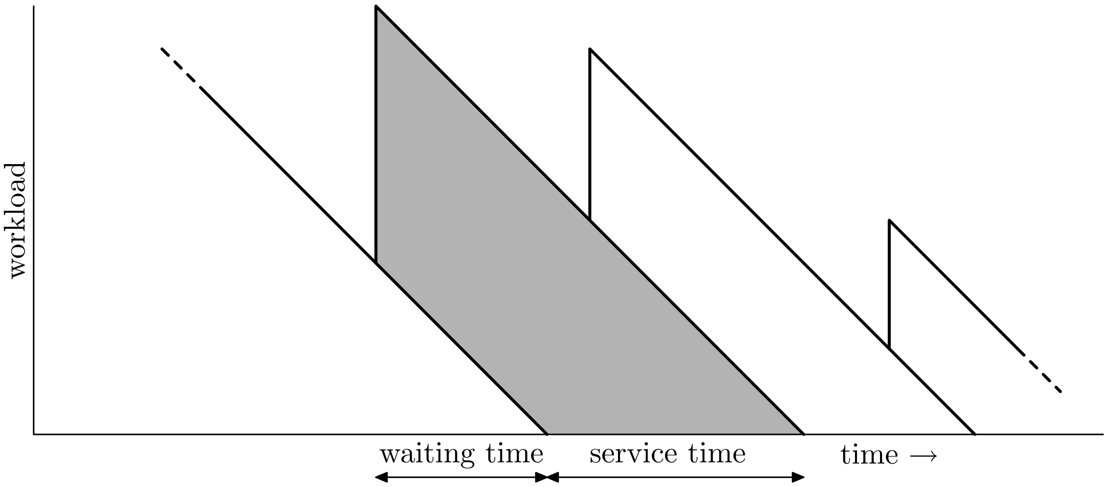

Figure 5.1: The workload process.

Let V be the stationary amount of work in the system. Then there is the "cost equation" $\mathbb{E}V = \lambda\mathbb{E}U$. Thus

$$\mathbb{E}V = \lambda\mathbb{E}S\mathbb{E}W_Q + \frac{\lambda\mathbb{E}S^2}{2}.$$

But PASTA tells us that $\mathbb{E}V = \mathbb{E}W_Q$, and finally we find $\mathbb{E}W_Q = \frac{\lambda\mathbb{E}S^2}{2(1-\lambda\mathbb{E}S)}$. The second expression for $\mathbb{E}W_Q$ is obtained as follows:

$$\frac{\lambda\mathbb{E}S^2}{2(1-\lambda\mathbb{E}S)} = \frac{\lambda(\mathbb{E}S)^2(1+c^2(S))}{2(1-\lambda\mathbb{E}S)} = \frac{\rho\mathbb{E}S(1+c^2(S))}{2(1-\rho)}.$$

The expression for $\mathbb{E}L_Q$ is derived using Little's law. By an argument using a cost equation (see Section 3.8) it can be shown that ρ is the fraction of the time that the server is busy, which gives $\mathbb{E}L = \rho + \mathbb{E}L_Q$. The expression for $\mathbb{E}W$ follows again from Little's law. □

Remark 5.3.3 Higher moments of W_Q (and therefore the whole distribution) can be derived from the Laplace transform of W_Q, which is the Pollaczek-Khintchine formula in its general form.

Let us interpret Theorem 5.3.2. If S is deterministic, then $c^2(S) = 0$; if S is exponential, then $c^2(S) = 1$. Thus the $M|M|1$ has an expected waiting time which is twice as high as the $M|D|1$ queue (where the D stand for deterministic) with the same ρ. In Figure 5.2 we see $\mathbb{E}W_Q$ as a function of λ for different values of $c^2(S)$.

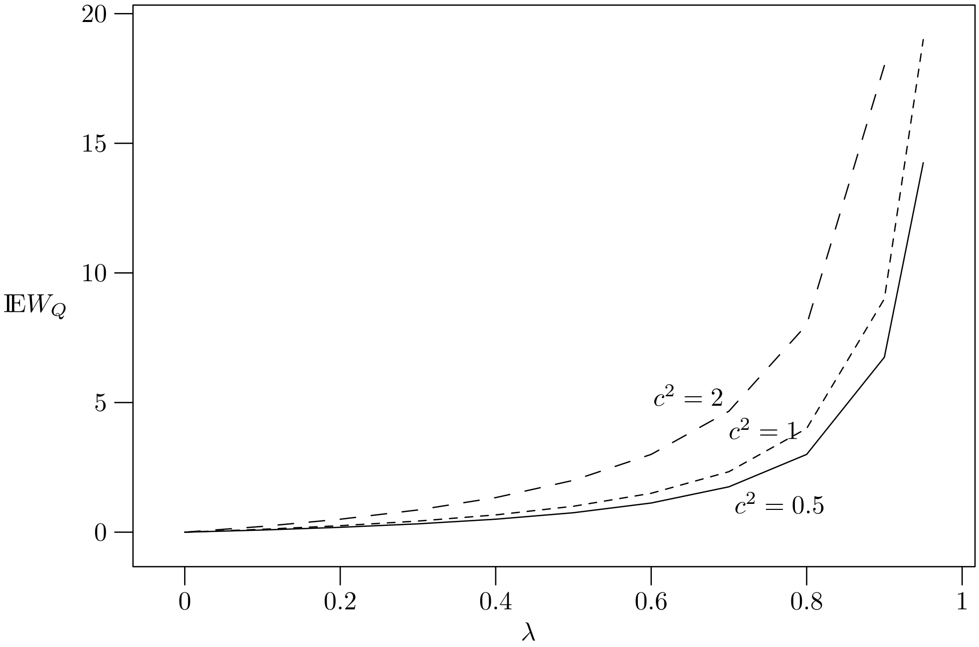

Figure 5.2: $\mathbb{E}W_Q$ as a function of λ and $c^2(S)$, for $\mathbb{E}S = 1$.

Note that $\mathbb{E}L_Q$ is again dimensionless, as it depends only on ρ and $c^2(S)$.

Example 5.3.4 Printers are typical examples of $M|G|1$ queues, and we all know that the size of printer jobs has a high variability. This leads to typical behavior of queues with a high $c^2(S)$ and a relatively low ρ: often we find no customers at all, but if there is a queue, then it is often very long.

Remark 5.3.5 There is no known expression for the $G|G|1$ queue. However, equation (5.4) can be used as the basis for an approximation of the expected waiting time of the $GI|G|1$ queue, which is:

$$\mathbb{E}W_Q \approx \frac{\rho\mathbb{E}S(c^2(A) + c^2(S))}{2(1-\rho)},$$

where A is the inter-arrival time and $\rho = \mathbb{E}S/\mathbb{E}A$. Note that the formula is exact for A exponential.

Remark 5.3.6 (The $M|G|1$ queue with processor sharing) Above we saw how uncertainty plays a role in the $M|G|1$ queue. An obvious way to reduce waiting times is reducing $c^2(S)$. If we assume the service times as given, and there are no ways to obtain information about the realizations of S, then changing the queueing discipline is an option.

So far we discussed the $M|G|1$ queue with the FIFO discipline. In the case of service times with a high variability this can lead to long delays because of customers demanding long service times blocking the server. A possible solution is changing the queueing discipline, for example to processor sharing (PS). Because under this discipline every customer is taken in service immediately at its arrival instant, we compare W and not W_Q for FIFO and PS.

First note that for the $M|M|1$ system the queue length is independent of the queueing discipline, which we denote by $\mathbb{E}L(\text{FIFO}) = \mathbb{E}L(\text{PS})$. From this it follows that $\mathbb{E}W(\text{FIFO}) = \mathbb{E}W(\text{PS})$, by Little's law.

It is well known that $\mathbb{E}L(\text{PS})$ is insensitive to higher moments of S (e.g., Section 4.4 of Kleinrock [57]), thus for the $M|G|1$ queue with the PS discipline holds

$$\mathbb{E}L(\text{PS}) = \frac{\lambda \mathbb{E}S}{1-\lambda\mathbb{E}S} \text{ and } \mathbb{E}W(\text{PS}) = \frac{\mathbb{E}S}{1-\lambda\mathbb{E}S}.$$

From this it follows directly that

$$\mathbb{E}W(\text{PS}) \leq \mathbb{E}W(\text{FIFO}) \Longleftrightarrow c^2(S) \geq 1.$$

Sometimes it is not possible to change the queueing discipline to PS; in some of these cases it is possible to break the service center up in smaller components. Then the $M|G|1$ queue is replaced by an $M|G|s$ queue with the same joint service capacity. Unfortunately there is no expression for the expected waiting time in the $M|G|s$ queue; for individual cases it should be examined whether splitting up the service capacity is an improvement or not. This only works for highly loaded systems and very highly variable service times, as the system works not at its full capacity if not all components are busy.

Example 5.3.7 Coming back to the printer example, it can be argued that it is advantageous to install multiple small printers instead of a single big one; of course all users should have access to all printers.

5.4 Multi-server single-type queues

The $M|M|s$ queue, with arrival rate λ and s servers that each can serve at rate μ, can be modeled as a birth-death process on $\{0,1,\dots\}$ (see Section 4.3), where state j represents the number of customers in the system (including the customers in service).

In the sequel we use the following notation: $a = \lambda/\mu$, the offered load (in Erlang), and $\rho = a/s$, the load (in Erlang per server).

Theorem 5.4.1 **($M|M|s$ queue)** *The following results hold for the $M|M|s$ queue with $\rho < 1$: The stationary distribution π is given by*

$$\pi(j) = \begin{cases} \dfrac{a^j}{j!}\pi(0) & \text{if } j < s, \\ \dfrac{a^j}{s!s^{j-s}}\pi(0) & \text{otherwise,} \end{cases}$$

with

$$\pi(0)^{-1} = \sum_{j=0}^{s-1} \frac{a^j}{j!} + \frac{a^s}{(s-1)!\,(s-a)};$$

$$\mathbb{E}W_Q = \frac{C(s,a)}{s\mu - \lambda}, \quad \mathbb{E}L_Q = \frac{\rho C(s,a)}{1-\rho},$$

and

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

with

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

$$\mathbb{E}W = \mathbb{E}W_Q + 1/\mu, \quad \mathbb{E}L = \mathbb{E}L_Q + a.$$

Note that the constant $C(s,a)$ can be interpreted as the probability of delay: $C(s,a) = \mathbb{P}(W_Q > 0)$. The r.v. $W_Q|W_Q > 0$, the waiting time distribution given that one has to wait, has an exponential distribution with the overcapacity $s\mu - \lambda$ as rate.

Proof of Theorem 5.4.1 Note that the $M|M|s$ queue is a birth-death process. Thus we can use the theory developed in Section 4.3. The transition rates are as follows: $\lambda_j = \lambda$ and $\mu_j = \min\{s,j\}\mu$, $j \geq 0$. The equilibrium equations are given by $\lambda\pi(0) = \mu\pi(1)$ and $(\lambda + \min\{s,i\}\mu)\pi(i) = \lambda\pi(i-1) + \min\{s,i+1\}\mu\pi(i+1)$ for $i > 0$. Summing these equalities for $i = 0$ up to j gives: $\lambda\pi(j) = \min\{s,j+1\}\mu\pi(j+1)$ for all j. From this it follows that:

$$\pi(j) = \begin{cases} \dfrac{a^j}{j!}\pi(0) & \text{if } j < s \\ \dfrac{a^j}{s!s^{j-s}}\pi(0) & \text{otherwise} \end{cases}$$

The value of $\pi(0)$ can be derived from $\sum_{j=0}^{\infty} \pi(j) = 1$, giving

$$\pi(0)^{-1} = \sum_{j=0}^{s-1} \frac{a^j}{j!} + \frac{a^s}{(s-1)!\,(s-a)}.$$

Now we calculate the waiting time distribution W_Q. Thanks to PASTA we can interpret W_Q as the time, starting at time 0 with the stationary queue length distribution L, until at least one of the servers stops working on the jobs that were initially present. This is the moment that the customer arriving at 0 would go into service. The probability $\mathbb{P}(W_Q > t|L = j+s)$ is then equal to the probability that there are less than $j+1$ departures in t time units. The probability of k departures for $k < j+1$ is given by the probability that a Poisson distributed random variable with parameter $s\mu t$ has outcome k. Thus

$$\mathbb{P}(W_Q > t|L = j+s) = \mathrm{e}^{-s\mu t} \sum_{k=0}^{j} \frac{(s\mu t)^k}{k!}.$$

We are only interested in $W_Q|W_Q > 0$. Note that W_Q has an atom at 0 of size $1 - \mathbb{P}(W_Q > 0) = 1 - C(s,a)$. For this reason we are interested in the distribution of $(L|W_Q > 0) = (L|L \geq s)$. This distribution is geometric with parameter ρ: $\mathbb{P}(L = s + j|L \geq s) = (1-\rho)\rho^j$, $j \geq 0$.

Now, using Equation (1.6),

$$\mathbb{P}(W_Q > t|W_Q > 0) = \sum_{j=0}^{\infty} \mathbb{P}(W_Q > t|L = j+s, W_Q > 0)\mathbb{P}(L = j+s|W_Q > 0) =$$

$$\sum_{j=0}^{\infty} \mathbb{P}(W_Q > t|L = j+s)\mathbb{P}(L = j+s|W_Q > 0) = (1-\rho)\mathrm{e}^{-s\mu t} \sum_{j=0}^{\infty}\sum_{k=0}^{j} \frac{(s\mu t)^k}{k!}\rho^j =$$

$$(1-\rho)\mathrm{e}^{-s\mu t} \sum_{k=0}^{\infty} \frac{(s\mu t)^k}{k!} \sum_{j=k}^{\infty} \rho^j = \mathrm{e}^{-s\mu t} \sum_{k=0}^{\infty} \frac{(\rho s\mu t)^k}{k!} = \mathrm{e}^{-(1-\rho)s\mu t} = \mathrm{e}^{-(s\mu-\lambda)t}.$$

Putting all together we find that

$$\mathbb{P}(W_Q > t) = C(s,a)\mathrm{e}^{-(s\mu-\lambda)t}.$$

From the interpretation it is immediately clear that

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

By Little's law, $\mathbb{E}L_Q = \lambda \mathbb{E}W_Q$, we also find

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

Note that $\mathbb{E}L_Q$ depends only on λ and μ through the quotient $a = s\rho$.

From a cost equation (see Equation (3.4)) it follows that $\mathbb{E}W = \mathbb{E}W_Q + 1/\mu$. □

Many implementations of the Erlang C formula can be found on the web. See, e.g., www.math.vu.nl/∼koole/obp/ErlangC.

Remark 5.4.2 (The $M|G|s$ queue) For the $M|G|s$ queue no closed-form expressions exist for the average waiting time or other performance measures. Many approximations and numerical methods exist in the literature.

Up to now we studied the $M|M|s$ queue, which is a delay model: customers that find all servers occupied are delayed until service capacity is available. We continue with the $M|G|s|s$ queue, which is a blocking model. We derive its stationary distribution, leading to the surprising fact that this is only a function of the mean of the service time.

Theorem 5.4.3 ($M|G|s|s$ **queue)** *The following results hold for the $M|G|s|s$ queue with s possibly ∞: The stationary distribution π is given by*

$$\pi(i) = \frac{(\lambda \mathbb{E}S)^i/i!}{\sum_{j=0}^{s}(\lambda \mathbb{E}S)^j/j!}; \tag{5.5}$$

$$\mathbb{E}L = (1 - \pi(s))\lambda \mathbb{E}S. \tag{5.6}$$

Note that for $s = \infty$ the expression simplifies to $\pi(i) = (\lambda\mathbb{E}S)^i/i!\,\mathrm{e}^{-\lambda\mathbb{E}S}$, a Poisson distribution. Note also that $\pi(s)$ for $s < \infty$ represents the blocking probability, thanks to PASTA. The model is also known as the Erlang B model, and the blocking probability $\pi(s)$ is sometimes written as $B(s, a)$, with $a = \lambda\mathbb{E}S$ the offered load. The number $aB(s, a)$ represents the load that is rejected, and $a(1 - B(s, a))$ is the load that enters the system, which is equal to the expected number of occupied servers.

Proof of Theorem 5.4.3 The proof of this result is not straightforward, except for $s = 1$ and ∞, or if the service time is exponentially distributed. For the general case we refer to the literature. For $s = 1$ the result follows readily from renewal theory (see Section 3.3). We continue with the $M|G|\infty(|\infty)$ system. Consider a Poisson process with rate λ on $(-\infty, \infty)$ and consider the $M|G|\infty$ queue at time 0. An arrival that occured at $-t$ for some $t > 0$ is still present at 0 with probability $\mathbb{P}(S > t)$. Thus the total number of customers still present at 0 has a Poisson distribution with parameter $\int_0^\infty \lambda\mathbb{P}(S > t)dt = \lambda\mathbb{E}S$. This corresponds with Equation (5.5).

Equation (5.6) follows from the fact that $(1 - \pi(s))\lambda$ is the average number of admitted customers per unit of time. It also follows directly from $\mathbb{E}L = \sum_{j=0}^{s-1} \lambda\pi(j)$. □

Example 5.4.4 The possible connections between two telephone exchanges can well be modeled by a loss system. Although it is generally thought that the length of telephone calls can be well approximated by an exponential distribution, it is of no influence to the availability of free connections, due to the insensitivity of the loss model.

An important property of the Erlang B model is that it shows economies of scale: when the load and number of servers are increased by the same percentage, then the blocking probability decreases. Similarly, when the servers of two parallel Erlang B models start pooling and effectively become a single Erlang B system, then that system performs better in the sense that the total occupancy is higher. These properties of the Erlang B models are formalized in the following theorem.

Theorem 5.4.5 *For the Erlang B model holds for every $a > 0$ that $B(s, sa)$ is strictly decreasing in s, $s \in \mathbb{N}$;*
for every $\lambda_1, \lambda_2, \beta_1, \beta_2 > 0$ and $s_1, s_2 \in \mathbb{N}$

$$(\lambda_1\beta_1 + \lambda_2\beta_2)B(s_1 + s_2, \lambda_1\beta_1 + \lambda_2\beta_2) \leq \lambda_1\beta_1 B(s_1, \lambda_1\beta_1) + \lambda_2\beta_2 B(s_2, \lambda_2\beta_2). \tag{5.7}$$

Proof The proof of the first expression involves analytical properties of the blocking probability; see the appendix of Smith & Whitt [86]. Equation (5.7) is equivalent to

$$(\lambda_1\beta_1 + \lambda_2\beta_2)(1 - B(s_1 + s_2, \lambda_1\beta_1 + \lambda_2\beta_2)) \geq \lambda_1\beta_1(1 - B(s_1, \lambda_1\beta_1)) + \lambda_2\beta_2(1 - B(s_2, \lambda_2\beta_2)).$$

Using the fact that $aB(s, a)$ is the expected number of occupied servers in the Erlang B model, we have to show that pooling increases the overall server occupation. This can be done using a *coupling argument*, which consists of comparing realizations of the processes in such a way that the pooled system always has a higher occupation. For details, see [86]. □

A further property is that $B(s, sa)$ is not only decreasing, but also *convex* in s. This means that increasing in size pays off less as the size increases: there are *diminishing returns*. There is ample numerical evidence for the correctness of this statement, but a formal proof is lacking.

Remark 5.4.6 (The $M|G|s|N$ system) Some models combine blocking and delay, such as the $M|G|s|N$ queueing systems with $s < N < \infty$. As for the $M|G|s$ queue there is no closed-form solution for the standard performance measures. The stationary distribution of the $M|M|s|N$ can be obtained by analyzing the corresponding birth-death process:

$$\pi(j) = \begin{cases} \dfrac{a^j}{j!}\,\pi(0) & \text{if } 0 \leq j < s \\ \dfrac{a^j}{s!s^{j-s}}\,\pi(0) & \text{if } s \leq j \leq N \end{cases}$$

with

$$\pi(0)^{-1} = \sum_{j=0}^{s-1} \frac{a^j}{j!} + \sum_{j=s}^{N} \frac{a^j}{s!s^{j-s}}.$$

For the $M|M|1|N$ this simplifies to

$$\pi(j) = \frac{\rho^j}{1 + \cdots + \rho^N}.$$

The waiting time distribution is a mixture of gamma distributions, tail probabilities of the waiting time can therefore be calculated easily.

Up to now we discussed models with Poisson arrivals. The motivation for Poisson arrivals is an almost infinite pool of possible customers, who all have a very small arrival rate. Thus the number of customers in service has no influence on the arrival rate.

If the number of potential customers is small, then the number of customers in service or at the queue influences the arrival rate. To be precise, if there are in total n customers in the model, and there are in total j customers at the queue or in service, then the arrival rate is $\lambda(n-j)$. Thus each customer joins the service facility after an exponentially distributed time with parameter λ.

If we assume exponential service times, then the state of the system is completely described by the number of customers in queue. Therefore the model is a birth-death process. On the other hand, the model can also be seen as a two-station queueing network. Thus the results of the next section on networks of queues can also be utilized.

We give the formulas for two situations: where there are enough waiting places next to the $s < n$ servers, and the situations where there are not. We indicate these systems by adding a fifth entry to the Kendall notation, indicating the size of the population.

In the literature finite source models are also called Engset models, in contrast with the Erlang models that have Poisson arrivals. (See also Remark 5.6.6.)

Theorem 5.4.7 (Engset models) *The $M|M|s|\infty|n$ or, equivalently, $M|M|s|n|n$ model (the* Engset delay model*) has as steady state probabilities*

$$\pi(j) = \binom{n}{j}\Big(\frac{\lambda}{\mu}\Big)^j \pi(0) \tag{5.8}$$

if $0 \le j \le s$ and

$$\pi(j) = \frac{n!}{(n-j)!s!s^{j-s}}\Big(\frac{\lambda}{\mu}\Big)^j \pi(0) \tag{5.9}$$

if $j > s$, with

$$\pi(0)^{-1} = \sum_{j=0}^{s}\binom{n}{j}\Big(\frac{\lambda}{\mu}\Big)^j + \sum_{j=s+1}^{n}\frac{n!}{(n-j)!s!s^{j-s}}\Big(\frac{\lambda}{\mu}\Big)^j; \tag{5.10}$$

The $M|G|s|s|n$ model (the Engset blocking model*) has as steady state probabilities*

$$\pi(j) = \binom{n}{j}\Big(\lambda\mathbb{E}S\Big)^j \pi(0) \tag{5.11}$$

for $0 \le j \le s$ with

$$\pi(0)^{-1} = \sum_{j=0}^{s}\binom{n}{j}\Big(\lambda\mathbb{E}S\Big)^j. \tag{5.12}$$

The states with $j > s$ have $\pi(j) = 0$.

Proof The results for exponential service times follow using the theory on birth-death processes. For the insensitivity of the $M|G|s|s|n$ blocking model we refer to the literature. □

If $s = n$ then all customers have their "private" server and behave independently, each with stationary probability of $\lambda/(\lambda+\mu)$ of being in service. Therefore the stationary distribution in this case is given by

$$\pi(j) = \binom{n}{j}\Big(\frac{\lambda}{\lambda+\mu}\Big)^j\Big(\frac{\mu}{\lambda+\mu}\Big)^{n-j},$$

the binomial distribution. This can also be derived from (5.11) and (5.12), using the fact that $\sum_{j=0}^{n}\binom{n}{j}(\frac{\lambda}{\mu})^j = (1+\frac{\lambda}{\mu})^n$. In the same spirit as for $s = n$, the stationary distribution for $s < n$ can be rewritten as

$$\pi(j) = \frac{\binom{n}{j}(\frac{\lambda}{\lambda+\mu})^j(\frac{\mu}{\lambda+\mu})^{n-j}}{\sum_{i=0}^{s}\binom{n}{i}(\frac{\lambda}{\lambda+\mu})^i(\frac{\mu}{\lambda+\mu})^{n-i}}.$$

This is a binomial distribution cut off at level s.

Note that like the Erlang loss model the finite source loss model is insensitive for the service time distribution.

5.5 Single-server multi-type queues

Now suppose that some knowledge on S is available, i.e., on arrival we know to which class the customer belongs. Customers in each class have their own service time distribution and arrive according to a Poisson process. In this subsection we study the influence of priorities between the classes on the waiting times.

Suppose we have P classes of customers, with service times S_p and arrival rate λ_p, $p = 1, \dots, P$. Then the service time S of an arbitrary customer is equal to S_p with probability λ_p/λ, with $\lambda = \sum_p \lambda_p$. If customers are served using a FIFO discipline, then the Pollaczek-Khinthine formula (5.4) still holds with $\mathbb{E}S = \sum_p (\lambda_p/\lambda)\mathbb{E}S_p$ and $\mathbb{E}S^2 = \sum_p (\lambda_p/\lambda)\mathbb{E}S_p^2$.

Instead of FIFO we study the head-of-the-line (HOL) discipline, which is defined as follows: customers within a class are served in a FIFO manner, and when the server has finished serving a customer then a waiting customer from the class with the lowest class number is selected.

Define $W_Q(p)$ and $L_Q(p)$ as the waiting times and queue lengths of class p. Define also $\rho_i = \lambda_i \mathbb{E}S_i$ and $\sigma_p = \sum_{i=1}^p \rho_i$.

Theorem 5.5.1 (priority queue)

$$\mathbb{E}W_Q(p) = \frac{\mathbb{E}R}{(1-\sigma_p)(1-\sigma_{p-1})}. \tag{5.13}$$

$$\mathbb{E}W_Q(HOL) = \sum_{p=1}^{P} \frac{\lambda_p \mathbb{E}R}{\lambda(1-\sigma_p)(1-\sigma_{p-1})}. \tag{5.14}$$

Proof Above we found the following formula for the $M|G|1$ queue:

$$\mathbb{E}W_Q = \lambda \mathbb{E}S \mathbb{E}W_Q + \frac{\lambda \mathbb{E}S^2}{2}.$$

Using Little's law gives $\mathbb{E}W_Q = \mathbb{E}S\mathbb{E}L_Q + \lambda\mathbb{E}S^2/2$. This term can be explained as follows: $\mathbb{E}S\mathbb{E}L_Q$ takes into account the customers ahead in the queue, $\lambda\mathbb{E}S^2/2$ is the remaining expected service time of the customer currently in service. This can also be shown as follows. Let R be the remaining service time of the customer currently in service. In Section 3.7 an expression was derived for the remaining time until a renewal in a renewal process. This corresponds to R given that the server is busy:

$$\mathbb{E}(R|R>0) = \frac{\mathbb{E}S^2}{2\mathbb{E}S}.$$

Note that $\mathbb{P}(R>0) = \rho$, following from a cost equation, see Equation (3.4). Using Equation (1.7) we find indeed

$$\mathbb{E}R = \mathbb{E}(R|R>0)\mathbb{P}(R>0) + \mathbb{E}(R|R=0)\mathbb{P}(R=0) = \frac{\rho\mathbb{E}S^2}{2\mathbb{E}S} = \frac{\lambda\mathbb{E}S^2}{2}.$$

In what follows we introduce the different classes of customers, and we rewrite the above expression for each class of customers.

Define $N_i(p)$ as the number of customers of class i that arrive during $W_Q(p)$. Note that every customer is served eventually (we assume a stable system), and thus R does not depend on the service discipline. Therefore $\mathbb{E}R = \lambda\mathbb{E}S^2/2$. We have:

$$\mathbb{E}W_Q(p) = \mathbb{E}R + \sum_{i=1}^{p} \mathbb{E}L_Q(i)\mathbb{E}S_i + \sum_{i=1}^{p-1} \mathbb{E}N_i(p)\mathbb{E}S_i.$$

By Little's law $\mathbb{E}L_Q(i) = \lambda_i\mathbb{E}W_Q(i)$, by the Poisson arrivals $\mathbb{E}N_i(p) = \lambda_i\mathbb{E}W_Q(p)$. Thus

$$\mathbb{E}W_Q(p) = \mathbb{E}R + \sum_{i=1}^{p} \lambda_i\mathbb{E}S_i\mathbb{E}W_Q(i) + \sum_{i=1}^{p-1} \lambda_i\mathbb{E}S_i\mathbb{E}W_Q(p).$$

Then the last formula is equivalent to

$$(1-\sigma_p)\mathbb{E}W_Q(p) = \mathbb{E}R + \sum_{i=1}^{p-1} \lambda_i\mathbb{E}S_i\mathbb{E}W_Q(i).$$

By induction to p it can now be shown that

$$\mathbb{E}W_Q(p) = \frac{\mathbb{E}R}{(1-\sigma_p)(1-\sigma_{p-1})}.$$

This is the waiting time for each class separately. An arbitrary customer belongs to class p with probability λ_p/λ. From this the expected waiting under HOL, Equation (5.14), follows. □

Example 5.5.2 As an example, we study the case $P = 2$, and we will analyze in which cases FIFO or HOL is better. From the above it follows that:

$$\mathbb{E}W_Q(1) = \frac{\mathbb{E}R}{(1-\rho_1)},\ \mathbb{E}W_Q(2) = \frac{\mathbb{E}R}{(1-\rho_1-\rho_2)(1-\rho_1)}.$$

Then:

$$\mathbb{E}W_Q(HOL) = \frac{\lambda_1}{\lambda_1+\lambda_2}\mathbb{E}W_Q(1) + \frac{\lambda_2}{\lambda_1+\lambda_2}\mathbb{E}W_Q(2) = \frac{\mathbb{E}R(\lambda_1(1-\rho_1-\rho_2)+\lambda_2)}{(\lambda_1+\lambda_2)(1-\rho_1-\rho_2)(1-\rho_1)}.$$

We already saw that

$$\mathbb{E}W_Q(FIFO) = \frac{\mathbb{E}R}{1-\rho_1-\rho_2}.$$

A simple computation shows that $\mathbb{E}W_Q(HOL) \leq \mathbb{E}W_Q(FIFO)$ if and only if $\mathbb{E}S_1 \leq \mathbb{E}S_2$. Thus if shorter jobs get priority the average waiting time decreases. Of course, the customers with long waiting times have to pay for this.

In the example we saw that scheduling customers with short service times first decreases the average waiting time. Thus the queueing discipline that schedules the waiting job with the shortest processing time first, the *shortest-job-first* (SJF) discipline, minimizes the average waiting time. The waiting time can be computed from the above formula, by making a class of each possible service time. This gives:

$$\mathbb{E}[W_Q(SJF)|S=x] = \frac{\mathbb{E}R}{(1-\lambda\int_0^x tf_S(t)dt)^2},$$

with f_S the density of S. For $x=0$ we find $\mathbb{E}[W_Q(SJF)|S=0]=\mathbb{E}R$, which makes sense: a customer requiring 0 service gets priority over all waiting customers. On the other hand we find $\mathbb{E}[W_Q(SJF)|S=\infty]=\mathbb{E}R/(1-\rho)^2$, the time until the first moment the system gets empty. The expected waiting time is given by

$$\mathbb{E}W_Q(SJF) = \int_{x=0}^{\infty} \mathbb{E}[W_Q(SJF)|S=x]f_S(x)dx = \int_{x=0}^{\infty} \frac{\mathbb{E}Rf_S(x)}{(1-\lambda\int_0^x tf_S(t)dt)^2}dx. \quad (5.15)$$

This expression is not easy to solve for specific examples; using Maple we found for exponential service times with $\mu=1$ and various values of λ the following results.

λ	0.1	0.5	0.75	0.9
$\mathbb{E}W_Q(SJF)$	0.10	0.71	1.55	3.20
$\mathbb{E}W_Q(FIFO)$	0.11	1.00	3.00	9.00

Table 1. Waiting times for exponential service times with $\mu=1$.

SJF is optimal even if the customer do not arrive according to a Poisson process. Only in special cases we can compute the waiting times.

Remark 5.5.3 There can be many reasons to use HOL. If one is interested in (weighted) waiting times, then the problem is to order the classes such that the expected costs are minimal. Thus: reorder $1,\dots,P$ such that

$$\mathbb{E}C_Q(HOL) = \sum_{i=1}^{P} \frac{c_p\lambda_p\mathbb{E}R}{\lambda(1-\sigma_p)(1-\sigma_{p-1})}$$

is minimal. By exchanging the order 1 by 1 (as in the example) it can be shown that the classes should be ordered such that $c_1/\mathbb{E}S_1 \geq \cdots \geq c_P/\mathbb{E}S_P$. For exponential service times this translates to $c_1\mu_1 \geq \cdots \geq c_P\mu_P$, which is the reason why this policy is commonly known as the μc rule.

5.6 Queueing networks

So far we studied queueing systems with a single service station. In this section we will make a start with the study of networks of queues. First we consider a tandem system consisting of a number of ∞-capacity single-server queues with exponential service times and Poisson input at the first queue. Let there be V queues, server j with rate μ_j, and arrival rate λ at the first queue.

Theorem 5.6.1 (tandem queueing system) *For a tandem queueing system with V queues and $\lambda < \mu_j$ for all j, the input to each queue is Poisson and the marginal queue lengths are each independent, with joint steady state distribution*

$$\mathbb{P}(N_i = n_i, i = 1, \dots, V) = \prod_{i=1}^{V} \Big(1 - \frac{\lambda}{\mu_i}\Big)\Big(\frac{\lambda}{\mu_i}\Big)^{n_i}. \tag{5.16}$$

Proof We give the proof for $V = 2$. Let us first consider *time-reversibility.* Consider a Markov process N with rates $\lambda(i,j)$ and stationary distribution π. Now we define the reversed process $\tilde{N}$ by taking its transition rates $\tilde{\lambda}$ such that

$$\pi(i)\lambda(i,j) = \pi(j)\tilde{\lambda}(j,i). \tag{5.17}$$

It can be seen that $\tilde{\pi}$, the stationary distribution of $\tilde{N}$, is equal to π. From (5.17) it follows that N moves as often from i to j as $\tilde{N}$ moves from j to i. This means that under stationarity $\{N(t), t \in \mathbb{R}\}$ and $\{\tilde{N}(s-t), t \in \mathbb{R}\}$ are stochastically indistinguishable. If $\lambda(i,j) = \tilde{\lambda}(i,j)$ for all i and j, then we say that N is time-reversible. This we use for the study of two $M|M|1$ queues in series.

For the $M|M|1$ queue it follows directly from the stationary distribution (5.1) that $\pi(i)\lambda(i,j) = \pi(j)\lambda(j,i)$. Thus the $M|M|1$ queue is time-reversible. (In fact, any birth-death process is.) Therefore, the departures before t in N are the arrivals after t in $\tilde{N}$. The arrivals in $\tilde{N}$ are Poisson with rate λ, and thus so are the departures in N. Furthermore, these arrivals in $\tilde{N}$ after t are independent of $\tilde{N}(t)$, and thus the departures in N before t are independent of $N(t)$. In conclusion: the departure process of an $M|M|1$ queue forms a Poisson process and past departures are independent of the current state. Now going to a system of two $M|M|1$ queues in tandem, we observe that the state of the second queue N_2 at t depends on the departures at queue 1 before t. Thus $N_1(t)$ and $N_2(t)$ are independent, and thus

$$\mathbb{P}(N_1(t) = n_1, N_2(t) = n_2) = \mathbb{P}(N_1(t) = n_1)\mathbb{P}(N_2(t) = n_2) =$$

$$\Big(1 - \frac{\lambda}{\mu_1}\Big)\Big(\frac{\lambda}{\mu_1}\Big)^{n_1}\Big(1 - \frac{\lambda}{\mu_2}\Big)\Big(\frac{\lambda}{\mu_2}\Big)^{n_2}.$$

□

Theorem 5.6.1 is a very useful result: expected queue lengths and thus also expected waiting times can be calculated from it. However, we cannot derive results for waiting time distributions from it, because for example the length of the second queue at $t + s$ depends on the length of the first queue at t. The independence holds only if the queues are considered at the same moment.

When there is feedback in the system the traffic between stations is no longer according to a Poisson process. Surprisingly enough the generalization of equation (5.16) holds as well when there is feedback in the system. We will show this for networks of general birth-death processes with Jackson routing. This is defined as follows. Assume we have V stations, and the service rate at station p is equal to $\mu_p(k)$ if there are k customers present. Of course $\mu_p(0) = 0$. There are outside arrivals at station p according to a Poisson process with rate λ_p. If a customer leaves station i then it joins station j with probability r_{ij}. We assume that $\sum_{j=1}^V r_{ij} = 1 - r_{i0} \leq 1$, where we see station 0 as the outside of the network.

Evidently on average more customers arrive at station p during a unit of time than λ_p, due to the feedback. We determine this arrival rate, notated with γ_i. It is given by:

$$\gamma_i = \lambda_i + \sum_{j=1}^V \gamma_j r_{ji}. \tag{5.18}$$

We call these the *routing equations.* If this system has as unique solution $\gamma_i = 0$ if $\lambda_i = 0$ for all i then we call the network open: all customers eventually leave the system. If $r_{i0} = 0$ for all i then the network is closed, in this case λ_i needs to be 0 for all i. otherwise the system is unstable.

Theorem 5.6.2 (open queueing network) *The joint steady state distribution of an open queueing network is given by*

$$\mathbb{P}(N_1 = n_1, \dots, N_V = n_V) = \prod_{i=1}^V \pi_i(n_i). \tag{5.19}$$

with

$$\pi_i(n_i) = \mathbb{P}(N_i = n_i) = \Big(\sum_{n=0}^\infty \prod_{j=1}^n \frac{\gamma_i}{\mu_i(j)}\Big)^{-1} \prod_{j=1}^{n_i} \frac{\gamma_i}{\mu_i(j)}, \tag{5.20}$$

if the denominator of the last expression exists for all i.

Proof It suffices to show that $\pi(n) = \mathbb{P}(N_1 = n_1, \dots, N_V = n_V)$ satisfies the balance equations, given by

$$\pi(n)\Big(\sum_{i=1}^V \lambda_i + \sum_{i=1}^V \mu_i(n_i)\Big) = \sum_{i=1}^V \lambda_i \mathbb{I}\{n_i > 0\}\pi(n - e_i) +$$

$$\sum_{i=1}^V \mu_i(n_i+1) r_{i0}\pi(n+e_i) + \sum_{j=1}^V \sum_{i=1}^V \mu_j(n_j+1) r_{ji}\mathbb{I}\{n_i > 0\}\pi(n + e_j - e_i).$$

However, this is a tedious task. When doing it, one realizes that in fact the balance equations can be decomposed in $V+1$ equations, which all hold separately. There are:

$$\pi(n)\mu_i(n_i) = \lambda_i \pi(n - e_i) + \sum_{j=1}^V \mu_j(n_j+1) r_{ji}\pi(n + e_j - e_i),\ i = 1, \dots, V,$$

and

$$\pi(n)\sum_{i=1}^V \lambda_i = \sum_{i=1}^V \mu_i(n_i+1) r_{i0}\pi(n+e_i).$$

These equalities are called the *station balance* equations. Note that we left out the indicator, as the corresponding stationary probabilities are 0. Using the routing equations and filling in $\pi(n)$ the proof that these equations hold is relatively simple:

$$\pi(n)\mu_i(n_i) = \pi(n-e_i)\gamma_i = \pi(n-e_i)\Big(\lambda_i + \sum_{j=1}^{V} r_{ji}\gamma_j\Big) =$$

$$\pi(n-e_i)\lambda_i + \pi(n-e_i+e_j)\sum_{j=1}^{V} r_{ji}\mu_j(n_j+1)$$

for station $1,\dots,V$ and

$$\pi(n)\sum_{i=1}^{V}\lambda_i = \pi(n)\sum_{i=1}^{V}\Big(\gamma_i - \sum_{j=1}^{V}\gamma_j r_{ji}\Big) = \pi(n)\sum_{i=1}^{V}\gamma_i r_{i0} = \sum_{i=1}^{V}\pi(n+e_i)\mu_i(n_i+1)r_{i0}$$

for station 0. □

For the special case of single-server queues the marginal distributions simplify to the expression for the $M|M|1$ queue (see Theorem 5.3.1) and expressions for expected waiting times and queue lengths can be given.

Theorem 5.6.3 (open network of single-server queues) *If $\gamma_i < \mu_i$ for all i, then we have for the open queueing network consisting of single-server queues:*

$$\mathbb{P}(N_i = n_i, i = 1,\dots,V) = \prod_{i=1}^{V}(1-\frac{\gamma_i}{\mu_i})\Big(\frac{\gamma_i}{\mu_i}\Big)^{n_i},$$

$$\mathbb{E}W_Q = \frac{\sum_{j=1}^{V}\frac{\gamma_j^2}{\mu_j(\mu_j-\gamma_j)}}{\sum_{j=1}^{V}\lambda_j}, \quad \mathbb{E}L_Q = \sum_{j=1}^{V}\frac{\gamma_j^2}{\mu_j(\mu_j-\gamma_j)},$$

$$\mathbb{E}W = \frac{\sum_{j=1}^{V}\frac{\gamma_j}{\mu_j-\gamma_j}}{\sum_{j=1}^{V}\lambda_j}, \text{ and } \mathbb{E}L = \sum_{j=1}^{V}\frac{\gamma_j}{\mu_j-\gamma_j}.$$

Remark 5.6.4 As we did for single queue models, we can approximate network performance under general service time assumptions, see again Chapter 5 of [43] for an overview. The crucial point is that the output of an $M|G|1$ is not Poisson; therefore the network becomes a network of $G|G|1$ queues. The output of one station is the input to the next, therefore we approximate as well the squared coefficient of variation c_d^2 of the output process:

$$c_d^2 = (1-\rho^2)c^2(A) + \rho^2 c^2(S).$$

We will not go into detail about the approximation; just note that the value for $\rho \approx 0$ and $\rho \approx 1$ is correct.

Now consider a tandem system, in which customers visit all queues one by one. Assume that queue i has service time S_i, and queue 1 has interarrival times A_1 with $c^2(A_1) = c_{a1}^2$ as squared coefficient of variation. Then we use as estimation for the squared coefficient of variation of the input to queue i the number

$$c^2_{a,i+1} = (1-\rho_i^2)c^2_{a,i} + \rho_i^2 c^2(S_i).$$

Now together with the approximation of the waiting time we can find the expected lead time and work-in-process.

Next we consider closed systems: thus $\lambda_i = 0$ for all i, and also $\sum_{j=1}^V r_{ij} = 1$, to avoid that the system empties. We define again γ_i by

$$\gamma_i = \sum_{j=1}^V \gamma_j r_{ji}.$$

Note that if γ_i is a solution to this equation, then so is $c\gamma_i$. To make the solution unique we assume that we take γ such that $\sum_i \gamma_i = 1$. (We will see later on that we only need that $\gamma_i > 0$.) The rest of the analysis seems to go as for the open networks, leading to the same stationary distribution. However, here we forget that this time the system is not irreducible: we have to condition on the number of customers M in the system. This leads to the following theorem.

Theorem 5.6.5 (closed queueing network) *The stationary distribution of a closed queueing network with M customers in the system is:*

$$\mathbb{P}(N_1 = n_1, \dots, N_V = n_V) = \Big(\sum_{m:\sum_i m_i = M} \prod_{i=1}^V \pi_i(m_i)\Big)^{-1} \prod_{i=1}^V \pi_i(n_i) \qquad (5.21)$$

if $n_1 + \cdots + n_V = M$, 0 otherwise, with $\pi_i(n_i)$ as in Theorem 5.6.2

Due to the form of (5.20) it is easily seen that multiplying γ with a constant gives the same solution.

Example 5.6.6 We consider a two-station model, where one is a single server queue with ∞ capacity (with service rate μ), the other is an ∞-server queue (with service rate α). The ∞-server queue models a finite customer source, as such this model is known under the name *Engset delay model*, and often considered as a single station model. Here we use the theory of closed networks to derive its stationary distribution. As solution to the routing equations we take $\gamma_1 = \gamma_2 = 1$. We already know that $\pi_1(i) = (1-\gamma_1/\mu)(\gamma_1/\mu)^i$; it can be easily verified that $\pi_2(i) = \exp(-\gamma_2/\alpha)(\gamma_2/\alpha)^i/i!$. Thus the stationary distribution is

$$\mathbb{P}(N_1 = n_1, N_2 = n_2 = M - n_1) = C\prod_{i=1}^V \pi_i(n_i) =$$

$$C\Big(1-\frac{\gamma_1}{\mu}\Big)\Big(\frac{\gamma_1}{\mu}\Big)^{n_1} \mathrm{e}^{\frac{-\gamma_2}{\alpha}} \frac{\frac{\gamma_2}{\alpha}^{M-n_1}}{(M-n_1)!} = \tilde{C}\frac{\Big(\frac{\alpha}{\mu}\Big)^{n_1}}{(M-n_1)!}.$$

The normalizing constant is given by

$$\tilde{C}^{-1} = \sum_{n=0}^{M} \frac{\left(\frac{\alpha}{\mu}\right)^n}{(M-n)!} = \left(\frac{\alpha}{\mu}\right)^M \sum_{n=0}^{M} \frac{\left(\frac{\mu}{\alpha}\right)^n}{n!} \approx \left(\frac{\alpha}{\mu}\right)^M \exp\left(\frac{\mu}{\alpha}\right).$$

Thus $\tilde{C} \approx (\mu/\alpha)^M \exp(-\mu/\alpha)$.

Equation (5.21) is less useful then it appears to be at first sight, because the normalizing constant is very hard to calculate. Thus computing expected queue lengths and waiting times on the basis of this formula is computationally not feasible for reasonably sized networks.

A solution to this is a recursive method to compute the mean waiting times, called *mean value analysis*. We write it out for a network of single server queues. Denote with $W^M(j)$ the sojourn time of an arbitrary customer at queue j, and with $L^M(j)$ the queue length. The following result, known as the *Arrival Theorem*, is crucial to our analysis: A customer in the network arriving at a queue sees the network as if he is not in the network. For open networks, this does not induce a change; for closed networks with M customers, this means that a customer who changes queue sees the network in equilibrium as if there are $M-1$ customers. This gives

$$\mathbb{E}W^M(j) = \frac{1 + \mathbb{E}L^{M-1}(j)}{\mu_j}.$$

A cost equation shows that $\mathbb{E}L^M(j) = \delta_j^M \mathbb{E}W^M(j)$, with δ_j^M the throughput of queue j, i.e., the number of customers that are served on average by queue j. We call $\delta^M = \sum_j \delta_j^M$ the system throughput; from the routing mechanism it is clear that $\delta_j^M = \gamma_j \delta^M$, if we assume that $\sum_j \gamma_j = 1$. Then

$$M = \sum_{i=1}^{V} L^M(i) = \delta^M \sum_{i=1}^{V} \gamma_i \mathbb{E}W^M(i).$$

Now we have all ingredients to write $\mathbb{E}W^M(j)$ in a recursive way:

$$\mathbb{E}W^M(j) = \frac{1 + \mathbb{E}L^{M-1}(j)}{\mu_j} = \frac{1 + \delta^{M-1}\gamma_j \mathbb{E}W^{M-1}(j)}{\mu_j} = \frac{1}{\mu_j} + \frac{(M-1)\gamma_j \mathbb{E}W^{M-1}(j)}{\mu_j \sum_{i=1}^{V} \gamma_i \mathbb{E}W^{M-1}(i)}.$$

This results in a $O(MV)$ algorithm to compute the average waiting times.

5.7 Further reading

Standard books on queueing theory are Cooper [26], Kleinrock [58], and Gross & Harris [45]. See also Chapter 8 of Ross [75] and parts of Tijms [93]. The latter book also discusses the *uniformization method*, a computational method for computing performance measures of transient Markov chains that can also be applied to (small) queueing models. More advanced books are Walrand [96], Cohen [25] and Kleinrock [57]. A practice-oriented introduction is King [54].

Handbook 2 on stochastic models [47] contains chapters on queueing by Cooper and Walrand, and Handbook 3 on computing [24] has a chapter by Mitrani [68] on queueing models of computer systems.

Most of the results of this chapter can be found in any of the above references; the part on priority queueing is based on Chapter 3 of Kleinrock [57]. Mitrani [68] also discusses priority queueing and processor sharing. Kelly [52] is the standard reference to reversibility, but the basic ideas can be found in many books (e.g., [96, 75]). Approximations for the $M|G|s$ queue can be found in Sze [89], see also Tijms et al. [94].

More information about the Scandinavian queueing pioneers Erlang and Engset can be found on Wikipedia. See also Myskja [69].

5.8 Exercises

Exercise 5.1 Calculate the expected waiting time in the $M|G|1$ queue for all 12 parameter combination of $\lambda = 0.5$, 0.8, 0.9 and $\beta = \mathbb{E}S = 1$, with S deterministic, exponential, hyperexponential, and gamma with shape parameter 2. A hyperexponential distribution is a random mixture of exponential distributions; choose the parameters as you like (but do not take the trivial case with both exponentials having the same parameter).

Exercise 5.2 A printer can be modeled as an $M|G|1$ queue. For a specific printer it was determined that the arrival rate was 1 and that the average service time was 0.5, and that the printing time of an arbitrary document has approximately an exponential distribution.
a. Calculate the expected time between the moment a printer job is submitted and the moment the printer finishes printing it (the "system time").
It is found that the time to print a job is much longer in reality. The answer lies in printer failures: about 1% of the jobs cause the printer to get jammed. Repair takes on average 30 time units. By lack of data the repair time is assumed to be exponentially distributed. For the sake of the calculation the repair time is added to the printing time.
b. Calculate the first and second moment of this new printing time.
c. Calculate the expected system time if repairs are included.

Exercise 5.3 Consider an $M|G|1$ queue with arrival rate 0.5 and service time distribution $S = X + Y$, with X and Y independent and both exponentially distributed with rates 1 and 2, respectively.
a. Calculate $\mathbb{E}S$, $\mathbb{E}S^2$, $\sigma^2(S)$ and $c^2(S)$.
b. Calculate the expected waiting time and the expected sojourn time for the $M|G|1$ queue.

Exercise 5.4 Consider a system of two parallel $M|D|1$ queues. Both have load 80%, but one has service times of length 1 and the other of length 10.

a. Calculate the system times in both systems, and the expected overall system time.
The manager of the system considers merging both queues to obtain economies of scale. We approximate the resulting $M|D|2$ queue by a single $M|D|1$ queue with double service speed. Customer are treated in the order of arrival.
b. Characterize the arrival process and the service time distribution of the resulting queueing system.
c. Calculate the system time in this new $M|G|1$ queue.
d. Compare the results found under a and c and give an intuitive explanation. How would you redesign the system as to obtain the lowest possible average system time?

Exercise 5.5 In Remark 5.3.5 an approximation is given for the $G|G|1$ queue. Use this approximation to answer the following questions.
a. Give an approximation for the waiting time in the $D|M|1$ queue.
b. Compute it for $\lambda = 0.5$, 0.8, 0.9 and $\beta = \mathbb{E}S = 1$.
Let A be the interarrival time in a $G|M|1$ queue.
c. Give the distribution of A for the following *compound Poisson process*: batches of orders arrive according to a Poisson process, each batch having a geometrically distributed number of customer orders.
d. Compute $\mathbb{E}A^2$.
e. Give the approximation for this situation and compute it for $\lambda = 0.25$, 0.4, 0.45, $\beta = \mathbb{E}S = 1$, and average batch size 2.

Exercise 5.6 Consider an $M|M|s$ queue with $\mu = 0.2$.
a. Compute, using a tool such as www.math.vu.nl/∼koole/obp/ErlangC, the number of servers needed to assure that $\mathbb{P}(W_Q \leq 0.5) \geq 0.8$, for $\lambda = 1$, 10, and 100.
b. Give the overcapacity in each of the cases.
c. Give a definition of productivity of the servers and compute it for the three cases.

Exercise 5.7 Consider the $M|M|2$ queue, choose some arbitrary λ.
a. Give a formula for $C(s, a)$, and calculate it for $\rho = 0.5$, 0.75, and 0.95.
b. Give a formula for $\mathbb{P}(W_Q > t)$, and calculate it for the same parameter values, for some $t > 0$.

Exercise 5.8 Recall that $B(s, a)$ is the blocking probability in the Erlang B queueing system.
a. Let $N \sim \text{Poisson}(a)$. Show that $B(s, a) = \mathbb{P}(N = s)/\mathbb{P}(N \leq s)$.
b. Use this to write a simple Excel Erlang B calculator with the help of the Excel Poisson() function.
c. Show that $C(s, a) = sB(s, a)/(s - a(1 - B(s, a)))$.
d. Use this to write a simple Excel Erlang C calculator.
e. Compare the results with existing calculators.

Exercise 5.9 Use an Erlang B calculator (www.math.vu.nl/∼koole/obp/ErlangB for example) to answer the following questions. A hospital has two wards with both 10 beds and

an offered load of 9 Erlang. Patients arrive according to a Poisson process.
a. Give the probability that a patient is rejected because no bed is available.
b. Both wards are merged into a single ward. What is now the rejection probability?
c. Another ward with the same parameters is merged. What is now the rejection probability?
d. Consider hospital wards with the same load (offered load divided by size), but varying size. What do you think about the signs of the first and second derivative of the rejecting probability as a function of the size? How would you call these properties in economic terms?

Exercise 5.10 A small town has a single ambulance for dealing with all emergencies in the area (24 hours a day, 7 days a week). Research shows that emergency calls arrive according to a Poisson process. Emergency handling time consists roughly of driving time to the site of the accident, handling time on the site, and driving time to the hospital.
a. Define the variables involved in this system.
b. Give a formula for the probability that the ambulance is busy at the moment an emergency call arrives. Did you make any assumptions?
c. Derive an approximation for the expected time between an emergency call and the moment the ambulance arrives at the site of the accident. Did you make any (additional) assumptions?
d. Indicate how you could answer questions b and c for the case of 2 ambulances.

Exercise 5.11 Consider the Engset model $M|M|1|3|3$.
a. Calculate the stationary distribution.
b. Calculate the distribution as it is seen by arriving customers.

Exercise 5.12 A production system consists of three machines in tandem with infinite buffer space. The second process step fails in 20% of the cases. For this reason there is a quality check (taking no time) after the second step that sends all parts for which the second step failed back to the queue at the second step for processing. Service times are assumed to be exponential, with rates that you can choose arbitrarily.
a. Model this production system as a queueing network.
b. For an arbitrary arrival rate, solve the routing equations.
c. What is the maximum production rate of this system?
d. For a Poisson order arrival process at 80% of this maximum, what are the expected waiting and response times?

Chapter 6

Inventory Models

In this chapter we study mathematical models for determining order sizes and moments. Although this usually results in inventory, a better name than the usual *inventory models* would be *order models*. We stick to the standard terminology. Inventory models are often used in distribution and sales environments, but sometimes they can also be used in other application areas: see for example Chapter 14 on revenue management. In this chapter we discuss the basic inventory models. Randomness plays again an important role. Next to that we pay attention to robustness.

6.1 Objectives and notation

The main characteristic of inventory models is demand that can be met by ordering items and keeping them on stock. We make a difference between single-order models and long-term problems for which orders are placed at regular intervals. Single-order models are direct applications of probability theory; long-term inventory models are examples of stochastic processes. The crucial difference with queueing models (see Chapter 5) is that the lead time, the time between the order moment and the delivery, does not depend on the order size. Thus we can assume that items are treated in batches. Capacity restrictions play a minor role. Instead, in inventory models there is a focus on costs.

Costs involve, in the first place, holding costs and sometimes order costs. Holding costs are usual linear in the number of items in inventory and the time that items stay in inventory; order costs can be linear in the order size, but often there is also a fixed component. We will denote the holding costs per item per unit of time with h, the order price per item with k, and the fixed order costs with K. In the case of long-term models it can either be the case that there are regular order moments or that orders can be placed at any time. We denote the stochastic demand in an interval with D, and the demand during the lead time L with D_L. Every item has a selling price p. The distribution function of D will be denoted by F_D, and D_L has distribution function F_L. Usually demand is of a discrete nature, and only an integer number of items can be ordered. Sometimes the goods are of a continuous nature, and often this is a good approximation of the discrete case. For goods of a continuous nature we assume that D has a density f_D and D_L density f_L.

It might occur that there is demand that cannot be met immediately because the inventory is 0. At that moment two things can happen: the item is *backordered*, or the sale is lost. The objective can either be cost minimization under constraints on the fraction of back orders or lost sales, or cost minimization where costs for back orders or lost sales are included. In the former we use α to denote the maximal fraction of lost sales or back orders as part of total demand, in the latter case we denote with q the costs per back order or lost sale.

In the next sections we discuss all possible models with the above characteristics. We start in Section 6.2 with single-order models, the so-called *newsvendor* or *newsboy problem*. Then we discuss a continuous-review deterministic demand model with fixed order costs leading to the central *Economic Order Quantity*. In that section we also discuss periodic-review models. The consecutive section discusses the extensions to stochastic demand. We conclude with a section treating possible extensions of the models.

We summarize the notation. The input parameters of our models are:
- h: holding costs per item per unit of time;
- k: order price per item;
- p: selling price per item;
- K: fixed order costs;
- L: lead time;
- D: demand (if relevant in an interval, usually of length 1);
- λ: the average demand per time unit, $\lambda = \mathbb{E}D$;
- D_L: demand during the lead time L;
- F_D, F_L: distribution function of D, D_L;
- f_D, f_L: density of D, D_L;
- α: maximal fraction of lost sales or back orders;
- q: costs per back order or lost sale (in which case it equals $p - k$).

We assume that all variables are non-negative.

Different types of inventory policies exist. The best known are the (s, S) and (r, Q) policies. The (s, S) policies are used for periodic-review models, the (r, Q) for continuous-review models. The meaning of the letters is as follows:
- s: the inventory level below which or at which an order is placed at an order instant;
- S: the order upto level (that is, the difference between S and the current inventory level is ordered);
- r: the inventory level at which an order is placed the moment it is reached;
- Q: the order quantity.

Example 6.1.1 Let $s = r = 5$, $Q = 20$ and $S = 30$. In the (r, Q) model or order of size 20 is placed the moment only 5 items are left over. In the (s, S) model we wait for the first order moment after the moment the inventory dropped to 5 (these moments have to be specified). Let this level be at 4; then $30 - 4 = 26$ items are ordered. In the (s, Q) model always 20 items are ordered at an order instant at which the current inventory is 5 or lower.

6.2 Single-order model

We discuss in this section the single-order model. We use the notation introduced above: demand D with distribution function F_D, costs q per order that cannot be met, and costs h for each left-over items. We have order costs K and a current inventory of y. S is the inventory after ordering, the *order up to level.* The actual order size is thus $S-y$. Assume $S>y$. Then the total costs $C(S)$ are:

$$C(S) = K + h\mathbb{E}(S-D)^+ + q\mathbb{E}(D-S)^+.$$

We could interpret $h\mathbb{E}(S-D)^+ + q\mathbb{E}(D-S)^+$ as the price we have to pay for the randomness in demand. If D were deterministic and $S=D$, then this term would completely disappear.

In the next theorem we calculate the optimal value of S.

Theorem 6.2.1 (Single-order model)
i) The optimal order size S^* that minimizes total holding and backorder costs in the single-order model in case $K = 0$ and no initial inventory is given by

$$S^* = F_D^{-1}\Big(\frac{q}{q+h}\Big) \tag{6.1}$$

in the case of D continuous and

$$S^* = \arg\min_S \{F_D(S) \geq \frac{q}{q+h}, S \textit{ integer}\}$$

in the case of D discrete.
ii) The optimal order size S^* that minimizes total holding and backorder costs in the single-order model in case $K = 0$ for initial inventory y is given by $(S^* - y)^+$;
iii) If $K > 0$ and $y < S^*$, then it is optimal to order $S^* - y$ if $C(S^*) < C(y) - K$, otherwise no order should be placed;
iv) In the case of minimizing holding costs with the fraction of lost sales below or equal to α the optimal order size S^* is given by the solution of

$$\mathbb{E}(D - S^*)^+ = \alpha\mathbb{E}D$$

in the case of D continuous and

$$S^* = \arg\min_S \{\mathbb{E}(D - S)^+ \leq \alpha\mathbb{E}D, S \textit{ integer}\}$$

in the case of D discrete.

Proof If the inventory after ordering but before sales is equal to S, then the sum of inventory and lost sales costs are equal to $C(S)$. Let us consider first the case that D is continuous, thus f_D exists. Then

$$\frac{d\mathbb{E}(S-D)^+}{dS} = \frac{d}{dS}\int_0^S (S-x)f_D(x)dx = \frac{d}{dS}S\int_0^S f_D(x)dx - \frac{d}{dS}\int_0^S xf_D(x)dx =$$

$$\frac{d}{dS}SF_D(S) - Sf_D(S) = F_D(S).$$

Similarly,

$$\frac{d\mathbb{E}(D-S)^+}{dS} = -(1 - F_D(S)).$$

Differentiating again gives

$$\frac{d^2\mathbb{E}(S-D)^+}{(dS)^2} = \frac{d^2\mathbb{E}(D-S)^+}{(dS)^2} = f_D(S) \geq 0.$$

Thus $C(S)$ is convex, and the global minimum can be obtained by solving $\frac{dC(S)}{dS} = 0$ which leads to $hF_D(S) - q(1 - F_D(S)) = 0$, resulting in (6.1). To arrive at this situation (if at all possible)

$(S^* - y)^+$ items should be ordered. If $K > 0$ then the same order quantity is optimal, but then total costs with ordering $C(S^*)$ should be compared with $C(y) - K$, which corresponds to not ordering.

If D is discrete then $\mathbb{E}(S+1-D)^+ - \mathbb{E}(S-D)^+ = F_D(S)$ for S integer. Again, $C(S) : \mathbb{N} \to \mathbb{R}$ is convex, and $C(S+1) - C(S) = hF_D(S) - q(1 - F_D(S))$. For the convexity it follows that the optimal order quantity S^* is given by $S^* = \arg\min_S\{C(S+1) - C(S) \geq 0\} = \arg\min_S\{F_D(S) \geq q/(q+h)\}$. For point ii) and iii) the same arguments apply as for the continuous case.

Concerning iv) it should be noted that holding costs are increasing in S, and thus S^* should be the minimal value for which the constraint is satisfied. □

Equation (6.1) has a nice interpretation. Consider some S for which

$$hF_D(S) < q(1 - F_D(S)). \tag{6.2}$$

Should we increase S or decrease S? If we increase S by δ, then $F_D(S)$ is the probability that the additional δ are left-over. Thus the marginal left-over costs are $\delta h F_D(S)$. On the other hand, the back-order costs are reduced by ordering more. $1 - F_D(S)$ is the probability that back-orders will occur if the order size is S, thus $-\delta q(1 - F_D(S))$ are the marginal back-order costs. If (6.2) holds then we can reduce costs by increasing S. We do this until we found S^* for which

$$hF_D(S^*) = q(1 - F_D(S^*),$$

which is equivalent to Equation (6.1).

The best-known single-period or single-order model is the *newsvendor* or *newsboy* problem. For the newsvendor there are order costs k per item, a selling price p and *salvage value* v, the amount received for left-over items. When the newsvendor orders S then his profit is given by:

$$P(S) = (p - k)\mathbb{E}\min\{D, S\} + (v - k)\mathbb{E}(S - D)^+.$$

Now take $q = p - k$, the profit per item sold, and $h = k - v$, the amount paid for left-over items. Then $P(S) = q[\mathbb{E}D - \mathbb{E}(D - S)^+] - h\mathbb{E}(S - D)^+ = q\mathbb{E}D - C(S)$. Maximizing $P(S)$ corresponds to minimizing $C(S)$. Thus Theorem 6.2.1 applies also to this model, and the optimal order level is given by

$$S^* = F_D^{-1}\Big(\frac{p-k}{p-v}\Big).$$

6.3 Multi-order deterministic-demand models

Now we continue with multi-order models, starting with the deterministic-demand continuous-review model. We also discuss the differences with the periodic-review model. In the next section we deal with models with stochastic demand.

The most famous result from inventory theory is the *Economic Order Quantity* (EOQ). One of the reasons for its popularity is its simplicity. In the standard version it concerns a deterministic-demand continuous-time continuous-product model without lost sales or back orders and with fixed order costs $K > 0$, where the objective is to minimize the long-run average sum of order and inventory costs. Note that because shortselling is not allowed there is no need for parameters related to backordering, purchasing, and selling: all demand needs to be satisfied immediately. Thus the only relevant parameters are λ, h, and K.

Note the difference between physical and economic inventory: economic inventory includes orders that are not yet delivered.

Theorem 6.3.1 (EOQ model) *In the deterministic-demand model with continuous review and lead time L the optimal policy orders a quantity Q^* (the so-called* Economic Order Quantity*) given by*

$$Q^* = \sqrt{\frac{2K\lambda}{h}} \tag{6.3}$$

at (economic) inventory level λL; the total average inventory and order costs are given by $C(Q^*) = \sqrt{2K\lambda h}$.

Proof Due to the lack of randomness in the model the optimal order point is 0, there is no need for *safety stock*. The parameters do not change, thus the order quantity Q is equal for each order cycle. We study the average costs over a single cycle, which is, by renewal theory (see Section 3.3), equal to the long-term average costs. Denote with T the time between two consecutive orders. Then $T = Q/\lambda$.

Denote the average costs for order quantity Q with $C(Q)$. It consists of order costs K/T and inventory costs. The inventory level decreases linearly from Q to 0, therefore the average inventory level is $Q/2$, and the average inventory costs $hQ/2$. Thus

$$C(Q) = \frac{K}{T} + \frac{hQ}{2} = \frac{K\lambda}{Q} + \frac{hQ}{2}.$$

We readily see that C is convex, and that $\lim_{Q\downarrow 0} C(Q) = \lim_{Q\to\infty} C(Q) = \infty$. Therefore $\frac{d}{dQ}C(Q) = 0$ has Q^* as solution which leads to Equation (6.3).

When we order when there are $L\lambda$ items on stock, then the order arrives when the inventory reaches 0. When $L > Q^*/\lambda$, then we should consider the economic inventory, because there is still an order underway. □

The model of Theorem 6.3.1, to which we shall refer as the EOQ-model, has many interesting consequences. We discuss a number of them.

Economies of scale The inventory that is held in the EOQ model is called *cycle stock*. When $K = 0$ then ordering infinitely often would be optimal, and no stock would he held at all. Thus cycle stock exists because of the economies of scale due to ordering large quantities.

It is interesting to calculate the inventory costs per item. This is for example useful when determining the price of a product. The minimal inventory costs per item are given by $C(Q^*)/\lambda = \sqrt{2Kh/\lambda}$. This is a decreasing function of λ, thus we see economies of scale: if λ is big then we can frequently order big quantities, making order and inventory costs per item go to 0.

Robustness Let us study the robustness under changes of the parameters, first for the case $L = 0$. Assume for example that we underestimated λ by a factor 2, i.e., we took as order size $Q' = \sqrt{K\lambda/h}$ instead of $Q^* = \sqrt{2K\lambda/h}$. Then we find as relative error:

$$\frac{C(Q') - C(Q^*)}{C(Q^*)} = \frac{\frac{K\lambda}{\sqrt{K\lambda/h}} + \frac{h\sqrt{K\lambda/h}}{2} - \sqrt{2K\lambda h}}{\sqrt{2K\lambda h}} = \frac{3}{2\sqrt{2}} - 1 \approx 0.06.$$

The same result holds if we overestimate λ by a factor 2, thus if we take order size $Q' = \sqrt{4K\lambda/h}$:

$$\frac{C(Q') - C(Q^*)}{C(Q^*)} = \frac{\frac{K\lambda}{2\sqrt{K\lambda/h}} + h\sqrt{K\lambda/h} - \sqrt{2K\lambda h}}{\sqrt{2K\lambda h}} = \frac{3}{2\sqrt{2}} - 1 \approx 0.06.$$

From the form of the formula it is readily seen that the same result also holds for K and h. Thus we come to the surprising conclusion that if make an error of less than a factor 2 in estimating any of the input parameters λ, K, or h then the relative error does not exceed 6%.

Now consider the case that more than one parameter changes. The effects might cancel out, but in the worst case the costs increase quickly: if λ and K are both twice as high as foreseen then the error is 25%.

If $L > 0$ then the result for variations in K and h remain the same. However, when λ is over or underestimated it can have dramatic consequences. Overestimating λ in this case leads not only to a wrong choice of Q, but also to orders arriving when there is still stock left. Underestimating λ leads to lost sales and/or backorders. To be more precise, when λ is underestimated by a number λ_d then there are no items to meet the demand for a total of $\lambda_d L$ items per order cycle. Thus a fraction $\lambda_d L/Q^*$ of the demand is not met. On the other hand, if λ is overestimated by a number λ_d then the order arrives when there is still $\lambda_d L$ inventory. If no measures are taken then the average costs are increased by $\lambda_d L h$.

Periodic models Variations in order lead time are also interesting to study, because of the connection with periodic models. Define $\hat{C}(T) = C(T\lambda) = C(Q)$, i.e., $\hat{C}$ are the costs as a function of the lead time T instead of the order size Q. Consider $T^* = Q^*/\lambda$. First note that in general, for $\alpha > 0$, $\hat{C}(\alpha T^*) = \frac{1}{2}(\alpha + \frac{1}{\alpha})\hat{C}(T^*)$. It follows that $\hat{C}(\frac{1}{\sqrt{2}}T^*) = \hat{C}(\sqrt{2}T^*)$ and that

$$\frac{\hat{C}(\frac{1}{\sqrt{2}}T^*) - \hat{C}(T^*)}{\hat{C}(T^*)} = \frac{\hat{C}(\sqrt{2}T^*) - \hat{C}(T^*)}{\hat{C}(T^*)} = \frac{1}{2}(\sqrt{2} + \frac{1}{\sqrt{2}}) - 1 \approx 0.06.$$

We see that choosing a lead time T within the interval $[\frac{1}{\sqrt{2}}T^*, \sqrt{2}T^*]$ can only increase costs by 6%. This can be very convenient. Assume for example that the required order lead time in a certain situation is 10 days, but that we prefer to order once a week or once every two weeks. The 6%-interval is $[7.07, 14.14]$, thus by ordering every week we increase the costs by a little over 6% and by ordering every two weeks we increase them by a little less than 6%. This way of reasoning is known in the literature as the *powers of 2* solution, because the intervals considered have the form $[T, 2T]$. Note that the result can also be formulated in terms of Q: its 6%-interval is given by $[\frac{1}{\sqrt{2}}Q^*, \sqrt{2}Q^*]$.

Backorders Backorders or lost sales are usually a consequence of stochastic demand: inventory costs would be too high if the probability of backorders or lost sales had to be reduced to (almost) 0. However, also in the deterministic EOQ model it can be advantageous to have a small fraction of back orders. See Exercise 6.5 for an example with a fixed order size.

Remark 6.3.2 (discrete demand) So far we assumed that products are indivisible and that demand is linear. Even if products are discrete (and demand is a step function) Q^* is often a very good approximation, especially if Q^* is big. If we want to model the discrete nature of the products explicitly, then we have to decide when the order is placed: at the moment inventory becomes 0, or $1/\lambda$ time units later when the next demand is placed? The difference between the two is having, on average, half a unit of product more or less in stock compared to the continuous demand model. The optimal order size remains equal; however, it should be rounded to an integer. As long as this integer falls within the interval $[\frac{1}{\sqrt{2}}Q^*, \sqrt{2}Q^*]$, then the error is limited to 6%.

Remark 6.3.3 (production model) The main difference between queueing and inventory models is that in the former capacity plays a major role, and lot sizes in the latter. Let us extend the EOQ-model to deal not only with lot sizes but also with limited capacity, by introducing a production rate p, with $p > \lambda$. If the order size is Q, then production occurs during the first Q/p time periods. The highest inventory position is reached when production stops, with level $Q(p - \lambda)/p$. This leads to average costs

$$\tilde{C}(Q) = \frac{K\lambda}{Q} + \frac{hQ(p-\lambda)}{2p}.$$

Similar arguments as in the proof of Theorem 6.3.1 lead to the optimal order quantity $\tilde{Q}^*$:

$$\tilde{Q}^* = \sqrt{\frac{2pK\lambda}{(p-\lambda)h}}.$$

Note that $\tilde{Q}^* > Q^*$.

6.4 Multi-order stochastic-demand models

In this section we assume, in contrast with the previous section, that the demand is stochastic and also stationary. If the order lead times were 0, then there would be no reason to change the re-order policy: if the inventory becomes 0, then an order can be placed immediately. Of course the time between orders and the costs become random variables: therefore we take the average expected costs as criterion. There are no crucial differences with the deterministic-demand model: ordering the EOQ when inventory is 0 is still optimal. It is not that simple anymore if we assume a non-zero order lead time L. In this case demand might occur while the inventory level is already 0. In this section we first assume that this demand is back ordered to the moment that the next order arrives. We consider both the cases where there is a constraint on the fraction of back orders and where the costs of back orders are part of the overall cost function. Both situation will result in ordering such that the expected stock level is non-zero when the order arrives. This stock is called the *safety stock*.

We assume that during the lead time L the demand D_L has the distribution function F_L with expectation λL. In the deterministic model of the previous section the order has to be placed when the inventory level is at λL. We call this the order level r. For the current stochastic model r is not necessarily equal to λL; often we take r higher to avoid back orders, and sometimes it can be optimal to take r smaller than λL, for example if back ordering is cheaper than keeping safety stock.

Theorem 6.4.1 *In the stochastic-demand model with continuous review and lead time L the policy that minimizes order and inventory costs under a constraint on the fraction of backorders orders or lost sales a quantity Q^* at (economic) inventory level r^* approximated by*

$$Q^* \approx \sqrt{\frac{2K\lambda}{h}} \text{ and } b(r^*) \approx \alpha Q^*$$

with α the maximal fraction of backorders or lost sales and $b(r) = \mathbb{E}(D_L - r)^+$ the number of backorders during L with initial inventory r;
The policy that minimizes order, inventory and backorder costs orders a quantity Q^* at (economic) inventory level r^* approximated by

$$Q^* \approx \sqrt{\frac{2\lambda(K + qb(r^*))}{h}} \text{ and } r^* \approx F_L^{-1}\Big(1 - \frac{2hQ^*}{2q\lambda + hQ^*}\Big);$$

The policy that minimizes order, inventory and lost sales costs orders a quantity Q^* at (economic) inventory level r^* approximated by

$$Q^* \approx \sqrt{\frac{2\lambda(K + qb(r^*))}{h}} \text{ and } r^* \approx F_L^{-1}\Big(1 - \frac{hQ^*}{q\lambda + hQ^*}\Big).$$

Proof Let us quantify the different aspects of the system with backorders, for a policy that orders Q units if the inventory level reaches r. The expected length of the time between two deliveries remains Q/λ. Therefore the fixed order costs are again $K\lambda/Q$. The expected number of back ordered items per cycle is denoted as $b(r)$. It can be derived from F_L: $b(r) = \int_r^\infty (x - r)dF_L(x)$. Determining the average inventory in the system is more complicated. First we make a distinction between the physical inventory, which is the inventory actually at stock, and the inventory level including back orders, which can therefore be negative. A good approximation of the average physical inventory is the average of the physical inventory at the beginning and the end of the cycle. At the end of the cycle this is $r - \lambda L + b(r)$. At the beginning of the cycle this is $r - \lambda L + Q$. (We assume that this quantity is positive.) Thus we approximate the average positive inventory with $\frac{1}{2}(r - \lambda L + Q + r - \lambda L + b(r)) = r - \lambda L + (Q + b(r))/2$.

We start with the service level formulation. We first consider the system where we minimize the sum of order and holding costs $C(r, Q)$ under the service level restriction that the probability that an arbitrary unit is back ordered is not bigger than α. This gives as minimization problem: $\min_{r,Q}\{C(r, Q)|b(r)/Q \le \alpha\}$, with $C(r, Q) = K\lambda/Q + h(r - \lambda L + (Q + b(r))/2)$. In general, this problem is hard to solve. However, if α is small, then $Q^* = \sqrt{2K\lambda/h}$, the EOQ, is a good approximation; r^* should be chosen such that $b(r^*) = \alpha Q^*$. The interpretation of the average inventory is straightforward: $Q^*/2$ is the cycle stock, as for the deterministic model, and $r^* - \lambda L$ is the safety stock that is present to avoid too much back orders. Note that safety stock can be negative; this can easily be seen if demand is deterministic and $\alpha > 0$.

Next we consider the single objective formulation with backorders. Now consider the case where the additional costs for each back ordered item are equal to q. Then the total costs can be approximated as follows:

$$C(r, Q) \approx \frac{K\lambda}{Q} + h\Big(r - \lambda L + \frac{Q}{2}\Big) + \Big(\frac{q\lambda}{Q} + \frac{h}{2}\Big)b(r).$$

Differentiating to r and Q gives as minimal values r^* and Q^*:

$$Q^* = \sqrt{\frac{2\lambda(K + qb(r^*))}{h}},\ r^* = F_L^{-1}\Big(1 - \frac{2hQ^*}{2q\lambda + hQ^*}\Big).$$

When compared to the EOQ, we see that K is replaced by $K + qb(r^*)$ in the expression of Q^*. Thus the fixed order costs K are augmented with the back order costs per cycle. Again, these optimal values are not simple to calculate. However, an iterative scheme, starting with Q^* approximated by the EOQ, converges fast to the optimal solution.

Lost sales can be modeled similarly: q are now the costs of lost sales. The main difference in the approximation is the level after the order arrival: as there are no back orders to fulfill this is on average equal to $r - \lambda L + b(r) + Q$. Therefore

$$r^* = F_L^{-1}\Big(1 - \frac{hQ^*}{q\lambda + hQ^*}\Big).$$

Another difference is the average cycle length, which becomes $(Q + b(r))\lambda$. Modeling this would complicate the solution considerably. □

Periodic review with $K > 0$ As for the deterministic models, we can consider periodic review models. The principle change is that the safety stock is not only used to bridge the order lead time, but also the order period. Thus F_L has to be replaced by F_{L+T}, where T represents the remaining time until the next order moment. This gives a new level s instead of r, such that as soon as the inventory drops below s then an order should be placed at the next order moment. It is optimal not to order a fixed quantity Q, but up to a level S. Such a policy is called an (s, S) policy.

Periodic review with $K = 0$ Periodic models with $K = 0$, backorders, and lead time L shorter than the period (assumed to be 1) are equivalent to single-period models. Consider an order moment. Then the order after this order arrives at $1 + L$ from now, and there is no reason to order now for demand occuring after $1 + L$, because it is free to order at 1. Theorem 6.3.1 can be used with F the distribution function of the demand during $1 + L$.

6.5 Multi-stage and multi-item models

The sections above give an overview of some of the most important results and concepts in single-stage single-item models.

Often items are stored at different levels. In these cases order policies depend on the inventory at all stages, not just the next stage. Understandably, optimal order policies become very complicated. A good practice is to base decisions on the total downstream stock, the so-called *echelon stock*.

Models with multiple items are also interesting. By combining orders cost reductions can be achieved. Again, optimal policies are very hard to calculate. Often there are two re-order levels for each item. When the lowest level is reached items are ordered. When an order for a certain item is placed, then all items for which there is less at stock than the higher level are ordered as well.

6.6 Further reading

An excellent source for inventory models is Zipkin [104].

The Chapters 1, 2, and 4 of O.R. Handbook 4 Graves et al. [43] deal with subjects related to the ones discussed in this chapter. The same holds for Chapter 12 of Handbook 2 Heyman & Sobel [47], which considers stochastic inventory theory.

Many books on production or logistics consider also inventory models, such as Hax & Candea [46] and Bramel & Simchi-Levi [16].

When it comes to inventory systems, most of the above references deal with periodic models. In this chapter we decided to give a central role to continuous review models, and to deal with periodic models through the powers of 2 approximation. Results on stochastic continuous review models can be found in Johnson & Montgomery [49] and in Chapter 1 of Graves et al. [43].

For multi-stage models we refer to Chapter 2 and 4 of Graves et al. [43].

6.7 Exercises

Exercise 6.1 Consider a newsboy problem with a demand that is normally distributed with expectation $\mu = 20$ and variance σ^2.

a. Find the optimal order quantity for the parameter values $p = 1$, $v = 0.1$, and $k = 0.5$ for variance 1 and 2.

b. Find the optimal order quantity for the parameter values $p = 1$, $v = 0.1$, and $k = 0.6$ for variance 1 and 2.
c. Calculate the costs for each of the four situations that we considered.
d. Give an intuitive explanation of the results.

Exercise 6.2 An agricultural firm harvests K kilograms of a certain product. The company has two ways to sell their product: to a supermarket at a price p_r per item or at a market at a price p_m. The supermarket will buy all the firm is willing to sell them, the demand at the market D is random. Leftover products are worthless.
a. Formulate your expected income as a function of the amount of product that you sell to the supermarket.
b. Give the policy that maximizes your expected income.
c. Calculate the policy for $K = 1000$, $p_r = 0.9$, $p_m = 1.0$, and D is normally distributed with expectation 1100 and standard deviation 300.
d. The management is not only interested in maximizing expected income, but is also risk-averse. What should management do in your opinion? Explain yourself using heuristic arguments.

Exercise 6.3 Consider a continuous-time multi-order deterministic-demand continuous-product inventory model with $\lambda = 5$, $K = 10$, $h = 1$ and $L = 1$.
a. Compute the optimal re-order level and re-order size.
Now demand is stochastic; it occurs according to a Poisson process with rate $\lambda = 5$. For the rest the system is the same. Items that are not available are backordered.
b. We use the same re-order policy. Estimate the probability that backorders occur in a cycle.
c. It is the objective to avoid backorders in at least 9 out of 10 cycles. How should we choose the re-order policy to achieve this?

Exercise 6.4 It is stated for the EOQ model at page 79 that the error in costs is 25% if λ and K are both twice as high as foreseen. Show this.

Exercise 6.5 A person receives a monthly salary S on a bank account. Each month is assumed to have 30 days. From this bank account she pays each day her daily expenses d. We assume that $30d < S$. She has the option to put money on a savings account. The savings account has a daily interest rate r, there is no interest rate on the account where her salary arrives, unless the amount is negative: then she pays an interest rate of p, $p > r$. Interest is payed at the end of the month (the same day the salary arrives). The day the salary arrives she decides how much money to put on the savings account and the money is imddediately transfered. The first month we start with 0 on the savings account and S on the other account.
a. Model this problem as an inventory model: classify the model and determine the necessary parameters.
b. Calculate the amount to put on the savings account that maximizes the interest at the

end of every month. You need not do this exactly, a good approximation is fine. This amount is not equal for every month!
c. Give the definitions of safety stock, cycle stock, and seasonal stock.
d. What type of stock is the money on the standard account?
She uses the money on the savings account to pay for her summer holidays.
e. What type of stock is the money on the savings account?

Exercise 6.6 Consider an inventory model with Poisson(10) demand, lead time 1, $K = 100$, $h = 1$, and maximal 5% backorders. Estimate Q^* and r^*.

Exercise 6.7 A shop sells goods. When ordered at the beginning of day n, the ordered goods arrive at the beginning of day $n+1$, and they can be sold from that day on. A unit of goods costs p, and is sold for r ($r > p$). Every night that a unit spends in stock costs h. There are no order costs, orders are therefore placed every day.
a. Model this as an inventory model, by introducing random variables for demand, stock, and order sizes. Give the relations between the variables.
b. Express the expected sales at day $n + 1$ and the expected inventory costs at the end of day $n + 1$ as a function of the stock at the beginning of day n (including the arriving order) and the order placed at the beginning of day n.
c. Suppose that the demand on every day are uniformly distributed on $[0, 1]$. Calculate the expected sales at day $n+1$ and the inventory costs of the stock at the end of day $n+1$, given the stock at the beginning of day n (including the order that just arrived), and the order placed at the beginning of day n.
d. Let R be the order policy that miximizes for each day n expected profit minus inventory costs at day $n + 1$. Do you think that this order policy maximizes the average expected profit minus inventory costs? Motivate your answer!
The same shop also sells perishable goods. When ordered at the beginning of day n, the ordered goods arrive at the beginning of day $n+1$, and they can be sold during days $n+1$ and $n + 2$. After that they are thrown away, without cost nor reward. A unit of goods costs p, and is sold for r. There are no order costs.
e. Model this as an inventory model, by introducing random variables for demand, stock, and order sizes. Give the relations between the variables.

Exercise 6.8 Consider a company with 10 outlets. Demand of a certain products occurs at each outlet according to independent Poisson processes, with average 10 each day at each outlets. Orders can be placed once a week, replenishments are immediate (they are done overnight). Only 1% of lost sales are accepted.
a. How many items should each outlet have in stock after each replenishment?
Consider a situation where all deliveries are done from a central location, which is like having one outlet with a daily expected demand of 100.
b. How many items should be in inventory in this situation?
Consider finally the more realistic situation where there is a central warehouse that delivers to the outlets every day. These replenishments occur overnight: what is ordered at the

end of the day can be sold at the beginning of the next day. The central warehouse is replenished every week. Thus the outlets should only keep one day of stock, the warehouse for one week.
c. Give an approximation for the stock at both location such that the solution under a is both outperformed in stock costs and order reliability.
(Of course transportation costs can be higher: ten weekly long-distance shipments are replaced by one weekly long-distance and daily short-distance shipments.)

Exercise 6.9 A distribution system consists of 10 outlets and 1 DC. Outlets order in lots of 10, on average once a week (a week consists of 5 working days). Demand from the outlets is approximately Poisson. Delivery is immediate. The order lead time to the DC is 1 week, and is done with an order quantity of 400.
a. What should be the safety stock at the DC to have less than 1% backorders?
The outlets change policy: they order with a lot size of one, but with the same average total demand.
b. What should, in this case, be the safety stock at the DC to have less than 1% backorders?
It is determined that the optimal lot size is 5 for orders from the outlets at the DC.
c. Propose a way to manage the inventory at the DC such that the safety stock is as low as possible.

Part II

Modeling

Chapter 7

The Modeling Process

Modeling is a word used in many contexts and sciences. In this monograph we deal with *mathematical* modeling, the process of solving real-world problems using mathematical techniques. This process involves much more than just mathematics. In this chapter we try to lay a theoretical foundation of the whole problem-solving process.

Before going into the details, let us pose ourselves the question what we expect to learn from this chapter. Indeed, many people state that, while solving mathematical models is a science, modeling is an art! With that they mean to say there is no theoretical foundation to modeling, as there is to model solving, but that good modeling needs a combination of talent and experience. However, in the current chapter we try to convince you that there are some general rules to learn and pitfalls to be avoided. It remains true however that practical experience is indispensable for good modeling.

7.1 Introduction and definitions

Managers solve problems. These problems can be of many different natures, with human aspects, organizational aspects, etc. Some of the problems have quantitative aspects that go beyond simple calculations. These lecture notes are about this type of problems. Rarely a problem with quantitative aspects can be solved taking all its quantitative and non-quantitative aspects into account. Therefore a *model* of reality needs to be constructed first.

A model is a (simplified) description of a real-world process or phenomenon. This is often a written description, but it can also be a physical construction, of for example a building. The object which is to be modeled need not exist (yet). To avoid confusion we use the neutral term *system* to designate the object of our study. When we want to stress the dynamic nature of the system we also use the term *process*. We see a process as a related group of activities with a common goal.

Example 7.1.1 A machine or a group of machines is an example of a (production) system. If the main interest is not the group of machines but the production or transformation of goods then we speak of a production process.

With the *environment* of a system we mean all other systems and processes with which the system can interact. The formal definition of a model that we use is as follows.

Definition 7.1.2 *A* model *is a description of a part of a system or process and its interaction with its environment that allows an analysis of certain aspects of that system or process.*

The extent to which system details are taken into account and the choice of the details depend on the objectives of the modeling phase. Differences in depth of the analysis will be discussed later. Here we stress that the model depends also on its (possible) use in the organization. This is reflected in the definition by the fact that a model is such that it allows an analysis of aspects that are of interest to the model builders.

Example 7.1.3 A model of a data communication line used for throughput estimations could abstract from cell losses. This model would evidently be useless for estimating losses.

The term modeling in a narrow sense is just the process of constructing the model. However, usually we designate by modeling the whole method of solving (business) problems using models.

Definition 7.1.4 Modeling *is a methodology for problem solving in which the use of models plays a crucial role.*

We realize that the word ‘problem’ has a somewhat negative connotation, but we use it because we think that it describes the concept best. Alternatives are ‘challenge’ and ‘question’.

A *mathematical model* is a model in which the relations within the model are given in mathematical terms. These notes deal with mathematical modeling, but some of the ideas are also applicable to models in a more general context.

The next section describes the modeling process by breaking it up into several steps.

7.2 Steps in the modeling process

In a typical modeling project we can distinguish several phases. They are shown in the figure below. The phases correspond to arrows, the starting situation and the products of the phases are represented by ellipses.

Starting from a system and a research question or problem we begin with the *model construction phase*. The result of this phase is a model. This step is qualitative in the sense that relations between quantities are given. An important aspect of the model construction phase are decisions concerning which details to model and which to leave out.

The resulting model is solved using one of the mathematical techniques to be discussed later in Part I. However, finding the right solution technique and executing it is not all that happens in the *model solving phase*. To execute the model data has to be available.

The necessary data collection and analysis is an important and time consuming part of this phase.

The solution to the model solving phase does not directly give us the solution to the system problem. Indeed, as the model is a simplified representation of the system under study, translating the model solution back to a system solution is not always easy, and sometimes not even possible. Making this possible is one of the main concerns when modeling (the other ones being the availability of relevant data and the possibility to solve the model). For a well chosen model this translation is not that difficult, most of the time of the reporting phase goes into convincing the problem owners of the correctness and feasibility of the proposed solution.

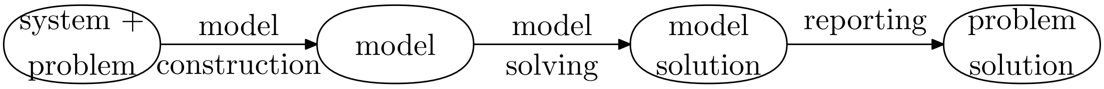

Figure 7.1: The modeling process.

We assume that the objective is part of the model, therefore the objective is not noted apart. This is for two reasons: it is common in the literature on solution techniques that the objective is part of the model, and it shows well that, while modeling, the objective should be taken into account.

From Figure 7.1 we might well get the impression that modeling is a linear process, i.e., it finishes after having dealt with the various steps consecutively. This is a wrong impression. In any stage of the modeling process there is feedback possible, the most important one is from the system solution back to the modeling phase. This is for example the case if the system solution is not implementable due to reasons that were abstracted from in the modeling phase. Another example of feedback is from the model solving phase to the modeling phase, if it is discovered that the model is too hard to solve and thus needs simplification.

The implementation can start as soon as we have a satisfactory outcome of the modeling phase. Often it is done first on a small scale. This allows the problem owners to gain confidence in the model outcomes, and effects that were not modeled can be studied on a system level. Feedback to the different phases of the modeling process occurs again.

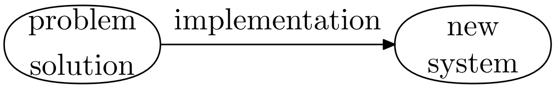

Figure 7.2: The implementation.

Note that modeling and implementation are usually done by different people. Modeling is often done by consultants specialized in modeling, the implementation is done by those who manage the systems and processes involved.

Example 7.2.1 In a large retail organization management is considering to change the distribution policy, that comes down to reallocating certain activities in the supply chain. The model contains all activities in the supply chain from the distribution center to the store. Due to the complexity of the model it was decided to use simulation as solution technique. Data collection was a major issue: certain handling times had to be measured on the spot! When the final results of the modeling process were presented to a large group of managers (only after several iterations of improvements based on feedback from a smaller group of logistics managers), they were considered to be counter-intuitive. It was decided that the proposed policy was to be tested first on a small but representative groups of products. Based on this a decision concerning all products was to be taken.

Remark 7.2.2 Improving business processes, through modeling or otherwise, involves more than solving a single problem once. Nowadays businesses improve their processes continuously. Our modeling process can be seen as steps of iterative problem-solving programs such as the *Deming cycle* or *Six Sigma*'s *DMAIC*. The Deming cycle (”Plan-Do-Check-Act”, PDCA) is an iterative problem-solving method developed by W.E. Deming who was a statistician working on quality control. Six Sigma is a statistical improvement method originally developed at Motorola. DMAIC stands for Define-Measure-Analyse-Improve-Control, a variant of PDCA.

7.3 Business problems

So far we gave definitions of a model and of modeling, and we discussed the various steps of a modeling project. Here we describe some general aspects of business problems and we classify the different types of problems that we encounter. We will see for which type of problems modeling can help us find a solution. In the next section we take a closer look at the structure of a mathematical model.

To formulate problems such that they are amenable for analysis and that different solutions can be compared we need to make the process under consideration *measurable*, i.e., we need to find ways to *quantify* the aspects of the process that are important. These numbers are called *(key) performance indicators* (KPI's), and are often easily translated into a model. One might even say that KPI's themselves are models of the company's performance. KPI's should be formulated with care and their strict use can sometimes lead to unwanted situations.

Example 7.3.1 A call center wants to give friendly service with a short waiting time. Both can be quantified: the friendliness is measured by asking customers to rate it on a scale of 1 to 5, the KPI for waiting time is the percentage of callers that wait less than 20 seconds. Strictly following this latter objective means that callers who have waited more than 20 seconds should be ignored and left waiting ”forever” (of course, they abandon after some time). This is in contrast with the general objective of a short waiting time. However, call centers are sometimes tempted to use this policy, certainly if the payment they receive is a function of the waiting time KPI.

Example 7.3.2 Similar issues can occur in health care. A striking example, where adhering to service level came in the place of quality health care, has been reported in a British hospital: ”In an attempt to meet the target four-hour Accident & Emergency waiting time, patients were sometimes ”dumped” in a ward without nursing care” [1].

We give a number of ways to classify business problems.

Programmed and unprogrammed problems The most important classification is that between programmed and unprogrammed problems (also called structured and unstructured). Problems are called programmed if "they are repetitive and routine, to the extent that a definite procedure has been worked out for handling them" (Simon [84], p. 46). Note that this procedure can be some highly suboptimal heuristical rule which has nothing to do with modeling. Problems which are not programmed are called unprogrammed. Of course there is a whole continuum between programmed and unprogrammed problems. Modeling can play a role in the solution of both programmed and unprogrammed problems, but its impact on the final solution is often bigger for programmed problems. There are several reasons for that, among which are the fact that for repetitive problems data is often available, and that programmed problems are usually easier to model.

Strategic, tactical and operational decisions A second way to classify problems is by its level of management decision. According to Anthony [9], there are three levels of decisions:

- strategic decisions, dealing with the determination of long-range goals and the means to achieve these;
- managerial or tactical decisions, concerning the realization of the long-term goals and the management of the resources;
- operational decisions, dealing with the short-time planning.

Of course, top management is concerned with strategic decisions, and low level management or the production employees themselves are concerned with the short-term planning. We see that the lower the decision level, the shorter the horizon over which the decision takes effect, and thus low level decisions are more often repetitive. Therefore problems at an operational level lend themselves better to a modeling approach, but sometimes strategic or tactical decisions can also be supported by mathematical models. "On the average, the decisions that the president and vice-president face are less programmed than those faced by the factory department head or the factory manager" (Simon [84], p. 31).

Internal and external coordination We can also classify problems by the fact whether they have only internal or also external aspects. Thus problems that are concerned with *external coordination* are related to the way the firm should react to changes from the outside. Examples are the behavior of competitors (new products or prices), changes to the workforce due to union and government decisions, etc. *Internal* coordination is concerned with problems that exist totally within the interior of the firm. External coordination problems are very often at the strategic level, internal problems at the tactical/operational level. Sometimes this is even used as a definition of strategic and tactical/operational.

Classification by objective Problems can also be classified by the goal which is to be achieved. This objective can range from an educated guess to the solution of a certain problem to a computer system that takes automatically and independently its decisions. Even if the goal is to find an optimal decision, it can be better to build a tool able to answer "what if" questions by which the managers involved can find the solution by experimentation. The goal should be taken into account when modeling. It is clear that a modeling approach is more successful in situations where a detailed solution is wanted.

Example 7.3.3 To fulfill a bottleneck analysis in a production system a simple spreadsheet model may suffice. To produce an optimal production schedule one often has to solve a complicated model.

Design and control problems A final distinction is between *design* and *control* problems. Design problems are those related to setting up a system or process, control problems are those dealing with operating systems or processes. Design problems are often at the strategic or tactical level, control problems are often operational.

Example 7.3.4 In a distribution setting, choosing the location of warehouses is a design problem at the strategic level. Selecting the daily routes from the warehouse to the customers is a control problem at the operational level. Setting up the information and other systems to be able to select these routes in an efficient way involves decisions at the tactical level.

Planning and scheduling What are called control problems above can be split up in *planning* and *scheduling* problems. Planning is concerned with the long-term control issues, often at the tactical level. Scheduling deals with the operational short-term control. Sometimes the word control is also used to indicate the activity consisting of checking whether the foreseen plans or schedules are met and taking the appropriate actions when necessary.

Example 7.3.5 In Part III we make the distinction between production planning and production control. Production planning deals with building production plans taking into account various constraints. Production scheduling deals with the actions that allow the production system to keep to the production plans.

7.4 General model structure

In this section we describe the general form of models. While discussing modeling phases we already stated that the system problem, translated to the model level, is part of the model. Thus the model needs to be *solved*. Models are often identified by their solution techniques: e.g., models with a linear objective and linear constraints are often called linear programming models.

The solution to the model is the output of the model solving phase. The input is the model itself. This input can be split in two: the form of the mathematical description and the parameter values. It is important to make this distinction: the form of the input determines the solution techniques, and if the problem is repetitive, then only the parameter values change. Having in mind problems with this repetitive nature, it is logical to think of a model as only comprising the mathematical rules specifying the relations, and to see the actual parameter values as input. The same distinction shows the two driving forces behind modeling as a way to solve business problems. The first is the availability of enough computational power on the desktop to solve complicated models. Powerful standard software packages for models such as linear programming make this computational power easily accessible. The second driving force is the availability of enough relevant data concerning the problem at hand. Indeed, companies nowadays gather enormous amounts of data, all having the potential to be turned into useful information for improving business processes. Seen as such modeling really turns data into information! Data collection is the subject of the next section.

The output can have different forms. Sometimes it is just a number, which could give the answer whether targets set by the management will be met or not. It can also be a complete policy specifying what to do in all possible situations. The last type of output is often on-line, meaning that (using some computer system) at any moment in time, given the current situation, the optimal control action can be determined.

Example 7.4.1 A model of a service center can well have as output the expected waiting time before service for arriving customers. A similar model where the number of servers can be controlled could have as output rules such as "if the number of waiting customers is higher than n, call for assistance". In call centers this is often implemented on-line, for example using a text display.

Every system or process has one or more objectives or goals and utilizes resources to achieve these goals. Thus modeling (and business problems in general) is always a balance between service (the extent to which goals are met) and costs for the use of resources, or, stated differently, between product and production costs. We quantify the quality of service with the *service level*. Often the service level is of a statistical nature, a certain fraction of the products should satisfy a strict quality constraint. This is because often no guarantees about all products or to all customers can be given. The typical objective for a model with costs and service level as separate entities is to minimize costs under a service level constraint or vice versa.

Example 7.4.2 In a call center with a Poisson arrival process queueing can always occur, no matter how many agents there are. Service level constraints are therefore of the form: a fraction x of all calls should be answered within s seconds.

OR professionals tend to integrate both service level and resource utilization in a single objective. The reason for this is that it becomes possible to optimize the system to this objective. However, from a practical point of view it is often better not to integrate service and costs, for several reasons. In the first place it is difficult to choose good weighing factors, necessary for the construction of the single objective. These weighing factors represent how much management is willing to spend to increase the service level with one unit. Of course this is very hard to quantify, if at all possible. In the second place, even if it would be possible to determine these weighing factors, it is often not desirable. This is because often we want to keep the service level explicit, for example for marketing reasons. Solutions for which one of the objectives cannot be improved without decreasing the other are said to lie on the *efficiency frontier*.

Example 7.4.3 The general service manager of a large company selling copy machines described the goal of the maintenance activities as giving the customer "a good service for a reasonable price". Although the value of service is hard to quantify, the manager had a good idea of what he found acceptable for which price. A queueing model helped making the trade-off between service level and labor costs by estimating the time between a maintenance request call and the arrival of a technician, as a function of the number of service personnel.

Example 7.4.4 The standard linear programming example is the determination of the optimal production mix. Suppose a plant can produce M products, each product of type m that is produced can be sold for a price p_m. There are N resources. Of resource n there is c_n available, and product m demands a_{mn} of resource n. The product mix that maximizes profit can be obtained by solving

$$\max\left\{\sum_{m=1}^{M} p_m x_m \;\middle|\; \begin{array}{ll} \sum_{m=1}^{M} a_{mn} x_m \le c_n, & n = 1, \dots, N \\ x_m \ge 0, & m = 1, \dots, M \end{array}\right\},$$

where x_m is the production level for product m.

Here the service level is translated into profit, and costs for resource utilization are translated into upper bounds for the amount of available resources.

Example 7.4.5 A bank uses call centers as its main means to communicate with its customers. For marketing reasons the waiting times of calls must be very short. Thus labor costs are minimized under the condition that the service level is higher than a certain level.

7.5 Data collection and analysis

Gathering and preparing data is an important activity in a modeling project. The mathematical aspects are partly outside the scope of these lecture notes, but can be found in many text books on data analysis and statistics. Here we discuss some practical aspects.

First we have to realize the importance of obtaining correct data. "Garbage in—garbage out" is not without reason a well known phrase. Getting good input for your model is more than doing statistics well: above all it is acquiring, measuring and estimating data correctly and then using it correctly.

Due to advances in information technology companies register more and more business transactions. This simplifies the task of gathering data enormously. Still, this does not mean that this data can be used right away. Often data is aggregated or gathered in another way that is not appropriate for immediate use in a model. Sometimes another model is needed to generate data that can be used as input for the original model.

Example 7.5.1 All sales of a retail organization were automatically stored in an information system. However, at the end of each month, these sales were aggregated to monthly sales numbers. To perform a simulation study on the impact of a new inventory policy detailed sales over a year were needed as input. These numbers were generated based on the monthly sales and the detailed sales over the last month. This model could be verified when new data became available after a month.

Example 7.5.2 In a call center log files with many statistics are available. However, often they are aggregated over 15 minute periods. To analyse the performance of a call center one often needs the call length distribution. This distribution is hard to obtain as only 15 minute averages are available.

For other issues on data analysis we refer to the literature on this subject.

In what we described so far we assumed that data is already available. This is not always the case; sometimes the necessary input has to be estimated or measured, if at all possible. It is clear that this can be an extremely time-consuming activity.

Another distinction is between internal and external data. Internal data is relative to the firm, external data needs to be acquired externally.

Example 7.5.3 To assess the profitability of an investment one often needs to estimate the interest rate for future years. This is a typical example of external data.

The enormous amounts of data that are currently being gathered by businesses have stimulated people with IT backgrounds to think what can be done with this data. This has stimulated the development of new data analysis and optimization techniques, partly in parallel to the mathematical fields of statistics, stochastic modelling and combinatorial optimization. The notions used in this context are analytics, data mining, computational intelligence and business intelligence. According to Davenport & Harris [29], optimization is the final and most sophisticated part of business intelligence. It is interesting to note that, although OR/MS is hardly mentioned, many of the examples that are described in [29] are part of "traditional" OR/MS.

7.6 Verification and validation

It goes without saying that it is of crucial importance that solution techniques are implemented correctly, and that the system behavior is represented well by the model. Checking these carefully convinces not only the modeler of the correctness of his or her model and data, it also can play an important role in convincing the problem owners of the chosen approach.

The term verification and validation are often used in a simulation context, but they can and should be used for any type of model. Verification means verifying that the implementation of the model is correct; validation means that it is checked that the model outcomes correspond with those of the system up to a certain extent. As the system need not exist already this is not always possible. It is always possible however to check parts of the model, or to predict outcomes in some other way.

Validation not always means comparing a model with a system; it can also be the case that a model is validated with another model, one which has more detail for example.

Example 7.6.1 In call centers data is not always reliable. Validation and if necessary parameter tuning (also called *calibration*) can be done on the basis of a simulation model. The resulting parameter values can then be used in some optimization model, whose outcomes can be checked with simulations.

Validation of large-scale stochastic models is often difficult. This is even more so if human behavior is involved. This is a serious objection against the modeling of processes. The possibility of a failure to validate the model should be taken into account before starting the modeling process.

7.7 Suboptimization

Modeling is always a compromise between the scope of the model and the complexity. If the model is too complex to solve satisfactorily then decreasing the model scope is an option. This has the risk that the influence on system parts that are not modeled is ignored. This influence can be important enough to change the proposed solution to the problem.

Example 7.7.1 In logistics it was common business for each participant of a supply chain to optimize its own processes. However, by seeing the supply chain as a whole, considerable improvements can be obtained. Therefore supply chain planning is currently one of the hot issues in operations management.

Another possibility for checking the influence of a small scope optimization procedure on larger scope issues is using simulation as a large scope model. Thus the small scope optimization is *validated* by a large scope simulation. Often however the influence of suboptimization on other systems is hard to quantify, and modeling cannot help us to assess the consequences.

Example 7.7.2 In a production line it was decided that large production batches were more efficient. The upstream systems however were confronted with increasing demand sizes and were forced to increase stock levels. Due to this total costs went up.

7.8 To model or not?

Modeling is a “white-box” approach: it requires that the behavior of the system under study is completely specified. Modeling is therefore a time-consuming activity, that demands much of the knowledge and skills of the modeler(s). An important question to be asked before starting a modeling process is therefore: can we solve the problem by an alternative approach that requires less time and that is (at least) equally reliable in giving the right answers? As modelers have the tendency to apply their knowledge and skills, this question is not asked often enough.

A good candidate for a "quick and dirty" solution method that is often overlooked by OR/MS professionals is a statistical black-box approach. Here the inputs of the system are directly related to the output, and on the basis of this conclusions can be drawn. Of course, this works only if the system already exists and if data on the behavior of the whole system is available. This approach can be compared to simulation, where the output is also analysed in a statistical way. The advantage of simulation is of course that it allows to study non-existing systems or not yet implemented scenarios, the disadvantage is that modeling is necessary.

Example 7.8.1 A hospital department was trying to find out what the main causes where of waiting times of patients. During a month they collected data on waiting times and activities of medical personnel. At the time both a modeling study was started and a statistical analysis was done on the outcomes. The statistical analysis readily gave useful results. The modeling study gave unreliable answers that could not be improved on due to a lack of reliable data concerning the activities of the doctors.

An intermediate approach is to analyze a system first through a statistical approach and then model only the part(s) where most of the improvement can be obtained. This avoids going through a long modeling study, and often simple analytic models suffice. The main advantage is that the probability of finishing the project is much higher, in much less time: the invested time is spent much better, the average return on invested time is higher. An additional advantage is that the proposed solution is focused on the issues where most can be gained.

Example 7.8.2 A production system can be modeled in its totality, giving superior results, but only if complete data of the entire process is available. This is not often the case. Instead, one could focus on the production step that is the bottleneck and/or on the one that is responsible for most of the delay. Modeling only this node is much less work. In a subsequent project another node, perhaps the new bottleneck, can be analyzed.

Philosophically speaking, statistics can be seen as an *inductive* method: on the basis of a number of observations general conclusions are drawn. Modeling, on the other hand, is a *deductive* approach: on the basis of known behavior of components conclusions for specific models are drawn. Note however, that the behavior of the components is often obtained through statistics. Therefore, statistics also plays an important role in almost any modeling project.

7.9 Model builders and problem owners

Modeling is an activity that can be applied in many fields. Similar mathematical models are suitable for production as well as administrative processes, and a solution technique such as linear programming has so many applications that it is a standard option in any major spreadsheet. Thus modeling can be seen as a *generic* discipline. Applying it successfully therefore demands people specialized in modeling.

The modelers or model builders usually have no responsibility for the systems they model. This responsibility lies with the managers that are concerned with the daily operations of the systems, the *problem owners*. The modelers are internal or external consultants. The differences between internal and external consultants are disappearing, as most internal modeling groups become responsible for their own results, and because they have to compete with external consultants.

A third group of people that might be involved in a modeling project are IT specialists. Here as well we see different constructions: programmers can come from different departments in the problem owner's or model builder's organizations. Sometimes the implementation is done by an independent IT firm, which can be the main contractor or the subcontractor. That this can both occur is understandable if we know that on one side there are companies which are specialized in modeling that do not implement, and that on the other hand modeling is sometimes a small part of a big IT project, in which the IT company involved as main contractor is not specialized.

It is clear that communication between the project partners is of crucial importance. This demands from the modeler, besides good communication skills, insight in the problem domain and the implementation aspects.

A crucial part in the relation between modeler and problem owner is convincing the last one that the solution proposed by the model is correct. Verification and validation only allow to check the modeler's work, it does not allow the problem owner to participate in the modeling. Making managers and employees participate in the modeling can greatly increase the acceptance of the final solutions.

Example 7.9.1 A logistics company restructures its European distribution network. To convince local managers of the cost-effectiveness of the proposed solution a computer system is built in which the effects of changes can easily be calculated, during a meeting. This way they can convince themselves of the correctness of the solution.

Being responsable for the implementation of the solution, the problem owner has to deal with the organizational consequences of it. Often, a simple rule with little organizational consequences is preferred over a less simple slightly better ("optimal") rule. This should be taken into account during the modeling process.

Example 7.9.2 A hospital wanted to maximize at the same time throughput in one of its care chains (from the operation room to the intensive care to the normal care) and the occupancy of the beds. A complex model-based information system was proposed. However, a simple rule (always keeping a few beds empty at the normal care unit for patients dismissed from the IC) performed almost as good and avoided communication overhead between the different departments.

When modeling projects fail this is usually not due to a lack of modeling skills of the modeler, but moreoften due to implementation problems. For this reason good project management is of crucial importance to the success of modeling projects. Several business improvement frameworks strictly describe how the relations between the different project members should be and which hurdles are to be taken. Examples are the statistical quality program *Six Sigma* and Goldratt's *Theory of Constraints* which focuses on finding bottlenecks in all kinds of processes.

The strict division between modeling and management skills in the firm makes that modeling is used mainly for large-scale projects. However, the quality of every-day decision making can be improved as well by modeling insights and techniques. Therefore managers in relevant areas should have modeling knowledge. On the other hand, one can pose the question why modelers can only be found in specialized firms and dedicated departments. The main reason is that managers do not believe in the usefulness of modeling for problems other than low-level operational planning tasks. This is partly due to the fact that managers rarely have a background in modeling. However, the modeler is to be blamed as well: he or she does not always speak the same language as the manager, and is often more interested in sophisticated mathematical methods than in solving real-world problems. The expulsion of the modelers to specialized OR departments (sometimes even privatized), where they are financially responsible for their results, guarantees the usefulness of their work to the company.

This situation does not stimulate the every-day use of modeling in the firm: the modeler is only called for if the management has a clear-cut task to be done which involves, in the management's opinion, modeling. This task is often of an operational nature; without modeling knowledge or experience a manager will not easily rely on modeling for strategic or tactical decisions.

7.10 Skills and attitudes of model builders

The business environment in which a model builder is working forces him or her to have certain skills and attitudes in order to be successful. Certainly, the right scientific knowledge about models and their solution techniques is the starting point. Note that this knowledge need not only have some depth but also a certain broadness: the modeler should be aware of the different techniques and model types that are at his or her disposition. Next to mathematical knowledge the model builder should have relevant knowledge of information technology, as nowadays models are rarely solved by hand.

Apart from scientific knowledge the modeler should be aware of the specific issues that play a role in the area he or she is working in and of the generic way to solve certain problems in this specific area. Next to a quicker understanding of problems it gives confidence to the problem owners in the skills of the model builders. It also makes the model builder less dependent of the model owner, in terms of requiring information about the problem. The mathematician John Dennis put it this way: "Being an applied mathematician and working with people from other fields you have to be an anthropologist. You have to learn their languages." If the modeler is well aware of the situation, the roles can even be reversed: then it is the modeler who indicates problems and directly proposes solutions.

Some financial knowledge is also useful, certainly when it comes to issues at the tactical and strategic level, such as investment decisions. To understand and anticipate on the decision process it is necessary to speak the manager's financial language: to understand what ROI means, where a computer tool shows up on the balance sheet, etc.

Communication is a key issue in the modeling process, and therefore the modeler should have the right communication skills, both orally and written. Most of the time the modeler is member of a specialized staff department or of an external firm bringing in specialized knowledge. Thus communication is needed in both directions, to understand the problem owner's problem, and to communicate the proposed solution. This situation as external specialist demands a special attitude from the modeler. He or she should give the problem and the problem owners a central place. This shift from technically oriented to problem oriented is sometimes hard to make for mathematically educated consultants. It also requires that the analysis should not be more complicated than necessary to solve the problem.

The idea of giving a central place to the problem owner, the customer, comes back when reporting. Of prime importance when writing a report, preparing a presentation, or even designing an interface, is to put yourself in the customer's place. You should ask yourself questions as to the (scientific) level of your audience, the types of problems they are interested in, and so forth. Often a report is not written for a single person, but for different people with different interests in the problem solution. Therefore a report contains often a short executive summary for those who want very quickly an idea of the content and main conclusions. The main text is written with the typical problem owner in mind, the persons with whom is mostly communicated. Usually mathematical details are avoided as much as possible. These mathematical (and also non-mathematical) details are reported on in the appendices. These sections are meant for employees directly working with the modeled systems (typically line management), and they give the modeler the possibility to clarify on technical modeling issues.

When you put yourself in the customers place, it becomes possible to ask yourself the customer's next question(s). This advantage of being "one step ahead" can be of crucial importance.

Finally, a model builder is asked to execute a modeling project for his or her specific technical capacities. However, he or she works within the environment of the firm, with its political issues and different interests of the stakeholders. It is important always to keep an eye on these issues, and to report on results in a way that is not confronting. At the same time it should be clear that honesty and scientific soundness are never to be tempered with.

7.11 Further reading

An elaborate text containing a lot of background information on problem solving and especially modeling is Turban [95]. Most books on Operations Management give information on the Deming cycle and Six Sigma. See, for example, Chapter 10 of Van Mieghem [66] on this subject. Seddon [81] discusses how abberations like in Example 7.3.2 can occur. Chapter 2 of Simon [84] develops a theory of managerial decision making, in which the distinction between programmed and unprogrammed problems plays a central role. The distinction between strategic, tactical and operational control comes from Anthony [9]. A nice discussion of the three levels, in the context of logistics and with emphasis on the type of decision, can be found in Hax & Candea [46]. A recent view on modeling, mostly dealing with simulation, is given in Pritsker [72].

The failure of OR to support strategic decisions is well described in a series of papers by Ackoff (see, e.g., [2]). A recent paper, still elaborating on the question why mathematical modeling is less successful than can be expected, is Meredith [65]. It blames the lack of validation for this failure. A possible way to successful modeling of strategic and tactical decisions is making the managers participate in the modeling process. An approach to this is *Participative business modeling*, developed by J.A.M. Vennix. See Akkermans [4] and the papers (e.g., [3]) which form this thesis. Most of the time simulation is used as solution method.

Warner [97] is an accessible text that helps understanding the (financial) way of thinking of upper management.

A Dutch text on planning and scheduling from a managerial point of view is Jorna et al. [51]. Any text book on management and organization can be helpful in putting mathematical modeling in the right business perspective.

Davenport & Harris [29] gives a business-oriented view of analytics and business intelligence.

Instructive insights on how OR/MS professionals work and perceive themselves are given in Willemain [99].

A useful text on verification and validation is Kleijnen [56]. Although it focuses on simulation, most of it applies also to other solution techniques. Gallivan [36] discusses validation, induction/deduction, and sensitivity analysis (see Section 8.6) in a health-care context, but his argument apply in general as well.

Galbraith [35] discusses the implications of installing overall planning tools in complex organizations. "Local" solutions require adding slack capacity (and are therefore mathematically speaking suboptimal). Senge [82] argues that a small change in a process can have a big impact ("leverage").

A good starting point to get information on the process improvement frameworks Six Sigma and the Theory of Constraints is Wikipedia. Next to that we want to mention De Mast et al. [31] and Goldratt & Cox [42].

Many of the issues discussed here that are not related to models are part of *project management*. More information on project management, focused on process improvement projects in health care, can be found in Belson [15]. Note that project planning (Section 11.6) is part of project management.

7.12 Exercises

Exercise 7.1 Consider Exercise 7.3.1.
a. Consider the average waiting time as performance indicator. What would be the best order or orders in which to handle calls?
b. Formulate one or more different PI's that are closer to the objective of measuring short waiting times. Explain to which extend they are sensible to changes in average waiting time and variability in waiting time.

Exercise 7.2 A consultancy company has a tool to help companies place their warehouses throughout Europe. For each possible allocation it is capable to calculate the total annual costs. Classify the problem that can be solved with this tool using all classifications of Section 7.3.

Exercise 7.3 Relate verification and validation to the figure of the modeling process (Figure 7.1). Which step(s) have to be done again if the result of the verification is negative? Answer the same question for validation.

Exercise 7.4 Some people say that modeling is "art, not science". What do you think of this opinion?

Chapter 8

Model and System Properties

The number of mathematical models that can be found in the literature is huge. Choosing the right model and solution technique for a certain system problem is not always easy, even if one is familiar with the most common models. In Part I we discussed different classes of models. In this chapter we discuss certain generic aspects of models *and* systems, which can be useful in making modeling decisions and choosing the right model.

8.1 Generating versus evaluating

In Chapter 7 we treated all output of a model similarly, whether it consisted of just a number or a whole policy. Here we make a difference between *evaluating* and *generating* models and systems.

Generating models generate decisions. Typical examples are linear programming and dynamic programming. This decision can be a single number (or even "yes" or "no"), some vector or a range of decisions. These are all special cases of *policies*, which specify at each point in time and for each situation what to do. Therefore we say that generating models generate policies.

We have chosen the term generating models instead of, for example, optimizing models, for the following reason. When a policy is sought, we often search the optimal one (with respect to some objective). For certain classes of models this is computationally not feasible, and we have to stick to heuristics that approximate the optimal policy. Therefore the term generating is more appropriate.

Evaluating models evaluate decisions. They just give a number (or a set of numbers), the policy (if something as a policy can be identified at all) is thus part of the input. Simulation is the prime example of an evaluating modeling technique (although simulation can also be used for optimization).

The distinction between generating and evaluating is even more clear at the system level. Is there already a proposed solution to the problem available that only needs to be evaluated, or do we want that the solution is generated by the modeling process?

Many models can be formulated as a generating model and a evaluating model. Often the generating version has a special case that is evaluating. This means that generating models are often harder to solve. It also means that evaluating models allow for more details to be modeled. In this sense there is a trade-off between the quality of the model solution and the validity of the model.

If the system problem is of an evaluating nature, then of course an evaluating model is to be used. But even for a system problem that consists of finding an optimum it can be better to build an evaluating model. This is the case if the objective or the constraints are difficult to model, or if "playing" with the model can help the problem owners to get confidence in the model. This is often the case with strategic decisions. New business concepts cannot be created through a modeling approach, but their business value can be validated by it. Programmed problem at the operational level that are of a generating nature usually are solved with a generating model.

Example 8.1.1 A large logistics company wants to change certain processes, without having a clear idea what the alternatives are. Modeling the processes as a dynamic program forces unrealistic simplifications in the model. Therefore it is decided to simulate the systems, and to test various scenarios by changing system parameters in the simulation.

Finally, it can be the case that formulating a satisfying generating model is not wanted or possible, for example because of time constraints, or because of the complexity of the model. In such cases an evaluating modeling technique such as simulation might be an alternative.

8.2 Fluctuations and uncertainty

Without changes that occur inside or outside our systems there is no need to implement any changes. As such, management can trivially be seen as adequately dealing with changes or fluctuations within systems and their environment. In this section we make the important difference between fluctuations that are predictable and fluctuations that are not (fully) predictable. Whether certain fluctuations are unpredictable or *uncertain* depends on the information available; different managers can have different information.

Example 8.2.1 The number of calls offered to a call center fluctuates over the year, during each week, and during each day. The long-term fluctuations are to some extend predictable. The minute-to-minute fluctuations are very hard to predict. A manager who is aware of a new advertisement campaign can predict an increase in call volume; for a manager who is not informed this increase will come as a surprise.

Galbraith [35] defines uncertainty as the difference between the amount of information required to perform a task and the amount of information already possessed by the organization. It is thus the amount of information that must be acquired during task execution.

Uncertainty is unwanted. "Managers attempt to avoid uncertainty as much as possible" (Turban [95], p. 163). On the other hand, the environment in which companies operate is becoming less and less predictable. For this reason the central issue in strategic management is how to deal with uncertainty (Ansoff & McDonnell [7]). On top of that, managing under uncertainty is more difficult than in a deterministic setting, because the result of an action can change from time to time. Or, as Senge (see Chapter 17 of [82]) puts it: "'Learning by doing' only works so long as the feedback from our actions is rapid and unambiguous."

Uncertainty with respect to certain aspects of our systems is translated in our mathematical models as random variables. We can prove that avoiding randomness, i.e., unpredictable variations, is better. Assume that our performance is represented by a function p, which takes an action a and a random variable X as input. Then it is readily seen that $\max_a \mathbb{E}p(a,X) \leq \mathbb{E}\max_a p(a,X)$, assuming that the maximizations exist. At the r.h.s. the maximizing action depends on the realization of X, thus the value of X is known to the decision maker: the variation was predictable and the decision maker could react. On the l.h.s. only its distribution is known, and one action has to be taken for all possible situations at once, independent of the realization.

Example 8.2.2 Sales persons usually have little information about their company's production planning. Therefore, while negotiating a new order, they cannot take the "state" of the production system into account. Nowadays ERP systems can give this type of information to sales persons. Thus they can negotiate orders depending on the remaining production capacity, thereby improving capacity use and decreasing the probability that an order cannot be met.

Uncertainty has many sources. On a system level, we make a distinction between internal or external uncertainty. External randomness enters the model through the interaction with the environment, internal randomness comes from the system itself.

Example 8.2.3 When modeling a service center the stochastic arrival times of customers and their stochastic service times are external randomness. In a production system the stochastic *time-to-failure* of a machine is internal randomness.

We saw that avoiding randomness (in the sense of already knowing the realization) improves performance. For internal randomness this can often be done, although this demands an additional effort. This gives, in principle, a means to quantify the value of management information.

Example 8.2.4 In an industrial environment machine break downs can lead to very high break down costs. By automatic *condition monitoring* the degradation of machinery can be observed and preventive maintenance can be efficiently scheduled.

By definition, external randomness cannot be influenced. However, this is not totally true. On a model level, we can make some part of the environment part of the model. On a system level this translates into trying to make the environment part of your own system, for example by negotiating with suppliers.

Example 8.2.5 Suppliers in a logistics chain have to keep high stocks as to be able to deal with the unpredictable high volume orders that retailers place. Currently, using ICT (Information and Communication Technology), production companies sometimes even know the stock level of the retailer and are thus able to produce at the right time just the amount of products needed.

In the example we see well how information can reduce unpredictable elements in business processes. Many companies nowadays have tremendous amounts of data, with a great potential for cost reduction. (Despite this, certain managers are so overwhelmed by the amount of available data that they do not want to have access to it, trying to avoid "information overload".)

So far we identified system uncertainty with model randomness. However, uncertainty in a system is not always reflected by randomness in the model, it can be that during the modeling process some uncertain parameter is replaced by a constant, for example its average. This is a modeling choice, it can simplify the model enormously.

In fact, randomness in models is of such importance in OR/MS that it splits the research community in two: those that occupy themselves with stochastic models and those that consider non-stochastic models, a branch which is often called combinatorial optimization.

Example 8.2.6 Navigation software used to use fixed times for traveling road sections, mainly based on length and maximum speed, ignoring random travel times due to the possibility of congestion. Nowadays, manufacturers are adding real-time congestion information to their systems. The next step is using statistical methods to predict congestion and to integrate this in the routing software.

We have to stress that we interpret "avoiding uncertainty" as knowing beforehand what will happen. Having less uncertainty is always better. Avoiding fluctuations is not always preferable, although it usually is. For example, less variability in service times leads to shorter waiting times in queueing systems (Theorem 5.3.2). This is even more so in the case of production networks with multiple stations and finite buffer space between stations (see Chapter 11).

Example 8.2.7 In a production environment, management often prefers a smooth demand. However, situations exist where it is preferable that orders arrive batched together. This is for example the case for machines with long switch-over times from one product to another.

Confronted with fluctuations, there are two positions possible. The first is accepting the fluctuations and dealing as good as possible with it, perhaps using the type of advanced planning methods that are discussed in this monograph. This is the typical mathematician's approach. A second approach however, extensively used in practice and discussed in the business literature, is the one where one tries to reduce as much as possible the (internal) fluctuations. Next to a reduction of fluctuations this leads to an improvement of quality, as fluctuations and quality are intimitely linked. Thus, instead of reducing the influence of fluctuation by a smart planning method, one wants to emphasize the influence of fluctuations as to force workers to reduce them, for example by reducing buffer spaces. These are central ideas in the revolutionary Toyota Production System (see Section 11.2).

8.3 Computational complexity

Models can be complex in different ways: their size can be large, with many details, the time to build an algorithm can be long, and the time to execute an algorithm can vary depending on the model and the solution technique. To appreciate a discussion of scope of size of models we have to understand the notion of computational complexity first.

Computational complexity has to do with running times. For many models there exist algorithms which are guaranteed to finish execution in a time with is bounded by some polynomial function of the problem size. These problems are said to have a polynomial complexity. This guarantees that the execution of the algorithm takes a reasonable amount of time, even for large problem instances. For other problems there is no such algorithm known, only methods with exponential running times are known. These problems are called non-polynomial. Thus computational complexity is not related to the problem size, it has to do with model *properties*.

Example 8.3.1 For a given graph, finding the shortest path between any two vertices can be done in polynomial time (of the order n^3 with n the size of the problem), for finding a minimal length tour that visits each vertex once (the traveling salesman problem) no polynomial-time algorithm is known.

Of course, for any realistic problem an algorithm that finds the optimal solution can be constructed. However, solution methods for these non-polynomial problems are often equivalent with an enumeration of all solutions. For realistic model sizes this leads to unacceptable running times and (computer) resource utilization. To overcome this computational hurdle, algorithms are constructed for many non-polynomial problems that have acceptable running times, without guaranteeing optimality. Some of these algorithms can be seen as procedures that search the solution space in an intelligent way. See the discussion in Section 9.2.

One might wonder if advances in hardware technology will make non-polynomial algorithms have acceptable running times for reasonable problem sizes. However, it is the exponentiality of these algorithms that assures that this is not reasonable to expect. The only hope is for a mathematical breakthrough (e.g., a polynomial time algorithm for the traveling salesman problem), but experts consider this very unlikely. On the other hand, getting the most out of a polynomial algorithm becomes less and less cost-efficient: instead of optimizing your code it is cheaper to let the computer run a little longer. Thus basically any polynomial-time algorithm will do fine, while problems without polynomial-time algorithms should always be solved using heuristics. (There are some exceptions to this rule, but these are merely academic issues.)

It should be noted that for many standard problems (such as the traveling salesman problem) heuristics are developed that perform very well. Thus computational complexity is only of concern to those who design and implement algorithms. A user of a decision support system (discussed in Chapter 9) or a model builder that uses existing routines for model solving can safely assume that very good algorithms exist for most standard problems. That a heuristic does not necessarily lead to an optimal solution, is of less concern in practical applications. A reason for this is that a model is already a description of reality, thereby introducing modeling errors that are probably bigger than the difference between the optimal solution and the value found by the heuristic.

Having introduced the concept of computational complexity we can now consider other types of complexity.

8.4 Scope and size of models

It is not hard to formulate a model with many system details. Solving such an enormous model on the other hand can be very difficult. We already saw that for very complicated models only evaluating solution techniques such as simulation can be used. And, even if we manage to solve the model using a generating technique, the solution might be too complex to implement: a model solution is often as complex as the model.

Example 8.4.1 Systems to control traffic at freeways use detection loops in the road surface. Placing more loops gives more information. However, designing an algorithm that uses this information in an intelligent way is less simple, and depends necessarily on many variables.

However, sometimes, to avoid suboptimization, or because we need a very detailed policy, we need to take a highly complex model. This complexity has two dimensions: the degree at which details are modeled, and the *scope* of the model, i.e., whether or not many different aspects of a system or process are modeled. While small size models often fall within one of the standard classes, with algorithms or even software readily available, if models get bigger special solution methods need to be developed. Models can even get so complex that they cannot be solved in a single optimization step, and a multi-stage procedure is necessary. Often the outcome of later phases is used to improve the objectives of earlier phases.

Example 8.4.2 In the *vehicle routing problem* goods have to be distributed by a group of vehicles (see Christofides [21] for an overview). Several algorithms first assign the goods to the vehicles, by grouping together destinations which are geographically close. What then remains to solve is a traveling salesman problem (TSP) for each vehicle. This initial solution is improved by making goods change vehicles in some smart way, and by solving the corresponding TSPs afterwards.

It is not always the case that a large-size model leads to a complicated solution technique. One of the strong points of simulation is that we can model virtually any level of detail, and nowadays also linear programs with thousands of constraints can be solved.

Another important factor while modeling is the time it takes to find the right solution technique and to implement it. For many projects this is one of the main cost factors. The availability of standard software for some well-known solution techniques such as simulation or linear programming makes implementation times short. For many other models first a solution technique has to be selected (or even developed!), which has to be implemented afterwards. Evidently this takes an enormous amount of time and scientific knowledge, both of which are not always available in companies. Indeed, most of the development of algorithms is done at universities. One can ask the question whether this is a good situation. In any case, communication between companies and universities needs to be very well for the new techniques to be useful in practice. Testing and tuning can also take a lot of time.

In general one can state that the modeling time increases as the model size increases. But there are examples of models which are very simple to formulate, but which are very hard to solve. For example, for certain easily formulated queueing networks that are no analytical solutions known for say the waiting time distribution.

A great advantage of simulation over other models (typically queueing models), even if the model size makes programming necessary, is that it is not impossible to give good estimates of its implementation time. For commercial purposes this can be a major advantage. Also LP models and certain heuristics are relatively easy to implement, but as said before, for example queueing models can be extremely hard to solve. In such cases implementation takes longer, and the implementation time has greater variability. Sometimes this investment pays off compared to simulations, for the simple reason that once the model is solved and implemented results are obtained fast and accurately. Sometimes the model is already optimizing. If the model is of an evaluating nature then it is often easier to use it as the basis of an optimizing procedure compared to simulation, because of the long running times that it can take for simulation to give reliable output. Additionally, the structure of an analytic solution teaches us much about the model; simulation is merely a black box, giving random results. (Although this randomness can be reduced by increasing the number of runs that is made; see the relevant literature on simulation.)

The above holds especially when we compare queueing and simulation for models that describe the system in much detail. For more high level analyses (for example at a tactical or strategic level) queueing models can be very useful.

8.5 Team problems and games

Up to now we tacitly assumed that there is a single problem owner interested in solving problems using modeling. In certain problems however there is no central decision maker, but the decision is decentralized, i.e., there are several decision makers that each control a part of the system. Depending on the objectives of the decision makers these problems are called team problems or games.

Splitting the model up in parts which are each controlled by a single decision maker is not always possible, and if it is possible, then optimizing each part separately can lead to suboptimality. This can however be a viable option, as the model complexity might be reduced significantly by concentrating on parts. The distinction between team problems and games for the problem as a whole is defined as follows.

In games the different decision makers or players have a different objective. Note that we consider objectives as winning a game as different: the players want different players (often themselves) to win. In a business context games can for example be used to model markets. In the literature one also finds the destinction between games where players can cooperate and where they cannot.

In team problems we assume that all decision makers have the same objective. If they also have the same information, then each decision maker can calculate its own globally optimal decision (and the decisions for all other decision makers). Therefore, from a modeling point of view, it is better to let a single decision maker decide for all others. For a problem to be a real team problem there must thus be different information available to each decision maker.

Example 8.5.1 A node in a computer or telecommunication network has no up-to-date information on the state of other nodes and lines. Dynamic load balancing of the link loads is therefore a team problem. Note that such a network is an interesting example of a system where uncertainty can be reduced. This can be done by sending packets to other nodes informing them of the state of the originating node. The price of this information is the extra charge of the network due to the control packets.

Note that problems are not only team problems because of the impossibility to share information. It might be too costly, or the resulting overall policy might be too complex. In this sense a team problem might be a compromise between a centralized policy that takes all information into account (which is therefore a function of all system details, and thus extremely detailed) and completely decentralized decisions (with the risk of suboptimalization). Note however that it is often easier to compute centralized policies than decentralized policies.

8.6 Robustness

An important issue in modeling is the *robustness* of the model. We call a model robust with respect to a certain input parameter if a small change in that parameter induces only a small change in the output.

Example 8.6.1 In linear programming the robustness of the solution can be quantified, using *sensitivity analysis*. The dual variables, automatically generated by the simplex algorithm, play a major role in the sensitivity analysis. Most LP packages include sensitivity analysis.

Robustness has important consequences for the data analysis. If a parameter is robust, then the accuracy of its estimation is of less importance than for a parameter that is not robust. This should be reflected in the statistical analysis of the input.

Robustness can take many forms. Often it is considered with respect to a number (say next month's interest rate), but it can also be with respect to the form of the distribution of a random variable.

Example 8.6.2 The Erlang B model, discussed in Section 5.4, is robust under changes of the service time distribution. This means that the performance of the model depends only on the average call lengths. For related models, such as the Erlang C model, this is not the case.

Knowing for which parameters the model is sensitive is one side of the coin; knowing which system parameters are likely to change is another. This requires domain knowledge. Here we see again the importance of domain knowledge for the modeler.

Example 8.6.3 In a call center call lengths change only very slowly, while the offered call volume can show drastic changes on any time scale. Therefore it is more important to monitor changes in arrival intensities than in call holding times.

Example 8.6.4 Inventory control is often a trade-off between costs and risk of lost demands. Usually the stock level is controlled by two numbers: if the stock goes below a specified *minimum*, parts are ordered up to the *maximum*. In a model with daily reordering opportunities this policy can lead to high ordering costs, especially if the minimum is close to the maximum. This is less the case for weekly reorder opportunities, as we order at least the demand over a full week. However, if the minimum is not well chosen, we risk lost sales. Which aspect is prevailing depends on the system under study.

We should be careful with basing conclusions on a model which has certain parameters that are likely to change and for which the model is not robust. If it is not possible to come up with a satisfying modeling solution, then this fact should be communicated to the problem owners.

In this section we clearly saw the interplay between data analysis and statistics on one hand and mathematical models on the other. Therefore data analysis should not be considered as an independent activity, but as an integrated part of the modeling activities. Modelers should therefore also be trained in data analysis.

8.7 Choosing the solution technique

For many models different solution techniques exist. Reasons for choosing a technique are the accuracy of the results, the run times, the time to implement the technique, and the possibility for using the technique for other problems. Which one prevails depends on the problem that is to be solved.

Practically speaking, one usually searches for the technique that gives the desired accuracy and run times with the shortest implementation time possible. This often rules out mathematical sophistication.

Example 8.7.1 Many problem are linear in nature, and can therefore be solved using linear programming. For many subclasses of linear programs special-purpose algorithms exist. Often there are no standard implementations; as long as run times remain accaptable it is better to use standard LP software. An additional advantage is that the more general technique allows for more flexibility in case the problem changes.

Special-purpose algorithms should only be used if the run times or accuracy requires it. In other cases general-purpose algorithms are preferably because they allow for more system changes and are easier transferred to other systems.

Example 8.7.2 Standard manpower scheduling problems in call centers can be formulated as integer linear programming (ILP) problems. In practice other techniques such as heuristics are used; ILP lacks the flexibility that local search and other heuristics have.

8.8 Dynamic versus static

System problems often involve time, they are often *dynamic*. Non-dynamic problems are called *static*. For example, the product mix problem modeled in example 7.4.4 is static, no time is involved. Routing and sequencing problems are good example of dynamic problems: in which order should jobs be scheduled or locations visited?

Also for solutions techniques we can make a distinction between dynamic and static methods. Stochastic processes are dynamic, while those that solve problems of the form $\min\{f(x)\big|x \in S\}$ usually are static. It is important to note that there is no 1-to-1 relation between dynamic system problems or models and solution techniques. For example, dynamic routing problems that can be formulated as a TSP are often solved using static techniques.

Note that a problem that involves repeatedly solving a similar problem without interaction between the different moment is not a dynamic model, it is merely a sequence of similar problems. Thus dynamic models involve interaction between the decisions at different points in time.

Example 8.8.1 A transportation company has to deliver goods daily. If goods need to be delivered at a fixed day and if the drivers return each day to the depot this is not a dynamic problem. If goods can be delivered at different days or if drivers can stay overnight at different places it becomes dynamic: earlier decisions influence the situation at a later point in time.

Dynamic problems usually also involve decision making over time. For a deterministic model the future behavior can be exactly predicted, and therefore there is no issue of online decision making. This is not the case if behavior over time is stochastic. In this case there are two possibilities: either the decision takes the evolution into account, or it does not. For these cases we use again the terms static and dynamic: in the former case we call the policy dynamic, in the latter case static. Note that they coincide if there is no randomness involved in the problem.

Example 8.8.2 Most routing software use deterministic estimates of travel times. Letting you guide in the car by the software or taking a printout with you is (mathematically speaking) equivalent. However, if road conditions are taken into account, and if the software in your car updates its travel time estimates, then the optimal route is dynamically adapted. It might even mean making a U-turn, for example if a traffic jam is being formed ahead. If you print your route before traveling using road condition estimates then your routing policy is static.

8.9 Further reading

An excellent source of information on OR models is the series *Handbooks in Operations Research and Management Science* (North-Holland), edited by Nemhauser & Rinnooy Kan. This is a series of books containing high level introductory texts to most areas of operations research, written by experts. Volume 1 [70] deals with combinatorial optimization, Volume 2 [47] with stochastic models, Volume 3 [24] with computing, and Volume 4 [43] with models of logistic systems. Most of the models and subjects discussed in the next chapters have a chapter in one the handbooks dedicated to them.

There are many accessible undergraduate level text introductory OR/MS text books, more appropriate to people new to the field. Without any claim concerning quality or completeness we give two of these books: Taha [90] and Winston [101].

Zimmermann [103] discusses aspects of modeling uncertainty. Next to probability theory he also discusses the possibility of other frameworks such as fuzzy sets. In this monograph we focus on probability.

The observation that the time spent on a modeling project involving simulation is rather short comes from Buzacott & Shanthikumar [20], p. 15. In Chapter 1 of this book some general observations on modeling are made.

The standard text book on complexity is Garey & Johnson [39].

A list of models, with little emphasis on the mathematical aspects, can be found in Chapter 5 of Turban [95].

The difference between generating and evaluating models is introduced in Anthonisse et al. [8].

A recent, quite interesting article on the Toyota Production System is Spear & Bowen [87].

8.10 Exercises

Exercise 8.1 A newsvendor buys newspapers for 0.70 euro and sells them for 1 euro a piece. Leftover newspaper are worthless. Historic data over a year shows that 90% of the sales are in the range $[150, 250]$.

a. Calculate the optimal order level and the expected revenue for demand having a Poisson distribution with expectation 200.

b. Is this a good assumption? What would you take as an approximation for the demand distribution?

c. Calculate the expected revenue for the order level you just found and the new demand distribution.

d. Calculate the optimal order level and the expected revenue for the new demand distribution.

e. Relate your finding to the concept of robustness.

Exercise 8.2 a. Show $\max_a \mathbb{E}p(a, X) \leq \mathbb{E}\max_a p(a, X)$ for X discrete.

The previous exercise implies that uncertainty is always unwanted. This does not hold for

fluctations.
b. Give an example where fluctuations lead to lower costs, and one where fluctuations lead to higher costs.

Chapter 9

Model-based Computers Systems

Few models are solved nowadays without the help of computers. Sometimes this is limited to data analysis or the implementation and the execution, once, of a solution technique. At other times the computer is used to *support* the whole decision making process. For certain models solution software is readily available, for other models algorithms have to be implemented entirely. We discuss the different possibilities in this chapter.

9.1 Classification

Model-based computer systems basically come in three flavors, when looking from the user's perspective. The main distinction is whether the user is human or not, and whether the human user is a modeler or a problem owner.

When the user is a problem owner, often a planner within a company, then he or she probably wants a software tool with a good user interface that is specially developed for the type of task that the planner is doing, and that allows the planner to create solutions interactively with the software (otherwise there is no use in having the planner). This allows the planner to take care of aspects of the problem that are not modeled.

Such systems are called *decision support systems* (DSSs). Many definitions of DSSs can be found in the literature, going back to the early 1970s. Keywords in most of them are *computers*, *models*, *unstructured problems* and *human interaction.* We use the following definition.

Definition 9.1.1 *A* DSS *is an interactive model-based computer system that helps humans solve a certain class of business problems.*

Note the use of "a certain class of": we do not want to include in the definition systems with some modeling capacities such as standard LP solvers or spreadsheets. It becomes a DSS when it is especially adapted to a certain class of business problems. Note that with this definition already a small LP model in Excel can be a DSS.

The user of a DSS is usually a problem owner; therefore it is problem oriented. On the contrary, modelers often have tools at their disposition that are focussed on solution techniques, not on application areas. They have a tool for simulation, and one for mathematical programming, and so forth. We will call these *modeling tools.* For convenience they also have (sometimes graphical) user interfaces.

Definition 9.1.2 *A* modeling tool *is an interactive model-based computer system that helps humans solve models with a certain class of solution techniques.*

It should be noted that the user interface of a modeling tool usually gives more freedom to the user than that of a DSS. This makes sense, as a modeler usually has better knowledge of modeling issues than the user of a DSS, he or she has the knowledge to use this freedom. In fact, certain modeling tools can be used to build DSSs.

Modeling tools exist for various techniques. For Monte Carlo simulation (discussed in Section 1.9) there are several add-ins for spreadsheets. They allow you, simply said, to turn spreadsheets cells into random variables for which you can draw a value repeatedly. Powerful graphical reporting tools are supplied.

Let us now consider discrete-event simulation discussed in Section 3.2). Tens of tools exist, usually with a graphical user interface that allows the user to build the model in a intuitive way. Often there is an underlying programming language that can be accessed by the user to model features that cannot be entered using the graphical interface. One attractive feature is to follow the dynamic evolution of the system in a visual way. Indeed, simulation tools allow you first to model the system you want to simulate in a visual way, and then to follow the simulation visually. This is great for debugging and for making a nice presentation.

Although strictly speaking not (always) model-based, we would like to mention that also for forecasting quite a number of special tools exist.

Also for mathematical programming modeling tools exist, for example AIMMS. Actually, in AIMMS different solution methods can be chosen, the modeling environment is to a certain extent independent of the implementation of the algorithm, the solution module. See the next section for a discusssion of mathematical programming solution modules.

Finally, there are systems that take model-based decisions without human interference. This can be the case because there is no time to consult a human decision maker, because there are too many decision to be made for humans to handle, or because the model-based decisions are good enough and need no human improvement. Let us call these systems *automatic decision systems.*

Definition 9.1.3 *An* automatic decision system *is a model-based computer system that takes decisions without human interference.*

Example 9.1.4 A company is setting up a call center. The lay out is decided upon after a simulation study by a consultant for which he used a simulation *modeling tool.* For workforce scheduling a *decision support system* is bought from a company specialized in call centers. The call routing is done online by an *automatic decision system* that is part of the software of the telephone switch.

9.2 Optimization modules

It can occur, for all three different types of model-based systems introduced in Section 9.1, that they have the same solution generating module in it. These solution modules can often also be bought independently of the user interface, for example by people who want to develop a decision support system for a specific class of problems. This is in particular the case for mathematical programming modules. As mathematical programming falls outside the scope of stochastic modeling we did not discuss it in Part I. However, for many of the applications we discuss in Part III, (deterministic) optimization is part of the solution. For this reason we describe the types of mathematical programming problems here.

A mathematical programming model has as goal finding the solution of

$$\min\Big\{f(x)\Big|x\in S\Big\}, \tag{9.1}$$

for some closed set $S\subset\mathbb{R}^n$ and $f:S\to\mathbb{R}$.

Example 9.2.1 In Chapter 6 we studied the so-called EOQ model. The costs were a function of the order size Q. Formulated as a mathematical programming model the problem of finding the optimal order size becomes:

$$\min\Big\{\frac{K\lambda}{Q}+\frac{Qh}{2}\Big|Q\geq 0\Big\}.$$

One can easily think of useful additional constraints. Q might also have an upper bound representing the shelf space for the product, or the re-order period T might be restricted to multiples of the re-order period t. This last constraint can be modeled by requiring that $T=kt$ for $k\in\mathbb{N}$. Using $T=Q/\lambda$ leads to $Q=\lambda kt$ for $k\in\mathbb{N}$.

Sometimes Equation (9.1) can be solved easily. The EOQ formula is an example: the objective is differentiable and is minimized in the interior of S, i.e., for some $Q>0$ (assuming $K,h>0$). Finding the minimum is more complicated when f is not differentiable or when the condition $x\in S$ is restrictive (i.e., the unrestricted minimum of f lies outside S). Moreover, in many situations, f has a multi-dimensional domain.

Most methods for solving (9.1) in its general form rely on locally finding a feasible direction (i.e., a direction in which we can make a little step without leaving S) that makes f decrease. This gives a method to find a local optimum. Conditions (related to convexity) can be given such that these optimization procedures are guaranteed to converge to the global minimum.

For many special forms of the objective and the feasible region special solution methods exist. If f is a linear function, and S can be written as

$$S=\Big\{x\in\mathbb{R}^n\Big|g_i(x)\geq 0,\ i=1,\dots,m,\ g_i\text{ linear}\Big\},$$

then the problem of (9.1) becomes a linear program. Efficient solution methods exist (given that there is an optimal solution), the simplex method being the best known. An important property of the simplex method is that it always terminates in a vertex of S. Many software packages exist that can solve linear programs with up to thousands of variables and constraints.

Example 9.2.2 In Theorem 4.2.3 linear equations are given for finding the stationary distribution π of a Markov process. Now suppose that the transition rates λ are linear functions of one or more parameters. Then minimizing a linear function of the stationary distribution is an LP, with the linear equations of Theorem 4.2.3 as constraints.

Another important choice is the linear model with the additional constraint that there is an index set $J \subset \{1, \dots, n\}$ of variables that need to be integer, thus

$$S = \Big\{ x \in \mathbb{R}^n \Big| g_i(x) \geq 0, \; i = 1, \dots, m, \; g_i \text{ linear}, \; x_j \text{ integer for } j \in J \Big\}.$$

By adding constraints of the form $x_j \geq 0$ and $x_j \leq 1$ we see that 0-1 programming is a special case of integer programming. In theory, integer programs are hard to solve, but excellent numerical methods exist.

The feasible set S of a mathematical program is a subset of $\mathbb{R}^n$; although S can be finite, the solution methods that we discussed up to now are based on connected sets. For most problems with $|S| < \infty$ or countable it is better to use a method especially designed for these kind of state spaces. A very popular method is *local search*. The idea is simple. Again we have an optimization problem of the form (9.1), but now S is finite or countable. For each $x \in S$ we define its neighborhood $N(x) \subset S$. We say that a x^* is a local minimum with respect to N if $f(x) \geq f(x^*)$ for all $x \in N(x^*)$. Now a simple algorithm to find a local minimum is as follows: if the current state is x, then its whole neighborhood is searched. If a state y is found with $f(y) < f(x)$, then y becomes current. This is repeated until a local minimum is found.

Mathematical programming software is in many forms available to the modeler: as spreadsheet add-in, as stand-alone program, or as routine that can be called from within a specially written program, in a way that is transparent to the user. Indeed, most of the scheduling programs that many companies use have in the background some commercially available mathematical programming routine running. Software surveys are available on the Internet (see Section 9.6).

9.3 Platforms

For the construction of model-based software the modeler and/or software developer has a number of possibilities at his or her disposition. The choice is often a compromise between the result and the time that needs to be invested. Evidently, a modeling tool for personal use has less requirements on the user interface than a DSS with a large user group. The major types of platforms are spreadsheets, general mathematical (symbolic manipulation) tools, off-the-shelf modeling tools, and regular programming environments. Evidently the former three are mainly employed if there are few users, a choice for the latter can be motivated by the requirements on the user interface (e.g., when there are many users). Which one of the first three is preferred depends partly on the match between platform and solution method.

Traditionally, DSSs as all software tools were tailor-made computer programs completely developed in a high-level programming language such as C. This demands an enormous development effort: not only the mathematical algorithms need to be implemented, but one also has to take care of issues such as the user interface. Indeed, it happens often that modelers building DSSs complain about the little time they spend on modeling and the large amount of time that they are programming. Of course, even if a firm builds everything itself, that does not mean that user interfaces and algorithms need to be developed from scratch for each modeling project: specialized firms and departments try to reuse software they made before. Next to that there free and commercial routines available for solving standard models such as linear programming (see Section 9.2). It should also be noted that the construction of user interfaces becomes easier and easier with the advent of programming languages such as Visual Basic and Delphi. Building dedicated DSSs is useful and cost efficient for large modeling projects with many users and a model that is often executed (with different data).

We continue with discussing the possibilities in case it is not considered necessary to build the model-based tool in a high-level programming language. This is the case when we are building a DSS or modeling tool with a small user group, often consisting of experts.

If possible off-the-shelf modeling tools are preferred. If there is no suitable tool available, then general mathematical packages such as Maple or Matlab are sometimes preferable to low level programming. It takes less time to implement, but running times are usually longer.

A relatively recent development that does not fall within one of the classes previously discussed is the use of spreadsheet, notably MS Excel. At its base a spreadsheet, a matrix of cells between which simple mathematical relations can be defined, Excel is rapidly becoming the mostly used modeling tool. This is because of a number of reasons. The first is its availability: being part of MS Office Excel is installed on almost any PC. Due to its usefulness for basic (financial) calculations many know how to use Excel's basic functions. This assures that users quickly learn to work with Excel.

The second reason for the popularity of Excel is the availability of model solvers and the flexibility to add your own algorithms. Many statistical functions and even a mathematical programming tool are standard available in Excel. Additional add-ins can be build or bought for performing all kinds of tasks. Finally there is an underlying programming language (similar to Visual Basic) with which virtually everything can be done.

The third reason is the flexibility of the user interface: using standard functions one can quickly build a simple DSS for private use by the modeler. On the other hand, Excel offers many possibilities to make the interface more user friendly and fool proof: one can add buttons, graphics, etc. Altogether, Excel is an extremely versatile modeling tool with which almost any prospective DSS user is already familiar. It allows the modeler to start with just a simple model, and to add features and it user friendly while going through the modeling project.

Example 9.3.1 A firm was competing for a large maintenance project at an airport. To determine the labor costs of the project an LP model was implemented in Excel. The user interface was kept as simple as possible because the DSS was only used by the model builders. When the project was granted the same DSS was used fro scheduling personnel. Of course the user interface had to be changed to adapt to the new users.

For standard business environments there is also the possibility to acquire a dedicated DSS. For example for call center manpower planning problems there are several DSSs on the market (see Chapter 13).

9.4 Decision support systems

DSSs are widely used within companies. They often play a role in which model-based decision making is only of minor importance. In this section we take a closer look at the desired functionality of DSSs.

The definition of Turban [95, p. 85–87] of DSSs is not a definition, but more a list of features. One of them is especially relevant to our view of DSSs: *DSSs support all phases of the decision making process.* This is evident by the fact that a DSS helps solving a business problem, not just a model instance.

The other 13 points are less relevant, sometimes even absurd: E.g., demanding that end-users, only with minor assistance from specialists, build DSSs limits its capacities to only simple LP and simulation models. (This is reflected in the model list in [95, Ch. 5], where only the very basic models are discussed.)

More useful to us is the classification of Anthonisse (based on personal communication). He defines the functionality of a DSS as follows: A DSS is capable of

- selecting,
- generating,
- manipulating,
- evaluating,
- presenting,
- memorizing, and
- in and exporting

data, models, and schedules.

In Anthonisse's point of view DSSs are restricted to scheduling problems, at the operational level. This explain the word schedule in his definition. If we replace schedule by a more general term such as solution his definition could well be used for other types of problems.

Let us discuss some important aspects of DSSs on the basis of this classification. The basic functionality of a DSS is the possibility to *generate solutions* automatically. However, the user can create or changes solutions as he or she wants, by the possibility to *manipulate solutions*. Thus the user is free to use the solution techniques that are implemented in the DSS as much or as little as he or she likes. The user interface should be such that it is very easy to manipulate solutions. Only this should make the user prefer the DSS. Next to that the DSS is capable to *evaluate solutions*, for feasibility and efficiency. Using a DSS is often a repetition of generating, manipulating, and evaluating solutions.

Example 9.4.1 A DSS is developed for scheduling employees of the catering service of a large company. This catering service is characterized by employees with many different skills and short tasks at different locations in the building. Although the DSS is capable of generating solutions itself, this option is used little. Due to personal preferences and capacities that are extremely hard to implement the proposed solutions are of low quality. However, the option to evaluate solutions is very useful. The planner can use this to see if people are scheduled double, if they have to work in overtime, etc. The real gain is in the reporting phase, where a single push on a button generates reports concerning numbers of hours worked, overtime, etc., for each employee. When this work was this done by hand this took the planner two days a week.

The possibility to *manipulate models* does not mean that the user can implement other solution techniques; it means that it is possible to tunes models, for example by setting certain parameters.

Example 9.4.2 A call center employee is responsible for planning and scheduling of the agents working in a call center. For this she uses a DSS especially designed for call centers. There are certain legal restrictions that need to be satisfied all the time. There are also personal preferences (weekly night off, car pooling, etc.) that cannot be satisfied all the time. By changes parameters the relative importance of these personal preferences can be changed.

Presenting solutions is of course of prime importance, to communicate solutions to users and others.

Example 9.4.3 A DSS was built to determine the new location of the warehouses of a firm. The decisions to move warehouses had a high impact on the workforce, because many of them had to move. The DSS was used on a tour along all the warehouses to convince the employees of the necessity of the intended decision. During the meetings new solutions could be entered, evaluated and presented.

9.5 Integration and interaction with other systems

A modeling study is rarely executed without the use of other computer systems. The most important reason is the determination of the input parameters, something that usually needs to be done every time the model is executed. Data relevant for modeling is stored in different types of computer systems. Some of these systems are used for operational and administrative reasons. An important example are ERP systems. ERP stands for Enterprise Resource Planning, a class of software programs that administrate all processes within a firm. The origin of ERP systems is the computer implementation of MRP, a computational method to release jobs in a production environment (see Chapter 11 for the MRP logic). Nowadays modules of ERP systems span diverse activities such as sales, finance, etc. The big advantage of such a system is that it is company-wide. If a representative wants to sell an order, he or she can immediately verify current stock and/or production capacity and take that into account when negotiating the order parameters. After the order has been placed, purchasing can immediately react by buying lacking raw materials, etc. The information available in these ERP systems has a huge potential. Currently the major ERP vendors are developing control tools on top of the transaction data that extract relevant data from the transaction database. Using this they are building data warehouses, with mathematical modeling as one as the possible applications. This facilitates the use of OR techniques in firms with ERP systems.

Systems to support operational processes exist also in other areas. For example, for customer contact there is so-called Customer Relationship Management (CRM) software, and we see a movement in hospitals and health care in general to move from isolated systems (for insurance information, radiology, etc.) to systems that cover all aspects of the health delivery process, including the Electronic Health Record that contains all information concerning the patient.

ERP systems and its equivalents in other sectors are constructed to support the operational processes. For this reason they contain many details of for example current customers or work at hand, but less historical data. This makes them less suitable for the extraction of management information and therefore also for modeling purposes. This explains the existence of *Management Information Systems*. They allow for the extraction and aggregation of large quantities of data to give new insights and spot trends. A term that is also often used in this context is Business Intelligence, although that also includes the extraction of new relations using statistics and data mining. See Section 7.5 for more on Business Intelligence and its relation to other terms.

9.6 Further reading

The literature on mathematical programming is enormous. Any introductory text book to Operations Research will give the basic methods. For a higher level overview we refer to Volume 1 of the series *Handbooks in Operations Research and Management Science*, Nemhauser et al. [70]. Williams [100] deals nicely with all aspects of solving problems using mathematical programming.

A lengthy text on DSSs is the already cited Turban [95].

Anthonisse et al. [8] discuss what they call *interactive planning systems*.

Jones [50] is a text on user interfaces from an OR point of view; DSSs are also discussed, and many references are given. In the same handbook [24] there is a chapter on mathematical programming systems.

For an overview of vendors and a comparison of mathematical programming tools, see the software surveys published in OR/MS Today, the INFORMS membership magazine, which can best be viewed on the internet at `lionhrtpub.com/orms/ormssurveys.html`. There are special surveys for linear and non-linear optimization. The survey of spreadsheat add-ins also includes mathematical programming tools. There is also a survey on forecasting tools.

A recent book emphasizing the importance of automatic decision systems is Taylor & Raden [92].

9.7 Exercises

Exercise 9.1 A call center is open from 8 to 5. It has a full-time shift from 8 to 5 with a break from 12 to 1, and 2 part-time shifts, from 9 to 1, and from 12 to 4. Between 10 and 4 there should always be at least 5 employees available. At the beginning and the end of the day there are few calls, requiring only 2 employees. It is the objective to minimize the total number of agent hours.
a. Formulate this as a mathematical programming problem.
b. Solve this problem using the Excel solver leaving out the integer constraints.
c. Solve the problem with the integer constraints.
d. Repeat the same questions for the problem where we replace the full-time shift by 2 part-time shifts, from 8 to 12 and from 1 to 5.

Exercise 9.2 Consider three parallel single-server queues with exponential service times. Arrivals occur to the system according to a Poisson process. Arriving customers are assigned to one of the queues independently, each using the same assignment probabilities. Formulate this as a mathematical programming problem and use the Excel solver to find the assignment probabilities which minimize the average waiting time for your choice of parameters.

Exercise 9.3 A high school is considering buying a tool for making their annual schedule at the beginning of the year (when do which lessons take place as to make the best schedule for classes and professors). Make a list of features for this system, using the classification of Anthonisse.

Exercise 9.4 The same question for the following system: an airline uses multiple fares for each connection. For each flight the number of seats available in each fare class has to be determined as to maximize expected profit. A system is developed that predicts the demand for each fare class and that can compute expected revenue.

Exercise 9.5 For the high school scheduling tool: Is it evaluating or generating? And how about the airline tool?

Exercise 9.6 Does uncertainty or randomness play a role in one of these two systems?

Exercise 9.7 Is uncertainty always caused by a lack of information? Try to find examples of uncertainty that cannot be predicted.

Exercise 9.8 In which of the two systems (the high school and the airline reservation tool) does robustness play a role?

Part III

Applications

Chapter 10

Application Overview

In this chapter we give an overview of the application areas that are discussed in the next chapters. The areas covered are, in our opinion, the most important ones. There is no claim on completeness: virtually any business area has been the subject of OR/MS studies. We do think that the majority of all OR/MS projects falls within one of the application areas that are discussed in one of following chapters.

The goal of the current chapter is to give a short introduction to each area by identifying its place in the typical business process. While doing so the dependencies between the chapters will become clear.

10.1 Service operations management

Applications of mathematics can be found in almost any place within every company. This is for example the reason why every engineering curriculum contains a substantial amount of mathematics. When looking at the way mathematics is used we can make the following distinction. Sometimes mathematics is used in the product that a company makes itself. Examples are the Markov chain underlying Google's search algorithm and differential equations needed to understand the aerodynamics that makes aircraft fly. There is also mathematics involved in the generic science that concerns managing processes within a company. This science is called operations management, and Operations Research is largely (but not entirely) focused on the quantitative side of operations management. The applications of the following chapters all stem from operations management.

Operations management is a generic science: both hospitals and car manufacturers have limited processing capacity, and need models to analyze decisions concerning the capacity. Models used in different areas are similar, but not always the same. The following chapters discuss a number of these more specific models, in different application areas.

Economic activity can be split up in several sectors: production of primary goods, manufacturing, and service. Sometimes a fourth sector is identified, which includes intellectual activities such as research and education.

The products in the service sector are characterized by the fact that the customer is part of the process itself: production and consumption occur at the same time, because the product cannot be put on stock. Health care institutions deliver services to their patients, consulting companies deliver services, and so forth. In a traditional manufacturing plant goods are produced, put on stock, and shipped to the customers. Although manufacturing still plays a crucial role in today's economy, the service sector currently represents the majority of economic activity in Western countries.

The characteristics of services have important implications for planning issues in companies. Fluctuations in demand cannot be smoothed by building stock, as it is often done in manufacturing. Therefore not inventory, but capacity plays a crucial role in services. Selling this capacity for the highest price possible, taking into account demand fluctuations, is called *revenue management* (see Chapter 14).

Example 10.1.1 Skiing equipment is manufactured during the whole year, and sold almost exclusively during the winter. Skiing resorts see most of the tourists coming during a few weeks a year; in these weeks services such as hotel rooms are usually more expensive than during off-season weeks.

The word service is also used in the context of the *service economy*, which refers to the fact that tangible products, produced in the manufacturing sector, are treated more and more as services. Instead of getting all the same product from the same location, we see also in the manufacturing sector that the customer needs takes a central place: the product is made or assembled exactly according to his or her wishes (*customization*), it includes delivery at the location, extensive after sales, etc. This means that also companies in the manufacturing sector deliver services. These are sometimes called *product services*, to differentiate from the *service products* delivered by the service sector.

Example 10.1.2 Ricoh is a big Japanese manufacturer of electronic equipment. They used to sell their copiers to customers. Nowadays the copiers remain their own property; instead, they charge per copy, which is the actual need of the customer. Thus instead of manufacturing copiers they now focus on delivering copying services.

In these lecture notes we deal both with application areas from the service and manufacturing sectors, with an emphasis on services, i.e., the role of customers in the business processes.

10.2 The primary process

Activities within a company can be divided into those that are part of the *primary process* and those that are not. The *primary process* is the collection of activities that together make the product(s) of a company.

Within manufacturing companies the primary process can often be identified easily. It is part of the *supply chain*, which is the system, often spanning multiple companies, that allows production from raw materials up to the final product including its distribution and after-sales activities. Nowadays even return flows exist where the depreciated product is dismantled and parts are used again. Then the chain becomes a cycle.

A supply chain is a chain of different activities, often executed by multiple companies. Management has to deal with these different activities, but also with the coordination between these activities. This is what we will call *supply chain management*. Main activities of the supply chain manager have to do with ordering, production and distribution, and their smooth interaction. Many people would also use the term *logistics management*. Originally a military term, the definition of *logistics* is not always clear: it can range from almost all operational activities to only distribution issues.

An important role is played by the *customers*. With a customers we mean the buyer of a product. A customer need not always be a person, it can also be another company. Neither need it be the end user: the customer can use the company's product to add value to it and make a new product. We will make a difference between orders and sales. Sales always occur at the end of supply chain (not taking after-sales service and depreciation into account). Orders can occur anywhere within the chain.

For many products orders and sales coincide. This is for example the case if you buy something in a supermarket. If production and distribution is done without order, then the product is put on stock waiting for the customer. This way of producing is called *make-to-stock* (MTS). *Make-to-order* (MTO) exists also: production is initiated by the customer order. Good examples are administrative processes, for example buying a loan. A complex administrative procedure has to be undertaken as soon as somebody requests a loan for a house or another important expenditure. Services are always MTO: production can only start when the customer is present.

A mixture between MTS and MTO is *assemble-to-order* (ATO). ATO means that in an MTS fashion parts are produced and stocked. After that, in an MTO fashion, parts are assembled to finished products. The *Customer order decoupling point* (CODP) is defined as the point in the supply chain where production is started that is initiated by the customer order.

Make-to-order allows the product to be made according to customer specifications. Thus from the customer order decoupling point the production process has the nature of a service. In the early days of industrialization we saw almost no customization. To illustrate this, Henry Ford is believed to have said about the T-Ford: "You can paint it any color, so long as it's black". Nowadays customization is quite common, necessitated amongst others by the global competition that manufacturing companies encounter. It is a big challenge to combine this possibility of customization with the efficiency of mass production (*mass customization*).

It is clear that stock or inventory plays an important role in MTS systems. But also in MTO systems stock will prove to be crucial as 'lubricant' in the production process.

In Chapters 11 we study all aspects related to manufacturing: short-term scheduling issues in the first sections, long-term planning issues related to on one hand minimizing inventory and and the other on meeting demand as good as possible, and finally at the complete supply chain.

Supply chain management became possible as soon as companies were able to exchange detailed information on a routine basis. The internet is the platform at which these exchanges take place; E-commerce is the language that uses the internet to communicate.

10.3 Aspects of a product

In the above we focused on the production of a tangible article, characteristic for the manufacturing sector. Of course, a product can be a service: a decision (will I get the loan?), medical treatment, a hotel room, a seat in an airplane, etc. For some of these services general logistics principles apply, for some of them they do not. And not even for all physical products supply chain management is useful; think of one-time products such as the construction of a large building. For these type of one-time activities project planning is useful, logistics principles and supply chain management are less relevant. Project planning is also useful in regular production companies for non-primary processes such as the development of a new product. Section 11.6 is devoted to this subject. Supply chain principles apply to systems where *similar* items are produced in the sense that (almost) the same production steps are followed by the products.

The product offered by for example airline and railway companies and hotels is also different in nature. They actually offer *capacity*: whether this capacity is used or not does hardly make any difference to the activities of the company. It does make a difference to the income of the company, which therefore tries to maximize its income given its available capacity. This activity is called *yield* or *revenue management*. It is the subject of Chapter 14.

Now let us focus on some arbitrary product, physical or not. An important point to make is that a product consists of more than the actual item that one buys; depending on its nature, it has also some other aspects such as time to delivery, its delivery location, after-sales service, etc. The time to delivery plays an important role in the design of the production process, such as deciding between MTO or MTS: see Section 11.5. Some of these activities are geographic in nature, such as delivery and after-sales repair on the customer location, field service. We dedicated Section 11.8 to planning problems with a geographic aspect.

During ordering and after-sales many contacts between customer and company can occur. Nowadays these contacts are often bundled in a single *customer contact center* (also known as *call center*, by their main activity). Waiting and scheduling issues in call centers are the subject of Chapter 13.

10.4 Inventory

Physical parts and products can be made before the order occurs, up to the CODP. This means that there will be inventory of these parts. Also during the production of these parts and in the later part of the process in which the customer-specific order is produced it can be helpful to stock items. There are different reasons possible for doing this. We classify the types of inventory and discuss their reasons of existence.

Cycle stock Cycle stock has to do with economies of scale: the fact that activities done in larger quantities have lower costs per item, or, equivalently, that marginal costs are decreasing. The economies of scale can have different sources. A few common examples are:
- in production processes in which machines have to be set up this usually has to be done once in the beginning of a production run, independent of the length of the run. This increasing the batch size leads to lower overall costs and/or time per item;
- many companies charge a fixed overhead per order, next to costs linear in the number of parts ordered;
- transportation costs usually have a considerable fixed component, and a much smaller variable component.

Safety stock Safety stock is a reaction to variability in demand that cannot be predicted. To avoid backorders or lost sales one tries to have, often on top of the cycle stock, some inventory that is there just in case the demand is unexpectedly high. Usually backorders or lost sales cannot be avoided entirely, and some service level (e.g., "not more than 5% backorders") has to be defined. It can also be the case that costs can be assigned to backorders or lost sales. In that case there a trade-off must be made between different types of costs.

Seasonal stock Building up seasonal stock is a way to react to long-term predictable variations in demand. Typical examples are clothing that is sold in a certain season, and Christmas gifts that are procuded in an MTS fashion during the whole year and sold during only a short period. To use the available production capacity in an efficient way it is often cost-efficient to produce the whole year round and therefore to build up seasonal stock.

Next to these reasons for keeping inventory there are several good reasons why we should not keep (too much) inventory. The main ones are listed below.

Investment and handling costs One of the downsides of inventory is that it requires investments. Thus every Euro invested in stock should increase the profit or decrease costs in such a way that it is worth investing it. Next to the investment costs there are costs for having and maintaining inventory. Costs in this category are the costs of warehouses, etc. Both types of costs are sometimes hard to calculate. How should we calculate the value of a finished item on stock? It is clear that we should count the costs that were made in purchasing raw materials. But should we already count the added value? In financial reporting it is common to do this. However, to avoid expensive stock this can lead to the decision of not producing and keeping raw materials while there is ample production capacity available. This is a form of suboptimization that needs to be avoided by either giving the right local objectives or to plan globally.

Obsolescence and depreciation Sometimes parts have a limited lifespan and should be shipped before some date. But even parts that do not deteriorate over time can decrease in value over time because of technology improvements, fashion changes, and so forth. This is called depreciation. Note that certain items at stock risk to be stolen. This needs also be taken into account. The measures to avoid theft are part of the investment and handling costs.

Physical limitations In any system the amount of inventory places is limited. If the inventory exceeds this level than a possible consequence in for example a manufacturing plant is a production stop on one or more machines.

Next to these reasons it might be that keeping inventory is a solution to a problem that can be addressed more effectively in another way. For example, instead of increasing inventory levels to cope with varying production times one should perhaps focus on reducing the variability in the process (see also the last paragraph of Section 8.2). This is the central idea behind the Just-in-Time (JIT) production method; see Section 11.1.

There has been an enormous focus on reducing inventory to reduce costs during the last decades. It is our experience that this often has been done without a solid motivation: regularly we encounter processes where total costs can be reduced by investing in *more* inventory. The reduction of inventory is often motivated by the successes of the JIT method as part of the Toyota Production System (TPS). However, it is a misconception that the TPS promotes the removal of all or almost all inventory ([87, sidebar on p. 104]).

A final disadvantage of inventory, in the context of MTO production, is the following. Due to Little's Law the average response times are proportional to the average work in process, that is, the total in-process inventory. Thus a high work-in-process is a symptom of a system with a long response time. This is not only valid for many production processes, but also for most administrative processes.

10.5 Resources

Production is made possible by the presence of a number of resources. These resources consist of raw materials and parts on one hand, of production capacity on the other. The production capacity is determined by the availability of machines and the availability of people. Decisions concerning the availability of raw materials and parts are part of supply chain management, discussed in Chapter 11, especially Section 11.5. In this Chapter usually the production capacity, in terms of machines and personnel, is given, with the exception of Section 11.7, where we are concerned with issues related to the availability and repair of machines. Chapter 13 contains sections on scheduling employees and related longer-term problems that are also, to a certain extend, relevant to other areas.

10.6 Further reading

To have a good understanding of how firms function and the way they are organized it can be quite helpful to read at least once a general book on management or to follow a course on it. Nickels et al. [71] is such a book that is easy to read.

10.7 Exercises

Exercise 10.1 a. Explain the difference between “service product” and “product service”. b. Give an example of both in the context of planning software.

Chapter 11

Manufacturing

This chapter deals with models for production facilities. These models play a role in the design and in the every-day operational control of the plant. Design and scheduling cannot be considered separately, as (some of) the operational decisions already play a role when designing the system.

The most important distinction we make in this chapter is between *flow lines* and *job shops*. The lay-out, the objective and the modeling questions for flow lines and job jobs are very different. We start with modeling flow lines, which can best be characterized as production facilities for large quantities of similar products. Car manufacturing is the prime example. Next we discuss job shops, which are facilities suitable for more heterogeneous products.

After that we discuss project planning. The main conceptual difference between job shops and project planning is that the latter deals with one project at a time, while the challenge with job shops is to make a plan for all products simultaneously. Also reliability and maintenance is part of this chapter, as it is especially relevant to the production sector. The same holds for the final subject of this section, field service, because this concerns tangible objects.

11.1 Flow lines

A flow line is essentially a production facility consisting consisting of a number of machines and jobs going from one machine to the next in a next order. There are three aspects that need to be specified: the way in which new jobs are initiated at the first machine, the size of the buffers between the different machines, and the distributions of service time durations.

When outside orders initiate the production process (MTO) then it is not unreasonable to assume arrivals according to a Poisson process. However, flow lines are most often used for MTS production systems. In that case a production plan (a so-called Master Production Schedule) is made to which the facility should adher. In that situation the flow line arrival process can be chosen in the best way, for example as a deterministic process.

CONTINUE buffers-service time variability

We start this section with the basic mathematical model for a machine with random processing times and random order arrival times. As order arrival times we start with the Poisson process (see Chapter 2). This is usually a good choice, certainly in the case of an MTO system. Obviously, often the rate will vary over time, thus we have an inhomogeneous Poisson process. This makes the analysis quite complicated. Instead, we consider the system at peak level performance, thus for the highest possible arrival rate. It is therefore not unreasonable to take a constant arrival rate, and thus to consider a homogeneous Poisson process.

Let us first relate to Section **??** on MRP by considering a model without capacity restrictions, modeled by an infinite number of machines or servers (in queueing terminology). Assuming deterministic processing times of length β, and a due date for all orders d ($d \geq \beta$) time units after arrival, we see that jobs are scheduled $d - \beta$ time units after arrival, resulting in an $M|D|\infty$ queue with arrival rate λ and $\mathbb{E}S = \beta$ (we denote service times with S). According to formula (5.5) the probability of i jobs in the system $\pi(i)$ is given by:

$$\pi(i) = \frac{(\lambda\beta)^i}{i!}e^{-\lambda\beta},$$

a Poisson distribution. Jobs never wait for available capacity, and are always ready right on time: The response time (the time between the order and the delivery) is always equal to d. The price to pay is that there is no upper bound to the processing capacity, any number of busy machines has a positive probability of occuring.

Evidently this is an ideal situation that will hardly ever occur in practice. In general production capacity is limited. Let us assume that there is a single machine, the situation that occurs most frequently in practice. We model this situation as an $M|D|1$ queue. We have to assume that $\lambda\beta < 1$, otherwise the queue is unstable and will build up to ∞. Waiting occurs, no matter how small $\lambda\beta$ is. Consider the response time W. By the famous *Pollaczek-Khintchine formula*, Equation (5.4), its expectation equals

$$\mathbb{E}W = \frac{\lambda\beta^2}{2(1-\lambda\beta)} + \beta.$$

The same formula can also be used to show the effects of varying service times. These variations can be caused by irregularities in the production process, or by the fact that different orders are placed with different requirements, leading to different processing times. For example, assume that the service time is exponentially distributed with mean β. Note that in this case $\mathbb{E}S^2 = 2\beta^2$. Therefore the expression for $\mathbb{E}W$ becomes:

$$\mathbb{E}W = \frac{\lambda\beta^2}{(1-\lambda\beta)} + \beta.$$

The part of the response time that is spent waiting has doubled!

From Equation (5.4) we see that $\mathbb{E}W$ is an increasing function of λ, $\mathbb{E}S$, and $\mathbb{E}S^2$. Otherwise stated: $\mathbb{E}W$ increases with the load $\lambda\mathbb{E}S$ (which is related to the (over)capacity) and with $\mathbb{E}S^2$, that quantifies variations. Note that $\mathbb{E}W$ tends to ∞ as the load approaches one. This is an important consequence of the formula.

The variability in processing times can have several reasons, one of them being failures of the machine. If these failures depend on the operation of the machine then they are called *operation-dependent* failures (as opposed to time-dependent failures). Operation-dependent failures can well be modeled by randomly extending service times with the repair time. It is clear that long repairs that occur infrequently can significantly increase $c^2(S)$, thereby leading to high waiting times. (Modeling maintenance in this way integrates the maintenance with the operation of the system, as opposed to what we do in Chapter **??**, where reliability and maintenance is studied without taking the operation of the system directly into account.)

Up to now we assumed Poisson arrivals. However, it is also interesting to consider arrival processes with non-exponential interarrival times. Unfortunately there are hardly any theoretical results for the $G|G|1$ queue. Therefore we use the approximation given in Remark 5.3.5. The approximation rightly suggests that the waiting time decreases as the variability of the interarrival time distribution decreases. In the extreme case of the $D|D|1$ queue there is no waiting at all and $\mathbb{E}W = \beta$. Otherwise said: the only way to have a reasonable $\mathbb{E}W$ under high loads is to have interarrival and service times with a very low coefficient of variation.

Now we consider flow lines in various situations. We always assume single servers. Extensions to multiple servers are of practical interest, but lead to little additional insights. We will start with infinite work-in-process (WIP) buffers, exponential service time distributions and Poisson arrivals. We will discuss the influence of finite WIP buffers and other service-time distributions in the context of the Just-in-Time (JIT) paradigm.

Also set-up times are not explicitly modeled, but if present, are assumed to be part of the service time. This is fine in the context of flow lines. Only in the case of batches of different jobs the set-up times should be modeled explicitly, at the beginning of each batch. This is therefore more relevant for job shops.

Infinite WIP buffers Let us start with the case of infinite work-in-process inventory. Making the usual assumption of Poisson order arrivals, we arrive at a queueing network model that is known as a tandem model: there are multiple stations in line and finished parts are transfered from one station to the next. A surprising result from queueing theory states that the queue lengths on any moment in a tandem system are independent and that the transfer processes from one queue to another constitute again Poisson processes; see Section 5.6. Therefore total queue length and average delay can be computed as the sum over multiple single-station models.

We give the formulas. Let there be K stations, with station k having service rate μ_k. The orders arrive at a rate λ. We assume that $\lambda < \mu_k$ for all k, and we define $\rho_k = \lambda/\mu_k$. Then, according to Equation (5.2), the expected queue length $L_Q(k)$ and waiting time

$W_Q(k)$ at station k are given by

$$\mathbb{E}L_Q(k) = \frac{\rho_k^2}{1-\rho_k} \text{ and } \mathbb{E}W_Q(k) = \frac{\rho_k}{\mu_k(1-\rho_k)}.$$

Summing over k gives the total stock waiting for production L_Q and the overall waiting time W_Q:

$$\mathbb{E}L_Q = \sum_{k=1}^{K} \frac{\rho_k^2}{1-\rho_k} \text{ and } \mathbb{E}W_Q = \sum_{k=1}^{K} \frac{\rho_k}{\mu_k(1-\rho_k)}.$$

Equivalently, using equation (5.3), we find as total work in process L and as expected response time W

$$\mathbb{E}L = \sum_{k=1}^{K} \frac{\rho_k}{1-\rho_k} \text{ and } \mathbb{E}W = \sum_{k=1}^{K} \frac{1}{\mu_k(1-\rho_k)}.$$

Example 11.1.1 We consider an administrative process at a financial institution. On average there are 10 requests coming in daily, requiring 3 consecutive processing steps, with daily capacities of 15, 12, and 20, respectively. The average delays are therefore for the consecutive workstations estimated by 0.13, 0.42, and 0.05 days. If we compare the total delay of 0.60 days with the total processing time of 0.20 we see that the request spends 75% of the time in process waiting for processing! This is not an unusual number, and in this example the maximum utilization $10/12 = 83\%$ is relatively low.

In the example we saw that the average queue length and therefore also the expected response time is dominated by the slowest production step, the *bottleneck* of the system. This bottleneck also determines the maximal production rate of the system: *the service rate of the bottleneck is the maximal production rate.* Because of the importance of bottlenecks we can often restrict the analysis to bottlenecks only. Note that in the case of non-exponential service times it is not necessarily the slowest server that has the longest queue, as follows from the discussion in the previous section. The slowest does determine the maximal production rate, also in this case.

Remark 11.1.2 In the above we used results from queueing theory. Queueing network models allow for more general routing mechanisms (allowing for example for some of the jobs to visit a certain station), but for more general models (e.g., non-exponential service time distributions) simple expressions are lacking in general. Indeed, the output of an $M|G|1$ queue is not Poisson anymore. Thus a production facility with general service times has to be modeled as a network of $G|G|1$ queues. There are no simple formulas for performance measures of these networks, but approximations exist (see Remark 5.6.4). Using the same approach we can also model order arrivals that are not Poisson.

Flow lines with finite WIP buffers Up to now we considered models with the possibility to stock any amount of work in process (WIP) between the production steps. For the administrative process of the example this is perhaps realistic; for most standard production environments it is not. For this reason we study next a flow line or transfer line with finite in-process buffers.

Finite buffers have upstream and downstream consequences. Upstream, because machines sometimes have to stop producing due to the lack of in-process inventory space: *blocking*. Downstream, because finite buffers can reduce the overall output, leading to *starvation* of machines. Finding the response time of a transfer line with finite buffer space between the process steps is a complex problem, even if service times are exponential. The only general method that is available (and only in the case of exponential service times) is solving the multi-dimensional Markov chain, but the dimensionality of the state space prevents us from doing so for most models.

To get an idea of the performance of the system, let us isolate a single (bottleneck) machine. First we analyse the maximal *throughput* of this system. *Throughput* is defined as the average items produced per time unit. We can analyze the maximal throughput by assuming a very high order rate. Then the flow line up to the bottleneck is saturated, from the bottleneck on production occurs at the maximal throughput, assuming that the buffer after the bottleneck is big enough to avoid blocking of the bottleneck. Let μ_b be the service rate of the bottleneck machine, and let there be a total of N buffer places. As soon as a buffer space becomes available, replenishment occurs from the upstream machine. As this machine is also saturated, this occurs at its service rate μ_u. If the bottleneck buffer is full and the upstream process step finishes then the completed part (due to the blocking mechanism) stays at the upstream machine until there is room in the buffer. Thus there is actually room for $N+1$ finished parts. We redefine the buffer as containing all finished parts; a machine is blocked if it finds the downstream buffer full.

From the above discussion we conclude that the bottleneck behavior can be approximated by a $M|M|1|N+2$ queue, with μ_u as arrival rate and μ_b as the service rate. (Note that the Kendall notation we denote with the last number the maximal number of jobs in the system, not just in the queue.) We assume $\mu_u > \mu_b$, as we are considering a bottleneck. The maximal throughput of this system can be calculated as follows. Define $a = \mu_u/\mu_b$. The stationary probabilities of the total number of customers at the buffer and in service at the bottleneck are approximated by (see Section 5.4):

$$\pi(i) \approx \frac{a^i}{1+\cdots+a^{N+2}}.$$

The maximal throughput T_N, given by $\mu_b(1-\pi(0))$, can be written as

$$T_N \approx \mu_b \frac{a+\cdots+a^{N+2}}{1+\cdots+a^{N+2}} = \mu_b - \mu_b \frac{1}{1+\cdots+a^{N+2}}. \tag{11.1}$$

Thus the maximal throughput μ_b of the system with infinite WIP buffers is approximately reduced with a factor $(1+\cdots+a^{N+2})^{-1}$, which has its highest value when $\mu_u \approx \mu_b$, namely $(N+3)^{-1}$. Thus buffers should be biggest when the flow line is balanced (by which we mean that processing rates are approximately equal).

The approximation of T_{N-1} can be used to decide on the WIP inventory space. This way the bottleneck can be protected against starvation. Note that it should also be protected against blocking. For this reason the output buffer of the bottleneck, which is the input buffer of the downstream machine (in the case of neglectable transportation times), should be big enough. This results in a concentration of buffer space around the bottleneck machine. How much exactly can best be analyzed using simulation.

Simulation is also the best technique to use in the case of non-exponential service times. From the results of Section **??** it follows that a higher variability of service times lead to bigger fluctuations in queue length. Therefore highly variable service times necessitate high WIP buffers.

Our conclusion is that high inventories help smoothing the production process, they "decouple" the production steps. However, it can be costly, and it covers irregularities in the production process. For this reason the production paradigm *Just in time* (JIT) was designed: by reducing inventories irregularities are discovered. We discuss JIT next.

CONWIP The motivation for implementing JIT has mainly organizational reasons, related to reducing irregulaties in the production process and motivating employees. From a planning point of view reduced inventory and pull systems have also advantages, but it should not necessarily be implemented by limiting the WIP at each station. Another option is CONWIP: *constant work-in-process*. Here the restriction on the number of items is not for each process step separately, but for the whole production process. Of course coordination becomes more complex, and some of the qualitative advantages, such as the need of quick responses to disruptions, are partially lost. We analyze CONWIP.

When modeling we make a difference between how to treat finished products. Either we assume that finished products are consumed directly, or we have a Poisson order process where order are lost when the finished product buffer is empty. From an application point of view having the possibility of orders waiting is perhaps more interesting, but the resulting model is much harder to analyze. In the former case finished products are immediately replaced by unfinished products at the start of the flow line. In the latter case there is a Poisson process that takes away finished goods, at which they are replaced by unfinished products at the start of the flow line. Thus in both cases the beginning and the end of the flow line are attached, forming a cycle, where in the case of a Poisson order process an extra station, modeling this order process, is added.

This cycle allows for an exact mathematical analysis, see Section 5.6.5 on closed queueing networks. Let there be a total of K stations, with N_k the number of products queueing for or in production at station k. The stationary probabilities of the CONWIP model are given by:

$$\mathbb{P}(N_1 = n_1, \dots, N_K = n_K) = C^{-1} \prod_{k=1}^{K} (1 - \mu_k^{-1}) \mu_k^{-n_k}.$$

Here we assumed that the WIP is equal to some constant N, and therefore $n_1 + \cdots + n_K =$

N. The constant C is called the normalizing constant, it is equal to

$$C = \sum_{n_1+\cdots+n_K=N} \prod_{k=1}^{K} (1-\mu_k^{-1})\mu_k^{-n_k}.$$

Numerically this formula is rather attractive, an efficient algorithm exists for computing C (see Section 5.6.5). A closed-form expression does not exist. Queueing mainly occurs at the slowest server, certainly if the line is unbalanced. In this case already a small number of N suffices to obtain the highest possible throughput. In fact, the total WIP is lower than under JIT to obtain the same throughput. It can be shown that, for the same total inventory, CONWIP performs always better than JIT; thus the throughput of JIT is bounded from above by the throughput of CONWIP.

11.2 The Toyota Production System

There are several important objections against MRP, although it is at the heart of most current-day production scheduling systems. The first objection is that in reality processing times are not deterministic. The second is that processing capacity never is unlimited. Finally, physical goods take up space and therefore the amount of in-process inventory is limited. Thus, strictly obeying the MRP logic will lead to coordination problems within the production process: certain process steps are delayed because of capacity or inventory constraints or random events prolonging production, effects that often grow worse further down the line. Within MRP there is no mechanism to deal with this, although extensions to MRP (known as MRP II and ERP, see Section 11.5) are designed that can deal to some extent with capacity restrictions. Random processing times can only get a place within MRP by taking lead times long enough such that with a high probability processing times are shorter. This makes production times longer, leading to increased costs, long idle times of machines and a decrease in flexibility. Evidently, smarter production scheduling methods are sought for.

Just in time JIT can be seen as the Japanese answer to the American MRP systems, in the context of flow lines. It is decentralized and requires highly motivated employees. It is a logistics concept that controls production in a totally different way than MRP does. The *means* to control production is by reducing inventory space, which results in just-in-time production, by Little's law. These reduced inventory spaces make the production system react quickly to disruptions: blocking and starvation readily occur and propagate in both directions in the production line. The only way to react to this is by reducing the variations in the production time, and this is exactly what the goal of the reducing buffer spaces. The maximum stock between process steps is controlled using cards call *kanbans*. Every item in production should have a kanban attached to it. Kanbans are freed when the parts to which they are attached are taken into production at the next step. Reduction of inventory is obtained by taking away cards.

JIT is not the only management method focused at reducing irregularities. Another well-known example is *Six Sigma*. Consider some measure of a process step that has to stay within an upper and a lower bound. Six Sigma refers to the objective that the average outcome plus or minus six times the standard deviation should fall within the upper and lower bound, leading to less than two errors in a billion measurements.

Push versus pull With JIT production can start only if there is work in the input buffer and place in the output buffer (through a kanban). Thus production can and will be initiated by parts taken into production further down the line. Orders are satisfied in the same manner: finished products are taken out of a finite "finished products buffer". Such a system is called a "pull" system, in contrast with MRP or the flow lines studied earlier that are examples of "push" systems.

New set-up:
Flow lines
- tandem queues with finite buffers
- JIT/Kanbans

Job shops
- basic concept (BOM, lot sizes, etc)
- MRP
- rough cur/aggregate capacity planning
- production planning?

Project planning
Reliability
Field service

11.3 Terminology and characteristics

We consider first the production facility itself. We assume the order process as given; decisions concerning this (e.g., whether we produce in an MTO or MTS fashion) are discussed in Section 11.5. This also implies that when subassemblies are made to stock and final assembly is done to order, then we consider these process steps separately. Their coordination is part of supply chain management, which is discussed in Chapter 11.5.

We see a production system as consisting of machines, people, and inventory locations. The production process is concerned with the production of orders according to their specifications; management is needed to assure that this is done as good as possible. We discuss the characteristics of machines, orders, and production management.

Machines Each machine is capable of executing certain process steps. Characteristics of machines include:
- set-up times. This refers to the fact that when the type of process step is changed then it might be necessary to take some time to set the machine up for the new operation. This reduces the flexibility of the plant, explaining the desire to build *flexible manufacturing*

systems (FMSs), consisting of machines that have no or very short set-up times. Note that set-up times forces the management, for efficiency reasons, to group together production steps of the same type. Such a batch is called a *lot*;
- up and down times. Machines can be up and down, and this can either be planned (e.g., planned repair), or unplanned (e.g., a failure);
- processing times can vary, either because of different types of production steps or due to irregularities in the production process.

Example 11.3.1 A firm produces rolls of plastic in different colors and with different prints on it. The main machines are so-called *calenders* that produce the actual plastics. After that printing and cutting operations can be necessary. Changing from one product to another involves always a phase in which the raw materials of both products are mixed. Because of this set-up time it is preferred that production runs are long. However, this decreases flexibility and increases inventory levels.

Orders Every individual order or job has the following parameters or characteristics:
- arrival time and due date;
- bill of materials (BOM). The BOM, usually in the form of a tree, specifies the different production steps of an order and the parts that are needed to execute the process steps. The tree form with all its details is necessary for administrative purposes; for scheduling purposes it is usually sufficient to consider a line.

When we consider all orders together, we differentiate between *flow lines* (also called *transfer lines*) and *job shops*. In flow lines there are few types of orders, all following the same routing through the machines, and all requiring (almost) the same processing steps. The prime objective of flow shops is efficiency, to optimize production in such a way that costs are minimized. In job shops there are many different jobs following different routes. Controlling such a job shop can be a complex task, and therefore the scheduling aspects are very important. Resource utilization cannot be expected to be very high, for example because of set-up times. A good service level (in term of meeting due dates) is the prime objective.

Example 11.3.2 In the automotive industry, the BOM consists of all parts of a specific type of car. For planning purposes however, it usually suffices to conceptualize it as a (production) line. Car production is a typical example of a flow line.

Example 11.3.3 The plastics production plant of the example above is an example of a job shop: jobs are heterogeneous. They require different processing times, and do not all visit the same (types of) machines.

An important issue when matching machines and jobs is the load on each machine (group); the machine (or group of machines) with the highest load is called the bottleneck. How to deal with bottlenecks is one of the main issues in production scheduling. This brings us to the subject of production management.

Production management Production management, or, more specifically, production scheduling, has as goal to schedule production in such a way that the objectives are met as good as possible. The objectives are similar to those in supply chain planning, so we postpone the discussion until there. However, it is obvious that meeting due dates is important; minimizing in-process inventory often is as well.

Optimal production scheduling leads, mathematically speaking, to a policy where for every moment it is decided for every machine which order should be handled next. This means that decisions can vary with the number of orders, their due dates, machine calendars (e.g., indicating when maintenance is planned), etc. However, it is safe to say that certain rules of thumb exist that are followed by many good production schedules. These rules of thumb deal with:
- *lot sizes*, the number of similar jobs that is processed as a batch on a machine. Often lot sizes are predetermined, and independent of other circumstances;
- in-process inventory, the way to deal with jobs ready to be processed on the next production step. To what extend do we allow these inventories to build up, if at all possible due to space restrictions?
- production methodology, i.e., *push* or *pull*: is there some mechanism by which production is initiated at each step by taking away inventory from the output buffer, thereby pulling the products out of the factory, or is there some centralized processing system that initiates production orders at the first processing step, thereby pushing it slowly out of the plant? In the rest of this chapter we will focus on methods that can deal with stochastic processing times, capacity restrictions, and/or finite in-process inventory. First we make an important classification of production systems, between *flow lines* and *job shops*.

Flow lines and job shops When modeling production systems with the objective to analyse or optimize the performance not all dependencies and production steps of the BOM have to be taken into account. The BOM is necessary to execute a job correctly and completely, and for administrative reasons. However, when analyzing performance or making scheduling decisions, we can often restrict to a single path in the tree that the BOM essentially is. If this path is roughly the same for all orders, then we have a flow line; if this path is often different then we have a job shop.

There is a big difference in modeling objectives and tools between flow lines and job shops. In the former questions are often of a tactical nature: what is the capacity of the production system, how to place in-process inventory locations, how to maximize bottleneck performance, and so forth. For a given flow line the main performance indicator is the time an order spends in the process. The maximal production rate is also of interest, certainly in those cases where the long-run demand exceeds the production capacity. The models for analyzing these questions are often of a stochastic nature. Job shops on the other hand have interesting operational problems: how to schedule the different jobs on the machines such that due dates are met as good as possible, switch-over times are minimized, etc. These models are mostly deterministic.

In what follows we first discuss models for flow lines, after that we consider job shops.

We start with a model of a single production step. This will give us insight on the impact of limited capacity, random production times, and changes in load on response times.

Throughout we assume that the demand process is given. In a make-to-order environment this can be a Poisson process, for make-to-stock it can be more regular (even deterministic) or it can be that orders are placed in batches. When to initiate production in an MTS setting is part of production planning; see Section 11.5.

ERP systems In this section we discuss a standard production scheduling method, which is called material requirements planning (MRP). To do so, assume that there is ample processing capacity, and that processing times are deterministic. To avoid set-up times process steps are executed in fixed-size batches, the lot sizes. Lot sizes can change from process step to process step. This is a Make-to-Order ssystem (see Section 10.2), and orders arrive on time, i.e., due dates are such that they can be met without special measures.

Under these conditions an optimal (with respect to most possible criteria) production schedule is easily constructed. The first step is calculating back from the due date the gross requirements by *explosion* of the BOM one level. Of certain parts there is already inventory; taking account for this is *netting*, what results are net requirements. From the net requirements, using the known lead times, we calculate at which moment certain items should be produced. This is called *offsetting*. We cannot order items with any batch size, taking account of batch sizes is called *lot sizing*. Net requirements, together with offsetting and lot sizing, leads to the planned orders. This procedure is repeated for each level in the BOM, until all orders for all parts and fabrication steps are planned. This process is called material requirements planning. A detailed description can be found in every book on production logistics. Here we just give a simple example.

Example 11.3.4 It takes 2 hours to produce a lot of size 4 of a certain product, and 2 parts of a certain type are needed when production starts. There is an order for 10 products, 3 are still on stock. We need to produce 7 items, thus 2 lots of 4. To do so, we need 16 parts as soon as production starts, 2 hours before the due date.

11.4 Job shops

Up to now we considered flow lines. The main difference with job shops is that job shops have orders with highly heterogeneous manufactering requirements. For manufacturing purposes, orders in a job shop have the following characteristics:
- heterogeneous routes through the manufactering facility;
- set-up times and service times depending on the job type and the current operation;
- different due date requirements and/or different *holding* costs for keeping inventory (as discussed in Section 10.4).

These characteristics make it interesting not to treat jobs at a FCFS basis at the machines, but to schedule on the basis of for example set-up times or due dates. This we discuss in the rest of this chapter. In this section we look at models for a single machine, in the next section we look at real job shops with multiple machines.

Few production systems consist of a single machine. However, often one machine can identified for which planning is more important than other machines, because it is the bottleneck of the manufacturing system. In this section we concentrate on such a machine. Note that the maximal throughput of the whole system is determined by the bottleneck, and that most of the waiting occurs at the bottleneck.

First we consider the case of zero switch-over times, which occur in flexible manufactering systems. After that we consider non-zero switch-over times.

No switch-over times Consider scheduling heterogeneous jobs with known processing times on a single machine with an objective related to due dates. This is a difficult but well-studied optimization problem that needs to be solved on a daily basis, requiring a special-purpose decision support system with a module that solves the combinatorial optimization problem.

In the case of holding costs instead of due dates the optimal scheduling policy can be characterized, even if the processing times are random. We assume there are P classes of different jobs, with jobs of type p, $1 \leq p \leq P$, having service-time distribution S_p and total inventory or holding costs h_p. The objective is to minimize the total inventory costs. For this case the optimal scheduling policy is known. It is a *priority rule*, meaning that jobs of a higher priority are scheduled before jobs of lower priority. Renumber the job types such that $h_1/\mathbb{E}S_1 \geq \cdots \geq h_P/\mathbb{E}S_P$. Then the optimal schedule gives priority to classes $1, \ldots, p$ priority over the remaining classes $p+1, \ldots, P$, for all $1 \leq p \leq P$ (see Remark 5.5.3). Thus if the machine becomes available, then a part of the lowest class available should be processed. Note that if all holding costs are equal, then priority should be given to jobs with low expected processing times. This is called *Shortest Job First* in Section 5.3.

In case the jobs arrive according to a Poisson process, then we can also compute the average waiting times and response times of the jobs. The last part of Section 5.3 is devoted to these types of problems. Equation (5.13) gives the expected waiting time for any class, and Equation (5.14) gives the average waiting time over all classes. In Example 5.5.2 it is shown that the average waiting time is minimized by the policy that gives higher priority to lower expected service times. In Remark 5.5.3 weighted waiting times are studied.

Non-zero switch-over times For zero switch-over times, the short-term schedule has no influence on the long-term throughput or waiting times, as long as the server is utilized fully. Simply said: changing the order of two jobs has no consequence on the jobs scheduled afterwards. This is not the case anymore when there are non-zero switch-over times. Changing the order of jobs might also mean a change in switch-over times, and then the whole schedule might change. In the case of non-zero switch-over times, productivity and long-run response times are optimized when the time spent switching-over is minimized. For this reason it is preferably to schedule batches. In the case of equal priorities and similar service times for different types of jobs it is even preferable to work on a type until there are no more jobs left. This type of policy is said to be *exhaustive*. The queueing models that model these systems are called *polling models*.

Delay at bottleneck nodes in job shops is the main source of late deliveries. For this reason there is much pressure on the planner to change priorities regularly and to schedule often emergency orders. These are often small batches, and improve in-time deliveries in the short run, but lead to a decrease of productivity, an increased backlog and less in-time deliveries in the long run. For this reason bottleneck nodes should be scheduled such that productivity is maximized. Upstream nodes should be scheduled such that the bottleneck can produce optimally. At downstream non-bottleneck nodes the delivery dates should determine the schedule.

In practice the batch size or *lot* size is often fixed for each type of job. Note that we saw that as well in Section **??** when discussing MRP. This only makes sense in the context of MTS, where an entire lot can always be produced, even if the demand is lower. A lot, together with its set-up time, can now be considered as a single job at the machine. Using this idea the theory of Section 5.3 can again be applied.

Remark 11.4.1 In most production systems a job cannot be interrupted once it has started. In some systems however one can put aside a job, start working on another job, and finish the first job at some later moment. Examples are certain administrative processes. Contrary to the intuition this reduces the average waiting in certain situations. Whether or not this is the case depends on the distribution of the production times: e.g., in the case of an DHR service time distribution (see Section 1.5) the average waiting time is minimized by working on the job that has received the least amount of service.

In yet other systems a machine can split its processing capacity and work on multiple jobs simultaneously, while keeping the same total processing rate. (This puts it aside from multi-server systems, where the total processing rate depends on the number of jobs present.) A similar effect is reached when jobs are assigned short time slots in a cyclical manner, as it is done in certain multi-tasking computer systems. This model is therefore useful for information processing systems. The policy that consists of sharing the processing capacity in a equal way between all jobs present is called *processor sharing* (PS). From results in Section 5.3 it follows that PS reduces the average waiting time if and only if $c^2(S) > 1$, where $c^2(S)$ is the squared coefficient of variation of S.

Extending these ideas and models multiple machines is extremely complicated. For MTO environments decision support systems exist. The schedule is easlity represented by a Gantt chart: a table with the machines on the vertical axis, time on the horizontal axis and colored blocks indicating when a job is scheduled on a machine. By arrows the consecutive processing of a shop is shown. By click and drop the planner can change the place of an activity. Additional colors can be used to show when a job is late. Different optimization methods can be used to improve a solution, such as local search or approaching that plan the machines one by one while keeping the orders on the other machines fixed.

A complicating factor in certain production systems is the fact that certain operations can be done on one machine out of a pool of machines. Then the schedule should also decide on which machine to do which operations. Note that it can occur that machines have partially overlapping capacities, making that certain jobs can only be executed at a subset of the machines at a pool of machines.

Let us now discuss systems with set-up times and fixed lot sizes, in the context of job shops. High lot sizes make machines more efficient, but makes the arrival process for downstream machines more peaked, which leads to longer queues at downstream stations.

Thus the choice of lot size is determined by the place of the processing step in the production process. If the machine forms a bottleneck, then high lot sizes are advisable; if there is ample capacity, then lower lot sizes are better as to guarantee a smoother operation of downstream processing steps.

Example 11.4.2 We illustrate the influence of lot sizing on multi-stage job shop models with the following deterministic example. Consider two machines, both needed for the production of a certain item. Machine 1 is used for a range of N types of products, changing product causes set-up costs. After processing at machine 1 the items that we consider are processed by the dedicated machine 2, that needs no set-up time. The service time at machine 1 is $1/N$, at machine 2 equal to 1; thus when we take the other times at machine 1 in consideration then loads are comparable. Customers arrive in all classes at the first machine according to a Poisson process with rate $\lambda < 1$. There are holding costs at both stations. (We can assume that for the other $N-1$ product classes there is a similar dedicated machine after machine 1; we concentrate on one of them.)

First consider machine 1 in isolation. Then high lot sizes are preferable, to avoid high set-up costs. The next class should be the one which has the most items waiting to be processed. (It can even be optimal that the machine idles while there are still products.) Let the lot size be denoted by Q. Under moderate loads and high N we have for the total costs of a single type of items $c(Q)$:

$$c(Q) \approx \frac{K\lambda}{Q} + \frac{hQ}{2},$$

with, equivalent to the EOQ model, K the set-up costs, and h the holding cost rate. It is an approximation, because we did not take into account the interference between the different classes. The optimal value is the EOQ.

Now consider both machines. If N is high, then if a lot is being processed it is to the second machine as if a batch of Q items arrives. This causes per batch holding costs of $hQ(Q+1)/2$, on average thus $h\lambda(Q+1)/2$. Thus the total costs over both machines $c^T(Q)$ can be approximated by

$$c^T(Q) \approx \frac{K\lambda}{Q} + \frac{h(1+\lambda)Q}{2} + \frac{h\lambda}{2}.$$

We see that the holding costs h are multiplied by a factor $(1+\lambda)$; thus the lot size should be smaller than is optimal for a single machine.

11.5 Production planning

Even with an optimal production schedule it can occur that deadlines are not met, due to production capacity restrictions. Avoiding this requires production planning, which refers to the fact that we have to plan the moments at which we initiate production. That is the first subject of this chapter. Then we consider the interaction between different (production) facilities, for the whole supply chain. Here we do not look at the internal processes of a facility. In the coordination between facilities inventory plays a crucial role.

Capacity and demand In production scheduling the jobs in the system are given: what can be controlled is the processing order. As soon as we try to plan the release of jobs to the system as to account for the production capacity then we deal with production planning. The two different production systems MTO, make-to-order, and MTS, and make-to-stock, were introduced in Section 10.2. In both production planning needs to be done.

Under MTS we use forecasts on the basis of which we plan future stock levels. Under uncertainty we can never avoid short-selling; using the inventory theory of the next chapter stock levels can be chosen such as to balance costs for holding stock and costs related to short-selling.

Under MTO the planning possibilities are rather limited. Due dates can sometimes be negociated with the customer, taking into account the remaining production capacity. Production planning is mostly done in MTS environments, or in ATO (assemble-to-order) settings, for those steps lying before the customer order decoupling point.

Production decisions are taken at different levels, with the end-item at the top level and raw materials at the lowest level. MTO and MTS are thus different methodologies for initiating production, the first initiated by orders, the other by forecasts. This distinction can be made at all levels. The equivalent of MTO is then called *pull*, because the customer (whoever that may be) pulls the product out of the production facility. The equivalent of MTS is called *push*, because production is initiated at the beginning of the production process, and production continues until there are no more items to process at any of the machines.

A pull system pulls the part out of the production facility: associated with each production step or machine there is a stock level for the output buffer, as soon as this drops below the level new items are produced. As such, taking demand from the final output buffer gives a chain reaction that makes machines along the whole line start production (if they were not producing already).

Seen as such ATO is a combinating of pull and push: parts are produced using a push methodology, and assembly happens in a pull fashion.

MRP II In Chapter **??** we saw how capacity constraints (and variability in processing times) influence inventory levels and, as a consequence, response times. This is relevant to any company that owns part of the supply chain: due to capacity constraints production times can vary. Thus even if the lead time predicted by the MRP logic is acceptable, then the capacity constraints might prolong this lead time to a higher unacceptable level. This has consequences for the production schedule and, equivalently, the inventory policy. Let us study this in the context of MRP, material requirements planning (which was introduced in Section **??**).

We saw in Section **??** that MRP indicates only which resources are necessary, there is no adjustment between demand and resource availability. Indeed, using the production plan produced by MRP can give highly fluctuating resource utilization, that can exceed the capacity or give rise to high overtime costs.

This led to the introduction of *manufacturing resource planning*, often called MRP II (the old material requirements planning then becomes MRP I). Part of MRP II is CRP, *capacity requirements planning.* Next to the BOM, we can construct the BOC, the *bill of capacity.* The BOC gives, for each resource, the needed capacity. Using this we can compute easily the resource requirements, and see when they exceed current resources. Responding to these signals is more complicated, planning is necessary.

Central in MRP II is the MPS, the *master production schedule.* The MPS determines at what moment how many end-items should become available. The MPS is based on forecasts and/or outstanding (back) orders. Thus MRP can be part of an MTO environment where end-items show up in the MPS as soon as the order is made, or of an MTS environment where end-items are produced in advance to meet future demand. Of course mixed forms can be found as well.

To assure relatively low lead times stock should be kept within the company, although these need not necessarily be finished products: each company within the supply chain can work an in an MTO (make to order), ATO (assemble to order), or MTS (make to stock) fashion (see Chapter 10 for an explanation of these terms). With respect to the stock this just means that stock is placed at a different position. When a company works in an MTO fashion, then raw material is usually on stock. For ATO half-finished products form the inventory, for MTS of course the final products. The production fashion not only determines the location of inventory, it also influences strongly how much inventory should be kept. That inventory control can be useful, follows from the enormous amount of money caught in inventory; reducing inventory only by a small percentage can give enormous profits.

Which production fashion is best depends on a couple of facts, among which are the following:
- whether orders are client specific, in the sense that production or assembly depend on customer specifications (the service aspect);
- to which extent a transformation adds value to the item;
- the time between the order and the due date.

The more orders are client-specific the better it is to initiate transformations by orders, as it avoids big inventories of finished items. Instead of that a smaller inventory of unfinished parts or raw material can be kept. If a transformation adds little value to the item, than the price is in the parts. When there are few end items, then the transformation should be done and end-products should be sold from stock, thereby having small lead times and generating income early. An exception is when the lead times are such that even the parts can be bought at order. When the transformation adds a lot of value to the item then it depends on whether these costs are fixed costs or not. In the case of fixed cost the transformation should also be initiated as soon as possible, thereby producing in an MTS fashion.

The MRP steps constitute the basic MRP I. We saw that as soon as demand management is included, with forecasts and the MPS, we speak of MRP II. Also concerns related to available resources are part of MRP II. As soon as even more functionality is added (such as financial modules), we speak of *Enterprise Resource Planning* (ERP).

ERP systems can be seen as transaction systems, that record all transactions within the company using a fixed logic, without any form of intelligence and/or optimization. It has been recognized that this is a missed opportunity. Currently ERP system builders are constructing tools on top of the transaction software that will allow for an intelligent use of the available data. These tools are called *Advanced Planning Systems* (APSs). The idea is that multiple optimization methods are implemented in these systems, often based on mathematical programming. A typical example is *Aggregate Production Planning*. This planning method tells you, given capacity constraints, when to produce as to meet the MPS. It leads to finished-products inventory, and minimizes the total inventory costs.

Aggregate production planning While discussing the basic MRP we saw that a major drawback is the inability to control resource utilization. Top-level ways to manage resource utilization are assembled under the name *aggregate production planning*. It is aggregate in the sense that not all details are modeled. We assume that the production for each item demands utilization of certain resources, without temporal constraints.

We assume there are N different products, M resources (machines, labor), and T time periods to plan. Costs consist of a linear function of production costs and inventory. Typical costs related to inventory are discussed in Section 10.4, but it will be clear that early production can mean big investments.

We use the following notation:

- x_{nt} is the amount of products n produced in interval t (the decision variable);
- d_{nt} is the demand for product n at the end of interval t;
- s_{nt} is the amount of inventory of product n at t;
- h_n are the costs of holding one unit of product n one unit of time in inventory (the *holding costs*);
- r_{mt} is the amount of resource m available in interval t;
- u_{nm} is the amount of resource m needed for the production of one unit of n.

Now the production and inventory costs are minimized by the following linear program:

$$\text{minimize} \sum_{t=1}^{T}\sum_{n=1}^{N} h_n s_{nt}$$

subject to

$$s_{nt} + x_{nt} - d_{nt} = s_{nt+1},\ n = 1,\dots,N,\ t = 1,\dots,T;$$

$$\sum_{n=1}^{N} u_{nm} x_{nt} \le r_{mt},\ t = 1,\dots,T;$$

$$x_{nt} \ge 0,\ n = 1,\dots,N,\ t = 1,\dots,T;$$

$$s_{nt} \ge 0,\ n = 1,\dots,N,\ t = 2,\dots,T+1,\ s_{n1} \text{ given.}$$

Many more features can be added, including overtime (thereby weakening the strong resource capacity constraints), time dependent costs, etc. Note that this is a linear program, and thus the answers need not be integer. If necessary integer constraints can be added.

The supply chain In the last sections we looked at the planning within a single facility given the Master Production Schedule. It is the MPS that shows the interaction of the facility with the downstream activities in the supply chain. In the rest of this chapter we consider the interaction between the different actors in the supply chain. To one actor the internal processes of the other are irrelevant. All that counts for a supplier is the demand process, which it translates in the MPS, taking its internal processes into account. What counts for the customer is the lead time of the supplier. Deriving the MPS consists of multiple steps, depending on the way a supplier works. In the case of MTO and MRP I current demand is entered into the MPS by calculating back from the due dates using MRP logic. Different possibilities exist if we want to take capacity restrictions into account. Given that due dates are already determined one could use aggregate production planning. In case the due dates are not yet determined yet, a technique like simulation could help to determine what a realistic due date would be given current order levels. In the case of MTS order fulfillment should be immediate. Thus the required stock level should be calculated using an appropiate inventory, in which the production time of the manufactering plant figures as the lead time in the inventory model.

From now on we consider the interaction within the supply chain, we do not look at the processes within the nodes. Thus the capacity constraints that played a role in determining the production times are now replaced by lead times. This also makes it possible to allow for other activities in the supply chain next to production: assembly, transportation, or a combination of these. They all change one or more characteristics of the item or (future) product. Therefore we call them *transformations*. Inbetween transformations items have to be stored, therefore it is assumed that there are stock points between activities. Thus a supply chain can be seen as an alternating sequence of activities (represented by lead times) and items on stock.

The linear representation of a supply chain is a little too simple. When looking at a single end-product at the end of the supply chain, then we see in general not a chain but an *in-tree*; i.e., we see a directed tree where each node has only one outgoing arc and possibly multiple incoming arcs. This is also called *converging*. When considering more than one end-product we see for example that they are distributed to different outlets: a *diverging* topology. Often we see both. When making the distinction between production and distribution we often see a converging topology up to the final assembly and after that divergence. This is the case when there is a single assembly point.

To profit from economies of scale transportation should be combined. This calls for combining deliveries and optimizing routes for these combined shipments. This type of routing problems is discussed in Chapter **??**. To make use of these scale advantages related to transportation most companies have *distribution centers* (DC). Additional advantages of scale can be obtained by outsourcing the distribution to a specialized company that does the distribution for multiple companies. In these DCs finished goods are shipped (sometimes after being assembled) together with products destined for a single location or a group of geographically close locations.

DCs carry cycle stock and safety stock: lead times and lot sizes tend to be longer upstream than downstream. The high lot sizes upstream and the low lot sizes downstream necessitate cycle stock. The long lead times upstream force companies to hold big stocks in DCs to avoid running out of products. As long as the DC has a high delivery reliability stock in sales outlets can be kept relatively low. The safety stock at the DC can be used more flexible (it is not yet decided to which outlet it will go, and therefore there are scale advantages), and holding inventory at a DC is less expensive than holding it at the outlet, because the DC is built and optimized for keeping stock (while the outlet is usually optimized for selling).

As a result the space in the outlets can be used for more different items and expensive inventory locations at outlets become obsolete. Deliveries to outlets should be frequent.

The coordinated supply chain In the supply chain as we considered it up to now every order is negotiated independently, and companies can change supplier whenever they want. This leads to orders of high quantities, i.e., big lot sizes: discounts can be negotiated and overhead is kept low. Also decisions about lot sizes are taken for each actor in the supply chain independently. This leads to suboptimization: what minimizes costs at a single station does not necessarily minimize costs for the whole chain. Thus when every actor in the supply chain minimizes its own costs there will be more inventory then is optimal for the whole supply chain. There will also be much safety stock: downstream demand will also occur in big batches, and there is no information exchange about future order moments.

The solution to these problems is two-fold: deliver frequently with short lead times small lot sizes and send inventory information to upstream actors such that they can anticipate on future demand. Current ICT technology allows companies to exchange information on their stock positions. Thus orders need not be unexpected anymore: they become predictable when one has access to other's stock levels and re-order policies. Of course this demands an extensive cooperation between the companies, including agreements on delivery service levels.

Using all information that comes available in an optimal way is extremely complicated: optimal production and order policies depend on all stock points. This is computationally infeasable and also impossible to implement. The solution to this the use of *echelon inventory*, which is all stock from a certain point on. Thus, when controlling inventory in a DC that delivers outlets, the inventory position of the outlets is added to that of the DC. This is the echelon inventory.

Responsabilities and cooperation between companies remains an issue. Therefore some companies have taken full responsability of their downstream inventory, even if outlets are not owned: 'vendor-managed inventory'. Thus in this situation the decisions concerning orders are not taken by the outlets themselves, but by their suppliers.

The idea of better managing inventory by using additional information can be taken a step further, namely to removing stock points. This is also made possible by decreased lot sizes that made more frequent deliveries necessary, and the improved cooperation that made deliveries more reliable. An example of this is cross-docking. This lets the supplier or producer prepare the shipments directly for the outlets, depending on their inventory levels. The whole shipment is delivered at the DC, from where is it immediately sent to the outlets. Thus the DC loses its inventory function, but keeps its transportation function.

Example 11.5.1 A retail organization uses cross-docking mainly for its fast movers. Slow movers are kept on stock at the DC. For products on stock there is a lead time of 1 day, for the cross-docking products 2 days (given that the product is on stock at the DC or at the supplier).

By removing stock points along the whole supply chain we see that the old image of every company within the supply chain having its own CODP disappears, and what remains is a completely coordinated chain with a single customer-order decoupling point. Due to reduced lead times this is pushed higher upstream, thus assembly to customer specifications is better possible.

11.6 Project planning

In logistics the objective was to produce (and distribute) *similar* items in a *similar* way. If the way items are produced is *unique*, then we call it a *project*. The management of projects is known as project management, part of which is project planning, the subject of this chapter. Of course, project planning can be used for all types of projects, not just in the area of production systems.

We define projects as follows.

Definition 11.6.1 *A* project *is a set of nonroutine activities and their interrelations meant to reach a specific goal.*

It is safe to assume that also the goal of the project is nonroutine: if the goal were routine then it is probably better to achieve this using routine activities. It is however possible to obtain a nonroutine goal through routine activities. In a production setting this is typically the case in a job shop (see Chapter **??**). Projects have in common with production processes that they consist of several process steps or activities.

The equivalent of the bill of materials (see Chapter **??**) in projects is the *Work Breakdown Structure* (WBS). The WBS specifies the project activities and their relations. It is often depicted using a graph model, in which the nodes represent the activities, and the edge the relations between the activities. There are a start and an end node indicating the begin and the end of the project.

In normal top-of-the-shelf products price and quality are the main attributes of a product. In projects (just as in MTO logistics) *time* also plays a very important role. In this chapter we concentrate on the time aspects of project management. The most important moment in a project is its finish time. In the next section we learn how to calculate the earliest finish time (EF) of a project, based on activity durations and resource availability.

Critical path calculation The EF of a project is determined by its starting time (ST) and the starting and finish times of the activities. To be able to schedule these we have to know activity durations and resource availabilities. For the moment we assume deterministic activity durations and no resource restrictions. Consider a project consisting of N activities, numbered such that activity i has only lower numbered predecessors. This implies that node 1 is the start node, and node N the finish node. Let d_i indicate the duration of node i. (It can be that d_1 or $d_N = 0$, to assure a single start and finish node.)

For each node we calculate the earliest finish time (indicated with EF_i for activity i). Assuming that $ST = 0$, and thus $EF_1 = d_1$, we can calculate the EF_i for $i = 2, \dots, N$ with the formula

$$EF_i = d_i + \max\{EF_j | j < i, i \text{ has predecessor } j\}.$$

Of course, $EF = EF_N$.

Thus the EF_i give the earliest times at which the activities can be ready. Then $ES_i = EF_i - d_i$ represents the earliest time at which activity i can start. It is also interesting to determine the latest times at which activities should start, while keeping to the EF. Define LS_i as the latest time that activity i can start. Of course $LS_N = EF - d_N$. Now calculate LS_i, for $i = N-1, \dots, 1$, with

$$LS_i = \min\{LS_j | j > i, j \text{ has predecessor } i\} - d_i.$$

Now we can define for each activity its *slack*, defined as $S_i = ES_i - LS_i$. Activities with $S_i = 0$ are called *critical*. A set of k activities $\{i_1, \dots, i_k\}$ is called a *critical path* if all activities are critical and if $EF_i = ES_{i+1}$. For all critical activities the start times are known. For the non-critical they still have to be determined, the possibilities for activity i being all moment in the interval $[ES_i, LS_i]$. This completes the description of the *Critical path method* (CPM).

A schedule can be represented in a time-activity chart, where there is a horizontal line for the duration of each activity. Such a chart is called a *Gantt chart*. Traditionally, Gantt charts were the first means to graphically display project planning. They have the disadvantage that they immediately give a schedule and that the interdependencies between the activities are not or not very clear. In software packages we often see Gantt charts with additional features such the critical path in red, arrows indicating the precedence relations, and an additional color for the slack of the non-critical activities.

Random activity durations As along as activity durations are deterministic the project will come to an end at exactly EF. However, this is a very unrealistic assumption. While discussing production systems we already saw that routine activities can have random durations, and this holds even more for the non-routine activities that projects consist of. We discuss the influence of random activity durations on critical and non-critical activities.

The project duration is the length of a critical path which is simply the sum of the durations of the activities on the critical path. Thus if the duration of an activity is exceeded, then the project is also delayed with the same time. (If an activity is ready before time, then the project is ready early if the critical path is unique. However, this happens rarely.) That does not hold for the non-critical jobs, as long as the excess is smaller than the time that the activity started before the latest starting time. Thus we see how important it not to let activities start at their LS, because this makes this job critical. Thus from a mathematical point of view it is best to start each job at its ES. There can however be good reason not to do this. (E.g., ES_i corresponds to an activity end for an activity j with $j < i$. It might well be that a critical activity starts at the same time, and thus the project manager decides to delay the start of activity i to give all its attention to the critical activity.)

If we assume that the only project delay will come from the critical path, then it becomes interesting to estimate the probability distribution of the project duration. This distribution is often assumed to be normal, with as variance the sum of the variances of the critical activities. This method of estimating the distribution of project durations, together with the method to compute the critical path, is known as *Project evaluation and review technique* (PERT). PERT is an old technique from the fifties. Currently better techniques are available if precision is wanted. For example simulation can be used, avoiding the normal approximation, and also taking the influence of non-critical jobs into account.

By now it must be clear that estimations of activity durations play an important role in project planning. For completely human reasons employees have the tendency to add some safety time to these estimations. Next to this, employees have the tendency to delay working on an activity until it is really necessary, thereby fulfilling the overly prudent estimate. These two effects (amongst others), according to Goldratt, make that projects are never finished on time and take much longer than really necessary. His proposal is to remove these safety “buffers” and replace them by a single buffer at the end of the project. To avoid that non-critical activities become critical he also adds a safety buffer everywhere a non-critical activity precedes a critical one. This also solves the problem when to start non-critical activities. In this way there is no more safety time added than necessary and by surveying how much of the buffers is used one can decide whether or not the project is on schedule. This method is very attractive because it works with deterministic estimates and finish time while it takes account of the randomness in the activity durations.

Additional topics Just as in production planning is resource use of major concern. If multiple activities use the same resource, then it can be avoiding that they are simulataneously scheduled by adding a precedence relation among them. (Text to be added.)

We only discussed precedence relations where an activity could only start if another one was finished. Other relations (such as an activity can only start if another one has started) exist as well, requiring adaptations of the critical path calculations.

Finally, when managing projects, we should realize that there are always other projects in parallel requesting the same resources. Goldratt states that prioritizing among the activities from different projects shortens the project durations. However, in Chapter 5 we learned that this only holds for activities with a low variability.

There are many software tools for project planning. We name only a few. Microsoft Project is probably used most. The Primavera Project Planner (`www.primavera.com`) is one of the most extensive ones, and Prochain (`www.prochain.com`) is an add-in to Microsoft Project that allows the user to work with the critical chain methodology.

11.7 Reliability and Maintenance

In this section we study reliability and maintenance.

Reliability is the study of systems that can fail or stop functioning. Maintenance deals with replacing or repairing components of unreliable systems with the aim of improving the availability of the systems.

Reliability and maintenance becomes more and more important in our society. The systems around us become increasingly complex; at the same time individuals and companies depend on an increasing number of complex systems for their daily functioning.

First we study the general theory of reliability, then we turn to maintenance. Before that we give some definitions and we make some general remarks.

In reliability theory we assume that a system consists of *components*. Whether or not a system is functioning depends on the components. It is not the case that all components should function for the system to function; this dependence can be more general. The level at which we study the system can differ: sometimes we see the system as consisting of a single component, and sometimes we consider the functioning of components. Another appropriate term for component is *subsystem*.

Availability deals with the functioning of a component or system at a specific point in time (also called *pointwise* availability). Reliability is more general, and deals also with the time-dependent behavior.

Sometimes it is hard to define what the goal is of the study of reliability and maintenance policies. In many systems reliability strongly influences the overall objective, but it is hard to set a specific objective for the reliability. Incorporating reliability in the overall objective is desirable, but often impossible.

Example 11.7.1 In a production system machines are prone to failure. Ideally a model for production scheduling should take into account at the same time the parts that circulate at the factory floor and the failures of the machines. This model however would often be too big to solve. Instead, while making the production schedule, it is assumed that the machines are always functioning. For determining the maintenance policy an availability target is set.

For industrial applications such as in the example maintenance plays a crucial role in obtaining the desired reliability.

In other systems reliability is a goal by itself. Indeed, an airplane should function during its whole flight. As such, maintenance does not seem to play a role. However, seen over a longer period, maintenance on components is crucial to assure that the airplane is reliable during all of its flights.

Reliability of a single component In this section we consider single components, without maintenance. The usual way to represent the life time of a component is by its hazard rate. We define certain properties of positive r.v.s related to hazard rates, as defined in Section 1.5.

Definition 11.7.2 *We call a positive r.v.* IHR *(*DHR*) if its hazard rate is non-decreasing (non-increasing).*

Of course IHR (DHR) stands for Increasing (Decreasing) Hazard Rate. This definition holds only for distributions with a density; a more general definition can be formulated that would include more distributions. For example, under this more general definition a constant would also be an IHR distribution.

To get a better understanding for what IHR (DHR) means, we derive another expression for $\mathbb{P}(X \leq t+h|X>t)$:

$$\mathbb{P}(X \leq t+h|X>t) = \frac{\overline{F}(t) - \overline{F}(t+h)}{\overline{F}(t)} = \frac{e^{-\Lambda(t)} - e^{-\Lambda(t+h)}}{e^{-\Lambda(t)}} = 1 - e^{-\int_t^{t+h} \lambda(s)ds}.$$

If λ is increasing (decreasing), then $\mathbb{P}(X \leq t+h|X>t)$ is increasing (decreasing) as a function of t. Thus X IHR (DHR) states that the remaining life time will get shorter (longer) in distribution as a function of t, given that the life time is longer than t.

Life times, and thus the hazard rates are influenced by the following types of physical events:
- **burn-in**. This is the phenomenon that a new component might fail early, for example due to construction errors;
- **wear-out**. The phenomenon that a component deteriorates in time;
- **external events**. The fact that components can fail due to reasons not related to the component, but due to other components or the environment.

Let us consider different classes of components. If a component is not subject to burn-in or wear-out, but only to random failures independent of age, and to external events, independent of time, then it makes sense to choose a constant hazard rate. This implies that the life time is exponentially distributed. If there is only burn-in (and possibly failures independent of age and external events), then the life time distribution is an DHR distribution. Of course, most manufacturers of components try to avoid burn-in effects, by testing products before they are shipped (avoiding initial failures), or by introducing burn-in periods, before selling the products.

In case of only wear-out, the life time distribution comes from an IHR distribution. Of course, wear-out is hard to avoid, and many components therefore have an IHR distribution.

If a component is both subject to burn-in and wear-out, then its hazard rate will likely have a "bath-tub" shape. In the *burn-in phase* failures are mostly due to burn-in. After that we have a phase where failures are rare, called the *change phase*. Finally wear-out starts playing a major role as we enter the *wear-out phase*.

In many cases it is not sufficient just to predict the life time distribution of a component, instead we want to have a better approximation as time continues. This means gathering additional information on the component. There are in principle two (partly overlapping) ways to do this. The first consists of dividing the component in subcomponents and gathering information whether their life times are expired. The second is *condition monitoring*, where the level of wear-out of a component is measured. By obtaining in this way additional information the life time distribution is not changed (unless maintenance is performed), but while information is gathered, the moment of failure can be better estimated.

Up to now we tacitly assumed that the time to failure does not depend on the way the component is used, but only on the time. This is called *time-dependent failures*. Next to time-dependent failures we have operations-dependent failures. These failures depend on the way the component is used. Operations-dependent failures will not be studied in this chapter, because we do not take the interaction with for example the parts that are produced into account. Note that if we define the time as the time that the system is used, then time- and operations-dependent failures might well coincide.

Example 11.7.3 In a production line certain machines are always on, whether they are used or not. Therefore their failures are time-dependent. Certain tools only wear out when they are used for producing parts. Often we see that failures are neither fully time-dependent nor failure-dependent, but a mixture of both.

Availability of systems In this section we consider the relation between the pointwise availability of the system and its components at a specific point in time t, without taking the dynamics into account. Throughout this section, let y denote the state of the system: $y = 0$ means that the system is down, $y = 1$ means that the system is up. Similarly, let y_i denote the state of the ith component. If the state of a component is not known but a r.v., then it is denoted with Y_i. In this case the state of the system also becomes a r.v., denoted with Y.

Let $\phi : \{0,1\}^n \longrightarrow \{0,1\}$ be the function that indicates, based on the state of the components, whether the system is functioning or not. We call ϕ the *system function*.

If the availabilities of the components are r.v.s, then also $Y = \phi(Y_1, \dots, Y_n)$ is a 0/1-valued r.v. It is our objective to calculate the system availability $\mathbb{P}(Y = 1) = \mathbb{E}Y = \mathbb{E}\phi(Y_1, \dots, Y_n)$. This can be very cumbersome, certainly if there are dependencies between the components. For this reason we assume that the components are modeled in such a way that all the Y_i are assumed to be independent.

Example 11.7.4 The main office of a company is connected with it computing center through two separate connections hired from different companies. However, these connections share the same physical cable. Thus software problems occur independently, but hardware problems (e.g., an excavator breaking the cable) are dependent. To make all components independent the connections could me modeled by three components: one modeling the physical connection, two modeling the software connections. (This occured in real life, the connections shared the same deep-sea cable without the company knowing that. When a fishing boat broke the cable the companies fleet of trucks was grounded for a couple of days.)

$(Y_1, \dots, Y_n)$ is completely characterized by the probabilities $p_i = \mathbb{P}(Y_i = 1)$, because all the Y_i are independent. We define the function $\Phi : [0,1]^n \longrightarrow [0,1]$ as follows:

$$\Phi(p_1, \dots, p_n) = \mathbb{E}\phi(Y_1, \dots, Y_n) = \mathbb{P}(Y = 1).$$

Assuming that the Y_i are independent and that ϕ is known, then

$$\Phi(p_1, \dots, p_n) = \sum_{(y_1,\dots,y_n)\in\{0,1\}^n} \prod_{j:y_j=0} (1 - p_j) \prod_{j:y_j=1} p_j \, \phi(y_1, \dots, y_n).$$

Note that the summation has 2^n terms. Thus simply enumerating all possible values of $(Y_1, \dots, Y_n)$ is only an option for small systems. Later on in this section we study a method for finding Φ for general systems, but first we look at the system function for some special configurations.

Series and parallel systems Systems where reliability plays a minor role are often designed such that every component is crucial for the system to function. This is the simplest possible structure for a system, often called a series structure. Its system function is given by $\phi(y_1, \dots, y_n) = \min\{y_1, \dots, y_n\} = \prod_{i=1}^n y_i$. The last representation is also valid for independent random availabilities, i.e., $\Phi(p_1, \dots, p_n) = \prod_{i=1}^n p_i$.

Clearly a series system will often lead to a low availability, as every component needs to function.

Example 11.7.5 A series system consists of 100 independent components, each having a 99.9% availability. The availability of the system is thus $0.999^{100} \approx 1 - 100 \cdot 0.001 = 0.9$. Complex systems can have much more than 100 components.

An often used method to improve the availability of a system is adding *redundancy*. This can for example by installing the same component several times in parallel. Let us consider such a parallel system. Suppose we have n parallel components, and only one needs to function to assure that the whole system functions. For this system $\phi(y_1, \dots, y_n) = \max\{y_1, \dots, y_n\}$. For now and later use we derive the following lemma.

Lemma 11.7.6 For arbitrary $y_i \in \{0, 1\}$, $\max\{y_1, \dots, y_n\} = 1 - \prod_{i=1}^n (1 - y_i)$.

Proof $1 - \max\{y_1, \dots, y_n\} = \min\{1 - y_1, \dots, 1 - y_n\} = \prod_{i=1}^n (1 - y_i)$. □

Thus we also have $\phi(y_1, \dots, y_n) = 1 - \prod_{i=1}^n (1 - y_i)$. Taking expectations gives $\mathbb{E}Y = \Phi(p_1, \dots, p_n) = 1 - \prod_{i=1}^n (1 - p_i)$.

Example 11.7.7 A parallel system consists of 2 independent components, each having a 99.9% availability. The availability of the system is $1 - (1 - 0.999)^2 = 0.999999$. Now if we assume that in the previous example every component consists of two parallel subcomponents, then the availability becomes $0.999999^{100} \approx 1 - 100 \cdot 0.000001 = 0.9999$. An alternative would be to take two parallel systems, then the availability becomes 0.99.

A generalization of a parallel system is the so-called k-out-of-n system, which means that out of n components at least k should function. The system function ϕ is given by $\phi(y_1, \dots, y_n) = \mathbb{I}\{\sum_{i=1}^n y_i \geq k\}$.

Example 11.7.8 The k-out-of-n system with $k = 2$ and $n = 3$ has structure function

$$\phi(y_1, y_2, y_3) = \mathbb{I}\{y_1 + y_2 + y_3 \geq 2\},$$

and

$$\Phi(p_1, p_2, p_3) = p_1p_2(1 - p_3) + p_1(1 - p_2)p_3 + (1 - p_1)p_2p_3 + p_1p_2p_3 =$$
$$p_1p_2 + p_1p_3 + p_2p_3 - 2p_1p_2p_3,$$

which follows from enumeration.

For general k and n there is no expression known for $\Phi(p_1, \dots, p_n)$. An exception is when $p_i = p$ for all i. In this case it is easy to see that

$$\Phi(p, \dots, p) = \sum_{i=k}^n \binom{n}{i} p^i (1 - p)^{n-i}.$$

General systems Now we consider general unreliable systems. We will make the following assumptions on the structure functions:
- $\phi : \{0, 1\}^n \longrightarrow \{0, 1\}$ with n the number of components;
- $\phi(0, \dots, 0) = 0$ and $\phi(1, \dots, 1) = 1$;
- ϕ is increasing, i.e., $\phi(y) \leq \phi(y')$ if $y_i \leq y'_i$ for all i.

Let us introduce the following notation: e_S is an n-dimensional vector with entries $(e_S)_i = 1$ if $i \in S$, 0 otherwise. (Thus $e_{\{i\}}$ is the usual ith unit vector.)

Definition 11.7.9 *A* minimal path set *is a set of components $S \subset \{1, \dots, n\}$ such that:*
- *$\phi(e_S) = 1$;*
- *$\phi(e_{S'}) = 0$ for all $S' \subset S$ with $S' \neq S$.*

The following result holds.

Theorem 11.7.10 *For a specific structure function ϕ, let $S_1, \dots, S_m$ be the collection of minimal path sets. Then*

$$\phi(y_1, \dots, y_n) = \max_{i=1,\dots,m} \prod_{j \in S_i} y_j. \qquad (11.2)$$

Proof For a vector y let A be the set of functioning components, i.e., $e_A = (y_1, \dots, y_n)$.

Assume that the r.h.s. of Equation (11.2) equals 1. Then there is a minimal path set (m.p.s.) S with $A \supset S$ and $\prod_{j \in S_i} y_j = \phi(e_S) = 1$. Because ϕ is increasing we find $\phi(A) = 1$.

Now assume $\phi(A) = 1$. One by one we try to make the components non-functioning, in such a way that the system remains up. Call the resulting set S. It is readily seen that S is a m.p.s., and for this reason the r.h.s. of Equation (11.2) equals 1.

Thus the l.h.s. of Equation (11.2) is 1 iff the r.h.s. is 1. Equality holds because 0 and 1 are the only possible values. □

Thus every system can be seen as consisting of several series systems in parallel. Note however that the same component may occur in several different series of components!

Example 11.7.11 The 2-out-of-3 system has as minimal path sets $\{y_1, y_2\}$, $\{y_1, y_3\}$, and $\{y_2, y_3\}$, and thus as structure function

$$\phi(y_1, y_2, y_3) = \max\{y_1y_2, y_1y_3, y_2y_3\}.$$

We see that every component appears in two series.

We can compute $\mathbb{E}\phi(Y_1, \dots, Y_n)$ in the following way. By Lemma 11.7.6

$$\phi(y_1, \dots, y_n) = 1 - \prod_{i=1}^{m} \Big(1 - \prod_{j \in S_i} y_j\Big).$$

The product can be worked out (e.g., using a symbolic manipulation package such as Maple), which results in a sum of series. It should be noted that $y_i^2 = y_i$ as $y_i \in \{0, 1\}$, which simplifies the expression enormously. Now all y_i can assumed to be random variables, and taking the expectation is nothing else than replacing all Y_i by p_i. This way we obtained a general method for computing Φ.

Example 11.7.12 Applied to the 2-out-of-3 system this method gives

$$\phi(y_1, y_2, y_3) = 1 - (1 - y_1y_2)(1 - y_1y_3)(1 - y_2y_3) =$$

$$y_1y_2 + y_1y_3 + y_2y_3 - y_1^2y_2y_3 - y_1y_2^2y_3 - y_1y_2y_3^2 + y_1^2y_2^2y_3^2 = y_1y_2 + y_1y_3 + y_2y_3 - 2y_1y_2y_3,$$

using the fact that $y_i^2 = y_i$. Thus

$$\mathbb{E}\phi(Y_1, Y_2, Y_3) = p_1p_2 + p_1p_3 + p_2p_3 - 2p_1p_2p_3,$$

the same as what we obtained in Example 11.7.8.

Reliability of systems Up to now we just considered the availability of a system at some (unspecified) point in time. In this section we will also take account of the life time distributions of the components.

Series systems Let X denote the life time of the series system, and X_i the life times of the independent components, for $1 \leq i \leq n$. Denote with λ the hazard rate of the system and λ_i the hezard rates of its components. Again, let Y (Y_i) be the state of the system (component i) at some time t. Then $\mathbb{P}(X \geq t) = \mathbb{P}(Y = 1) = \prod_{i=1}^n \mathbb{P}(Y_i = 1) = \prod_{i=1}^n \mathbb{P}(X_i \geq t)$. This gives a simple way to compute the system life time distribution based on the component life times.

An interesting question is if X inherits certain properties of X_i. We have the following properties.

Theorem 11.7.13 *For series systems*

$$\lambda(t) = \sum_{i=1}^n \lambda_i(t),$$

and consequently X is IHR (DHR) if all X_i are IHR (DHR).

Proof From $\overline{F}(t) = \mathbb{P}(X > t) = \prod_{i=1}^n \mathbb{P}(X_i > t) = \prod_{i=1}^n \overline{F}_i(t)$ it follows that $f(t) = -\frac{d}{dt}\mathbb{P}(X > t) = \prod_{i=1}^n \overline{F}_i(t) \sum_{i=1}^n f_i(t)/\overline{F}_i(t)$. Therefore $\lambda(t) = f(t)/\overline{F}(t) = \sum_{i=1}^n f_i(t)/\overline{F}_i(t) = \sum_{i=1}^n \lambda_i(t)$.

If λ_i is increasing (decreasing) for all i, then so is $\sum_{i=1}^n \lambda_i(t)$. □

Parallel systems We continue our analysis for parallel systems. Here we find $\mathbb{P}(X \geq t) = \mathbb{P}(Y = 1) = 1 - \prod_{i=1}^n (1 - \mathbb{P}(Y_i = 1)) = 1 - \prod_{i=1}^n \mathbb{P}(X_i < t)$. However, a simple characterization of the hazard rate does not exist for parallel systems. Neither does IHR or DHR component implicate the same for the system.

Example 11.7.14 Let $n = 2$, and X_i exponentially distributed with parameter i. Then $\mathbb{P}(X \geq t) = 1 - (1 - \exp(-t))(1 - \exp(-2t)) = \exp(-t) + \exp(-2t) - \exp(-3t)$. Its hazard rate is easily calculated, and it can be seen that it is not always increasing.

A weaker property than IHR that does also hold at the system level is IHRA, "increasing hazard rate average". It even holds for general systems, and therefore it is discussed for these systems.

General systems A life time X with hazard function Λ (see Section 1.5) is called IHRA if $\Lambda(t)/t$ is increasing in t. Note that IHR is a stronger property than IHRA: if X is IHR, then X is also IHRA.

Without proof we mention that the life time of a monotone system is IHRA if all components have IHRA life times.

Maintenance of a single component Let us consider a single component that is not only prone to failure, but that can be repaired as well. The time U that the system/component is up has distribution F_U, the time R it takes to repair the system has distribution F_R. Assume that the system is repaired as soon as it fails. Then, from renewal theory (see Section 3.3, Example 3.3.2), we know that the long-run fraction of time that the system is up is given by

$$\lim_{t\to\infty} \mathbb{P}(\text{system up at } t) = \frac{\mathbb{E}U}{\mathbb{E}U + \mathbb{E}R}.$$

Maintenance does not always mean that the maintenance starts right after the failure of a component. In many systems it is possible to do maintenance while operating, thus prolonging the life time of the component. This is what we call *preventive maintenance* (PM). Let us first assume that preventive maintenance makes the component "as new", thus its life time starts all over.

A reasonable maintenance policy would be: conduct preventive maintenance if the component is running for T units. If it fails before T time units, conduct regular *corrective maintenance* (CM). Let the time P to execute preventive maintenance have distribution F_P. We assume $\mathbb{E}P < \mathbb{E}R$, otherwise it would not be useful to execute PM. We pay some attention to the choice of T.

Example 11.7.15 Let U be exponentially distributed with parameter λ. If we execute PM at T, then

$$\mathbb{E}U = \int_0^T t\lambda e^{-\lambda t} dt + Te^{-\lambda T} = \frac{1}{\lambda}(1 - e^{-\lambda T}).$$

Thus the long-run fraction of time that the system is up is equal to

$$\frac{\mathbb{E}U}{\mathbb{E}U + F_U(T)\mathbb{E}R + \overline{F}_U(T)\mathbb{E}P} = \frac{\frac{1}{\lambda}(1 - e^{-\lambda T})}{\frac{1}{\lambda}(1 - e^{-\lambda T}) + (1 - e^{-\lambda T})\mathbb{E}R + e^{-\lambda T}\mathbb{E}P}.$$

Taking the derivative to T shows take this fraction is increasing in T. Thus it is optimal never to undertake PM, giving an availability of $(1 + \lambda\mathbb{E}R)^{-1}$.

The above example illustrates that while the hazard rate of a system is decreasing, then it is not optimal to execute PM. This makes sense: by repairing the system the time to failure only becomes shorter! On the other hand, for components with hazard rates that increase at least part of the time it can be optimal to execute PM. Note that in this discussion we did not take into account the costs of PM or CM.

Earlier in this chapter we discussed condition monitoring. By using additional information on the condition or state of the component the PM decision can be improved. Using simple Markovian models the optimal PM policy can be computed. The typical PM decision now becomes: if the condition of the component reaches condition x, then PM should be executed.

Maintenance of systems For the maintenance of systems with multiple components there are two basic situations to consider: those in which the maintenance of one component does not influence the maintenance of another component, and those where this is the case. Examples of reasons why maintenance on one component influences other components are:
- there are not enough repairmen to repair all failed components at the same time;
- it is cost-efficient to repair all failed components at the same time.

First we consider the situation where the maintenance of different component is independent. Then all components are independent. Let a_i denote the long-run availability of component i, i.e., $a_i = \lim_{t\to\infty} \mathbb{P}(\text{Component } i \text{ is up at } t)$. E.g., if there is only corrective maintenance, and U_i and R_i are the up and repair times of component i, then $a_i = \mathbb{E}U_i/(\mathbb{E}U_i + \mathbb{E}R_i)$. Let a be the system availability. Then, using results of the previous section, we find $a = \Phi(a_1, \dots, a_n)$.

More often we find that the components become dependent because of the maintenance policy. In general these situations are hard to analyze, and simulation is often the only available method.

As an example of a system that can be analyzed analytically we consider the situation of a k-out-of-n system with s repairmen ($s < n$), with as special cases $k = 1$ (parallel system) and $k = n$ (series system). We assume equally distributed exponential up (repair) times U (R), with parameter λ (μ). If we identify the components with customers, and the repairmen with servers, then we see that this maintenance model is equivalent to a queueing model with a finite source of n customers, s servers, and queueing. Using results from Theorem 5.4.7 we can compute the long-run availability of the system. For the general k-out-of-n systems it is given by $a = \sum_{j=0}^{n-k} \pi(j)$, with $\pi(j)$ as in (5.8)-(5.10).

To do Cost-effective maintenance: better double parts that are reliable and cheap than parts that are expensive and unreliable.

11.8 Distribution and field service

Many services delivered by companies involve going physically to the customer location. By assigning these visits in a smart way to employees (often technicians) and by optimizing routes traveling times can be reduced to a minimum. This section deals with the different issues related to these systems.

We start with a taxonomy of these geographic problems. Then we deal with them one by one. There is still little quantitative scientific literature about these types of systems. Therefore this chapter is mostly descriptive.

The most important distinction between systems is the interval in which the service is planned, starting from the moment of the service request call. Basically we find three possibilities:
- the customer is visited during a fixed tour of an employee (often the first tour possible);
- the customer is visited at some unspecified moment, but before a certain due date;

- the customer is visited during a certain time interval (not necessarily the first), agreed on by the company and the customer.

Example 11.8.1 An example of the fixed tour service is mail or packet delivery. To guarantee overnight delivery each packet should be delivered at the next tour conducted by the employees. Maintaining professional machinery such as copiers is an example of visits with due dates. These due dates follow from service contracts in which the maximum time between the service call and the arrival of a technician is determined. Visits in agreed intervals happens for example in the maintenance of customer appliances such as kitchen equipment. The customer makes the service call, and then together with the employee a moment or time interval is decided during which a technician will visit the customer.

All three possibilities for customer service have different (although related) operational scheduling models that we will discuss. However, not only the scheduling issues are of interest. There are strategic and tactical issues related to the size of the workforce and how the employees are used to fulfill the service requests. An example is the relation between the service level (how many calls are serviced before the due date?) and the costs (the number of technicians). Other examples are related to a regional vs. a global approach and issues related to specialization.

Without software implementing the models described in the next section these system could not be operational. Central in the software is a module that gives the average travel time between any two locations.

The operational decisions in the current systems can be split in two phases: first calls have to be assigned to intervals or tours, and then tours themselves need to be scheduled. We will start with the last step.

Vehicle routing In this section we consider the following problem. We have V vehicles or technicians that need to service a set consisting of N service requests. Given are the expected time distances between different customer locations $d(i, j)$ and the time distances between home bases of each of the technicians and the customer locations $\tilde{d}(v, i)$. Note that the home bases of the technicians need not be equal. We concentrate on the simplest model where there are no restrictions on the order or times of visits. The service or visit duration is S. Although travel and service times are random, we made the usual simplification of taking the average. A visit takes on average $s = \mathbb{E}S$ time.

Example 11.8.2 In a distribution environment the vehicles usually start their tour at the warehouse, making d_{vi} independent of v. On the other hand, many maintenance technicians go directly from home to their first customer, to reduce travel times.

An important and difficult modeling step is the choice of objective. If there are no due dates then it is reasonable to minimize the maximum of the tour lengths. To express this mathematically, we have to define the V tours, for example as follows. Let $i_{v1}, \ldots, i_{vn_v}$ be the n_v customers assigned to technician v, in this order. Then the expected length $l(v)$ of tour v is

$$l(v) = n_v s + \tilde{d}(v, i_{v1}) + \tilde{d}(v, i_{vn_v}) + \sum_{i=1}^{n_v-1} d(i_{vi}, i_{vi+1}).$$

Thus the problem is to find V tours, in which each of the N service requests occurs once, such that $\max_v l(v)$ is minimized.

For $V = 1$ this problem is equivalent to the traveling salesman problem, which is a well-studied optimization problem, for which high-quality solutions can be found using local search (see Section 9.2). For $V > 1$ (the usual situation) it is much more complicated, and mainly studied for $d(v, i)$ independent of v, i.e., a single home base. Many algorithms can be found in the literature.

For a system with due dates the number of late customers should probably be incorporated in the objective function.

Remark 11.8.3 Many variants of this model exist, with different types of additional constraints. Typical additional constraints are: there is a volume attached to each service request, and the total volume served in a single tour may not exceed the capacity of the vehicle; goods have first to be picked up before delivered in the same tour, leading to order constraints; there are time windows between which deliveries should take place. This last type of constraint can be used to model due dates.

With this we finish the discussion of systems with fixed tour assignments. We continue with extensions to due data systems and systems with free assignment of calls.

Due date systems In due date systems the service should start some moment between the current time and the due date. Often encountered due dated are 8 or 16 business hours, meaning 1 or 2 working days. Naturally, due to variations in call arrival times, not all customers can be visited before the due date. To obtain a satisfactory service level some overcapacity should exist to overcome peak demands.

A simple way to schedule is Earliest due data first (EDDF). On a short term basis this avoids as much as possible late arrivals at the customer location. However, it can also mean long travel times: by servicing in another order long-term advantages can be achieved by reducing the total travel time. Thus, by exceeding the due dates for one or a few customers, we have less remaining backlog, and thus less overtime visits later on. The challenge is of course to find a schedule for which no due dates are passed and that has the lowest possible travel times.

Taking only the number of customers which are served after their due dates is not a good objective: if a due date date is passed, then there is no incentive left to visit that customer. A better objective is as follows. Let dd_i represent the due date of customer i, and let T_i be its visit time. Of course, T_i is a random variable, although the model is often simplified by replacing all random variables by their expectations. Then a possible objective is to minimize $\sum_i \max\{0, T_i - dd_i\}$, with the summation running over all current service requests. Thus the "costs" for customer i are $T_i - dd_i$ if the due date is past, 0 if the technician arrives before the due date. This is a reasonable objective, but it has the disadvantage that future service request are not taken into account. This should somehow be compensated for.

Free assignments Crucial to the operation of the due date system is that the customer can be visited at any moment. This is particularly suited for B2B (business-to-business) services. On the other hand, with private persons appointments have to be made, while the time constraint is often less important. This leads to systems where the planner negociates on a time window with the customer (e.g., next wednesday between 8 o'clock and noon). The planning should have the following properties: next day schedule should be completely filled up, travel times should be minimized, several slots should be proposed to the customer as to accomodate him or her. This is done by adding the customer to each technician-time window combination to see if this fits well into the existing schedule. Several effects can be seen: the system has the tendency to wait as long as possible with filling up new time windows. This should be made easier by lowering the costs in these cases. It also might happen that next day schedule is not always completely filled up. This can then be achieved by adding a parameter that makes assignments to next day's schedule cheaper.

Finding the right balance is a matter of adjusting parameters based on a particular situation.

The algorithm can be simplified and sped up by first determining regions for each technician in which all their customers should lie. Of course these regions should be overlapping, otherwise we loose economies of scale advantages.

As the planning should be done online it is important that the software tool has a very short response time. This can be achieved by doing the calculation offline, which are then used to update a table in which for each postal code the preferred time windows are listed.

Car stock management A problem different from tour assignment and routing problems is that of determining the car stock of a technician. This is an inventory problem (see Chapter 6). The car stock enables the technician to repair equipment by replacing parts without having to go back. Going back indirectly costs money: by dividing the total operational costs by the number of visits a price per visit v can be computed. A 'lost sale' therefore approximately costs v. On the other hand, there are holding costs for having items in the car stock. A value of 10% of the value of the item is not unrealistic.

The replenishment policy can differ, but often the stock is replenished overnight or in the morning when the technician visits the spare parts warehouse. When we do not take the delivery costs into account then the usage is replenished directly every day. In that case the days do not depend on each other and it suffices, for each possible item, to solve a single-day problem, with the following parameters:
- costs per lost sale v;
- holding costs c, typically equal to the price of the product times 0.1, the annual holding costs per unit price, divided by 200, the number of working days;

- demand D per day;
- car stock size S.

For these parameters the total expected daily costs are given by:

$$C(S) = Sc + q\mathbb{E}(D-S)^+ = c(\mathbb{E}D + \mathbb{E}(S-D)^+ - \mathbb{E}(D-S)^+) + v\mathbb{E}(D-S)^+ =$$

$$c\mathbb{E}D + c\mathbb{E}(S-D)^+ + (v-c)\mathbb{E}(D-S)^+.$$

Because $c\mathbb{E}D$ is a constant independent of S, the optimal S can be found using Theorem 6.2.1.

The model that includes replenishment costs is much harder to analyze, especially since a delivery consists of items of multiple types.

Another complicating factor is the fact that it might occur that the rational car stock does not fit into the car anymore.

Design and capacity planning issues Up to now we discussed the operational scheduling issues of several field service and delivery systems. At a strategic and tactical level there are several issues that are strongly related to the operational scheduling. At the strategic level we discuss discussions related to splitting up service areas in multiple separate service regions. At the tactical level we discuss methods to access the size of the workforce.

Usually we assume that calls arrive according to a Poisson distribution. This means that, in the vehicle routing model, the number of service requests for a period is Poisson distributed. This means that there is no upper bound to the number of calls for a period. Thus a balance should be found between overtime and/or delayed service and operations/personnel costs. In a way we see the same contrast between service level and productivity as in call centers. A possible solution is the introduction of different levels of service: a premium next-period service level agreement, and other service levels that allow for longer delays. The service time of a customer consists of the actual service time and the travel time. Note that a complete tour consists of a number of trip/service pairs and one additional trip. Whether the trips to and from the first and last customer are part of the work time of the technician has important consequences at the operational level.

Example 11.8.4 A delivery service has several vans that all start from the same warehouse. Travel times are 15 minutes on average, delivery times 5 minutes. The first and last travel leg take on average 30 minutes. Drivers are available for 8 hours a day, of which they use 1 hour for loading, briefing, etc. On average they visit 18 customer locations per day.

Example 11.8.5 A service organization makes its technician start from their homes (spare parts are delivered overnight). The 8 hour working day starts at the first customer. Travel times between customers are 45 minutes on average, service takes 60 minutes. On average 5 customers are visited each day.

For due date service systems we can assume continuous operation by leaving out the non-business hours. Assuming homogeneous Poisson arrivals, a $M/M/c$ queue can be used to give a rough estimate of the performance. Also this system can benefit from different customer classes (i.e., different maximum times between service calls and the actual visits).

Finally consider the scheduling model with time windows. In these systems deadlines play a minor role, and thus assuring that the workforce is capable of handling all service requests (stability, in queueing terminology) suffices. Thus makes a high productivity possible.

Of course, in all these models travel times and service times have a fixed distribution. With this we should be careful. The distance between two arbitrary customers is independent of the total number of customers, but the length of the shortest vehicle routing schedule decreases with the number of customers: here we also see economies of scale. Thus in these field service systems we see two types of economies of scale: those related to the Erlang formula, and those related to the decrease of travel time.

These issues play also a role in specialization. In service organizations with highly specialized maintenance we see a trend towards specialization. The price to pay for this is less scale advantages in the sense that one group of technicians cannot take over another group's peak service requests. It also leads to longer travel times.

Another decision where travel times play a role is whether to work with regions or not. A global approach has the advantage of being able to obtain the global minimal solution, thus minimizing travel time. A regional approach leads to somewhat longer travel times, but has many organizational advantages. In practice we see that there is no objection against working with regions, as long as these regions have enough overlap. This calls for coordination between the regions.

11.9 Further reading

An excellent book that discusses in much detail the issues discussed in this chapter and much more is Hopp & Spearmann [48].

Also Silver & Peterson [83] treats in Part V many issues related to production scheduling, including MRP and JIT. We also recommend Hax & Candea [46].

The "business novel" Goldratt & Cox [42] presents an accessible introduction to Goldratt's Theory of Constraints, that is centered around the treatment of bottlenecks.

For production scheduling O.R. Handbook 4 [43] contains some interesting chapters. Chapter 5 presents queueing network models, Chapter 9 deals with (mainly deterministic) machine scheduling problems, Chapter 11 deals with MRP, and Chapter 12 considers JIT. Also Gershwin [40] considers queueing network models for production systems.

Zangwill [102] gives another view on several aspects of JIT.

Wikipedia is a great source for information on methodologies such as Just In Time or Six Sigma.

Chapter 7 of O.R. Handbook 4 [43] gives an overview of production planning.

An excellent book on production and inventory management, which pays attention to all steps in the modeling process, is Hax & Candea [46]. See also Hopp & Spearmann [48] and Silver & Peterson [83].

An easy to read book containing the project planning methods CPM, PERT, and some additional topics, is Awani [12]. Goldratt's Theory of Constraints applied to project management is described in [41]. Belson [15] is a book chapter describing project management for health care process improvement projects.

The standard text on reliability is Barlow & Proschan [14], including a proof of the statement at the end of Section 11.7. An introduction to the mathematics of reliability is Chapter 9 of Ross [75]. An advanced mathematical text is Aven & Jensen [11].

A journal entirely devoted to reliability is *IEEE Transactions on Reliability*.

Part II of Bramel & Simchi-Levi [16] deals with the vehicle routing problem and some of its variants. Also several chapters in Handbook 8 [13] deal, in more detail, with different aspects of vehicle routing. There is no (or very little) scientific literature on scheduling aspects of the other problems.

Flow line modeling: Dallery & Gershwin [28].

Publisher McGraw-Hill maintains a list with virtual company tours at http://www.mhhe.com/omc/t
frames.htm.

11.10 Exercises

Exercise 11.1 What is the direct influence on machine failure to MRP? And what is the long-term consequence of regular failures? Answer the same question for JIT. Remember that MRP assumes infinite capacity and deterministic lead times.

Exercise 11.2 Consider a machine with infinite storage space space to which jobs arrive according to a Poisson process with rate 1. The service time distribution has density $f(x) = 2x$ if $x \in [0, 1]$, 0 otherwise. Jobs are processed in FCFS order.
a. Calculate the first two moments of the service time distribution.
b. Calculate the expected long-run waiting time in this system.
Service times are known upon arrival. It is decided to split the customers in two classes: those with service time below y, and those with service time above y, for some $y \in (0, 1)$. It is decided to give non-preemptive priority to the class with short service times. Within a class the processing order is FCFS.
c. Calculate the first two moments of the service time distributions of both classes.
d. Calculate the expected long-run waiting time in this system.
e. What is the value of y that minimizes the waiting time?

Exercise 11.3 A flow line consists of two machines with infinite in-process inventory space. Arrivals occur according to a Poisson process. Service times are assumed to be exponential with rates 2 and 3, respectively.
a. What is the maximum production rate $\bar{\lambda}$ of this system?
b. Make a plot of the waiting times at both machines for the arrival rate ranging from 0 to $\bar{\lambda}$.

Exercise 11.4 A flow line consists of two machines with no in-process inventory space in between. The order arrival process is Poisson.
a. What is the maximal production rate in the case of exponential service times?
b. What is the maximal production rate in the case of deterministic service times?
c. What is the "worst case situation" for given first moments of the service time distribution?
(Hint: Consider the flow line as a single station with a more complicated service time distribution.)

Exercise 11.5 Give some advantages and disadvantages of big lot sizes.

Exercise 11.6 Consider a machine that processes two types of parts. The parts have deterministic processing times, with average 1 and 2 hours for type 1 and type 2, respectively. Parts to be processed arrive according to a Poisson process, with an average of 0.1 and 0.2 per hour, respectively.
a. Calculate the long-run average queue length and waiting time if the processing order is FIFO.
b. Calculate the long-run average queue lengths and waiting times for each type separate and combined under both non-preemptive priority rules.

Exercise 11.7 We add holding costs to the system of Exercise 11.6. For type 1 they are equal to c per hour and part, for type 2 they are equal to $3c$.
a. Calculate the long-run average holding costs if the processing order is FIFO.
b. Calculate the long-run average holding costs under both non-preemptive priority rules.

Exercise 11.8 Consider two production lines, each consisting of two consecutive production steps. The production lines share the same resource for the second production step (but not the first). Production planning is on a MTO basis, and orders arrive according to a Poisson process. Assume that service times are exponential. The order arrival and service rates are given in the following table:

	Order arrival rate	Stage 1	Stage 2
Type 1	1	2	3
Type 2	1.5	2	α

Let the processing order at all stages be FIFO.
a. Calculate the expected total waiting and response time for both product types, for $\alpha = 3$.
b. Calculate the expected total waiting and response time for both product types, for $\alpha = 2$.
c. The same question as a., but now if type 1 has priority over type 2. (Hint: use Theorem 5.6.1.)

Exercise 11.9 Consider a production line with 3 machines and 2 types of jobs. A job of type i visits with probability 0.5 machines i and 3, and with probability 0.5 only machine i. Machine 1 (2) is thus visited by all jobs of type 1 (2), machine 3 is visited by half of all jobs. The arrival processes are Poisson, all service time are i.i.d. exponentially distributed. The average service times of type 1 (2) jobs on both machines they can visit is 1 (2). The arrival rate of type 1 (2) jobs is 0.6 (0.3). The service order at machine 3 is FCFS.
a. Calculate the load of each machine.
b. Calculate the expected waiting times at machines 1 and 2.
c. Describe the arrival process at machine 3. Describe also the service time of an arbitrary job at machine 3.
d. Calculate the expected waiting time at machine 3.
e. Calculate the expected total time that an arbitrary job spends in the system.

Exercise 11.10 Each job on a machine consists of two different operations that are executed consecutively. Each operation has an independent exponentially distributed processing time (with averages β_1 and β_2). Assume that input and output buffers can accomodate any number of parts. Job orders arrive according to a Poisson process with rate λ.
a. For which parameter values is the waiting time finite?
b. Give an expression for $\mathbb{E}(X+Y)^2$ for general and independent X and Y.
c. Calculate the waiting time for $\lambda = 1$, $\beta_1 = 1/2$, and $\beta_2 = 1/3$.
Now assume that it is possible to change the machine such that the two operations can be executed in parallel, i.e., they start at the same time.
d. Show that the service time is of the form $X + ZU + (1-Z)V$, with X, U, V, and Z independent and $Z \in \{0,1\}$.
e. Give an expression for $\mathbb{E}(X + ZU + (1-Z)V)^2$.
f. Calculate again the waiting time for $\lambda = 1$, $\beta_1 = 1/2$, and $\beta_2 = 1/3$.

Exercise 11.11 A production system consists of 2 production steps. Both take an exponentially distributed amount of time with parameter μ. Production times are independent. Orders arrive according to a Poisson(λ) process. Two different configurations for the system are considered.
a. In the first configuration the two production steps are executed consecutively. There is a large buffer space in front of each production step. Calculate the maximal production rate and the system time as a function of λ.
b. In the second configuration both production steps are executed at the same time. Production on a new order can only start if both steps are finished. There is a large buffer space in front of the combined production step. Calculate the maximal production rate and the system time as a function of λ.
c. Which system has the lowest system time for λ small? Explain this.

Exercise 11.12 The objective of aggregate production planning is to find a production plan that minimizes total weighted inventory costs for given order due dates, capacities, and inventory costs. If there is no feasible plan (i.e., it cannot be avoided that some orders

are late) then there is no feasible solution to the correponding linear problem.
a. Construct an example, as simple as possible, for which this is the case.
b. Extend the linear-programming formulation to the situation where penalties are paid for every time unit that a job is late. Make sure that the model remains linear.

Exercise 11.13 Generalize the aggregate production planning model as to account for overtime on the resources. Note that utilizing overtime costs extra money. Make sure that the model remains linear.

Exercise 11.14 A project has the following activities:

Activity	Preceding activities	Duration
A	-	2
B	A	3
C	A	2
D	C	1
E	B,D,G	2
F	-	3
G	C,F	2

Assume for the moment that there are enough resources.
a. Compute the earliest finish time of the project and all earliest and latest s tarting times of the activities. (Hint: renumber first the activities.)
b. Give the definitions of slack, critical activity, and critical path.
c. Compute in the example project the slack of each activity. What is the critic al path?
Suppose that activities B and C use the same resource. Therefore they cannot be scheduled at the same time.
d. What is now the earliest finish time of the project?
e. Prove that the solution to d. gives indeed the earliest finish time possible.

Exercise 11.15 Project with normal durations. Utilize PERT and simulation, possibly with importance sampling.

Exercise 11.16 Consider a system with 2 components A and B in series. There are two spare machines C and D. Machine C can replace A or B, but when C replaces B, then also D is necessary. (Thus D is only used to "help" C replace B.)
a. Find all minimal path sets.
b. Determine ϕ and Φ.

Exercise 11.17 We consider the availability of a system with 2 components, named A and B, in series. There are two spare components C and D. Component B can be replaced by component D. Component A can be replaced by component C, but only if D replaces B as well. Machines fail independently, whether they are used or not.
a. What are the minimal path sets of this system?

b. Give the functions ϕ and Φ (giving expressions for the availability in a deterministic and in a random environment).
c. If all components have an exponential time to failure with rate 1, and they are all up, what is the expected time to failure of the system?

Exercise 11.18 Consider a system with 2 identical components, with uniform life time distributions on [0,1].
a. Calculate their hazard rates. Is it IHR on [0,1)?
b. Calculate the hazard rate of the system if the components are placed in series. Is it IHR on [0,1)?
c. Calculate the hazard rate of the system if the components are placed in parallel. Is it IHR on [0,1)?

Exercise 11.19 Consider a k-out-of-n system with n identical machines. The time to failure of each machine is exponentially distributed with mean α.
a. Give a formula for the expectation of the time to failure of the system.
b. Give the failure rate of this system for $k = 2$ and $n = 3$.
We add a single repairman to this system, the repair time is exponential with mean 1.
c. Model this system as a birth-death process.
d. For arbitrary n, give an expression for the long-run fraction of time that the system is up.

Exercise 11.20 Consider a system consisting of n identical machines. The time to failure of a machine is exponential with mean 10. The system is up if at least one machine is up. If more than one machine is up, then these spare machines are in cold standby.
a. For $n = 2$, give the probability that the system is up at time 20.
b. What is the failure rate of this system?
We add a single repairman to this system, the repair time is exponential with mean 1.
c. For arbitrary n, give an expression for the long-run fraction of time that the system is up.
d. How many machines are needed to make this fraction at least 0.9999?

Exercise 11.21 Consider a system consisting of n parts that each fails, independently of the other parts, after a time that is uniformly $[0, 1]$ distributed, i.e., the density of the time to failure of each component is 1 in $[0, 1]$, 0 otherwise.
a. Give the definition of the failure rate and the system function.
b. Compute the failure rate of the life time of a single component.
c. Let the system consist of n parts in series. Compute the failure rate of the life time of the system.
d. Let the system consist of n parts in parallel. Compute the failure rate of the life time of the system.

Exercise 11.22 To increase the availability of a computer system a second was placed in parallel. Both have an availability of 98%. However, it was found that 1% of the

unavailability was due to a problem with a common power supply.
a. Model this system using independent components.
b. Formulate ϕ and Φ.

Exercise 11.23 Show that if X is IHR, then X is also IHRA.

Exercise 11.24 Make a plot of Λ of the system of Example 11.7.14 and convince yourself that the system life time is IHRA.

Exercise 11.25 Consider a 2-out-of-3 system with a single repairman, exponential times to failure, exponential repair times, and warm stand-by. Give a formula for the long-run probability that the system is up.

Exercise 11.26 Consider a system consisting of n identical machines. The time to failure of a machine is exponential with mean 10. The system is up if at least one machine is up. If more than one machine is up, then these spare machines are in cold standby.
a. For $n = 2$, give the probability that the system is up at time 20.
b. What is the failure rate of this system?
We add a single repairman to this system, the repair time is exponential with mean 1.
c. For arbitrary n, give an expression for the long-run fraction of time that the system is up.
d. How many machines are needed to make this fraction at least 0.9999?

Exercise 11.27 Consider a technician who needs a certain spare part on average once a year. A customer visit costs on average 40 euro. The part cost 100 euro. Is it economically rational to put the part in the car stock?

Exercise 11.28 A technician visits every day 4 customer locations. At each location there is a probability of 0.01 that a certain part is needed. Replenishments occur daily.
a. How many parts should the technician have in his car stock to have a probability lower than 0.01 per visit that he cannot replace the item when needed?
b. The item costs 50 euro. What is a rational car stock?

Exercise 11.29 A service system has 20 technicians. Average travel times between customer locations is 25 minutes, service times are on average 40 minutes. Technicians travel to the first customer and from the last customer in their own time; the average working day is 8 hours.
a. How many service requests can this system handle on average daily?
b. Describe a methodology for obtaining approximations for the time a customer must wait on average.
c. The technicians prefer to work 10 hours during 4 days instead of 8 hours during 5 days. What do you think of this proposal?

Chapter 12

Health Care

In developed countries the health care sector may account for up to 15% of the gross national product, and its share is still increasing. This has put cost reductions in health care expenses high on the political agenda in many countries. The idea that much can be gained without reducing the quality of care is shared by many, but seems hard to realize.

Although many general operations management principles apply, there are a number of aspects that make health care, from a planning and scheduling point of view, different from manufacturing. The way health care is financed and the fact that health care is a service are among the most important ones.

In this chapter we discuss those planning and scheduling problems in health care that occur most often, with a focus on hospitals.

12.1 Introduction

In manufacturing the goal is to make as much profit as possible. A company makes products that customers buy. As long as the total production costs are lower than the revenue from sales the company makes profit. Health care is fundamentaly different. Although commercial aspects play an increasingly important role, one of the aspects that are essential is that governments finance the health sector to a large extent in order to provide all citizens with a certain level of health care. In health care there are three conflicting objectives: price, quality and accessibility. The latter is new in comparison to manufacturing, and relates to the fact that all citizins should have access to high-quality care. Unfortunately, the way in which the sector is financed not always stimulates health care institutions to behave according to these general objectives.

Quality of health care services falls apart in two aspects: those that represent the medical outcome and those that are related to the process such as the length of waiting times. Note that both types of quality are related: think about an emergency patient who has to wait long before being seen by a doctor. In this chapter we mainly focus on the process-related definition of quality, in relation to efficiency of the health-care delivery process. Of course, efficiency is strongly related to costs. Thus, we focus on the quality-efficiency trade-off, as we did for manufacturing. However, the fact that health care is a *service* makes it crucially different from manufacturing.

Health care is a service, which means that the customer or patient is part of the process. Thus, if we see health care delivery as a production process, "inventory" equals patients waiting. Thus, even more than in manufacturing, there is a need to reduce inventory. However, this puts pressure on the productivity of the process, because inventory serves as lubricant in production chains, unless the variance in the process steps is very low. Radical solutions to avoid waiting, such as the combination of process steps (e.g., creating a "one-stop shop", combining multiple appointments in one visit), call for completely different planning approaches.

Another aspect of the human factor is that variability cannot be reduced to the extent as it can in manufacturing. People are different in all aspects of their behavior: some do not show up for an appointment and others come early, they require a different length of stay in the hospital to recover from an operation, and so forth. Fluctuations can be reduced and predicted to a certain extent (for example, by sending a reminder for an appointment, or by statistically differentiating between patients), but there will always remain fluctuations and uncertainty, more than in manufacturing. Planning in this situation is a major challenge for the health care sector.

Not related to the service aspect, but also different to manufacturing, is the organizational structure of most health care institutions. Other than in most production facilities there is a flat organizational structure, known as a *professional bureaucracy* (Mintzberg [67]). It is *professional* because the expertise of the hospital professionals is prevalent in the decision process; it is a *bureaucracy* because it is organized according to fixed rules and positions. To give an example of the latter, the process of becoming a doctor is completely regulated. This structure makes it difficult to implement management decisions that encompass different organizational units, and improvements that focus on an efficient use of resources usually do require a broad focus. This does not facilitate changes in the process.

In the last decades health care expenses exploded. It is commonly believed that the ageing population is the main cause, but technological advances are at least as important. This drove governments in developed countries to focus on cost reductions and the introduction of a regulated market. Formerly this was less of an issue. Because of the lack of competition and the availability of sufficient government funding the health care sector was, logistically speaking, in the nineties where manufacturing was in the seventies. Because of this, many concepts that had been proven useful in manufacturing have recently been adopted to health care. We discuss them in the next section. The rest of this chapter is focused on mathematical modeling of health care processes.

We introduce some useful terminology and concepts. The first distinction is between *scheduled patients* and *unscheduled patients*. Scheduled or *elective* patients are scheduled through an appointment system, unscheduled patients can be assumed to arrive according to a Poisson process, with an arrival rate that depends on the time of day (as discussed in Section 2.4). Unscheduled patients fall apart in two groups: *urgent* patients and *emergency* patients. The difference between the two is that, for example, urgent patients can wait for the next day to be operated, while emergency patients have to be operated in say 8 hours.

An elective patient has two types of waiting times: the *access time*, which is the time between the moment the appointment is made and the actual appointment, and the *waiting room time*, which is the time the patient spends waiting in the waiting room at the health care facility. Another distinction is between *clinical* patients and *ambulatory* patients, those that stay in a hospital, and those that visit during the day a health care institution. Ambulatory patients are also called *out-patients*, and clinical patients *in-patients*.

12.2 Improvement methods

In health care many different quality improvement methods are used, often inspired by what has been successful in manufacturing. Considerable improvements are achieved using these methods. Some of them involve mathematical modeling, usually simulations. The focus in these methods has been, up to now, on implementing process changes based on common logistical principles or statistical methods. The problems that remain to be solved are too hard to handle without the use of advanced methods such as mathematical modeling: common insights from other domains do not suffice. These problems will likely be addressed in the context of one of the usual improvement methods, if only to provide a framework for the implementation of the solution. We discuss those methods that are most often used in health care: Six Sigma, Theory of Constraints, Lean Manufacturing, Advanced Access, and the use of clinical pathways.

Six Sigma is a statistical quality control method, the sigma in the name refers to the common notation for standard deviation, and a process improvement method at the same time. In health care it is mostly used as a general improvement method, statistics are not used that often. Six Sigma is very result-oriented and focused on obtaining the financial targets that were set at the beginning of the project. At the same time, it dictates a well thought out organizational structure, with support from top management, involving terminology inspired by martial arts (for example, a Six Sigma specialist is called a *black belt*). While the statistical quality control aspects are less relevant, it is especially the organizational embedding that can be very useful to health care, besides the fact that Six Sigma is very much *data-driven*, that is, the process analysis is based on data from the process itself and not on subjective opinions of stakeholders.

The Theory of Constraints (ToC) is conceived by E. Goldratt around the idea that every process is constrained by a bottleneck. Finding and eliminating it will improve performance. This general principle can be applied in many different situations and has also succesfully been applied to health care.

Lean Manufacturing is a process management philosophy based on the Toyota Production System, centered around the idea of eliminating waste in the production process. Waste is everything that does not add value, that increases production time and costs. Different types of waste are identified, such as transportation, inventory and defects. A second focus of ”lean” is on reducing variations. Currently, Lean is sometimes combined with Six Sigma leading to *Lean Six Sigma*.

Now, we discuss in more detail how principles from lean manufacturing are applied in so-called "Advanced Access" projects. Many health care facilities, such as outpatient clinics or diagnostic departments, show constant access times of several weeks or months. For systems with independent arrival processes, and customers that wait until they get into service, queueing theory tells us that either the system should empty now and then or the access time should grow to infinity. In reality neither happens, which shows that the demand for these health care services depends on the access time: some of the patients do not enter the queue but go elsewhere or, while attending, their needs disappear and they leave the queue. It has been proven that the health condition of patients on waiting lists deteriorates, resulting in more severe disability and in increased mortality.

Long access times do not only have a diminishing effect on the workload. Some patients require immediate treatment and therefore ad-hoc solutions are necessary for them, such as the reservation of "emergency" slots. This results in inefficiencies in the planning, the necessity of *triage* (deciding as to which category a patient belongs), and so forth. Thus short access times not only increase the quality of care, it also frees capacity (removes waste) otherwise necessary to deal with the negative effects of long access times. Several institutes are specialized in helping health care institutions re-engineer their process and reduce the waiting lists. This works under the condition that the demand in the case of short access times is lower than the capacity of the re-engineered process. Examples of such institutions are the Institute for Healthcare Improvement (IHI, www.ihi.org) in the USA and CBO (www.cbo.nl) in the Netherlands.

Example 12.2.1 In 2003-04 the VU medical center in Amsterdam has done *Advanced Access* projects at three of its out-patient clinics. In all three cases access times were reduced considerably, from several weeks, or even months, to days. The processes were re-engineered. For example, in certain situations return appointments are not made anymore at fixed intervals: they are made at the initiative of the patient, when (s)he feels it necessary (so-called *patient-initiated care*). This is possible because of the short access times, and led to a reduction of the workload. A major point was the resistance to change in all clinics: crucial in all cases was the commitment of a doctor.

The type of system where this method applies is an isolated facility or unit with enough capacity. The access time is measured in days, which is suitable for the majority of patients. The situation changes when there are economical or medical reasons to distinguish between patient classes. Suppose for example that there are both clinical and ambulatory patients. Access times for clinical patients of one or two days are long: every day they spend in the hospital costs hundreds of euros. Therefore, clinical patients should have some form of priority over ambulatory patients. Similarly, there are often economical reasons to identify different types of clinical patients: for example, intensive-care patients tend to be very costly. Another complicating factor is the fact that patients often need a number of resources immediately after each other. When we are talking about resources we actually mean the staff to man these resources: an ICU might have 20 beds, but only personnel to work on 16. Then (at least) 4 beds have to remain empty.

Example 12.2.2 Open heart surgery typically requires an operating room for a number of hours and an IC bed at the day of surgery. When one of the two is not available the operation needs to be cancelled. After discharge from the ICU the patient needs a bed at the medium care, etc.

Health care delivery is often a sequential process consisting of different steps, such as visits to a doctor in the outpatient clinic, a diagnostic phase with exams such as X-rays, a surgical procedure, and time spent at possibly different clinical wards. Usually different departments are responsible for the different steps in the process, without any overall coordination. In the last decade in many different hospitals (and other health care institutions) *clinical pathways* or *integrated care pathways* have been created, following the ideas of the *coordinated supply chain* in manufacturing (see Chapter 11). It consists of coordinating the typical path of a homogeneous group of patients from admission to discharge, which stand perpendicular to the departmentental structure of most hospitals. If the group of patients is homogeneous and their resource usage predictable enough, then resources can be allocated over the whole path and not just step by step. Quality control is also easier because the group is well identified.

Certain types of treatment better fit in a clinical pathway then others: the more predictable the patient group, the easier it is. Setting up a clinical pathway is worth the investment when the patient group is big enough. It must be understood that some of the patient demand can never be organized via a clinical pathway: the paths that these patients follow are so unpredictable that resources have to be allocated step by step. It is an interesting question what the influence of the pathways is on the remaining group of patients that are not organized according to clinical pathways. Many hospitals struggle with this question.

Some institutions go a step further: they specialize on certain clinical pathways. The term *focused factory*, introduced by W. Skinner, is often used in this context, refering to the fact that a production plant, in his opinion, should focus on a few process steps and excel in these. In such a focused factory a patient-centered pull-strategy can be used: all steps in the care process are planned at once. Not the seperate organizational units (radiology, operation rooms, etc.) play the central role, but the health care delivery process does. This stands in contrast with the traditional push policy. Focused factories have been successful in manufacturing, and also in health care remarkable results have been achieved using these ideas.

Applying the methods introduced in this section has improved quality and efficiency in health care already considerably. They are based on general concepts that have been proven succesful in many situations inside and outside health care. However, sometimes general principles do not work and advanced modeling is required to come to more specific solutions. In the next sections we look at a number of these situations. We start with a classification of these problems.

12.3 Model and problem classification

In this section we give an overview of the next sections by classifying a number of planning and capacity management problems in health care, especially hospitals.

An important difference is between decisions concerning the available capacity and those that concern the way the capacity is used. The first type of decision, capacity planning, is at the tactical level; the actual scheduling is at the operational level. The underlying principles that guide the capacity planning and scheduling are decided at the strategic level, and concern problems like product mix, rejection percentages, etc. As an example, if a hospital decides to specialize in open heart surgery (a strategic decision), then there should be sufficient IC beds to accomodate the patients (a tactical decision), and for every bed it should be decided to what type of patient is is decided (an operational decision). Note that not taking an action, such as assigning beds to every patient as long as capacity is available, is strictly speaking also a way of deciding.

Another difference is between processes that allow for delay (such as making an appointment for something non-urgent) and processes that do not (such as the arrival of trauma patients, making appointments, or the transfer of patients from the operating room to the Intensive Care).

With these differences we come to the following subjects treated in the subsequent sections:
- Capacity decisions for clinical wards in Section 12.4;
- Dynamic bed allocation methods in Section 12.5;
- Outpatient scheduling in Section 12.6;
- Capacity decisions for outpatient departments in Section 12.7;
- Bottleneck analysis for shared resources in Section 12.8.

12.4 Capacity decisions for wards

Let us consider the size of clinical wards. Every ward contains a number of beds. Note that only a number of the physical beds might be available due to personnel issues. The beds that can be used because personnel is available are called *operational beds*. Capacity decisions and, eventually, admission decisions at wards are complicated by the fact that the time that patients spend on wards for medical reasons are highly variable. This time is called the *length of stay* (LOS), its average is denoted by ALOS. Arrival times are also random, up to a certain degree. Emergency patients can be assumed to arrive according to non-homogeneous Poisson processes with a daily cycle. Usually the ALOS is considerably longer than 1 day, making the fluctuating arrival rate of limited relevance for capacity decisions. For elective patients the moment that patients arrive at a ward can often be planned. The most common example is a patient who arrives at a ward the day before operation and enters a (possibly different) ward right after the operation. However, data analysis shows that the variation of the number of admissions per day on many wards is comparable to that of a Poisson distribution, making the Poisson process also a reasonable approximation for the case of elective patients. The randomness in both arrival moments and length of stay implies that either the bed capacity at wards cannot be fully used, or that there is a high blocking probability. The latter situation occurs regularly in hospitals: too often scheduled operations have to be cancelled because of lack of capacity at clinical wards. Next to that it often happens that a patient is admitted at the right ward, in the "wrong bed". The reason is that hospital professionals, responsable for capacity decisions, do not recognize the impact of the variation: they have the tendency to take the number of beds s equal to the average numbers of arrivals λ times the ALOS β. This is an example of the *Flaw of Averages* (see Section 1.2). In other situations they account for some slack capacity, but this percentage is usually not related to the size of the ward or the number of refused or delayed admissions. For example, the number of beds is considered to be right if the occupancy is close to 85%.

A reasonable model for helping to make ward capacity decisions is the Erlang B model. see Theorem 5.4.3 for the calculation of performance measures, and www.math.vu.nl/∼koole/obp/Erlang for an Erlang B calculator. Note that this model assumes Poisson arrivals but does allow for general service time distributions. Erlang B also assumes that arrivals that find all beds occupied leave the system, and are possibly redirected elsewhere. See Table 12.1 for some numerical examples. Note that the rejection probability is equal to $\pi(s)$, the fraction of time that all s beds are occupied. The average number of occupied beds $\mathbb{E}L$ is equal to $\lambda(1-\pi(s))\beta$ (cf. Equation (5.6)). In the Erlang B model it is customary to write $\pi(s) = B(s,a)$, with $a = \lambda\beta$. Note what happens when we double both the size of the ward and λ: the rejection probability $\pi(s)$ decreases and the number of occupied beds $\mathbb{E}L$ more than doubles. These are the economies of scale of the Erlang B model.

s	10	10	10	10	10	20	20	20	20	20
λ	2	4	5	6	8	4	8	10	12	16
$\pi(s)$	0.005	0.12	0.21	0.30	0.44	0.0002	0.06	0.16	0.26	0.41
$\mathbb{E}L$	4.0	7.0	7.9	8.4	9.0	8.0	15.0	16.8	17.8	18.8

Table 12.1: Erlang B examples for $\beta = 2$ (all units in days)

A complicating factor in practice, when applying the Erlang B formula to ward dimensioning, is that usually only admissions are counted. Refused admissions are often not registered and therefore we do not have a direct view of the actual demand. Let us define λ_e as the expected number of admissions per unit of time: the *effective* arrival rate. Usually we measure λ_e, β and s. From this λ can be computed, but there is no closed-form expression known. Using the fact that λ_e is increasing in λ we can find a simple numerical procedure to obtain λ. Another method to find λ is measuring $\pi(s)$. Using the PASTA property (see Section 3.6) we find that $\pi(s)$ is equal to the probability that an arbitrary arrivals is refused admission, and thus $\lambda = \lambda_e/(1-\pi(s))$.

To calculate $\pi(s)$ we need the actual arrival and departure times to calculate the number of occupied beds as a function of time. Note that we assumed s to be fixed. In reality s, the number of operational beds, fluctuates to a certain extent. This can be on purpose, for example in order to anticipate on a lower load during weekends, or unpurposely, because of illness of nurses.

Example 12.4.1 For a ward with 5 beds we analyzed the average numbers of arrivals as a function of the number of occupied beds. Data was collected for a year on a half-hour basis. The average number of arrivals for $s = 0, \dots, 7$ was as follows: 0.24, 0.21, 0.22, 0.17, 0.11, 0.05, 0.01, 0.00. These data show that sometimes there are as little as 3 operational beds, and that on the other hand there are sometimes as much as 7 beds occupied.

The Erlang B model gives us a tool to rationalize decisions concerning ward sizes. At the same time, this model can give us a lot of insight concerning decisions related to merging of wards or *bed pooling*, the idea that dynamically, on the basis of occupation, wards can share beds. We already saw in Table 12.1 an example of economies of scale: when a ward is doubled in both size and demand then the occupation goes up and, equivalently, the rejection probability goes down. Note that doubling the scale is equivalent to merging two identical wards. Using Theorem 5.4.5 we can see that this property holds in general: if wards with identical ALOS ($\beta_1 = \beta_2$) are merged then, due to Equation (5.7), the overall expected number of occupied beds goes up. If we divide (5.7) by $\beta = \beta_1 = \beta_2$, then we get:

$$(\lambda_1 + \lambda_2)B(s_1 + s_2, (\lambda_1 + \lambda_2)\beta) \leq \lambda_1 B(s_1, \lambda_1\beta) + \lambda_2 B(s_2, \lambda_2\beta). \tag{12.1}$$

Because $\lambda B(s, a)$ is the expected number of rejected patients per unit of time, we also see that the total number of refused patients decreases. This does not mean that the blocking probabilities of both patient flows go down; one might increase, but the weighted average decreases.

Example 12.4.2 Consider two intensive care units (ICUs) with equal ALOS: one with a load of 20 and also 20 beds, and a specialized ICU with a load of 8 and 12 beds. Assuming Poisson arrivals the Erlang blocking model can be used to determine the blocking probabilities: 0.16 and 0.05, with a weighted average of 0.13. A single ICU with load 28 and 32 beds has a blocking probability of less than 0.07: half the value for the separate ICU! However, the blocking probability for the specialized ICU increased.

Up to now we discussed merging wards with the same ALOS. Usually wards with different types of patients have a different ALOS. Equation (5.7) applies also in this situation: merging wards leads to a higher overall occupancy. If a hospital is being paid by the days that beds are occupied, then the revenue is maximized by merging as many wards as possible. On the other hand, Equation (12.1) does not hold anymore, because $\beta_1 \neq \beta_2$. Table 12.2 gives a counterexample. In column 3 we see the weighted averages over the two wards: $(\lambda_1 B(s_1, a_1) + \lambda_2 B(s_2, a_2))/(\lambda_1 + \lambda_2)$ is the weighted average rejection percentage, $(\mathbb{E}L_1 + \mathbb{E}L_2)/(s_1 + s_2)$ is the overall occupancy (which is also the weighted average occupancy, weighted with respect to the number of beds). Column 4 are the numbers for the merged ward. We see that not only the occupancy but also the rejection percentages increases.

	ward 1	ward 2	weighted average	merged ward
arrival rate (λ)	1.00	5.00		6.00
ALOS (β)	5.00	1.00		1.67
# of beds (s)	3	7		10
rejection % ($B(s, a)$)	53%	12%	19%	21%
occupancy % ($\mathbb{E}L/s$)	78%	63%	67%	79%

Table 12.2: Example of merging wards with $\beta_1 \neq \beta_2$

The counterexample is not very realistic, with a 53% rejection probability in the first ward. In most practically relevant cases merging leads to a decrease of the overall rejection rate. The fact that the merged wards do not necessarily profit equally from merging, or even have an increasing rejection rate, needs more attention. It is the subject of the next section.

12.5 Dynamic bed allocation

In the previous section we saw that merging wards leads to an overall increase in occupation and usually also to an overall decrease in refused admissions. However, it can happen that the blocking percentage of one of the types of patients increases. In most cases this is undesirable, as the number or percentage of refused admissions per type of patient is one of the key performance indicators of hospitals. Usually it is formulated as a constraint: it should remain below a certain specified percentage.

Example 12.5.1 An intensive care unit admits emergency patients from the Emergency Department and patients that underwent scheduled surgery. Emergency patients can be sent to another hospital, and scheduled operations can be cancelled. This is allowed to happen only for 1% of the emergency patients and 5% of the patients that have to undergo a scheduled operation.

The challenge is to find a way to assign beds of a ward to different types of patients having different parameters, in order to profit from the economies of scale and to satisfy the constraints on the rejection percentages. The way to do this is *dynamic bed allocation*: the idea that beds should be assigned to patients on the basis of current occupancy information. A possible consequence is that a patient can be refused while there are still beds free at the ward.

In this section we consider two types of dynamic bed assignment policies: one that uses *threshold policies*, often studied in the context of admission control to queueing systems, and one that uses assigned beds, an idea coming from the health care domain. With threshold policies a patient of a certain type is only admitted if the total number of beds occupied does not exceed a certain type-dependent number, the threshold. When assigned beds are used then a certain number of each type of patients is guaranteed a place, and when less places are occupied, then beds are kept free for these patients.

Let us go into more detail about threshold policies. We consider the case of two types of patients and (almost) equal ALOS. The occupation can be modeled as a birth-death process with states $\{0, \dots, s\}$ and departure rates $\lambda(x, x-1) = x\mu$ for $0 < x \leq s$, with μ as usual equal to the reciprocal of the ALOS. For a threshold policy the arrival rates are as follows: $\lambda(x, x+1) = \lambda_1 + \lambda_2$ for $0 \leq x < s'$ and $\lambda(x, x+1) = \lambda_1$ for $s' \leq x < s$, with $s - s'$ the number of beds that we try to reserve for type-1 patients. It can be proven that such a threshold policy minimizes a weighted sum of the rejection rates, assuming that the costs for rejecting type-1 patients is higher than rejecting type-2 patients. The optimal threshold level can be determined using a technique called *dynamic programming* or simply by comparing the costs for different threshold levels using the birth-death formulation. In practice, there are usually constraints concerning the precentages of refused admissions. A comparison for different threshold levels is then required to find the best solution. Outcomes of such an analysis can be found in Table 12.3.

situation	# of beds	rejections type 1	rejections type 2	overall occupancy
separate wards	{5,10}	11.01%	30.19%	78%
no threshold	15	18.03%	18.03%	84%
threshold = 1	15	4.21%	25.28%	79%
threshold = 2	15	1.06%	30.99%	75%

Table 12.3: Threshold policies with $\lambda_1 = 1$, $\lambda_2 = 4$ and $\beta_1 = \beta_2 = 3$

The threshold policy can be easily generalized to more than 2 patient types, every type having its own threshold level. Determining the right mix of threshold levels does become a little more involved. More challenging is the case when the ALOS of the different patient types are very different. In that case a one-dimensional birth-death process does not suffice to model the system: a separate state variable is needed for every patient type. With two types, the state is thus of the form (x_1, x_2), with x_i the number of patients of type i, and $\mathcal{X} = \{(x_1, x_2) | x_i \in \mathbb{N}_0, x_1 + x_2 \leq s\}$. The optimal admission policy is a function of the state and can be rather complicated. Simple threshold policies which depend only on $x_1 + x_2$ perform quite well and are much easier to implement.

Threshold policies are well suited for nursing wards with a single specialty and where no different skills are needed for different types of patients. In a ward shared by several specialties and different types of patients threshold policies have the disadvantages that:
- there is no upper bound (other than s' or even s) on the number of beds that can be occupied by a certain type of patients making that all nurses should be able to handle all patients, even if they are assigned to a subset of the beds;
- there is no lower bound on the number of beds available for a specialty, making it, for example, difficult to plan operations in advance that have the ward as next process step.

To avoid both disadvantages we can assign beds of the shared ward in the following way: there are number s_1 and s_2 with $s_1 + s_2 \leq s$ that are the beds reserved for type 1 and 2, respectively. The number of shared beds is $s - s_1 - s_2$. The admission policy is as follows: in state (x_1, x_2) with $x_1 + x_2 < s$ type 1 is admitted if $x_1 < s - s_2$, and vice versa for type 2. For a numerical example, computed using a 2-dimensional Markov chain, see Table 12.4.

s	s_1	s_2	overall occupancy
15	10	5	86%
15	8	5	90%
15	6	5	91%
15	6	3	92%
15	0	0	92%

Table 12.4: Assigned beds policy with $\lambda_1 = 4$, $\lambda_2 = 4$ and $\beta_1 = \beta_2 = 3$

Note that for the computations we have to analyze a more-dimensional Markov chain, even if the ALOS are equal, because the policy and thus the transition rates depend on the numbers of beds occupied by different specialties.

12.6 Outpatient appointment scheduling

Consider a health care facility with a single doctor, or another resource such as radiology equipment, for which appointments are made. We can also consider facilities with more than one resource, as long as appointments are made for a specific doctor: then we can regard it as one or more parallel single-server systems. The method described below can even be extended to the situation where the appointments are not made for a specific doctor, but we will not discuss the technicalities of such a generalization.

The appointment schedule that is most often used is called the *individual schedule*. It means that all appointments are made with equal distance. For example, consider a session of 3 hours where 12 patients are scheduled. Then they are scheduled at 15-minute intervals. Already in the fifties it was clear that this assignment rule is far from optimal. Often the doctor idles quite a lot the first hour, due to no-shows and possibly short treatment times. On the other hand, delays accumulate over the course of the session making the session often run late. For this reason a British mathematician and a British doctor, Bailey and Welch, worked together on this problem and came up with a rule that consists of taking the last patient of the individual schedule and putting it up front together with the first patient. This avoids the slow start that characterizes many clinics, and assures that there is on average some backlog without making the waiting room times much longer.

In this section we develop a mathematical model to evaluate different outpatient scheduling policies and to find a good one. It is the objective to schedule the appointments in such a way that the physician's idle time and the total patients waiting room time are minimized. These are conflicting objectives, and thus we have to find a trade-off between the two.

The model is as follows. We have a session consisting of T intervals, each of length d. d is a multiple of some time unit, say minutes. We want to schedule N patients during this period. We have a possibly random service time $S \in \mathbb{N}_0$, $\mathbb{P}(S = s) = p(s)$. No-shows can occur. We denote by q the no-show probability. The schedule is represented by a T-dimensional vector, x_t indicating the number of arrivals at the beginning of interval t.

Example 12.6.1 For $d = 5$, $T = 36$, and $N = 12$ the individual schedule is geven by $x = (1, 0, 0, 1, 0, 0, \dots, 1, 0, 0)$ and the Bailey-Welch rule by $x = (2, 0, 0, 1, 0, 0, \dots, 1, 0, 0, 0, 0, 0)$.

For a given schedule x, we denote with π_t^- (π_t^+) the distribution of the amount of work at t just before (after) the arrivals that are scheduled at t. Using ideas from discrete-time Markov chains we find that $\pi_{t+1}^-(k) = \pi_t^+(k + d)$, $\pi_t^+(k) = \pi_t^-(k)$ if $x_t = 0$, and $\pi_t^+(k) = q\pi_t^-(k) + (1 - q)\sum_{i=0}^k \pi_t^-(i)p(k - i)$ if $x_t = 1$. If $x_t > 1$ then the $p(k - i)$ in the last formula have to be replaced by a convolution. From π_t^- and π_t^+ all performance measures can be calculated: the average waiting time of arriving patients W, the idleness of the doctor, and also the *tardiness* of the doctor. The tardiness Z is the time the doctor finishes after the end of the session, earliness counting for 0. In our situation, the expected tardiness is equal to $\mathbb{E}Z = \sum_k k\pi_{T+1}^-(k)$ (note that $T + 1$ denotes the end of interval T, a moment at which no arrivals occur). Finally, let I denote the idleness of the doctor up to Td, the planned end of the session. In rows 1 and 2 of Table 12.5 we give the outcomes for the numbers and rules of Example 12.6.1, with exponentially distributed service times with expectation 15 minutes. For the sequence-entry the part between brackets is repeated the number of times indicated by the index. Thus $(100)^{12}$ means repeating 100 12 times, the individual schedule. $200(100)^{10}000$ is the Bailey-Welch rule. We see that the Bailey-Welch rule reduces the tardiness considerably, at the expense of an increase in waiting time.

rule	q	N	sequence	$\mathbb{E}W$	$\mathbb{E}Z$	$\mathbb{E}I$
individual	0.05	12	$(100)^{12}$	16:33	28:50	35:33
Bailey-Welch	0.05	12	$200(100)^{10}000$	20:29	21:30	23:06
$\min_x\{\mathbb{E}W + \mathbb{E}Z\}$	0.05	12	$110(100)^7(010)^3000$	19:32	22:15	25:27
$\min_x\{\mathbb{E}W + 1.15\mathbb{E}Z\}$	0.05	12	$1101(010)^3(001)^500010^4$	20:28	21:28	24:09
$\min_x\{\mathbb{E}W + \mathbb{E}Z\}$	0.3	12	$201(001)^90^6$	12:43	7:57	43:00
$\min_x\{\mathbb{E}W + \mathbb{E}Z\}$	0.3	15	$210(10100)^4(10010)^20^3$	20:10	18:25	31:31
$\min_x\{\mathbb{E}W + \mathbb{E}Z\}$	0.3	16	$2(10)^4(10010)^3(10100)^20^2$	22:49	23:22	27:53

Table 12.5: Outpatient scheduling policies with $d = 5$ and $T = 36$

It is clear that patient waiting time and tardiness are conflicting objectives. For any given trade-off the best schedule has to be found. The trade-off can be formulated in different ways: as a linear combination of the form $\min_x\{\mathbb{E}W + \alpha\mathbb{E}Z\}$ for some $\alpha > 0$, or by using a constraint, for example $\min_x\{\mathbb{E}Z|\mathbb{E}W \leq \beta\}$ for some $\beta > 0$. The policies that are found are all located on the *efficiency frontier*, discussed in Section 7.4. Rows 3 and 4 of Table 12.5 give outcomes of this type of analysis. Note that the outcomes of row 4 are slightly better than those for the Bailey-Welch rule. Thus the Bailey-Welch rule is not always on the efficiency frontier. However, the differences are so small that this has little practical importance.

The current schedule has a scheduled load of 100% and 5% no-shows. Without the no-shows the performance would be worse, the optimal schedule minimizing $\mathbb{E}W + \mathbb{E}Z$

has $\mathbb{E}W \approx 21$ minutes and $\mathbb{E}Z \approx 26$ minutes. Thanks to the no-shows the performance remains acceptable. In the examples we took 5%, but in practice the no-show percentage can be as high as 30%! Planning at 100% capacity now leads to very good performance, but evidently a low utilization. Thus we plan more than 100%, anticipating the fact that some will not show up, a practice known in airlines as *overbooking*: see rows 5, 6 and 7 in Table 12.5. It is interesting to see that more patients are scheduled early, anticipating no-shows.

To do:
- replace exponential by normal, average 15 and std dev 5
- study influence variability (from deterministic to normal with std dev 10)
- add emergencies: av 3 and 6 per session (same service time) and decrease scheduled patients accordingly

Remark 12.6.2 The scheduling of operations, *operating room scheduling*, has much in common with appointment scheduling as discussed in this section. However, there are a few differences, the most important one being the following. In the appointment scheduling problem, when a doctor is ahead of time, he or she idles while waiting for the next patient. When performing operations, it is usually possible to call the next patient earlier if the operations are ahead of the schedule. This means that the time to execute a schedule is simply the sum of the operation times. This makes that operation scheduling is of a really different nature than appointment scheduling.

12.7 Capacity decisions for outpatient departments

Elective care patients have to wait twice before seeing a doctor: the time between the moment the appointment is made and the time of appointment (usually measured in days or even months) and the waiting room time (measured in minutes). The waiting room time was discussed in the previous section. Here we discuss the time until appointment or, depending on the way in which the appointment system works, the time patients spend on a waiting list.

For elective patients we distinguish between two types of doctors (or other resources): those that have a fixed group of patients which they see regularly during a certain period of time (for example, general practitioners, gyneacologists) and those that treat in principle anyone who is contacting them. For the second group it has been observed that long waiting lists exist that are not only annoying and giving hospitals a bad reputaion, it is often also endangering the patient's health. See also the discussion of waiting lists, in the context of *Advanced Excess*, in Section 12.2. There it was observed that many health care facilities show almost constant access times. This is in contrast with the standard delay models: queueing theory tells us that either the system should empty now and then or the access time should grow to infinity. The main reason is supposedly that customers leave the queue before being seen. There can be many reasons for that: the patient found what she searched somewhere else, the need disappeared, or, on the contrary, the patient's condition became worse and the patient therefore fell in a different category and is perhaps being treated as emergency. In this section we consider a model for this situation. We start from the regular $M|M|1$ model that we extend by adding the possibility of patients abandoning the queue, similarly to the model studied in section 13.7. There it was used to model callers abandoning while waiting for a call center to answer their call.

We model the situation as follows. The unit in which we work is working days. Let μ be the average number of appointment on a day. Let λ be the average number of appointments made per working day. Define the time until patients leave the queue (due to whatever cause) supposing they are not being served as Y. Estimating the distribution or moments of Y can be done using the Kaplan-Meier estimator, see Section 1.8. Take $\gamma = 1/\mathbb{E}Y$, the rate at which a waiting patient leaves the queue. Now we can model this as a birth-death process with rates $\lambda(x, x+1) = \lambda$ for $x \geq 0$ and $\lambda(x, x-1) = \mu + (x-1)^+\gamma$ for $x > 0$. This queue is always stable, because the departure rate is increasing (see Section 13.7 for a formal argument). Numerical experiments show what we observe in practice: in overload $(\lambda > \mu)$ this queue hardly empties and the stationary distribution is concentrated around the state x with $\lambda \approx \mu + x\gamma$.

Insert figure showing this

Under Advanced Access (AA) one tries to offer very short access times by same-day or next-day appointments. In our queueing model, this requires that $\lambda < \mu$. In AA projects the goal is to increase μ by applying lean principles. However, short waiting times attract more patients: $\lambda(x, x+1)$ is probably not constant, but decreasing in x. This makes it hard to maintain the short access times, and apparently this is indeed the reason why many AA projects fail.

The way to keep short access times is by controling the arrival rate. This is possible in specializations where the doctors keep a fixed group of patients, the *panel*, which visit them regularly. Examples are gyneacologists, general practitioners and doctors who treat patients with uncurable diseases such as diabetis. For example, a general practitioner in the Netherlands usually has 2000 patients. Each of these patients has a small probability of making an appointment. The resulting arrival process is close to a Poisson process (see Section 2.1). The panel size can now be chosen in such a way as to avoid long waiting times.

12.8 Multi-unit scheduling problems

Delivering health care often goes through multiple stages, such as visits to a doctor, tests, surgery, time spent at wards, etc. We first consider the situation where it is allowed to wait between different stages. This occurs with outpatients where the different resources are planned one by one. It also occurs sometimes with inpatients: think of a patient lying at a ward waiting for a surgical slot, although this is usually undesirable, but not always avoidable (using *lean* terminology, waiting is considered waste).

Multi-stage processes with waiting in between ressemble flow lines or job shops, depending on their characteristics. Waiting is seldomly evenly distributed, most of the waiting usually occurs at the *bottleneck*. For this reason, attention should be focused at the bottleneck. At the bottleneck the available capacity should be used as effciently as possible.

As a result, waiting room time at the bottleneck resource is probably higher than at other resources. In the case the bottleneck is only used by a single type of patients, then it is possible to schedule using methods such as the one discussed in Section 12.6. In case of multiple types of patients we have to decide how many slots to allocate to different types of patients. If we do not do this then resources are used by the patients that make their appointment longest in advance, and these are usually not the most urgent or profitable. Especially if a resource is used by outpatients on one hand and emergency and/or inpatients on the other, then slots should be reserved for the latter categories. In the case of emergencies, there is a medical need for this; in the case of inpatients the costs for waiting for the first free slot is much too high because they are occupying expensive resources such as a bed.

Let us formulate a mathematical model for a single period. By this we mean that we incur costs for delayed inpatients, but we do not take into account the influence of that on subsequent periods. The demand for slots for inpatients is given by a random variable X, and we reserve S slots for inpatients. We assume that there is enough demand from outpatients such that all remaining places are scheduled. There is a reward r for each patient that uses the resource; there is a cost c for every inpatient who cannot be scheduled. For every slot a reward r is earned unless the slot remains empty. Thus the expected reward that is lost is equal to $r\mathbb{E}(S-X)^+$, the expected costs for delayed customers is equal to $c\mathbb{E}(X-S)^+$. It is thus our objective to find S^* that minimizes

$$C(S) = r\mathbb{E}(S-X)^+ + c\mathbb{E}(X-S)^+. \tag{12.2}$$

This problem is equivalent to the newsboy problem of Section 6.2. The optimal solution is given by Theorem 6.2.1: S^* is the smallest S for which $F_X(S) \geq c/(c+r)$.

Example 12.8.1 Example with realistic numbers

As the concept of planning the steps in the health delivery process one by one is abandoned ever more often, the need to plan the different step consecutively increases. Next to that, for inpatients, this always has been the case in many situations. For example, after open heart surgery an IC bed is necessary. If none is available at the time the operation is supposed to start, then the operation has to be cancelled. This leads to challenging planning problems for which the planning methodology is yet to be developed.

12.9 Further reading

Brandeau et al. [17] contains a collection of papers on OR and health care. A journal at the interface of health care and OR/MS is *Health Care Management Science*.

The Bailey-Welch rule was introduced in [98]. Gallivan et al. [37] considers a mathematical model for a ward with a single class of patients.

A paper in which the term overbooking is used in the context of patient scheduling is Laganga & Lawrence [63]. They also state that in the case they considered 30% no-shows is not exceptional.

The idea of determining the optimal panel size is introduced by Green & Savin, see Green [44] for a tutorial on this and other subjects.

The concept of a focused factory has been introduced by Skinner [85].

Information on project management for process improvement projects in health care can be found in Belson [15].

12.10 Exercises

Exercise 12.1 Consider Exercise 3.4. Under the Poisson assumption, calculate β and the occupancy.

Exercise 12.2 Reproduce the numbers of Example 12.2.

Exercise 12.3 Consider Example 12.4.2. We implement the idea of protecting type 2 patients by introducing a threshold policy.
a. Develop a birth-death process that models this situation.
b. Compute the blocking probabilities for various values of the threshold and report on the consequences.

Exercise 12.4 A small IC unit has 6 beds and two types of patients. The average LoS is 3 days, and on average one patient arrives per day in each class. Compute the blocking probabilities for each class in the following situations:
a. No admission control is used, i.e., every patient is admitted unless all beds are occupied;
b. When exactly 1 bed is available then type 2 is blocked and type 1 is admitted.

Exercise 12.5 Consider an outpatient department with patients that might leave the queue while waiting.
a. Model it by a birth-death process with s servers and a departure rate that is linear increasing in the number of waiting patients.
b. Find an expression for the stationary distribution.
c. Calculate it for $s = 1$, $\mu = 20$, $\lambda = 22$, and every day on average 1% of the patients that are waiting leave the queue.
d. What is the state with the highest stationary probability? Can you answer this question without computing the stationary distribution?

Exercise 12.6 The waiting list for a certain medical capacity is modeled as a waiting queue with a single server, exponential service times and an arrival process with an arrival rate that is possibly a function of the waiting time.
a. Compute the expected waiting time and the 10%, 20%, etc. quantiles of the waiting time for a system with constant arrival rate, an expected service time of 30 minutes, and a load of 95%.
b. Compute the same numbers for an approximation where the waiting times, given the number of customers on arrival, is based on deterministic service durations. (That is, when

there are x customers in the system, then the waiting time is not gamma with x phases, but equal to x times the service duration.)
c. Now suppose that the arrival rate is a linear function between 0 and 400 hours: at 0 the value is 1 per hour, at 400 it reaches 0 and stays 0 for higher values.
d. Approximate the expected waiting time and the waiting time quantiles for this situation.

Exercise 12.7 A reservation system for MRI slots works as follows. Every day k slots are reserved for semi-urgent patients. Patients (all seen in the morning) are booked for slots for the same day, or, if no slots are available the same day, then for the next day. When there are even no slots the next day then an ad hoc solution is sought with cost c_1. At the end of a day slots are possibly given back for the next day to make sure that the number of free slots does not exceed $m \leq k$. Given back a slot costs c_2. Any left-over slots at the end of the day cost c_3. Demand is assumed to be Poisson distributed.
a. Model this process as a Markov chain and indicate how to compute the costs from the stationary distribution.
b. Compute these numbers for $k = 2$, $m = 1$, an average demand of 1, and $c_1 = 20$, $c_2 = 1$, and $c_3 = 10$.
c. Formulate your ideas on how to find optimal values for k and m.

Exercise 12.8 Consider the problem of reserving slots for inpatients of Equation (12.2). Take $c = 300$, $r = 50$, and D Poisson distributed with average 10.
a. What is the optimal value of S using the single-period approximation?
Consider now the original problem where refused inpatients are treated later.
b. Do a simulation over say 50 periods to determine the optimal value of S. Explain how you set up the experiment, how many runs you did and why, and so forth.

Chapter 13

Call Centers

In this chapter we study call centers. In call centers there are many interesting modeling questions, often related, but not restricted, to queueing models. Studying quantitative issues of call centers is not just a matter of solving mathematically challenging problems: there is a big economic interest because of the large number of call centers in operation nowadays. To illustrate this, one of the many estimations states that in the US more than 6,000,000 agents (as call center employees are called) work in call centers. In Holland this number is around 100,000, with half of all companies having at least one call center. Model-based Decision Support Systems exist to assist planners with their work, especially in the area of *Workforce Management* (WFM). We discuss typical modeling problems in call centers, including those that are part of WFM. But before going into the modeling issues, we give a general description of call centers and go into the statistical aspects of call center customer behavior.

13.1 Introduction

A call center is a collection of resources (typically agents and ICT equipment) capable of delivering services by telephone. The number of call centers is increasing rapidly: many companies see a call center as a way to have a close relation with the customer. As such, setting up a call center is often part of a business process redesign or re-engineering (BPR) project in which work from the back-office is shifted to the front-office (the call center). One hopes that this results in increased customer satisfaction through more direct communication and reduced response times in administrative processes.

This shift to call centers is enabled by a number of technological advances, mainly in the area of information and communication technology (ICT). Here we have to think of PABX's (private automatic branch exchanges, the telephone exchanges within companies) that support ACD (automatic call distribution) functionality. To every PABX a number of extensions (telephones) are connected; the ACD switch is able to select an extension with a free agent for a call coming in over a certain line.

At least as important as advances in telecommunication are those in information technology. Instead of searching for a paper file in a central archive, that renders immediate handling of a task related to a file impossible, an agent has nowadays access to centrally stored computer files in a fraction of a second.

Current trends include CTI (computer telephone integration), that goes even further: when a client is calling, the PABX recognizes the phone number of the client, and using CTI the computer system searches automatically for the files of the customer with this phone number.

Inbound and outbound traffic The most important distinction between call centers is between those handling *inbound* and *outbound* traffic. Synonyms for inbound/outbound are incoming/outgoing. There are few call centers with only inbound traffic, usually a mix of the two occurs. Call centers with only outbound traffic are seen more often, they are mostly used for advertisement campaigns. Issues around an optimal use of the prime resource, call center agents, amount here to problems of another type than for mixed centers. In order to avoid waiting of the agent for the PABX to call the next customer the call center software makes the PABX phone while no agent is available at that moment, in the hope that an agent will be available when there is a customer on the line. These units are called *predictive dialers*. They have to balance between agent productivity (is there always a customer right away?) and customer dissatisfaction (because it might occur that no agent is available when the customer picks up the phone).

We concentrate on problems that occur for the majority of call centers that handle both inbound and outbound traffic, but for which the quantitative issues are dominated by the inbound character.

Different skills The traditional call center is the switch board of a company where someone switches all incoming calls to the right extension. Although highly automated, we still see this type of call center in many firms. But, next to these traditional groups, we find call centers with agents that are specialized professionals. Calls come in on different lines, or are redirected through the use of IVR (interactive voice response), to groups of agents. Agents can have multiple *skills*: the typical example is the help desk supporting multiple products in multiple languages. Agents are often *cross-trained*, i.e., they have multiple skills. We say that each agent has its own *skill set.* making sure that there are enough agents for each skill group and that the calls are routed in an efficient way is a highly complex problem, that is the subject of Section 13.8.

Objectives The objective of a call center is usually to obtain at least a certain minimal service level (SL) for minimal costs. The SL will in general consists of multiple criteria, some of which are related to the actual call, some to the time before, the waiting time. Although there are also some interesting quantitative issues related to the call itself, we will concentrate on SL metrics that have to do with waiting times and calls that abandon before reaching an agent. The most usual way to define service level is by taking a percentage α and a number a for which must hold: α% of the customers must have a waiting time shorter than a seconds. Often we see values such as $\alpha = 80$ and $a = 20$ (the 'industry standard', for what it's worth). Whether this service level must hold for every time interval, or on average over the whole day, is not always clear.

Another way to define service times is by the number of abandoned calls. Of course this number is strongly related to the waiting time; often call center managers will choose the service level parameters α and a such that abandonments occur rarely. Percentages up to 3 or 5% are often considered acceptable.

On the other side are the costs. Highest are personnel costs, estimations range between 60-70% of the total costs. Therefore we concentrate only on labor costs, and we assume that other resources (such as ICT) are not bottlenecks. A quantitative study of the optimal use of the resource labor is thus the main objective of this chapter. Main issues are here the scheduling of agents and skill-based routing. Of course the hardware forms a constraint in the qualitative sense. For example, it determines whether skill-based routing is possible, and how it can be implemented.

Remark 13.1.1 In some cases the expected profit of a call can be calculated. Then an overall optimal policy can be calculated that makes a trade-off between profit and costs from personnel and telecommunications (if the call center pays for this). The danger is that this may lead to customer dissatisfaction and reduced numbers of calls. We focus on making the cost-SL trade-off explicit.

13.2 Modeling a call center

In what follows we assume that we only have inbound traffic. Additional outbound traffic will be discussed later. A call center is modeled as (a set of) multi-server queues, where each server represents an agent. In a well-operated call center the number of lines by which the call center is connected to the public network is sufficient to deal with the load in almost all situations, also abandonments rarely occur. In this case the $M|G|s$ queue is the obvious model, given that the agents are the bottleneck and that an infinite waiting space is available. In other cases the number of lines and the possibility of abandonments should be modeled explicitly. This leads to what is sometimes notated as the $M|G|s|N-G'$ model, where N is the number of lines, and G' denoted the patience of the customers, i.e., the distribution of the time until abandonment.

Input to the above queueing models are the arrival process, the service times, the patience distribution, the number of agents, and the number of lines. The latter two are simple to obtain. In the rest of this section we first discuss in detail the arrival process, then the service times and the patience.

Typically we model the arrivals as coming from a Poisson process, with a parameter that depends on the time. Such a process is called an *inhomogeneous Poisson process*. Modeling in such a way is necessary if the analysis takes into account long time intervals, because the load of a call center fluctuates strongly over the day and between days. If one is interested in for example peak hour performance, then of course a standard homogeneous Poisson process suffices. Most queueing formulas assume a constant arrival rate, which is in conflict with the fluctuating rate of the inhomogeneous Poisson process. Therefore we take the rate function piecewise constant, during for example 15 minute intervals. Thus we do not find a stationary situation; however, experience shows that this type of stochastic processes converges fast to its equilibrium state, meaning that in general we can take the stationary situation as approximation. Another assumption not yet discussed is the fact that arrivals are Poisson. This is usually the case, but in case of doubt it should be verified statistically. A typical exception are calls coming in after a telephone number being shown on television.

A major problem is how to deal with the service times: there is no closed-form expression for the waiting time in the $M|G|s$ system. In commercial software service times are almost always assumed to be exponential. The resulting $M|M|s$ queue or *Erlang delay model* (to managers known as Erlang C) is relatively easy to analyse, closed-form expressions exists for the average waiting times and the service level (the percentage waiting longer than some a). In Chapter 5 these expressions are given. In general we find that taking exponential service times is a good approximation, especially for large call centers. This modeling choice should however be validated, for example by a simulation.

13.3 Workforce management

The basic Erlang formula, or one of its generalizations, used for calculating the performance in an arbitrary time interval, is the starting point of the management of the workforce. Managing this workforce in such a way that service level targets are met and that costs are minimized is the subject of this section. This operational task consists mainly of scheduling agents such that there are enough agents at each time interval to meet the desired service levels. DSSs exist to support this task. We discuss the typical operation of this type of tool.

The first step of workforce management (WFM) is predicting the load of the system. Predicting call volume on the basis of historical data is quite difficult due to the diversity of events that have to be taken into account, ranging from special holidays to the introduction of new products that influence the number of calls. The forecasting step results in predictions of the arrival rate for each interval that the call center is operational. See Section 2.5 for more details on this subject.

Based on these predictions the minimum number of agents needed to reach the service level for each interval can be calculated, using the Erlang formula. From this daily schedules can be made using a set of standard shifts. This is done using some optimization routine. There can be numerous different shifts, depending on starting times, lengths and the moments of the breaks.

If the minimum levels are translated into shifts, then these shifts have to be assigned to agents. This is a complicated task for which few useful tools exist. Complicating factors are here personal preferences of the agents and restrictions imposed by contracts and by law. For more details on making shifts and assigning them to employees, see Chapter **??**.

By now we have a complete picture of the standard approach in which the problem is decomposed into four steps: call volume estimation, calculation of minimum number of agents, determining shifts, assigning agents to shifts. We find this approach in many software packages that are especially designed for this task. But matching demand and personnel does not end with determining this say monthly schedule: unpredictable events create fluctuations in anticipated call volume, and the availability in personnel can change for a multitude of reasons of which illness is only one. For these reasons the schedules have to be updated continuously, and traffic loads and SLs are monitored continuously during the day to be able to adjust to changes. The importance to adopt to changes and the ways to adopt to them are discussed later in this chapter.

Example 13.3.1 A call center handling insurance claims anticipated on its on average 5% absence rate by scheduling always 5% additional personnel while setting up the schedule a month earlier. Schedules were not changed on the basis of new information. It appears to management the right number of agents was rarely there; instead there were either too many agents, or not enough. When management realized this they reduced the 5% additional staff and assured that there is enough personnel that can be scheduled on a flexible short-term basis.

The current approach to WFM can be improved in several ways. We discuss these issues after having gone into more detail about the Erlang C model.

13.4 Some properties of the Erlang delay formula

The Erlang C or delay formula plays a central role in call center management. It is derived in Section 5.4. In this section we study some properties of the Erlang C that are relevant to call centers.

A formula such as the Erlang formula is not just a black box that produces numbers on command; it is important to get a feeling for it, to understand how performance changes as certain parameters are changed, etc. Obvious and well-known properties are that the mean waiting times and productivity are decreasing convex functions of the number of agents s, as long as the system is stable, i.e., $\rho = \lambda/(s\mu) < 1$, with λ the arrival rate and μ the service rate. For the productivity this follows from the fact that it is equal to $\lambda/(s\mu)$. The form of the mean waiting time as a function of s can be guessed from the limiting behavior. (The mean waiting time is equal to $C(s, \rho)/(s\mu - \lambda)$ (see Section 5.4), but the analysis of this formula is complicated by the fact that the expression for the delay probability C is not that simple.)

To better understand the Erlang C formula it is advisable to experiment with a so-called *Erlang calculator*, next to reading this section. There are several available for free on the internet; ours can be found at www.math.vu.nl/∼koole/obp/ErlangC.

Scaling in time In Section 5.4 we saw that the average queue length $\mathbb{E}L_Q$ of the $M|M|s$ model depends only on ρ. This means that if we multiply λ and μ by a constant c, i.e., if we scale time, then $\mathbb{E}L_Q$ remains the same. This does not hold for the time in queue $\mathbb{E}W_Q$: by Little's law $\mathbb{E}W_Q = \mathbb{E}L_Q/\lambda$, and thus $\mathbb{E}W_Q$ is divided by the scaling factor c. This means, for two similar call centers with equal loads, that the one with longer service times has a lower service level. Waiting time distributions do not scale linearly in the scaling constant, as is readily seen because $C(s,\rho)$, the probability of finding all agents occupied, depends only on ρ (and on s of course). When queueing occurs then it takes more than c times as long.

Thus for short service times it is possible to have both a high service level and a high productivity; for longer service times we either have to sacrifice service level or productivity. A solution is to let agents do outbound work if many agents are on available (called call blending). This can be seen as a way to decrease uncertainty due to the arrival process. Other options are to have employees with other tasks on stand-by for periods with an unexpectedly high load.

Example 13.4.1 A help desk manager is confronted with a mean call duration of more than 10 minutes. He decides that it is not possible to have a high productivity and an acceptable service level. He decides to cancel the possibility to wait in queue, instead customers that find all agents occupied are asked to leave their telephone number. As soon as more than one agent is free then customers with recorded messages are called back. Productivity is high, and still a large proportion of customers are served immediately by a free agent.

Scaling in size Large call centers have a better productivity to service ratio, i.e., they show economies of scale. Let us first illustrate this with a simple but extreme example.

Example 13.4.2 An emergency service wants to guarantee that at least 99 out of 100 calls do not have to wait to be answered. Assume that $\mu = 1$. Then the maximum arrival rate for a single agent is approximately 0.01; for two agents this is about 10 times as much, thus an increase in maximal attainable productivity of a factor 5.

We consider in more detail scaling in size in another way, by considering merging two call centers. Thus suppose we have a $M(\lambda_1)|M(\mu)|s_1$ and a $M(\lambda_2)|M(\mu)|s_2$ queue which are merged to one $M(\lambda_1+\lambda_2)|M(\mu)|s_1+s_2$ queue. (Because the call centers could apparently be merged, we assume that the telephonic services are comparable and therefore we took the service times equal.) This is the way call center managers look at it: Many companies have physically separated call centers, but by using ICT they can be merged in a virtual way, calls that are to be put on wait are offered to the other call centers. The simplest way to prove that merging decreases waiting times is by *coupling*. This can be seen as writing out a simulation using the same samples of arrival and service times in such a way that the waiting time of each customer is lower in the merged system. This is relatively simple using an intermediate step: we look at the two separate centers as one center where for some reason certain agents are not used for certain calls. If we omit this restriction then customers are helped earlier, and waiting decreases.

The economies of scale are well illustrated in Figure 13.1. There the SLs for two call centers are plotted for varying λ. We took $\beta = \mu^{-1} = 5$, $s = 7$ (solid) and $s = 14$ (dashed), $a = 0.33$, and varying λ. The horizon axis is scaled such that both centers have the same

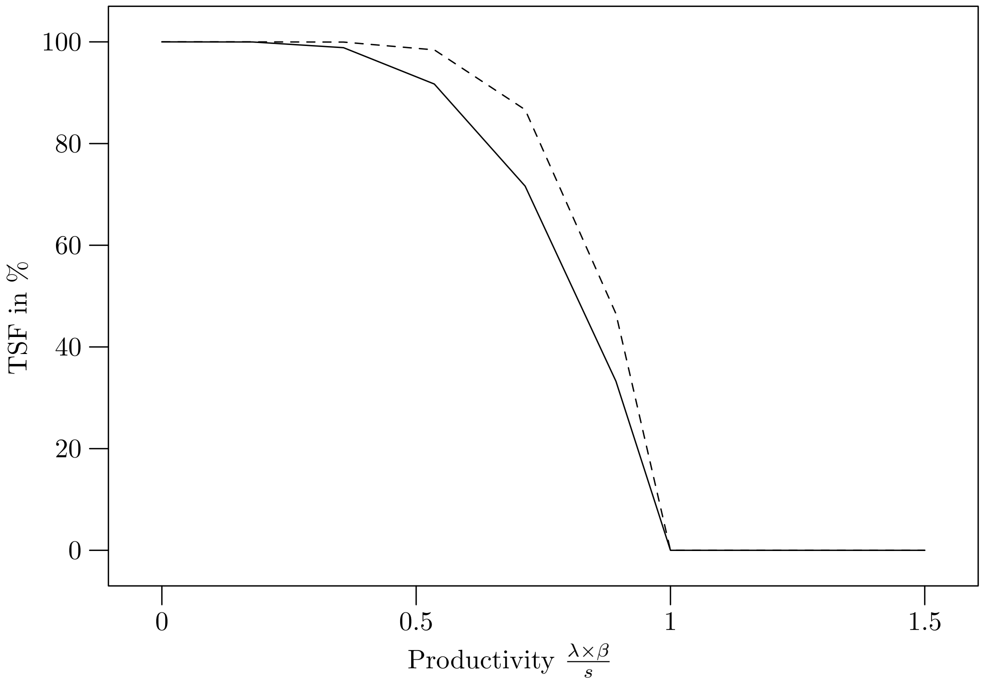

Figure 13.1: Two call centers with $\beta = \mu^{-1} = 5$, $s = 7$ (solid) and $s = 14$ (dashed), and $a = 0.33$

load. The fact that the dashed line should lie above the solid follows from economies of scale.

A way to quantify the economies of scale is by the *square-root safety staffing principle.* Define $s(\lambda)$ for given β by:

$$s(\lambda) = \min_s \{M|M|s \text{ satisfies SL constraint for } \lambda\}.$$

Then the square-root safety staffing principle (SRSSP) states $s(\lambda) \approx \lambda\beta + \alpha\sqrt{\lambda\beta}$ for some constant α, that does not depend on s, but that represents the level of service.

Example 13.4.3 Let us consider a numerical example. Take $\beta = 1$. If $\lambda = 16$ and $s = 20$ then the SL is equal to 93.25%, as is ealisy verified by an Erlang calculator. From the SRSSP it follows that $\alpha = 1$. Now suppose λ gets 25. What is $s(25)$, for the same SL? Now both the Erlang calculator (which gives the exact number) and the SRSSP (which is an approximation) both give $s(25) = 30$, which shows the accuracy of the SRSSP in this situation. This is no exception.

The SRSSP shows that us that the overcapacity should be proportional to the square root of the load to the system.

Robustness The load to the system and the available capacity, agents, are, in practice, subject to short-term, unpredictable changes. Take λ. Forecasting using a workforce management tool produces as a rule a single number. This number however should be interpreted as an average, the real value will rarely be exactly the estimated value. A deviation from the point estimate of 10% is no exception. An important question is therefore: how sensitive is the Erlang formula to changes in the arrival rate? This depends on the operational regime. Due to economies of scale call centers work with a productivity that is close to one while keeping to the required service level. In this regime small changes in parameters can cause dramatic changes in service level. A major challenge in call centers is therefore how to deal with unpredictable variations in load. The same applies to the number of agents s.

In practice the situation is not as bad as it looks: callers abandon after some time, leading to shorter waiting times for callers that do not abandon. Note that callers that abandon are usually highly dissatisfied with the service. This can be avoided to some extent by giving an estimation of the waiting time to callers as soon as they enter the queue. Then they can decide whether they consider it worth waiting or not.

Example 13.4.4 Take a call center with $\lambda = 10$, $\beta = 2$, and $s = 21$. Then only 36% is served within 20 seconds. If customers abandon on average after three minutes of waiting, then only 6% abandons and 76% of those that have enough patience wait less than 20 seconds.

13.5 Improving workforce management

The standard approach to WFM leaves much to be desired. In the first place, scheduling at least the minimum number of agents in each interval gives an overall service level that is well above the minimum. This is for two reasons: in the first place because of the integral nature of the number of agents, in the second place because of the fact that the optimal way of covering the highly varying staffing requirements with shifts of fixed length (e.g., 4 or 8 hours) might introduce some slack. This calls for an approach that considers the average daily service levels, that allows intervals with a low SL to be compensated by intervals with a high SL. Note that we should account for the arriving rate when calculating the expected daily service level. Let λ_i be the arrival rate in interval $i \in \{1, \dots, I\}$, $S_i(s_i)$ the SL as a function of the number of agents, then the expected daily SL S is given by

$$S = \sum_{i=1}^{I} \frac{\lambda_i}{\sum_{j=1}^{I} \lambda_j} S_i. \tag{13.1}$$

When shifting from interval SL constraints to overall SL constraints, we can profit from economies of scale in the following way. The offered load to a call center typically varies heavily over the day. This means that the agents-SL curve is much steeper during the busy hour than during less busy hours. This, together with the higher weighing factors in (13.1) for busy intervals, makes it interesting to overstaff during busy periods and to understaff during quiet periods. A small example with only two intervals is worked out in Table 13.1. By comparing the first two rows it can be seen that the service level improves

s_1	s_2	TSF interval 1	TSF interval 2	overall TSF
13	3	89.51	95.33	90.04
14	2	95.41	76.12	93.66
13	2	89.51	76.12	88.29
13	1	89.51	0	81.37

Table 13.1: The effect of rescheduling; $\lambda_1 = 10$, $\lambda_2 = 1$, $\beta = 1$, $a = 0.333$, $\alpha = 0.8$

considerably by shifting an agent to the busy interval, although the SLs in both intervals are less balanced.

Table 13.1 also nicely illustrates the effect of the integer constraints for numbers of agents on the overall service levels. From the last line it can be shown that with two agents less we still satisfy the overall SL constraint.

Another drawback of the current WFM practice is the fact that steps three and four, determining shifts and coupling them to agents, are separated. As agents often have different types of contracts, resulting in different shifts, it is not possible to determine the right mix of shifts before actually assigning agents to them. The same holds for the personal preferences, such as employees which have to start at the same time due to car pooling, or agents which are available only part of the time due to obligations such as meetings. This calls for an integration of step three and four, determining the shifts and assigning them to agents.

In summary, we see that ideally the whole planning process should be done in a single step. From a model-solving point of view this is impossible. Of course there are simple call centers with a single type of shifts and no additional preferences, in such a case the standard approach can be followed. But in our opinion, for a complex call center, scheduling should start from the personnel preferences and the expected call volume, and should consist of a DSS in the real sense, that allows the human scheduler to evaluate changes made to the schedule.

A final way to improve WFM is using a more sophisticated model than the Erlang C. This will be discussed later.

13.6 Variability and flexibility

In Section 13.4 the lack of robustness of the Erlang delay system was discussed. Although it looks worse than it is in reality, thanks to the abandonments, measures need to be taken to be able to react to changes in load and changes in workforce availability. In practice the latter is usually done by introducing *shrinkage*: this is the difference between planned workforce and the workforce actually needed. It often includes also training, meetings, and so forth. A usual value is 30%. But evidently, the actual shrinkage is unpredictable, for example illness cannot be predicted the moment the schedule is made. Thus it rarely occurs that exactly the right number of seats is occupied. For this reason another solution is needed.

By introducing flexibility at all time levels of the operation we can offer the required SL while keeping a high productivity at the same time, independent of changes in parameters: it makes the call center more robust. At the highest level we have flexibility in contracts. With this we mean that for certain agents we can decide on a very short notice (e.g., at the beginning of the day) whether we require them to work or not. This is an excellent solution to deal with variability in arrival rate and absence. For the latter this is obvious; for the former we have to realize that the arrival rate during the first hours of the day often gives a good indication of the load during the rest of the day. Thus early in the morning it can already be decided whether additional agents are needed.

When trying to quantify this, we start with a minimum number of fixed contract agent. This minimum is based on some lower bound on the arrival rate and a minimal absence. Then we assure that there are enough agents with flexible contracts such that we can get the number of agents equal to the number required in the case of a maximal arrival rate and maximal absence.

Example 13.6.1 A call center has an arrival rate that falls between 4 and 4.8, with 90% probability. For the lower bound 50 agents are needed, for the upper bound 9 more. Out of these 50 agents between 1 and 6 agents are absent, on average 3. Thus we schedule at least 51 agents, and in the "worst" case we have to hire 14 more, on average 6.

Introducing flexible contracts gives us the possibility to handle days with a higher than usual traffic load. If the peaks are shorter, in the order of an hour, then we cannot require agents to come just for this short period of time. In this it is possible to mobilize extra workforce by having personnel from outside the office work into the call center.

Example 13.6.2 Stocks trading lines of banks have scenarios in which many people from other departments can be mobilized, in case of for example a stock market crash.

A final type of flexibility is flexibility in task assignment. This is a method to react to load fluctuations that can even work at the finest level of fluctuations, that the Erlang formula accounts for. For this it is necessary that there are, next to the incoming calls, other tasks that have less strict service requirements. Examples are outgoing calls and faxes, and more recently emails and messages entered on Web pages. They have service requirements that range from hours to days, thus of a totally different scale than the requirements of incoming calls. To be able to satisfy the service requirements for these so-called channels it suffices to schedule just enough agents to do the work. Scheduling overcapacity, as for incoming calls, is not necessary. It also doesn't matter when outgoing calls or emails are handled, as long as they are handled in the required time interval. This makes it possible to use outgoing calls to fill in the gaps left by a low offered load, and allows in case of undercapacity agents originally scheduled for emails or outgoing calls to work on incoming calls. Thus instead of assigning in a fixed way agents to ingoing or outgoing calls, they are assigned dynamically (either by the supervisor or automatically) to a certain channel. This assignment should be done carefully. A free agents should obviously be assigned to a waiting incoming call if any are present. A way to maximize productivity is by assigning free agents to outgoing calls if there are no waiting incoming calls. However, then every incoming call has to wait for a free agent. In most situations this will lead to a very low SL. The solution is to keep a number of agents free for incoming calls when none are waiting. This rule works when changing from ingoing to outgoing calls takes relatively little time. It is known as call blending, as it was originally intended for call center dealing with inbound and outbound traffic. Simply *blending* seems a more appropriate name given the recent focus on communication over the internet.

The advantages of most other channels compared to inbounds calls are clear. therefore the robustness and the SL are increased if inbound call are exchanged to emails or outbound calls. An active policy on this might reduce costs significantly.

Example 13.6.3 To make reservations for international travel the Dutch railways has two options. The first is calling the contact center by dialing an 800-number. The second is entering your travel data and the moment at which you want to be called back (a four-hour interval) on a web page. Potential travelers are thus financially stimulated to enter their data on the web page, thereby turning an inbound call into an outbound call. This allows the contact center to contact you at some quiet moment during your preferred time interval. Often the call takes little time as the agent already known the travel options, based on the data that you entered.

13.7 Generalizations of the Erlang delay formula

An issue that we have not discussed yet is how to model abandonments. If we assume exponential times (with rate γ) until abandonments occur then it can be incorporated in the Markov process that constitutes the Erlang model. The arrival rate is $\lambda(x, x+1) = \lambda$ and the departure rate for state $0 < x \leq s$ is equal to $\lambda(x, x-1) = x\mu$, as in the Erlang C model. However, for higher states the departure rate is different: $\lambda(s+x, s+x-1) = s\mu + x\gamma$ for all $x > 0$. Using Equation (4.7) we can find the stationary distribution. Note that it always exists because the sum in Equation (4.8) always exists (as long as $\gamma > 0$). This means that the system is always stable, independent of the values of λ, μ and s.

The derivation of the waiting time distribution is more involved than that of the $M|M|s$ queue (Section 5.4), because the waiting time, conditioned on the state, does not have a gamma distribution as is the case for the $M|M|s$ queue. If a customer arrives in state $s+k$, i.e., there are k waiting customers in front of him or her, then this customer has to wait a sum of exponentially distributed random variables with rates $s\mu + k\gamma, s\mu + (k-1)\gamma, \ldots, s\mu$ before being served. Such a distribution, a sum of exponentials with different rates, is known as a *hypoexponential* distribution. We derive the tail distribution for hypoexponential distributions for which all rates are different.

Let $X_k \sim \exp(\mu_k)$, $k \leq K$. We define F_k as the distribution function of $X_1 + \cdots + X_k$, $\bar{F} = 1 - F$. The result we show is as follows:

$$\bar{F}_k(t) = \sum_{i=1}^{k} \textstyle\prod_{j \neq i} \frac{\mu_j}{\mu_j - \mu_i} \mathrm{e}^{-\mu_i t}, \quad 1 < k \leq K. \tag{13.2}$$

Of course, $\bar{F}_1(t) = \exp(-\mu_1 t)$.

We prove Equation (13.2). From properties of the exponential distribution we find for $h > 0$ small, $k > 1$:

$$F_k(t+h) = \mu_k h F_{k-1}(t) + (1-\mu_k h)F_k(t) + o(h). \tag{13.3}$$

Rewriting and taking the limit as $h \to 0$ gives

$$\bar{F}'_k(t) = \mu_k(\bar{F}_{k-1}(t) - \bar{F}_k(t)), \tag{13.4}$$

for $k > 1$. Differentiating (13.2) and plugging it into this formula tells us, after some rewriting, that Equation (13.2) gives the solution to $\bar{F}_k(t)$.

A web-based calculator for the Erlang X model in which this method is implemented can be found on www.math.vu.nl/∼koole/obp/ErlangX. Figure 13.2 gives SLs and abandonment percentages for several values of γ and varying λ.

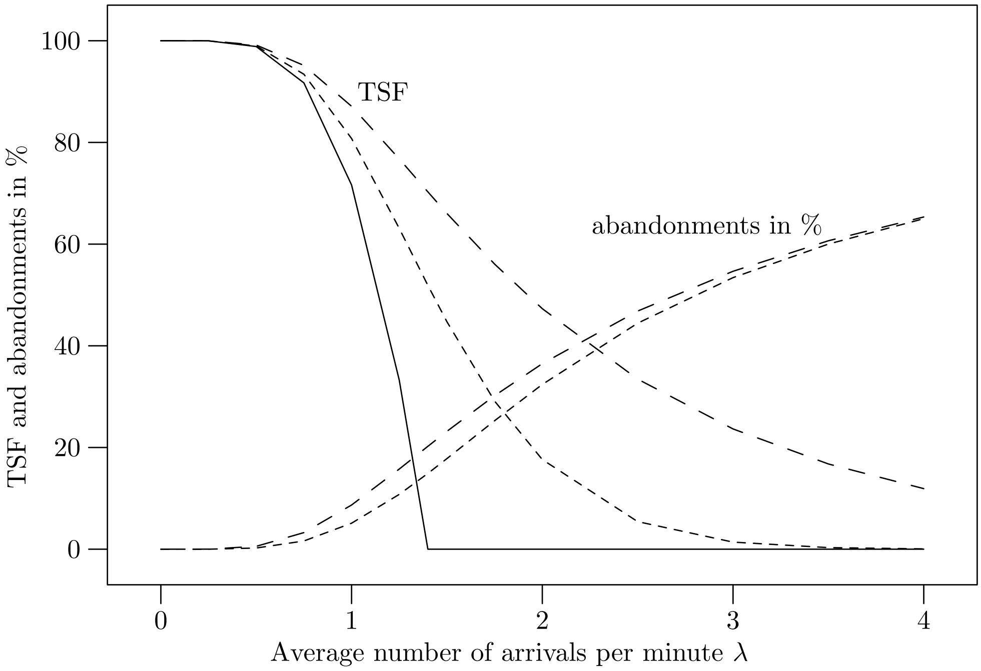

Figure 13.2: SL (also called TSF) and abandonment percentage for average patience ∞, 5 and 1 (from below), for $\beta = 5$, $s = 7$, $a = 0.33$, and varying λ

We see that adding abandonments to the Erlang model is a valuable extension in the case of call centers, certainly if the load is high compared to the number of agents. This model is sometimes called the *Erlang A model*. It models the situation where it is left to the caller to decide whether he or she is willing to wait for service. Note that an additional parameter has to be estimated. Estimating this parameter is a non-trivial statistical problem, for which the so-called Kaplan-Meier estimator gives the solution (see

Example 1.8.1). It has also been observed in the literature that the patience distribution (and not just the average patience) has a big influence on the performance measures.

From Figure 13.2 it is clear that the Erlang A model is less sensitive to parameter changes than the Erlang C model. Thus as long as we accept a certain number of abandonments, we need less flexible agents than if we base our calculations on the Erlang C model.

An alternative way to reduce the number of callers waiting for service is by limiting the number of waiting places. This $M|M|s|N$ queue is also useful if the number of lines connecting the ACD to the public network forms an important restriction. Note that this models the situation where the decision whether or not to give service to a caller is taken by the ACD. A smart way to avoid waiting times that uses a finite number of lines and call blending is asking blocked callers to leave their telephone number and asking for a convenient moment to be called back.

The next step in refining the Erlang C model is adding redials.

13.8 Multiple skills

Introducing a differentiation in skills has many managerial advantages. Training costs are lower compared to a call center where all agents should be able to deal with all calls, and the acquisition of new skills after some time offers call center agents a career path, that can help reduce turn-over. But there are also dangers related to the introduction of multiple skills. In the first place, it increases the complexity of the call center. Next, it has a lower flexibility compared to one big single-skill call center, and, finally, one might loose the economies of scale. To illustrate the latter point, let us look at the numbers in Table 13.2. Here we see a call center with two skills and a total of 24 agents, $\lambda_1 = \lambda_2 = 5$, $\beta = 2$, $a = 0.333$. The advantages of multiple skills but also of cross-training are obvious.

skill 1	skill 2	skill 1 & 2	SL
0	0	24	84.7%
2	2	20	84.4%
4	4	16	83.8%
6	6	12	83.1%
8	8	8	81.4%
10	10	4	77.5%
12	12	0	67.8%

Table 13.2: The effect of cross-training: $\lambda_1 = \lambda_2 = 5$, $\beta = 2$, $a = 0.333$

The results in Table 13.2 are obtained through simulation, except for the first and last line. This illustrates the lack of useful solution methods for these types of problems. In this section we give an overview of the problems and possible solutions.

In fact, there are two types of problems related to multiple skills. One is of a design nature, namely how many agents with certain skills are needed to obtain the desired service level for the skills. The other is an online control problem: to which agents to assign which calls. Both problems are extremely difficult, and only partial solutions and rough estimates exist in the scientific literature. We start by discussing online call routing, which is known as *skill-based routing*.

There are two types of skill-based routing: Static and dynamic. For both types of routing, each agent is member of one agent group. Groups are characterized by one or multiple skills that all agents in the group have (although it can occur that there are multiple groups with the same skills). Now if a call for a certain skill arrives, it is offered to one or more groups having this skill. The order in which this is done is determined by the skill-based routing. We call it the routing list.

Static routing means that the order in which calls are offered to groups is fixed and does not depend on current information, only on the call type. If all agents in all groups of the route are occupied, then the call is offered again to all groups in the same order, or, equivalently, to the first available agent within one of the selected groups. Of course the call center manager can change the routing at the beginning or even during the day; in practice we often see that the routing is changed only when groups are introduced or deleted.

No closed-form solutions exist for performance measures in static routing situations. Very good approximations exist if delayed customers abandon immediately. The delay case is still open. Also the scheduling problem is still open, although the approximation for the static case can be used as the basis of a local search algorithm. For certain standard situation it is observed in the literature that about 20% of cross-trained agents suffices to obtain most of the economies of scale.

Dynamic routing means that there is an online algorithm that determines how to route each call using current information such as availability of agents. A static algorithm is a special case of a dynamic algorithm; therefore dynamic algorithms have (in theory) a better performance. The dynamic routing algorithm depends on the numbers of busy agents in all groups. This state space description has therefore as many entries as there are skill groups. Thus the state space is, in general, high-dimensional. This makes that general methods to solve this type of dynamic decision problems cannot be used, due to the so-called *curse of dimensionality*. Approximation methods are currently being developed.

A type of multi-skill situation that desires separate attention is when different types of calls have different SL requirement. E.g., one might have B2B customer that require a faster answer than B2C calls; the same might hold for a group of premium customers, or sales calls require a faster answer than after-sales calls. In the case of a multi-skill operation one wants to pretect the SL of premium customers by reserving in some wat capacity for their calls. This can be done by placing cross-trained agents in the single-skill premium group. A more flexible way in which a better SL is obtained is by reserving a number of cross-trained agents to premium calls: if less than a certain number of cross-trained agents are available then regular calls are not assigned to multi-skill agents.

The numbers in Table 13.2 illustrated the gain obtained from cross-trained agents as compared to only having specialists, assuming that the call handling times were equal. However, often this is not the case: agents that receive multiple types of calls are often less efficient than agents that only receive a single type of call. This counterbalances the loss of economies of scale, certainly in the case of large call centers. Let us illustrate this with a numerical example.

Consider a call center with 2 skills. Specialists of either skill have $\beta = 5$, generalists have $\beta = 5.5$ (because they have to switch skill regularly). For $\lambda = 1$ for each skill we need or 15 generalists or 16 specialists, thus cross-training agents saves one agent. For $\lambda = 5$ these numbers are 62 and 60, and thus working only with specialists requires two agents less.

The implication of the numerical example is that, in big call centers, the advantages of skill-based routing are limited. However, this does not mean that call centers should only employ specialists: cross-trained agents increase the flexibility of the call center. This flexibility in task assignment (see Section 13.6) can be used in many different ways. Multi-skilled agents might be scheduled to work on different skills during the day, but sometimes it is used to deal with the seasonality of certain call types.

Example 13.8.1 An assurance company has a peak on travel insurance calls right after the summer, while at the same time there is a dip in call about housing assurances. This gives no problems because there is a group of agents that have both types of skills.

Cross-trained agents are also very useful in the case of unpredicted fluctuations. Peaks in the volume on one type of calls can be attenuated by decreasing the number of agents scheduled to work on other skills.

Example 13.8.2 Consider again two types of calls, with both $\lambda = 5$ and $\beta = 5$, and twice 30 agents. If they are all specialists, and $\lambda = 5.5$ for one of the skills, then the SL on that skill reduces to 54%. The average SL over both skills is 67%. Moving one or two agents from skill one to skill two (who therefore need to be cross-trained) moves both SLs to around 70%.

Evidently, the situation will become even more advantageous if peaks in one type of traffic are accompagnied by dips in other types.

13.9 Shift scheduling

Machines and other technical equipment is often constantly available for use within a company, with the exception of repair and maintenance (see Chapter **??**). This is not the case for the prime resource in most modern companies: people. This chapter deals with mathematical models for the (optimal) employment of personnel.

Standard shift scheduling We consider problems with T time intervals. For every time interval t a number s_t is given representing the minimum number of employees needed in interval t. The simplest shift scheduling problem has a constant shift length of K intervals without (planned) breaks. It can be solved by the following simple algorithm. Let x_t be the number of employees that start working at time t. To determine a schedule that uses the minimal number of employees one takes $x_1 = s_1$ and for $t = 2, \dots, T$ x_t such that there are at least s_t employees working:

$$x_t = \max\{0, s_t - (x_{t-K+1} + \cdots + x_{t-1})\},$$

where we assumed for convenience that $x_t = 0$ for $t \leq 0$.

For several reasons this situation hardly ever occurs. Usually we encounter more complicated shifts (with for example scheduled breaks) and variations in shifts (e.g., shifts with different lengths). We formulate a mathematical programming formulation of this problem. Let K now be the number of different possible shifts, and x_k the number of of people that are scheduled for shift type k. We define the constraint matrix A by $a_{tk} = 1$ if shift k works during interval t, 0 otherwise. Let c_k be the costs of shift k. Then an optimal schedule can be obtained from:

$$\min \left\{ \sum_{k=1}^{K} c_k x_k \;\middle|\; \begin{array}{ll} \sum_{k=1}^{K} a_{tk} x_k \geq s_t, & t = 1, \dots, T \\ x_k \in \mathbb{N}_0, & k = 1, \dots, K \end{array} \right\}. \tag{13.5}$$

Several solution methods exist for this type of integer programming problem. Note that the integer constraint is essential, without it non-integer solutions can be found (see Exercise 13.13).

This formulation leads to an optimal choice of shifts. This choice however is often infeasable in practice, due to limitations in the possible shifts. For example, an additional constraint could set the number of full-time shifts equal to the number of full-time employees. Many other types of constraints can be thought of.

Shift scheduling with a global constraint Previously we assumed that there is a separate constraint for each interval: in interval t at least s_t employee should be scheduled. In certain situations however the situation is different. Then the number of employees is allowed to be lower than s_t in certain intervals as long as this is compensated for in other intervals. This is for example the case in call center scheduling, if we take the daily average service level as objective. Then a low SL in certain intervals can be compensated for by high SLs in others. This allows for more flexibility when scheduling, leading to cost reductions.

Indeed, scheduling at least s_t employees in interval t gives an overall service level well above the minimum. This is the case for two reasons: in the first place the integral nature of determining s_t, in the second place the fact that the optimal solution of (13.5) might have $\sum_{s=t-K+1}^{t} x_s > s_t$ for certain t. This calls for an approach that considers the average daily service levels, but research on this type of solution is still in its infancy. Such an approach would integrate the second and third step of the decision process, namely determining the minimum levels s_t and determining the shifts.

To formalize this, let $L_t(s)$ be the service level at interval t if there are s employees working. Assume that $L_t(s)$ is increasing in s, this is for example the case if L is the percentage of customers waiting shorter than a seconds. We pose a restriction on the overall service level, which is defined as the weighted average of the interval service level. With weighting factor w_t for interval t, this gives:

$$\min\left\{\sum_{k=1}^{K} c_k x_k \;\middle|\; \begin{array}{l} \sum_{t=1}^{T} w_t L_t\Big(\sum_{k=1}^{K} a_{tk} x_k\Big) \geq \alpha \\ x_k \in \mathbb{N}_0,\ k = 1,\dots,K \end{array}\right\}. \tag{13.6}$$

Example 13.9.1 In call centers (see Chapter 13) we consider the service level of an *arbitrary* customer. Customer calls arrive in interval t with rate λ_t. With probability λ_s/λ (with $\lambda = \sum_{t=1}^{T} \lambda_i$) this call arrives in interval s, and thus $w_t = \lambda_t/\lambda$. The service level L_t is given by the Erlang formula or one of its generalizations. To avoid understaffing we probably want to add constraints of the form $\sum_{k=1}^{K} a_{tk} x_k > \lambda_t \beta$, with β the average service time.

Problem (13.6) is non-linear, as L_t is in general non-linear. (In the case of tail probabilities, L is concave.) This calls for solution methods that are based on other techniques than branching to non-integer values. Such a technique is local search, where shifts are added, deleted and shifted until a local minimum is found. Numerical results show that this can lead to significant cost reductions.

Another option is adding additional variables to make (13.6) linear. Introduce the variables n_{ts}: $n_{ts} = 1$ if during interval t exactly s employees are scheduled, 0 if this is not the case. Thus, for every t, exactly one n_{ts} should be one. This is obtained by requiring $n_{ts} \in \{0, 1\}$ and $\sum_{s=0}^{S} n_{ts} = 1$ for every t, with S the maximum number of employees that can be scheduled.

Problem (13.6) is now equivalent to:

$$\min\left\{\sum_{k=1}^{K} c_k x_k \;\middle|\; \begin{array}{ll} \sum_{k=1}^{K} a_{tk} x_k = \sum_{s=0}^{S} n_{ts} s, & t = 1,\dots,T \\ \sum_{t=1}^{T} w_t \sum_{s=0}^{S} n_{ts} L_t(s) \geq \alpha & \\ \sum_{s=0}^{S} n_{ts} = 1, & t = 1,\dots,T \\ x_k \in \mathbb{N}_0, & k = 1,\dots,K \\ n_{ts} \in \{0,1\}, & t = 1,\dots,T,\ s = 1,\dots,S \end{array}\right\}. \tag{13.7}$$

Note that this problem is linear in all its variables.

Assigning shifts to agents Making shifts is not the end of HR planning: shifts have to be assigned to people. The first requirement to make this possible is that the right numbers of the right shifts have been generated. In the total pool of employees there are often different types of contracts, which differ for example in the number of working hours per day. The shifts should be generated in the right numbers. This can be ontained by adding constraints of the form $\sum_{k \in \mathcal{K}} x_k = N_{\mathcal{K}}$ with $\mathcal{K} \subset \{1, \dots, K\}$ a set of shifts and $N_{\mathcal{K}}$ the number of shifts of this type that should be generated. $\mathcal{K}$ could for example be the set of 4 or 6-hour shifts start between 8.00 and 10.00.

Thus, for a single day shifts can be assigned to agents taking all constraints into consideration. However, the shift assigned to a particular agent on one day might influence the possible assignments on another day. For example, an agent might work 4 days per week, on days to be determined by the call center. Or an agent works 32 hours a week, assigned in a flexible way to the days. To deal with this scheduling sometimes needs to done at the agent level for a whole week at once. Work during weekends sometimes even leads to planning periods that are longer than a week. For example, in a certain industry sector in Holland there used to be a regulation that employees should have at least 4 free Sundays in every block of 13 weeks. To utilize fully the possibility of scheduling employees on Sundays the planning period should be 13 weeks in this case.

In practice we encounter situations where shift scheduling and shift assignment are completely separated, and situations where they are completely integrated. The former is computationally less demanding but gives suboptimal or even unfeasable solutions in many cases; the latter is computationally much more demanding, certainly if the planning period is long and if the intervals are short, i.e., T is big.

So far we described a way for an organization to plan its workforce given all constraints. A completely different approach, with its own advantages and disadvantages, is to have the employees choose their own shifts. Then the mathematical analysis stops after shift generation, and an especially designed (web-based) user interface allows the employees to choose their own shifts.

13.10 Long-term planning

Over a longer time period companies have to deal with changing need for employees and with *attrition*, the fact that people leave the company. The *turnover*, the ratio of new hires to the number of employees, can be well over 100% in for example call centers.

The challenge is the fact that offered load and turnover is unpredictable, and that hiring and training of new employees costs time. This requires stochastic models to determine the right moment to start hiring new employees. For costs reasons new employees are trained in groups.

13.11 Further reading

By now a considerable number of scientific papers on call centers exists. Overview of this scientific literature on call centers are Gans et al. [38] and Zeynep et al. [5]. All issues discussed in this chapter are discussed there in more detail, and all relevant references are given there. For this reason we only give a few general and/or easily accessible references. For recent work on multi-skill call centers, see Koole & Pot [62].

An elaborate text on the mathematics of call centers, that is also suitable for managers without a mathematical background, is Koole [61]. A general text on call center management from a managerial point of view is Cleveland & Mayben [23]. An introduction to general aspects of call centers is Dawson [30]. Brown et al. [18] contains a statistical analysis of data from one particular call center. All aspects of workforce management are discussed in Reynolds [73]. Fukunaga et al. [34] presents an overview of the scheduling modules of one of the major wfm systems. A more mathematically oriented text book that include the basic models but also some advanced skill-based routing models is Stolletz [88]. The underlying ideas of Section 13.6 hold for services in general, see Sasser [79].

Predictive dialers, in use in outbound call centers, are discussed in Samuelson [78]. The formula for quantiles of hypoexponential distributions comes from Ross [75, Section 5.2.4], the derivation resembles that of Koole [60].

Call center literature on shift scheduling, Ryan (INFORMS talk) on crew scheduling, Ortec tools.

13.12 Exercises

Exercise 13.1 An important aspect of call centers are abandonments by people who do not want to wait any longer in queue. What do you think that is the influence of abandonments on the service level of the customers? And on the productivity?

Exercise 13.2 Consider a call center with on average 1.5 arrivals per minute, an average service time of 5 minutes, and 10 agents. The Erlang C model is used to compute the performance of this call center.
a. Compute the expected waiting time.
b. Give examples of an increase in scope and an increase in scale in the context of call centers.
Suppose the call center doubles in arrival rate and in number of agents.
c. Compute the expected waiting time.
d. How many agents do you need to make the average waiting time less than 90 seconds?

Exercise 13.3 Consider a call center where every agent that becomes available is assigned to an inbound call if one is present, otherwise to an outbound call. In other words: agents are never idle, inbound calls have priority over outbound calls of which there is an infinite supply. Assume that inbound and outbound calls have the same exponential service times.
a. Construct a birth-death process with as state the total number of calls in the systems (i.e., outbound calls in process and inbound calls). Give the transition rates.
b. Determine the stationary distribution.
c. Give a formula for the waiting time distribution for inbound calls.
d. Give a formula for the fraction of time that the agent is busy with outbound calls, and the average number of finished outbound calls per unit of time.
e. Compute these numbers for $s = 12$, $\beta = 3$ and $\lambda = 2$, 3, and 3.75.

Exercise 13.4 An Erlang calculator can calculate the service level, defined as the percentage of callers that waits longer than t seconds, based on the arrival rate λ, the average service time β, and the number of servers s.

In a call center there are on average 10 calls per minute, that require each on average 3 minutes to answer. The acceptable waiting time is 20 seconds, and the time between the moment a call is assigned to an agent and the moment it is answered by the agent is around 3 seconds.
a. Give the parameter values for the Erlang calculator by which you can calculate the service level in the call center.
b. To obtain a service level of around 80% 35 agents are needed. Define productivity and calculate it.
c. A model is not an exact description of reality. Give 3 aspects in which the Erlang system does not model the call center exactly.
d. The arrival rate doubles to 20. The manager decides to double the number of agents. What do you expect to be the consequences for the costs and the service level? Motivate your answer!
e. Estimate without using the Erlang formula how many agents need to be scheduled to obtain a 80% service level. Motivate your answer.

Exercise 13.5 Consider a call center with the following parameters: $\lambda = 5$ per minute, $\beta = 3$ minutes. An acceptable waiting time is 18 seconds. Answer the following using the Erlang C calculator:
- What is the service level with 20 agents?
- How many agents agents are needed for a 95% service level?
- How many are needed if λ doubles?
- And what if the acceptable waiting time is only 9 seconds?
After each call the agents need on average 1 minute to enter data related to the call and they take on average 5 seconds to take up the phone. Answer the same questions as above for this new situation using the Erlang calculator. Note that the time that an agent uses to take up the phone is both waiting and service time!

Exercise 13.6 Consider a call center with the following parameters: $\lambda = 5$ per minute, $\beta = 3$ minutes, 18 agents. An acceptable waiting time is 20 seconds. Answer the following using the Erlang X calculator:
a. Without blocking or abandonments, what is the SL?
Customers abandon, on average after 2 minutes.
b. Without blocking, what is the SL?
The number of waiting places is limited to 8.
c. What is now the SL?
If blocking plus abandonment must be less than 5%, and the SL as high as possible, how many lines would you choose?

Exercise 13.7 Agents are absent (e.g., because they are ill) with probability 0.05. Assume that $\lambda = 10$ and $\beta = 3$.
a. How many agents do you need to have behind the telephone to assure a 80-20 service level (i.e., 80% answered within 20 s)?

b. For s agents, what is the probability that more than k of them are ill?
c. How many agents should you schedule, not knowing who will be ill, to have 95% probability that enough will show up to meet the service level?
d. How many agents with flexible contracts and with fixed contracts would you schedule? Explain. (Note that flexible contracts are somewhat more expensive that fixed contracts.)

Exercise 13.8 Consider a call center with on average 2.5 arrivals per minute, an acceptable average waiting time of 1 minute, and an average service time of 6 minutes. The Erlang C model is used to compute the number of agents.
a. Compute this number using the table.
It is observed that the service time consists of 2 minutes talk time and 4 minutes so-called wrap-up time. Two minutes of this wrap-up time needs to follow the call; the remaining 2 minutes can be done at another time, by another agent. Because of this two agent groups are created: one that handles calls (average service time 4 minutes) and one that does only the second half of the wrap-up (average service time 2 minutes).
b. Compute the number of agents needed in the first group to obtain an average waiting time of less than 1 minute.
c. How many agents are needed in the second group to have a 100% productivity?
Agents like to finish a call completely if possible. It is decided to implement this in the following way. All agents are in 1 group, and they handle a call entirely if the there are few calls waiting. When there are many calls waiting then only the first part of each call is done. The remaining second parts are distributed among free agents later on when the load is lower.
d. How many agents do you expect to need under this new situation? Motivate your answer!

Exercise 13.9 Consider a call center with 2 types of calls, A and B. Arrival rates are λ_A and λ_B, average service times β_A and β_B. There are specialists of both skills and generalists available.
a. Take $\lambda_A = \lambda_B = 10$, $\beta_A = \beta_B = 3$. There are 20 generalists. How many specialists with skills 1 and 2 would you schedule to minimize costs under a 80-20 SL? Motivate your answer. (An exact calculation is not necessary, a motivated estimation is sufficient.)
b. The same question for $\beta_A = 2$ and $\beta_B = 3$.

Exercise 13.10 A call center has 12 agents, calls arrive at a rate of on average 15 per minute, and the average call duration is 25 seconds.
a. Calculate the expected waiting time using the Erlang calculator.
It was found that this does not match with reality. Further research showed that it takes on average 5 seconds before an agent pick up the phone after a call is assigned to an agent.
b. Calculate again the expected waiting time of an arbitrary call.
Again, there is a considerable difference between reality and the prediction of the model. Further research showed that agents take short breaks, totaling on average 5 minutes per hour.
c. Give an approximation for the expected waiting time for this new situation.

Exercise 13.11 Prove Equations (13.3) and (13.4).

Exercise 13.12 A company is changing its call centers operations. Instead of a regional approach (there are currently 3 regional call centers), they decide to build a single call center with skill-based routing (i.e., different groups of agents handling different types of calls). A model for the new call center is needed to determine the consequences for the workforce. Based on the model outcomes decisions are taken with regard to the possible employment of new agents and with regard to the training of agents for specific skills.
The average call duration β is 2 minutes, independent of call center and call type. In the next table the number of agents in the regional call centers and the expected waiting times during peak hours are given. The arrival rates are not known.

Call center	1	2	3
Number of agents	12	9	7
Expected waiting time (seconds)	27	78	5.5

a. Determine the arrival rates, using the Erlang calculator.
There will be 2 call types, approximately 1 out of three calls is type 1. The average waiting time for both types should not exceed the current overall waiting time.
b. Determine the current average waiting time.
c. Determine the right number of agents for the two types.
d. How many agents would be needed in the case of only one single group?

Exercise 13.13 Consider a simple shift scheduling problem with $T = K = 3$, and $a_{tk} = 0$ if $t = k$, 1 otherwise, and $c_k = s_t = 1$ for all k and t.
a. Find the optimal schedule by solving the problem of Equation (13.5).
b. Do this again but without the integer constraint of Equation (13.5).

Exercise 13.14 A call center has inbound calls and emails. Every interval, agents either work on inbound calls or on emails. Emails should be answered within 24 hours. Suppose there are, over the whole day, u agent hours of work on emails to be done.
a. Extent the problem of Equation (13.5) to allow for emails that can be done by all agents.
b. Same as a, but assume that only agents doing shifts $1, \dots, K'$, with $K' < K$, can do emails.
Make sure that the constraints remain linear.

Exercise 13.15 Consider a call center that is to be operated during 5 periods. There are three possible shifts, each working 3 consecutive intervals. The arrival rates are 5, 8, 10, 7, and 3 per minute. The average service time is 3 minutes. As standard 80-20 SL is chosen, performance is estimated using the Erlang C.
a. Compute a schedule that minimizes the number of agents if the SL has to be met every interval.
b. Compute a schedule that minimizes the number of agents if the SL has to be met on average over the whole day.

Exercise 13.16 A call center needs 100 employees. The turnover is 120%, equally spread out over the year, otherwise unpredictable. Employees leave on a very short notice.
a. Formulate a model for the attrition.
b. Hiring new employees and training them takes 2 months. At which moment should the call center start hiring to make sure that the probability that there are less than 95 agents is less than 5% the moment new agents become available?

Chapter 14

Revenue Management

"Revenue management is the art and science of predicting real-time customer demand at the micromarket level and optimizing the price and availability of products" [27, p. 4].

Revenue or yield management deals with maximizing income when the number of products is fixed in advance. Its main application domain is aviation: airline companies maximize their income by selling similar seats to different types of customers for different prices. In this chapter we give an introduction to revenue management and the models that can be used.

In aviation a standard figure shown in business reports is the load factor, i.e., the number of occupied seats divided by the total number of seats "flown". However, a high load factor does not mean a high income, or v.v. The price at which the seats are sold is equally important. An interesting factor in aviation enabling revenue management is the fact that the same seat can be sold at different prices by changing the sales conditions. For example, low-priced tickets have to be bought long in advance, the traveller should stay a weekend at the destination, and the ticket cannot be changed. High-priced tickets have no such conditions. The objective of revenue management is to maximize revenue or yield by deciding regularly if or how many seats are available in each category.

Revenue management can also be applied to other business areas, such as hotels.

14.1 Pricing

Consider a certain product. Total production costs C are a function of the number of produced items n. There are fixed costs $C(0)$, variable costs $C(n)-C(0)$, marginal costs $C'(n)$. When the total costs are linear then $C'(n) = (C(n)-C(0))/n$: the marginal costs equal the average variable costs.

Example 14.1.1 Consider the rooms in a hotel for a particular night. n is the number of occupied rooms. Then it is not unreasonable to take $C(n)$ linear. $C(0)$ consists of the investment in the building, and staff that has to be there no matter how many customers there are. The marginal costs consist for example of cleaning costs. Actually, the marginal costs might be negative, because visitors, even though they pay for the room, generate additional income, for example by dining in the hotel's restaurant.

The (expected) demand D is a decreasing function of the selling price p. D is also called the *demand curve.* The profit W under a price p can be calculated as follows:

$$\text{profit} = W(p) = \text{revenue} - \text{total costs} = pD(p) - C(D(p)).$$

We define $E(p) = pD'(p)/D(p)$, the *price elasticity*, the relative difference in demand under relative changes in price. We can safely assume that $E(p) < 0$: when the price is higher, less items are sold. When $E(p) \in (-1, 0)$, then an increase in price will lead to a relatively small decrease in demand, thereby increasing total revenue. The revenue is maximized if $E(p) = -1$: when the price increases by 1%, then the demand decreases by 1%, keeping the total revenue equal. The condition $E(p) = -1$ can also be obtained by differentiating the total revenue $pD(p)$ to p.

The difficulty in applying the above is that it is often very hard to estimate the form of the demand curve. In few situations it is possible to experiment with different prices in such a way that a reliable estimate of D can be obtained. The demand curve also changes over time, and this can happen quickly if for example the competition changes its prices.

Example 14.1.2 For a parking lot it is possible to make advance internet reservations. By changing the price from week to week it is possible to estimate the demand curve and to determine the price that maximizes the revenue. The variable costs are almost 0, thus maximizing revenue equals maximizing profit.

When the variable costs are low, then maximizing revenue is (almost) equal to maximizing profit. When the costs are non-negligable but linear, then $E(p) = -1$ should be replaced by $(p - c)D'(p)/D(p) = -1$, with c the variable costs.

Market segmentation Suppose that a product is sold to different groups of customers, for example through different channels. Then, instead of selling the product for a single price, we could differentiate in price for the different segments of the market. Now for every segment the price should be maximized, leading to a higher total expected revenue than for a single price. A major concern is *diversion*, when customers from one segment, willing to pay the price for that segment, buy the product for the price targeted at another market segment.

Example 14.1.3 Consider again the parking lot of Example 14.1.2. There are internet reservations and customer who drive up without reservation. The internet price is lower than the drive-up rate. The numbers of parked cars are carefully monitored to make sure that the number of drive-ups does not decrease.

Up to now we considered pricing in the context of a potentially infinite supply of products. When the number of products or underlying resources is limited, below the demand for the revenue maximizing price, and consumption cannot be delayed, then *revenue management* comes into play.

14.2 Conditions for revenue management

Suppose you have a product or a range of products that all depend on a single resource, and the number of items that can be produced is limited to C. We assume that the items of the product cannot be produced in advance. Many tangible products can be produced in advance (called make-to-stock, see Section 10.2), but services cannot. We neither assume that production can be delayed, as it is the case with for example seeing a doctor (an appointment at a later time is made). Good examples for which the time of consumption is crucial are tickets (of different fare classes) for a flight. Then the resource is a particular flight with C seats at a particular day, which is shared by all passengers having booked for this flight. Other examples of resources where delayed consumption is not possible are hotel rooms or telecommunication lines.

For these examples the fixed costs are high compared to the variable costs: the airplane will fly anyway, the investments in the hotel and most of the personnel costs are fixed, investment in hardware (switches, laying cables) form the bigger part of the costs for a telecommunication network operator. The high fixed costs are related to the fact that capacity is limited: if the fixed costs would be low, then additional capacity would be cheap and capacity would not be a constraint anymore.

Evidently, every company wants to sell its valuable production capacity as good as possible. In the examples there is also the possibility of different product classes or market segmentation. Using the theory developed in the previous section we start by asking ourselves: is there sufficient capacity if one tries to maximize the revenue per class? If the total (maximal) demand over all classes under this policy is lower than the capacity, then these optimal prices should be used. If the capacity might not be sufficient (because the total expected demand over all classes exceeds the capacity, or because due to random fluctuations in demand this might occur regularly), then revenue management is a method to maximize income.

One can do revenue management in two ways:
- either one changes the price of the products, of which there is often only one;
- or the prices of the different products are kept fixed, but bying cheap products is made impossible under certain circumstances as to protect capacity for customers willing to pay higher prices.

Both situations can be treated in the same way, by interpreting a product with different prices as different products. However, one should realize that in this case customers make only reservations for the lowest price that is available, although they are perhaps willing to pay more. This also plays a role when the products differ not only in price, although to a lesser extend. In this situation it is called ‘diversion’.

Example 14.2.1 A low-cost carrier such as Easyjet has a single product on each flight. The price that is offered for this product is varied depending on many factors such the time until the flight and the number of reservations made already. Traditional carriers have different types of tickets with different conditions. Although most people choose for the ticket with the lowest price, there are still people willing to pay higher prices, for example because it is refundable.

Concluding, in revenue management we study problems that have the following characteristics:
- the product is perishable, i.e., there is a date before which and a date after which it cannot be consumed;
- the capacity is limited and variable costs are relatively low;
- the market can be segmented, i.e., customers pay different prices for essentially the same product, e.g., because of different additional characteristics of the product, or because the price changes over time and channel.

Many systems to which revenue management is applied have also the following characteristics:
- demand is random;
- customers that are willing to pay a high price arrive late.

The combination of the last two characteristics complicates effective revenue management considerably. To avoid late customers to buy cheap tickets these should not be available anymore by the time late customers arrive. Thus, in the simple situation with only two types of customers, we have to decide how many seats to reserve for the high-price customers. When doing so we have the risk to have less revenue as possible for the following two reasons:
- the demand of expensive products is higher than expected, and income is spilled because of already accepted low-price customers;
- the demand of expensive products is lower than expected, and income is missed because not all products are sold.

How to calculate for a simple model the optimal number of products to be reserved for high-paying customers is the subject of the next section. Also robustness and possible measures to increase revenue are discussed.

Reality is often more complicated than indicated above. Some important issues are as follows.

Demand diversion Diversion is an important issue. How many of our low-paying customer were willing to pay a higher fare? One usually tries to avoid diversion, but sometimes it can be advantageous. Many airlines indicate prices for multiple days such that potential customer can trade off day and price.

Multiple items In many practical situations a customer wants more than one item at a time. In aviation, a customer flies from an origin to a destination possibly using more than one *flight leg*. In hotels people stay often more than one night. This dependence between different resources makes modeling more complicated.

Cancellations In many situations capacity that is reserved can later be cancelled. To anticipate this more products are reserved than there is capacity. This is called *overbooking*. Forecasting cancellations is equally important as forecasting actual demand.

Overselling Overselling or shortselling occurs if there was enough capacity reserved for the customers who pay the most. They are not refused, but instead customers paying a low price are given a financial incentive to take an alternative products, such as a later flight. The financial incentive is lower than the difference between the two prices, making it financially advantageous.

14.3 Single-resource models

Let us start with a model with two classes, no diversion. Again, the discrete capacity is equal to C. The stochastic demand of class i is denoted by D_i, its revenue per sold item y_i, with $y_1 > y_2$. Type 2 demand occurs before type 1. The question to answer is how much capacity should be reserved for type 1? The demand distribution for type 2 is irrelevant, if the demand is small then there is no issue whatsoever. Thus we assume for the moment that $D_2 = C$. Our objective is to maximize expected revenue. Assume that the seats are numbered and we start selling them to type 2. For every next seat that we calculate the marginal expected revenue of selling the seat to type 2 or reserving it together will all remaining seats for type 1. Assume there are s seats remaining, thus we consider seat $C-s+1$. If we sell it to type 2, then the revenue is y_2. With probability $\mathbb{P}(D_1 \geq s)$ we can sell this seat (and all higher numbered seats) to type 1 with revenue y_1. Thus the maximal optimal number of seats reserved for type 1, s_1, is the biggest number s for which:

$$y_2 \leq y_1\mathbb{P}(D_1 \geq s),$$

thus

$$s_1 = \max_{0\leq s\leq C}\Big\{y_2 \leq y_1\mathbb{P}(D_1 \geq s)\Big\} = \min_{0\leq s\leq C}\Big\{F_{D_1}(s) \geq 1-\frac{y_2}{y_1}\Big\} = F_{D_1}^{-1}\Big(1-\frac{y_2}{y_1}\Big), \qquad (14.1)$$

where F^{-1} is the quantile function as defined in Equation (1.1). The number s_1 is called the *reservation* or *protection level* for type 1. The total revenue is given by

$$y_1\mathbb{E}\min\{D_1, \max\{C-D_2, s_1\}\} + y_2\mathbb{E}\min\{D_2, C-s_1\}.$$

Example 14.3.1 An airport parking lot has two classes of customers: those that drive up without reservation and those that make advance reservations over the internet. Let us simplify the problem and look at a single day (perhaps the most crowded day of the week). The discount internet rate is \$10, the drive-up rate \$15. Capacity is 200, the forecasted number of drive-ups is 150. Demand is Poisson distributed. How many internet reservations should we allow? Using a normal approximation of the demand we easily find using the inverse of the normal distribution (for example, using Excel) that we should reserve 145 places for drive-ups. Note that this is less than the expected number of drive-ups!

In reality type-2 customers do not all make their reservation before type-1 customers make theirs. To avoid diversion the possibility to book type 2 should be cancelled from a certain moment on, even if the booking limit has not yet been reached.

Overselling Consider the same model, but assume that type 2 customers, that already have ordered the product, are willing to change product (e.g., take a later plane) for an indemnity $z_2 < y_1 - y_2$. Then type 1 customers will never be refused, and the reservation level s_1' is the biggest number s that satisfies:

$$y_2 + (y_1 - y_2 - z_2)\mathbb{P}(D_1 \geq s) \leq y_1 \mathbb{P}(D_1 \geq s),$$

which is equivalent to

$$y_2 \leq (y_2 + z_2)\mathbb{P}(D_1 \geq s).$$

It is readily seen that $s_1' \leq s_1$. Note also that the expected marginal revenue per seat is higher, thus the expected total revenue is higher. Thus overselling is profitable! The total revenue is given by

$$y_1 \mathbb{E}\min\{D_1, C\} + (y_1 - z)\mathbb{E}\max\{0, \min\{D_1, C\} + \min\{D_2, C - s_1'\} - C\} + y_2 \mathbb{E}\min\{D_2, C - s_1'\}.$$

Multiple classes An extension of this model (without overselling) to multiple classes is the EMSR algorithm, which stands for *Expected Marginal Seat Revenue*. Assume there are n classes, with revenue $y_1 > \cdots > y_n$ and demand $D_1, \ldots, D_n$. The idea of booking limits is extended to multiple classes, as follows. Consider bookings of type $j+1$. How much of the remaining capacity should we reserve for type $1, \ldots, j$? For every type $i \in \{1, \ldots, j\}$ we compute a booking limit as follows:

$$u_i^{j+1} = F_{D_i}^{-1}(1 - \frac{y_{j+1}}{y_i}).$$

The total number of seats that have to reserved for types 1 upto j when considering selling type $j+1$, written as s_j, is now given by

$$s_j = \sum_{i=1}^{j} u_i^{j+1}.$$

The disadvantage of this method is that the booking limits are simply added, and we would expect (from the central limit theorem, see Section 1.7) that when we add demand over multiple classes that we need relatively speaking less seats to deal with the variation in demand. To account for this, a second version of EMSR is designed, called EMSR-b (the old one becoming EMSR-a). For EMSR-b we add demand over the j types and we calculate the weighted average price $\hat{y}_j$ over classes $1, \ldots, j$:

$$\hat{y}_j = \frac{\sum_{i=1}^{j} y_i \mathbb{E} D_i}{\sum_{i=1}^{j} \mathbb{E} D_i}.$$

Now we take

$$s_j = F_{\hat{D}_j}^{-1}(1 - \frac{y_{j+1}}{\hat{y}_i})$$

with $\hat{D}_j = \sum_{i=1}^{j} D_i$.

Example 14.3.2 Consider a revenue management instance with three classes, all with normally distributed demand of average 40. The prices are $y_1 = 100$, $y_2 = 80$, $y_3 = 70$. Let us first apply EMSR-a. Computations using Excel show that when selling type-3 seats we should reserve 37 seats for type 1 and 33 for type 2, thus 70 in total. When selling type 2 we should reserve only 35 for type 1. Under EMSR-b, when selling type-3 seats, in total 73 seats should be reserved. Now let the total capacity be 80. Using a Monte Carlo simulation Excel plug-in we find an estimation for the revenue under EMSR-a of 8313 and for EMSR-b of 8321.

14.4 Multi-resource models

Here we consider the situation where customers require services consisting of multiple products, such as multiple flight legs for an origin-destination pair or multiple night of a hotel room. A way to solve this is by using *bid prices*. A bid price is a way to implement the booking limits of the previous section. A bid price is the level above which booking are accepted. Thus if the booking limit for a class is reached then the bid price is increased above its price. A customer requiring multiple products has a bid price that is the sum of the bid prices of the products.

14.5 Further reading

The book Talluri & van Ryzin [91] is currently the main reference for revenue management. The EMSR algorithm, forecasting and many other relevant details are discussed at lenght. Cross [27] is a non-mathematical book about the impact of revenue management.

The two-class model is introduced by Littlewood (see Brumelle et al. [19]).

14.6 Exercises

Exercise 14.1 Consider a product with constant but positive marginal costs c.
a. Give an expression for $W(p)$ for this case.
b. Derive the optimal value for the price elasticity in terms of p and c.
c. Give an intuitive explanation for the result you found.

Exercise 14.2 A railway dedicated to freight has two types of trains: bulk trains (type 1, for example coal) and container trains (type 2). The demand for each hour is Poisson, both with average 3. The variable costs are negligable, the revenue per type is 150 and 200 Euro, respectively. Reservations for bulk trains arrive before container trains.
a. The capacity of the line is 12 per hour. What is the maximal expected revenue, and how can it be obtained? b. Due to safety regulations in tunnels the capacity is limited to 4. What is the maximal expected revenue, and how can it be obtained?
c. The same questions, but now under the assumption that the demand is deterministic with the same expectation.

Exercise 14.3 A small airplane has a capacity of 10 seats. There are 2 fare classes. The price of a type 1 ticket is E 200, of a type 2 ticket E 500. Type 1 customers book before type 2 customers. The demand for type 1 tickets is 10. The demand for type 2 tickets is distributed as follows:

demand	0	1	2	3	4	5	6	7	8	9	10
probability	0	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$	$\frac{1}{10}$

a. Without overselling, how many seats should you sell to type 1 customers to maximize expected revenue?
b. What is the expected total revenue?
A third type of customers (last minute) is introduced with demand 10, price E 100 per ticket, that book after type 2 customers.
c. What is the expected total revenue when you reserve as many seats as determined under a) for type 2?
d. In this new situation, calculate how many seats should be sold to type 1 customers to maximize expected revenue.

Exercise 14.4 Consider Example 14.3.1.
a. Use an Excel Monte Carlo plug-in to simulate this model for various values of s_1, and make a plot with s_1 on the horizontal axis and the total revenue on the vertical axis.
b. Management wants to avoid sending drive-up customers away. What would you advice as a value for s_1?

Exercise 14.5 Make an Excel sheet in which EMSR-a and EMSR-b implemented for 10 booking classes and normally distributed demand. Try different numbers and try to find an instance for which the difference between both models is maximal.

Bibliography

[1] Stafford hospital: what went wrong. Times Online, March 17, 2009. http://www.timesonline.co.uk/tol/news/uk/health/article5925945.ece. 7.3.2

[2] R.L. Ackoff. The future of Operational Research is past. *Journal of the Operational Research Society*, 30:93–104, 1979. 7.11

[3] H.A. Akkermans. Participative business modelling to support strategic decision making in operations — a case study. *International Journal of Operations & Production Management*, 13(10):34–48, 1993. 7.11

[4] H.A. Akkermans. *Modelling with Managers*. PhD thesis, Technical University of Eindhoven, 1995. 7.11

[5] O.Z. Akşin, M. Armony, and V. Mehrotra. The modern call-center: A multi-disciplinary perspective on operations management research. Working paper, 2007. 13.11

[6] B. Andrews and S.M. Cunningham. L. L. Bean improves call-center forecasting. *Interfaces*, 25(6):1–13, 1995. 2.7

[7] H.I. Ansoff and E.J. McDonnell. *Implanting Strategic Management*. Prentice Hall, 2nd edition, 1990. 8.2

[8] J.M. Anthonisse, J.K. Lenstra, and M.W.P. Savelsbergh. Behind the screen: DSS from an OR point of view. *Decision Support Systems*, 4:413–419, 1988. 8.9, 9.6

[9] R.N. Anthony. *Planning and Control Systems: A Framework for Analysis*. Harvard University Press, 1965. 7.3, 7.11

[10] S. Asmussen. *Applied Probability and Queues*. Springer, 2nd edition, 2003. 3.5, 3.9

[11] T. Aven and U. Jensen. *Stochastic Models in Reliability*. Springer, 1998. 1.10, 11.9

[12] A.O. Awani. *Project Management Techniques*. Petrocelli Books, 1983. 11.9

[13] M.O. Ball, T.L. Magnanti, C.L. Monma, and G.L. Nemhauser, editors. *Handbooks in Operations Research and Management Science, Vol. 8: Network Routing*. North-Holland, 1995. 11.9

[14] R.E. Barlow and F. Proschan. *Statistical Theory of Reliability and Life Testing*. Holt, Rinehart and Winston, 1975. 1.10, 11.9

[15] D. Belson. Managing a patient flow improvement project. In R.W. Hall, editor, *Patient Flow: Reducing Delay in Healthcare Delivery*, pages 429–452. Springer, 2006. 7.11, 11.9, 12.9

[16] J. Bramel and D. Simchi-Levi. *The Logic of Logistics*. Springer, 1997. 6.6, 11.9

[17] M.L. Brandeau, F. Sainfort, and W.P. Pierskalla, editors. *Operations Research and Health Care*. Kluwer, 2004. 12.9

[18] L. Brown, N. Gans, A. Mandelbaum, A. Sakov, H. Shen, S. Zeltyn, and L. Zhao. Statistical analysis of a telephone call center: A queueing-science perspective. *Journal of the American Statistical Association*, 100:36–50, 2005. 13.11

[19] S.L. Brumelle, J.I. McGill, T.H. Oum, K. Sawati, and M.W. Tretheway. Allocation of airline seats between stochastically dependent demands. *Transportation Science*, 24:183–192, 1990. 14.5

[20] J.A. Buzacott and J.G. Shanthikumar. *Stochastic Models of Manufacturing Systems*. Prentice-Hall, 1993. 8.9

[21] N. Christofides. Vehicle routing. In E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, and D.B. Shmoys, editors, *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, pages 431–448. Wiley, 1985. 8.4.2

[22] E. Çinlar. *Introduction to Stochastic Processes*. Prentice-Hall, 1975. 3.9

[23] B. Cleveland and J. Mayben. *Call Center Management on Fast Forward*. Call Center Press, 1997. 13.11

[24] E.G. Coffman, Jr., J.K. Lenstra, and A.H.G. Rinnooy Kan, editors. *Handbooks in Operations Research and Management Science, Vol. 3: Computing*. North-Holland, 1992. 5.7, 8.9, 9.6

[25] J.W. Cohen. *The Single Server Queue*. North-Holland, 2nd edition, 1982. 5.7

[26] R.B. Cooper. *Introduction to Queueing Theory*. North Holland, 2nd edition, 1981. 5.7

[27] R.G. Cross. *Revenue Management: Hard-Core Tactics for Market Domination*. Broadway Books, 1998. 14, 14.5

[28] Y. Dallery and S.B. Gershwin. Manufacturing flow line systems: a review of models and analytical results. *Queueing Systems*, 12:3–94, 1992. 11.9

[29] T.H. Davenport and J.G. Harris. *Competing on Analytics: The New Science of Winning*. Harvard Business School, 2007. 7.5, 7.11

[30] K. Dawson. *Call Center Handbook: The Complete Guide to Starting, Running and Improving Your Call Center*. CMP Books, 3rd edition, 1999. 13.11

[31] J. de Mast, R.J.M.M.Does, and H. de Koning. *Lean Six Sigma - for Service and Healthcare, publisher=*. 2006. 7.11

[32] F.X. Diebold. *Elements of Forecasting*. Thomson, 4rd edition, 2007. 2.7

[33] M. El-Taha and S. Stidham, Jr. *Sample-Path Analysis of Queueing Systems*. Kluwer, 1998. 3.9

[34] A. Fukunaga, E. Hamilton, J. Fama, D. Andre, O. Matan, and I. Nourbakhsh. Staff scheduling for inbound call centers and customer contact centers. *AI Magazine*, 23(4):30–40, 2002. 13.11

[35] J. Galbraith. *Designing Complex Organizations*. Addison-Wesley, 1973. 7.11, 8.2

[36] S. Gallivan. Challenging the role of calibration, validation and sensitivity analysis in relation to models of health care processes. *Health Care Management Science*, 11:208–213, 2008. 7.11

[37] S. Gallivan, M. Utley, T. Treasure, and O. Valencia. Booked inpatient admissions and hospital capacity: Mathematical modelling study. *British Medical Journal*, 324:280–282, 2002. 12.9

[38] N. Gans, G.M. Koole, and A. Mandelbaum. Telephone call centers: Tutorial, review, and research prospects. *Manufacturing & Service Operations Management*, 5:79–141, 2003. 13.11

[39] M.R. Garey and D.S. Johnson. *Computers and Intractability: a Guide to the Theory of NP-Completeness*. Freeman, 1984. 8.9

[40] S.B. Gershwin. *Manufacturing Systems Engineering*. Prentice-Hall, 1993 or 1994. 11.9

[41] E.M. Goldratt. *Critical Chain*. North River Press, 1997. Dutch translation: *De Zwakste Schakel*, Het Spectrum, 1999. 11.9

[42] E.M. Goldratt and J. Cox. *The Goal*. Gower, 1990. Dutch translation: *Het Doel*, Het Spectrum, 1999. 7.11, 11.9

[43] S.C. Graves, A.H.G. Rinnooy Kan, and P. Zipkin, editors. *Handbooks in Operations Research and Management Science, Vol. 4: Logistics of Production and Inventory*. North-Holland, 1993. 5.6.4, 6.6, 8.9, 11.9

[44] L.V. Green. Using Operations Research to reduce delays for healthcare. In Zhi-Long Chen and S. Raghavan, editors, *Tutorials in Operations Research*, pages 1–16. INFORMS, 2008. 12.9

[45] D. Gross and C.M. Harris. *Fundamentals of Queueing Theory*. Wiley, 2nd edition, 1985. 5.7

[46] A.C. Hax and D. Candea. *Production and Inventory Management*. Prentice-Hall, 1984. 6.6, 7.11, 11.9

[47] D.P. Heyman and M.J. Sobel, editors. *Handbooks in Operations Research and Management Science, Vol. 2: Stochastic Models*. North-Holland, 1990. 5.7, 6.6, 8.9

[48] W.J. Hopp and M.L. Spearman. *Factory Physics: Foundations of Manufacturing*. McGraw-Hill, 2nd edition, 2001. 2.7, 11.9

[49] L.A. Johnson and D.C. Montgomery. *Operations Research in Production Planning, Scheduling, and Inventory Control*. Wiley, 1974. 6.6

[50] C.V. Jones. User interfaces. In E.G. Coffman, Jr., J.K. Lenstra, and A.H.G. Rinnooy Kan, editors, *Handbooks in Operations Research and Management Science, Vol. 3: Computing*. North-Holland, 1992. 9.6

[51] R.J. Jorna, H.W.M. Gazendam, H.C. Heesen, and W.M.C. van Wezel. *Plannen en Roosteren*. Lansa, Leiderdorp, 1996. 7.11

[52] F.P. Kelly. *Reversibility and Stochastic Networks*. Wiley, 1979. 5.7

[53] W.D. Kelton, R.P. Sandowski, and D.A. Sandowski. *Simulation with Arena*. McGraw-Hill, 1998. 3.9

[54] P.J.B. King. *Computer and Communication Systems Performance Modelling*. Prentice-Hall, 1990. 5.7

[55] J.P.C. Kleijnen. *Statistical Tools for Simulation Practitioners*. Marcel Dekker, 1987. 3.9

[56] J.P.C. Kleijnen. Verification and validation of simulation models. *European Journal of Operational Research*, 82:145–162, 1995. 7.11

[57] L. Kleinrock. *Queueing Systems, Volume II: Computer Applications*. Wiley, 1975. 5.3.6, 5.7

[58] L. Kleinrock. *Queueing Systems, Volume I: Theory*. Wiley, 1976. 5.7

[59] R. Koenker and K.F. Hallock. Quantile regression. *Journal of Economic Perspectives*, 15:143–156, 2001. 2.7

[60] G.M. Koole. A formula for tail probabilities of Cox distributions. *Journal of Applied Probability*, 41:935–938, 2004. 13.11

[61] G.M. Koole. Call center mathematics. Draft of a book, 2006. 13.11

[62] G.M. Koole and S.A. Pot. An overview of multi-skill call centers. Work in progress, 2004. 13.11

[63] L.R. LaGanga and S.R. Lawrence. Clinic overbooking to improve patient access and increase provider productivity. *Decision Sciences*, 38:251–276, 2007. 12.9

[64] A.M. Law and W.D. Kelton. *Simulation Modeling and Analysis*. McGraw-Hill, 1997. 3.9

[65] J.R. Meredith. Reconsidering the philosophical basis of OR/MS. *Operations Research*, 49:325–333, 2001. 7.11

[66] J.A. Van Mieghem. *Operations Strategy: Principles and Practice*. Dynamic Ideas, 2008. 7.11

[67] H. Mintzberg. *The Structuring of Organizations*. Prentice-Hall, 1979. 12.1

[68] I. Mitrani. Computer system models. In E.G. Coffman, Jr., J.K. Lenstra, and A.H.G. Rinnooy Kan, editors, *Handbooks in Operations Research and Management Science, Vol. 3: Computing*. North-Holland, 1992. 5.7

[69] A. Myskja. The man behind the formula. Biographical notes on Tore Olaus Engset. *Telektronikk*, 94:154–164, 1998. 5.7

[70] G.L. Nemhauser, A.H.G. Rinnooy Kan, and M.J. Todd, editors. *Handbooks in Operations Research and Management Science, Vol. 1: Optimization*. North-Holland, 1989. 8.9, 9.6

[71] W.G. Nickets, J.M. McHugh, and S.M. McHugh. *Understanding Business*. McGraw-Hill, 2002. 10.6

[72] A.A.B. Pritsker. Modeling in performance-enhancing processes. *Operations Research*, 45:797–804, 1997. 7.11

[73] P. Reynolds. *Call Center Staffing*. The Call Center School Press, 2003. 13.11

[74] S.M. Ross. *Simulation*. Academic Press, 6th edition, 1996. 3.9

[75] S.M. Ross. *Introduction to Probability Models*. Academic Press, 7th edition, 1997. 1.10, 2.2, 2.7, 3.9, 4.7, 5.7, 11.9, 13.11

[76] S.M. Ross. *A First Course in Probability*. Prentice Hall, 6th edition, 2002. 1.10

[77] R.Y. Rubinstein. *Simulation and the Monte Carlo Method*. John Wiley, 1981. 3.9

[78] D.A. Samuelson. Predictive dialing for outbound telephone call centers. *Interfaces*, 29(5):66–81, 1999. 13.11

[79] W.E. Sasser, Jr. Match supply and demand in service industries. *Harvard Business Review*, 54:133–140, 1976. 13.11

[80] S. Savage. *The Flaw of Averages: Why We Underestimate Risk in the Face of Uncertainty*. 2009. To appear. See also www.flawofaverages.com. 1.10

[81] J. Seddon. *Systems Thinking in the Public Sector*. Triarchy Press, 2008. 7.11

[82] P.M. Senge. *The Fifth Discipline*. Doubleday, 1990. 7.11, 8.2

[83] E.A. Silver and R. Peterson. *Decision Systems for Inventory Management and Production Planning*. Wiley, 2nd edition, 1985. 2.5, 2.7, 11.9

[84] H.A. Simon. *The New Science of Management Decision*. Prentice-Hall, revised edition, 1977. 7.3, 7.3, 7.11

[85] W. Skinner. *Manufacturing in the Corporate Strategy*. Wiley, 1978. 12.9

[86] D.R. Smith and W. Whitt. Resource sharing for efficiency in traffic systems. *The Bell System Technical Journal*, 60:39–55, 1981. 5.4

[87] S. Spear and H.K. Bowen. Decoding the DNA of the Toyota Production System. *Harvard Business Review*, 77(5):96–106, 1999. 8.9, 10.4

[88] R. Stolletz. *Performance Analysis and Optimization of Inbound Call Centers*. Springer, 2003. 13.11

[89] D.Y. Sze. A queueing model for telephone operator staffing. *Operations Research*, 32:229–249, 1984. 5.7

[90] H.A. Taha. *Operation Research: An Introduction*. Prentice Hall, 6th edition, 1997. 8.9

[91] K.T. Talluri and G.J. van Ryzin. *The Theory and Practice of Revenue Management*. Kluwer, 2004. 14.5

[92] J. Taylor and N. Raden. *Smart (Enough) Systems*. Prentice-Hall, 2007. 9.6

[93] H.C. Tijms. *A First Course in Stochastic Models*. Wiley, 2003. 2.4, 2.7, 3.9, 4.7, 5.7

[94] H.C. Tijms, M.H. van Hoorn, and A. Federgruen. Approximations for the steady-state probabilities in the $M/G/c$ queue. *Advances in Applied Probability*, 13:186–206, 1981. 5.7

[95] E. Turban. *Decision Support Systems and Expert Systems*. Prentice-Hall, 4th edition, 1995. 7.11, 8.2, 8.9, 9.4, 9.6

[96] J. Walrand. *An Introduction to Queueing Networks*. Prentice-Hall, 1988. 5.7

[97] A. Warner. *The Bottom Line: Practical Financial Knowledge for Managers*. Gower, Aldershot, 1993. 7.11

[98] J.D. Welch and N.T.J. Bailey. Appointment systems in hospital outpatient departments. *The Lancet*, 259:1105–1108, 1952. 12.9

[99] T.R. Willemain. Insights on modeling from a dozen experts. *Operations Research*, 42:213–222, 1994. 7.11

[100] H.P. Williams. *Model Building in Mathematical Programming*. Wiley, 3rd edition, 1993. 9.6

[101] W.L. Winston. *Operations Research: Applications and Algorithms*. Duxbury Press, 1987. 8.9

[102] W.I. Zangwill. The limits of Japanese production theory. *Interfaces*, 22:14–25, 1992. 11.9

[103] H.-J. Zimmermann. An application-oriented view of modeling uncertainty. *European Journal of Operational Research*, 122:190–198, 2000. 8.9

[104] P.H. Zipkin. *Foundations of Inventory Management*. McGraw Hill, 2000. 6.6