Bayesian Zero-Failure Reliability Demonstration

Maha Rahrouh

A thesis presented for the degree of
Doctor of Philosophy

Statistics and Probability Group
Department of Mathematical Sciences
University of Durham
England, U.K.

2005

In the memory of my Father,
to my lovely family and to my country, Syria.

Bayesian Zero-Failure Reliability Demonstration

Maha Naef Rahrouh

Submitted for the degree of Doctor of Philosophy
April 2005

Abstract

We study required numbers of tasks to be tested for a technical system, including systems with built-in redundancy, in order to demonstrate its reliability with regard to its use in a process after testing, where the system has to function for different types of tasks, which we assume to be independent. We consider optimal numbers of tests as required for Bayesian reliability demonstration in terms of failure-free periods, which is suitable in case of catastrophic failures, and in terms of the expected number of failures in a process after testing. We explicitly assume that testing reveals zero failures. For the process after testing, we consider both deterministic and random numbers of tasks. We also consider optimal numbers of tasks to be tested when aiming at minimal total expected costs, including costs of testing and of failures in the process after testing. Cost and time constraints on testing are also included in the analysis. We consider such reliability demonstration for a single type of task, as well as for multiple types of tasks to be performed by one system. We also consider optimal Bayesian reliability demonstration testing in combination with flexibility in the system redundancy, where more components can be installed to reduce test effort. For systems with redundancy, we restrict attention to systems with exchangeable components, with testing only at the component level.

We use the Bayesian approach with the Binomial model and Beta prior distributions for the failure probabilities. We discuss the influence of choice of prior distribution on the required zero-failure test numbers, where these inferences are very sensitive to the choice of prior distribution, which requires careful attention to the interpretation of non-informativeness of priors.

Declaration

The work in this thesis is based on research carried out at the Statistics and Probability Group, Department of Mathematical Sciences, University of Durham, England. No part of this thesis has been submitted elsewhere for any other degree or qualification and it is all my own work unless referenced to the contrary in the text. Chapters 3 and 4 are based on joint work with my supervisors, Dr. F.P.A. Coolen and Dr. P. Coolen-Schrijner, which is published in [16] and [17], respectively. A summary of results of these two chapters also appears in [15], which was presented at the 5th IMA international Conference on Modelling in Industrial Maintenance and Reliability, Salford, April 2004. Part of Chapter 5 has also appeared in [51], which was presented at the 16th ARTS Advances in Reliability Technology Symposium, Loughborough, April 2005.

Acknowledgements

I owe an immeasurable debt to my supervisors, Frank Coolen and Pauline Coolen-Schrijner, who have been a tremendous source of inspiration and knowledge throughout this Ph.D. Many thanks for the support and careful orientation of my research. I am also grateful to Aleppo University, Syria, for sponsoring my Ph.D.

I would like to thank the whole Mathematics department, and the Statistics group in particular, for its friendliness and support. Special thanks to Iain MacPhee, Sharry Borgan, Wojtek Zakrzewski and Anne Taormina. Many thanks for the friendship and support of Magda Carr, Antony Hill, Michael Watson, and Georgina Titchener.

The sudden and tragic death of my dearest father at a crucial time of my Ph.D. would have been extremely difficult to cope with without the constant support and love of my friends and my dearest sister. I am eternally indebted to my soul mates: my closest and dearest friend Karima Laachir, my sister Hanan, and my lovely niece, Nemeir Moukhaiber, for making life so joyful with their humour, and England so homely with their care and affection.

To my friends who kept me going, I cannot express my gratitude in words, Tatsuya Ueno, Ray Mumford, Christian Brand, Jim House, Patrick Dorey and Samer Jarkasy. I am extremely indebted for your wonderful friendship.
Special thanks also to Farid Tari, and Maison and Ahmed Eqab for their wonderful support and care during the last period of my Ph.D.

My greatest thanks to my family, especially my loving father, for all the sacrifices, for never failing to be there with their unrelenting support, persistent encouragement and love, that helped me much more than I can ever express. I love you all very much.

Contents

List of Tables

Chapter 1

Introduction

1.1 Overview

In many situations a technical system or unit can be tested before actually being used in real-life processes. Such testing is particularly important if high reliability is the aim, for example if failures (the unacceptable performance caused by a fault) have severe consequences, such as losses of life or business. The focus in this thesis is explicitly on the system's required performance during a specified period after testing, with tasks that the system has to perform during that period either arriving as stochastic processes, the *random* case, or known in advance, the *deterministic* case. We first consider a *unit*-system in Chapters 2 to 4, and then we extend our approach to include systems with redundancy, where components are considered to be exchangeable, in Chapter 5. We use predictive formulations for reliability requirements and related optimisation problems. Prior to the system's use in a process it can be tested, where we can use tasks as inputs without relying on a particular arrival process for the tasks [32, 39, 40, 54, 55]. We explicitly assume that testing reveals zero failures, which we share with Martz and Waller [37] and which has also been considered by Sandoh [53], in order for the system to be used in the process after testing. This assumption is mostly used for safety-critical systems where high reliability is required.

Throughout this thesis, we consider such situations in the following setting. We assume that the system has to perform one or multiple types of tasks, which are

independent both in the manner in which the system performs the tasks and in the arrival of these tasks in the processes considered. We also assume that the system does not wear-out, and indeed performs tasks of one type similarly, in the sense that failures occurring in tasks of the same type can be represented by exchangeable Bernoulli random quantities, enabling a simple Bayesian model [33, 38] (Section 2.3). In addition, the functioning of the system for such different types of tasks is tested independently for each type of task. These independence assumptions may, in some situations, be considered to be unrealistic, for example if some system functionality is shared by all types of tasks or if one strongly feels that 'one-test-tests-all' for some type of task. Many such situations would require more detailed modelling, with positive correlations between probabilities of the system dealing successfully with different (types of) tasks explicitly included in the model. Such included positive correlations would lead to fewer required tests without failures in order to demonstrate reliability, hence the results in this thesis could be seen as an upper bound for required test numbers in case of lack of knowledge about such possible positive correlations. From the point of view of observing test results possibly to support such (possible) positive correlations, we would like to remark that with the assumption of stopping testing at the first observed failure, the test data could not statistically support any claim of such possible positive correlations, and, alternatively, if there are no failures observed in the test this could also not provide conclusive statistical support for such correlations. Hence, not only are the results here important as they correspond to a kind of 'worst-case scenario' for the number of tests required, the test observations in this setting could also not provide conclusive statistical support in favour, or against, assumptions on such correlations had one included these in the model. Study of required test effort, as we present in this thesis, for the more general situations with positive correlations as presented in Coolen, *et al.* [19] is an interesting topic for future research.

Throughout, we are interested in the required numbers of tasks, of each type, to be tested without failures to achieve a certain required level of reliability for the process in which the system is used after testing. We will express such reliability in terms of probability of failure-free periods (Chapters 3 and 5) and in terms of the

expected number of failures (Chapter 4) in the process after the zero-failure testing. The solutions of our formulated problems depend explicitly on the number of tasks, or the length of the period after testing considered, which does not occur in the more established approaches. We will aim for minimal total number of tests needed and for minimal total expected costs, both of testing and possible failures in the process after testing. We consider both types of process failures; *catastrophic* failures, in the sense that the process would be discontinued, and *non-catastrophic* failures, which allow continued use of the system in the process. Test budget and, in some situations, time constraints, to achieve a predictive reliability requirement for the system's functioning in the process, are also taken into consideration. We consider this from a Bayesian statistical perspective [6, 24], which has the advantage that predictive inference is relatively straightforward. Bayesian statistics also offers the opportunity to take expert judgements explicitly into account by eliciting an appropriate prior distribution [5, 49]. However, if one requires reliability to be *demonstrated* via zero-failure testing, one would probably not wish to rely on assumed information other than the test results. In the Bayesian setting, this suggests the use of a so-called 'non-informative' prior, choice of which is far from trivial for zero-failure testing, which will be shown in this thesis.

The optimal solutions to the problems considered in this thesis depend on the, possibly random, numbers of tasks of different types in, or length of, the considered process after testing. This does not occur in the more established approaches in which the optimality criteria do not tend to take such process characteristics explicitly into account. A motivation here is the observation that frequently, when one speaks about testing to demonstrate reliability, it is rather vague which optimality criterion is deemed most appropriate.

1.2 Reliability demonstration

A natural measure of reliability, especially for critical systems (or components), is the probability that the system will perform successfully (without failures) for a specified period of time and in a specific environment, according to the prior knowl-

edge and the information obtained from the testing process. In most applications in the literature of reliability, the focus tends to be on unobservable characteristics such as the mean time between failures (MTBF), which is considered to be a critical parameter to measure reliability, see e.g. Meeker and Escobar [43, Sec.10.6] and Martz and Waller [38]. Such applications do not take into account important information about the process after testing, e.g. the required number of tasks in the process after testing, as we consider in this thesis, has not been given much attention in the reliability metrics literature. Kvam and Miller [32] state that inference for the unobservable parameters, especially if they have no operational meaning to the analyst, can be unnecessary, if not misleading in the applications of probabilistic safety assessment. In such applications, using observable parameters allows the focus to be on future failures without the necessity of estimating failure rates. Kvam and Miller present methods for predicting future numbers of failures for probabilistic safety assessments, where failures are modelled with the Poisson and Binomial distributions. They also introduce prediction intervals for future numbers of failures based on zero failures in the test phase, in the case of highly reliable test items, where they support Bayesian methods to be used in such applications. However, they do not study the required number of tests to demonstrate a required reliability level.

Barlow [4] presents a historical overview of reliability theory from 1961, the year of publication by Birnbaum, *et al.* [8], when reliability theory started to be treated as a separate subject, to the year 2002. He also emphasizes the need of adopting the Bayesian approach through reliability history. A brief historical overview to reliability can also be found in Andrews and Moss [1].

Reliability testing offers not just improvement of the system but also provides a basis for reliability assessment. It is best for reliability assessment to be carried out during every stage of a project, with reliability demonstration the final stage of reliability assurance. Balaban [3] emphasizes the importance and effectiveness of reliability demonstration as a reliability assurance activity, near the end of the system development phase, that provides real, measurable data on reliability performance.

As testing is an important way to learn about the performance of a system, we

discuss the general meaning and importance of testing in reliability below.

1.3 Testing and reliability

Failure of large projects, some causing disasters, including the loss of human life and financial loss, highlighted the importance of testing in order to find out how reliable the system is to be run safely and according to the user's specifications. For example, the European Space Agency's Ariane 5 exploded 40 seconds after lift-off because of a software error [52],

In order to learn about the performance of any system, testing is a useful way to gain insight into the overall system quality and to find errors in order to correct them. Testing can be regarded as a tool for learning about the quality of the system by examining it with the aim to find errors, given the user's specifications and environment, before releasing the system for practical application. In other words, test results can be used to deduce the probability of the system failing over a specified period of operation in a process after testing, if the process is assumed to be known. Moreover, testing gives the opportunity to correct errors that are found in the testing process, with the aim of increasing the quality and the reliability of the system. We emphasize the difference between testing and the correction process. Testing can never ensure that the system is free of faults. If testing results in zero failures then it may suggest good reliability of the system, but it cannot confirm that you have a system that is free of faults. As Redmill [52] states, "Testing can prove imperfection by finding a single fault, but it cannot prove perfection".

Generally speaking, the more we test a system, the more confident we can be about its level of reliability. However, it is usually not possible to test exhaustively due to time, budget and other constraints imposed by the real world [31]. Myers [47] claims that by testing, which is often a costly activity, one can add some value to the system by raising its quality or reliability, which could be performed by finding and removing errors. The number of tests, or when testing should stop, should be determined carefully and according to a suitable reliability requirement. This depends on how confident the testers are that possibly remaining faults do not form

a high risk, either to the customer or the producer.

It is important to emphasize the role that testing plays, especially when there are safety issues related to a system's performance. In that case, high reliability is often necessary, and extensive testing may be required.

In this thesis, we consider what we learn from zero-failure testing, and what we can infer about reliability for a system used in a process after testing. In the reliability literature, several methods have been considered to demonstrate a specific reliability level based on different models and assumptions providing different results, for calculating the required testing duration or number of tests. Tal, *et al.* [56] give a comparison of eight methods for calculating the duration of software reliability demonstration testing for safety-critical systems. They divide these methods into two groups: fixed duration testing methods and non-fixed duration testing methods.

Table 1.1, which is taken from [56], gives a brief comparison between these eight methods with respect to the following points.

- If it provides a point estimate, E, and variance,V, and a confidence level, c, for a required reliability value. This is denoted by +;
- The number of failures, F, one should find in order to use the method;
- Test duration: fixed, unknown or unknown but limited.

Tal, *et al.* [56] prefer the TRW method [57], from all the fixed duration testing methods, because it provides an upper confidence bound for the probability of software failure on an arbitrary input, and which can be used for any number of failures, including zero. They argue that Laplace's method is not useful for safety-critical software as it requires a relatively small number of tests, in comparison to the other methods, and applies only when no failures are found, and provides only a point estimate of unreliability. The Bayesian reliability estimation method considers all tests to be independent and the number of failures is Binomially distributed, with a Beta prior distribution for the failure probability. Tal, *et al.* [56] claim that the Bayesian estimation method is very useful for safety-critical software as it provides a point estimate and a variance for failure probability, and enables one to calculate a

Method	E	V	c	F	Duration	Remarks
Laplace	+	-	-	0	Fixed	Finite number of inputs; of historical value only
TRW	+	+	+	≥ 0	Fixed	Best fixed duration method
Bayesian estimation	+	+	+	≥ 0	Fixed	Results almost identical to TRW method
Life testing	-	-	+	≥ 0	Fixed	Many more tests than according to TRW method
MTBF assurance	-	-	-	≥ 0	Fixed	Producer-oriented; not for Safety-critical software
BAZE	+	+	+	≥ 0	Fixed	Very complex calculations and very specific prior knowledge about probability of failure
PRST	-	-	+	≥ 0	Unknown but limited	Shorter than life testing but more stringent; more tests than according to TRW
$t-$Distribution	+	+	+	≥ 1	Unknown	Demonstrated reliability smaller than according to TRW method; not efficient for safety-critical systems

Table 1.1: Methods for calculating system reliability/testing duration.

confidence interval. The BAZE method [37] (Bayesian zero-failure reliability demonstration testing) also considers all tests to be independent, but assumes zero failures and uses the Gamma prior distribution for failure probability. This method requires the following inputs: maximum permissible failure rate, required confidence level c, upper prior limit UL, and lower prior limit LL, for failure rate and p_0, which is the prior probability that the interval (LL, UL) contains the failure rate of interest, for detail see [37]. Tal, *et al.* [56] claim that this method requires very complicated calculations and very specific prior knowledge. The life testing method [36] requires the following: producer's risk, consumer risk, acceptable quality level and rejectable quality level. Tal, *et al.* [56] claim that the life testing method provides a confidence interval for any number of failures and it requires a very large number of failures, which makes it inapplicable for safety-critical systems. The MTBF assurance method [38, 43, 45, 56] demonstrates that the MTBF is not smaller than a specific required value. Tal, *et al.* [56] claim that this method is unsuitable for

safety-critical systems as it does not take into account failures that might happened after a successful certain number of zero-failure missions in the test.

From the non-fixed duration testing methods, Tal, *et al.* [56] prefer the PRST method (probability ratio sequential testing) to the t-distribution method, which is not efficient when safety-critical systems are of concern (e.g. it cannot be used unless at least one failure is found in the test and the testing duration cannot be predicted nor bounded, see [56] for detail). The PRST method [45, 57, 58], requires the following inputs: producer's risk, consumer risk, upper test MTBF and lower test MTBF, which should be specified in the contract. The procedure for this method is usually carried out with the help of a graphical aid. This method is shorter, in terms of the required testing duration, than those for the fixed duration test, which makes it more practical, but may reject systems that would have passed fixed duration tests, especially those that revealed more failures at the beginning of testing than at the end.

Sandoh [53] proposes two procedures for reliability demonstration testing to determine test duration and the maximum acceptable number of failures during the test. The procedures are based on two risks, namely producer and consumer risks, the second procedure is for zero-failure reliability demonstration testing considering a constraint on either the producer-risk or consumer-risk. He assumes that the time to failure follows an Exponential distribution and both the specified MTTSF (Mean Time To Software Failure) in the contract and the acceptable lower level of MTTSF for the consumer are specified prior to demonstration testing. He also assumes that the system is not debugged during demonstration testing but failures are removed after the test. Although he does not use the Bayesian approach, he states clearly that using it can reduce the amount of testing required when good prior knowledge is available. Moreover, he claims that his procedures hold for both software and hardware systems as long as the corresponding assumptions hold. We also believe that our approach holds for both software and hardware systems also as long as our assumptions hold.

In the approach presented in this thesis, we do not explicitly consider point estimates, variance, or confidence levels, which are unobservable characteristics, but

we consider the optimal number of tasks to be tested (of each type of tasks), taking into account aspects of the process after testing, which none of the previously mentioned methods considered explicitly. Restricting attention to zero-failure testing, we consider our method to be intuitively attractive when concerning safety-critical systems. However, we should acknowledge that extending our approach to allow failures in the testing phase, maybe with repairing these failures, may increase its practical applicability and efficiency.

1.4 Bayesian statistics

Bayesian statistics [29, 33], which has been used increasingly in the reliability literature, provides a suitable way for dealing with uncertainties taking prior knowledge into account. The Bayesian approach can be used to combine subjective opinion with data obtained from testing, to compute a system's reliability [26, 49]. Bayesian inferential analysis is straightforward when one represents prior information via a conjugate prior distribution. Without conjugacy, the simplicity and elegance of the predictive densities are lost. However, non-conjugate priors might sometimes be more appropriate, e.g. in probabilistic safety assessments of domestic nuclear power plants where log-normal distributions are sometimes used for Poisson rate parameters [32]. In such situations, modern computation methods, such as MCMC [22] and numerical integration methods can be used. If prior information is not available, a non-informative prior distribution can be used. However, the non-informative prior distribution is not always clear, which is the case in this thesis due to our zero-failure assumption. A further advantage of the Bayesian framework is the direct link to decision theory, via optimisation of expected utilities [21]. Although we use such decision theoretic reasoning in Chapter 5, we do not focus explicitly on utilities but use assumed cost figures instead.

Bertolino and Strigini [7] demonstrate the use of Bayesian statistics as a high level quality way to predict software reliability. Guida and Pulcini [27] use the Bayesian approach for analysing failure data for mechanical components in a reliability demonstration, emphasizing that a correct use of accurate prior information

allows one to draw inferences on the reliability of the new product even before performing the demonstration tests. However, we would not normally support the idea of relying totally on prior information when reliability demonstration for highly reliable systems is the concern.

Throughout this thesis, we use the Bayesian approach, which we share with Mastran [41], Mastran and Singpurwalla [42], and Martz and Waller [37, 38], to reliability predictions, using a Binomial model for the number of failures in testing and a Beta prior distribution for the unknown failure probability, as in Martz and Waller [39], Martz *et al.* [40], and Springer and Thompson [54, 55]. Grohowski and Hausman [26] present the Bayesian approach which quantifies prior information in order to estimate reliability. They support the selection of Gamma distributions as prior distributions for their conjugacy to the Poisson distribution. Rai [50] uses a similar approach to ours in the Poisson case, but used a Poisson distribution, with an unknown parameter following a Beta distribution to get a posterior distribution with a Poisson-Beta distribution.

The two hyperparameters in the Beta prior distribution can be interpreted in terms of results of an imaginary earlier test, in which the system failed to deal with a number of tasks, but performed some other tasks successfully. For reliability demonstration, we will assume that one wishes to use very little prior information, which can be modelled by taking all the hyperparameters small. Often in the statistical literature, the choice of giving the hyperparameters the value 1 is proposed as a so-called non-informative prior, but also other values for these parameters between 0 and 1 are advocated, such as Jeffries' non-informative Beta priors with parameters equal to 1/2. In our setting, the choice of non-informative prior is far from trivial. We will discuss this later, and, also on the basis of our analysis in this thesis, we think that there are good arguments for the choice Beta$(1, 0)$ for reliability demonstration. The influence of these hyperparameters can be understood from their interpretation in terms of numbers of failures and successes in hypothetical earlier tests. Effectively, the number of required tests without failures should counter the prior information of 'imaginary test failures' to demonstrate a specific level of reliability.

Nonparametric predictive inferential methods [18] for the same setting as studied in this thesis have been presented by Coolen and Coolen-Schrijner [14], with which we share the perhaps surprising fact that the deterministic case is a worst-case scenario in terms of the number of tests needed for reliability demonstration (see Section 2.4.2), however they restricted attention to a single type of task.

1.5 Outline of the thesis

In this thesis we study optimal numbers of tasks to be tested as required for Bayesian reliability demonstration with regard to the system's use in a process after zero-failure testing, where the system has to function for different types of tasks, which we assume to be independent.

Chapter 2, which forms the basis for Chapters 3 to 5, presents how we can learn from testing a system in order to predict its reliability, first in a process of one type of tasks, and secondly in a process of multiple types of tasks. We explicitly assume that testing reveals zero failures (Section 2.2). Reliability is considered in terms of failure-free periods (Section 2.4) and in terms of the expected number of failures (Section 2.5) in the process after testing. Using Bayesian statistics, we analyse the prediction for a known number of tasks and the prediction for tasks arriving randomly, with special attention to the Poisson process.

In Chapter 3 reliability is expressed in terms of the probability of a failure-free period, and we will aim for the minimal total number of tests needed (Section 3.3), and for minimal total expected costs of testing and possible failure in the process after testing (Section 3.4), assuming that a failure during the process would be catastrophic in the sense that the process would be discontinued. For the process after testing, we study in detail the cases where the numbers of tasks to be dealt with by the process are known (the deterministic case) and where these numbers are random quantities with Poisson distributions (the Poisson case). We also discuss the influence of the choice of prior distribution on the required test numbers.

In Chapter 4 reliability is expressed in terms of the expected number of failures in the process after testing, and the optimal numbers of tasks to be tested are

derived by optimisation of a cost criterion, taking into account the costs of testing and of failures in the process after testing. We assume that such failures are not catastrophic to the system, in the sense that failures do not stop the use of the system in the process. Cost and time constraints on testing are also included in the analysis (Section 4.2). We focus on the study of the optimal numbers of tests for different types of tasks, depending on the arrival rate of tasks in the process and the costs involved. In Section 4.4, we briefly compare the results of this chapter with optimal test numbers in Chapter 3. For these two different optimality criteria, the dependence of the optimal numbers to be tested, for different types of tasks, on the costs of testing per type, and on the arrival rates of tasks in the process after testing, is very similar.

In Chapter 5, we extend the study by considering reliability demonstration for systems with redundancy, including the two extreme cases of parallel and series systems. Reliability prediction for such systems is considered in terms of failure-free periods, where the process failures are, as in Chapter 3, considered to be catastrophic (Section 5.2). Reliability demonstration for a single task (Section 5.3) and for m tasks, both for one type and for multiple types of tasks are considered (Sections 5.4 and 5.5). The results of this chapter are also compared to the results of Chapters 3 and 4. General conclusions, and some suggestions for future research, are briefly discussed in Sections 5.8 and 5.9.

Chapter 2

Reliability prediction after zero-failure testing

2.1 Introduction

In this chapter, which forms the basis for Chapters 3 to 5, reliability is considered in terms of failure-free periods (probability of zero failures in process) and the expected number of failures, in the process after testing. We consider processes with a known number of tasks, which we call the 'deterministic case', and processes with a random number of tasks, which we call the 'random case'. We study the Poisson case as one of the random cases in more detail. In both the deterministic and random cases, we consider processes with a one-type of task and with multiple independent types of tasks.

Throughout this thesis, we assume that the system's performance for tasks of one type is exchangeable over these tasks and that the system's failure of performing one task does not affect its ability to perform other tasks. We also assume that tasks arrive independently in the process and can also be tested independently. Learning about the performance of a system from test results enables prediction of the number of failures in the process for which the system is used after testing. The random quantities representing failures (or successes) of the tasks are assumed to be conditionally independent (for given failure probabilities per type of task) and identically distributed per type of task. We restrict our attention to zero-failure

testing before the process, so we assume that the system is only released for use in the process if testing reveals zero failures. This assumption is reasonable if systems must be highly reliable. We also assume that the system does not wear-out and will not be repaired during the period over which the process is considered.

We use a Bayesian approach to reliability predictions, with a Binomial model for the number of failures in testing (before the process) and Beta prior distributions for the unknown parameters of the failure probabilities. We also show that, in our Bayesian approach to this problem, the deterministic case is a worst-case scenario with regard to the achieved level of reliability, and hence with regard to the number of tests needed.

In Section 2.2 we introduce the zero-failure testing assumption. Section 2.3 introduces the model and the method used throughout this thesis, where the Bayesian approach with Beta prior distributions is used. In Section 2.4 reliability prediction is introduced in terms of failure-free periods for both the deterministic (Section 2.4.1) and the random (Section 2.4.2) cases, after zero-failure testing. In Section 2.5 reliability is considered in terms of the expected number of failures in the process after testing, again for the deterministic and the random cases.

2.2 Zero-failure testing assumption

Throughout this thesis, we assume explicitly that testing reveals zero failures, which is a common requirement for safety-critical systems, in the sense that the system is only released for use in the process if testing has revealed zero failures [37]. The underlying idea is that, if a failure occurs during testing, the system will be redesigned, after which testing may have to start all over again. Such testing results are, for example, realistic in situations where high reliability is required, and where failures during testing have severe consequences or may lead to redesign of the system.

We do not consider aspects of retesting after fixes or redesign, in case of failures during testing, which would increase practical applicability of our approach but would require detailed modelling of reliability growth (see e.g. Gaver and Jacobs [23] who consider such aspects for items that require destructive testing). As examples

of systems for which such a setting may be realistic, one can think of systems that must trigger alarms for a variety of possible warnings in industrial applications, or software systems that control the use of different databases and enable information transfers between them.

2.3 Learning from tests

In this section, we present the standard Bayesian method of updating a conjugate prior for the parameter of a Binomial distribution, representing how we learn from testing. To learn about a system's performance, we can test the system prior to its use in a process. During testing, we can use tasks as inputs without relying on a particular arrival process for the tasks. Assume that we carry out a test consisting of n tasks ($n \geq 1$). Throughout this thesis we will, for mathematical convenience, assume that the uncertainty about the probability of a failure, say θ, before testing, is represented by a Beta prior distribution, $\theta \sim Beta(\alpha, \beta)$, with probability density function (pdf)

$$p(\theta) = \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{B(\alpha,\beta)}\,, \quad \text{for } 0 \leq \theta \leq 1, \tag{2.1}$$

where $B(\alpha,\beta) = \dfrac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$, and $\Gamma(.)$ is the Gamma function (which is defined as $\Gamma(\alpha) = \int_0^\infty u^{\alpha-1}e^{-u}du$, $\alpha > 0$) for which $\Gamma(\alpha+1) = \alpha\Gamma(\alpha)$ for any positive value of α, and $\Gamma(\alpha+1) = \alpha!$ for positive integer values of α. So, if α and β are positive integers, the Beta-function simplifies to

$$B(\alpha,\beta) = \frac{(\alpha-1)!(\beta-1)!}{(\alpha+\beta-1)!}. \tag{2.2}$$

Here, α and β are positive constants ($\beta = 0$ is also possible in such settings), the choice of which we discuss in detail later in this thesis, in Section 2.4 in particular, and $\theta \in [0,1]$. The Beta prior is conjugate to the Binomial model, see e.g. Lee [33] or Martz and Waller [38]. Moreover, the Beta distribution allows a reasonably large variety of shapes to model expert beliefs [33]. Weiler [59] showed that assuming a Beta prior in Binomial sampling, when it is not the true prior distribution, often has only a small effect in practical applications. In addition, the parameters of Beta distributions can easily be interpreted, an important advantage when we consider

the influence of the prior distribution on the required test numbers later in this thesis. Dyer and Chiou [20] found in their research that the Beta family is the most suitable, comparing to other families of priors, when prior information about the system is not available. Colombo and Constantini [11, 12] support the Beta distribution as Bayesian prior with Binomial sampling. However, their approach does not support the choice of $\beta \leq 1$ which we suggest in our approach. Moreover, their analysis is limited to integer values for α and β, whereas in our approach α and β can be any positive constants, while we advocate the choice of $\alpha = 1$ and $\beta = 0$, which we discuss in detail later in this thesis. We also share the use of a Binomial model for the tests, and a Beta prior distribution with Martz *et al.* [40], Martz and Waller [39], and with Springer and Thompson [54, 55].

We use the test results to update the prior distribution for the unknown failure probability, θ. The posterior distribution of θ, after learning that a test of $n \geq 1$ tasks has revealed $r \geq 0$ failures, is characterised by the pdf

$$p(\theta|(n,r)) = \frac{L(\theta|(n,r))\ p(\theta)}{\int_0^1\ L(\theta|(n,r))\ p(\theta)\ d\theta}\ , \quad \text{for } \ 0 \leq \theta \leq 1, \tag{2.3}$$

where the term $L(\theta|(n,r))$ is the likelihood function. Based on the assumption that the number of failures in the n tests follows a Binomial distribution, the likelihood function is

$$L(\theta|(n,r)) = \binom{n}{r}\ \theta^r\ (1-\theta)^{n-r}. \tag{2.4}$$

The posterior distribution, with probability density function $p(\theta|(n,r))$, quantifies our uncertainty about θ based on our background knowledge and the results of the test. Thus, when using the Beta(α, β) prior distribution for θ, and taking the zero-failure testing assumption into account, denoting such data by $(n, 0)$, the resulting posterior distribution is $\theta|(n,0) \sim \text{Beta}(\alpha, \beta + n)$, which has pdf

$$\begin{aligned} p(\theta|(n,0)) &= \frac{(1-\theta)^n\theta^{\alpha-1}(1-\theta)^{\beta-1}}{\int_0^1 (1-\theta)^n\theta^{\alpha-1}(1-\theta)^{\beta-1}d\theta} \\ &= \frac{\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{\int_0^1 \theta^{\alpha-1}\,(1-\theta)^{\beta+n-1}d\theta}\ . \end{aligned} \tag{2.5}$$

Using the Beta function $B(\alpha, \beta+n) = \int_0^1 \theta^{\alpha-1}(1-\theta)^{\beta+n-1}\ d\theta$, (2.5) is equal to

$$p(\theta|(n,0)) = \frac{\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{B(\alpha, \beta+n)}, \tag{2.6}$$

which is the pdf of the Beta distribution with parameters α and $\beta + n$.

The constants α and β can be interpreted as the number of failures and the number of successes, respectively, in $\alpha + \beta$ imaginary observations which reflect our judgements before the real test results become available. We can also interpret $\beta + n$ as the total successes in both the imaginary data set and the test before the process. Indeed, we will see that there is a good reason actually to favour the choice $\beta = 0$. Due to our assumption that testing reveals zero failures in order for the system to be used in the process, we cannot use a Beta prior with $\alpha = 0$, as (2.6) would not be a proper pdf [6]. The actual choice of α will later be shown to play a crucial role in determining the required test numbers. Clearly, if we wish to use very little prior information then we should take both α and β very close to zero. In this thesis we will use the choice of $\alpha = 1$, which still requires very many zero-failure tests to achieve the required reliability level. We consider this choice to be conservative; however, it is needed because of the absence of clear non-informative priors due to the zero-failure testing assumption, and we discuss this in more detail later.

2.4 Reliability prediction in terms of failure-free periods

In this section we express reliability in terms of probability of a failure-free periods in the process after testing. Reliability in terms of failure-free period is particularly important if high reliability is the aim, for example if failures have severe consequences. This section is the basis for Chapters 3 and 5, where failures are considered to be catastrophic. We consider such situations in the following setting. We assume that the system has to perform multiple types of tasks, which are independent both in the manner in which the system performs the tasks and in the arrival of these tasks in the processes considered. In addition, the functioning of the system for such different types of tasks is tested independently per type of task. We consider two types of processes after testing, namely processes with known numbers of tasks, the deterministic case, which is presented in Section 2.4.1 and processes where these numbers are random quantities, the random case, presented in Section

2.4.2. Again, we assume that the system does not wear-out and performs tasks of one type similarly, in the sense that failures can be represented by exchangeable Bernoulli random quantities, enabling the use of a simple Binomial model.

2.4.1 Known number of tasks

In this section, we first consider a single type of task, where tasks are dealt with independently of each other, independently of the system's performance, and also independently of testing. We then generalize the results for the case where multiple independent types of tasks are required to be performed successfully by the system.

Process with a single type of task

Let m be a known number of tasks in the process, after testing, that the system has to perform, and let $P(0|m,\theta,(n,0))$ be the probability of zero failures in this process, after zero-failure testing of n tasks, for given θ. As the tasks that need to be performed in the process are assumed to be conditionally independent of the tasks in the tests, for given θ, we have

$$P(0|m,\theta,(n,0)) = P(0|m,\theta).$$

By using the theorem of total probability we can derive the unconditional probability distribution for zero failures in the process. Using the posterior pdf $p(\theta|(n,0))$, as given in (2.6), the posterior predictive probability for a failure-free process is

$$\begin{aligned} P(0|m,(n,0)) &= \int_0^1 P(0|m,\theta,(n,0))\; p(\theta|(n,0))\; d\theta \\ &= \int_0^1 \frac{(1-\theta)^m\;\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{B(\alpha,\beta+n)}\; d\theta = \frac{B(\alpha,\beta+n+m)}{B(\alpha,\beta+n)} \\ &= \frac{\Gamma(\beta+n+m)\Gamma(\alpha+\beta+n)}{\Gamma(\alpha+\beta+n+m)\Gamma(\beta+n)}. \end{aligned} \tag{2.7}$$

The role of β is interesting in this probability, as it always appears in the form $\beta+n$. If we define $z=\beta+n$, then it is clear that any reliability requirement based solely on this $P(0|m,(n,0))$ will give a requirement on z, hence the immediate effect of choosing a larger value for β is that the required test size n will be smaller, by the same margin. In this case, $\beta+n$ can be interpreted as the 'combined total number of

successes in the imaginary data set, representing prior judgements, and in the test'. Hence, and from the zero-failure assumption in n (> 0) tests, we can use $\beta = 0$. Also, because of this assumption (no failures are observed) we cannot use $\alpha = 0$, as the corresponding posterior distribution would be improper.

We consider two ways to simplify (2.7). First, for general values of $\alpha > 0$ and $\beta \geq 0$,

$$P(0|m,(n,0)) = \prod_{j=1}^{m} \frac{j+\beta+n-1}{j+\alpha+\beta+n-1}. \tag{2.8}$$

Let us denote the right-hand side, as a function of m, by $g(m)$. Clearly, $g(m)$ is positive (for $n > 0$) and decreasing as a function of m, and from the fact that

$$g(m) - g(m+1) = \prod_{j=1}^{m} \frac{j+\beta+n-1}{j+\alpha+\beta+n-1}[1 - \frac{m+\beta+n}{m+\alpha+\beta+n}],$$

is decreasing in m, $g(m)$ is a convex function of m. If we restrict attention to integer values $\alpha > 0$ and $\beta \geq 0$, then a second expression for (2.7) is

$$P(0|m,(n,0)) = \prod_{j=1}^{\alpha} \frac{j+\beta+n-1}{j+\beta+n+m-1}. \tag{2.9}$$

Let us denote this right-hand side, as a function of m, by $f(m)$. Although this form is less general than $g(m)$, the effect of different choices of α might be clearer from $f(m)$. It is clear that $P(0|m,(n,0))$ is an increasing function of n, and a decreasing function of m, which agrees with intuition. From $f(m)$ it is clear that increasing α has a big effect on $P(0|m,(n,0))$, and hence on the minimal required test size n.

Process with multiple types of tasks

It is easy to generalize this approach to a system that has to perform multiple types of tasks, which arrive independently in the process after testing, and which can also be tested independently. We assume that the system must deal with $k \geq 2$ types of tasks, which we assume to be fully independent, in the sense that the system's failure of performing one task does not affect its ability to perform tasks of other types. We use notation as above, but with index i, $i = 1, \ldots, k$, added to indicate the particular type of task. We denote k-vectors by underlining the notation used above, further notation is straightforward and is not explicitly introduced. We should emphasize

that all random quantities related to different types of tasks are assumed to be independent throughout our analysis.

Assuming a Beta(α_i, β_i) prior distribution for the parameter θ_i of a task of type i ($i = 1, \ldots, k$), and that n_i tests of this type revealed zero failures, the posterior predictive joint probability of zero failures in m_i tasks of type i after testing is

$$P(0|\underline{m}, (\underline{n}, \underline{0})) = \prod_{i=1}^{k} \prod_{j=1}^{m_i} \frac{j + \beta_i + n_i - 1}{j + \alpha_i + \beta_i + n_i - 1}, \tag{2.10}$$

for general values of $\alpha_i > 0$ and $\beta_i \geq 0$, and similar to (2.9) we can also write this as

$$P(0|\underline{m}, (\underline{n}, \underline{0})) = \prod_{i=1}^{k} \prod_{j=1}^{\alpha_i} \frac{j + \beta_i + n_i - 1}{j + \beta_i + n_i + m_i - 1}, \tag{2.11}$$

for integer values $\alpha_i > 0$ and $\beta_i \geq 0$, $\forall i = 1, \ldots, k$. Clearly, $P(0|\underline{m}, (\underline{n}, \underline{0}))$ is decreasing in k, the number of types in the process, which is intuitively logical: the more types the system has to perform successfully in the process, for fixed n_i and m_i for all i, the less reliable the system gets, hence the more tests are needed to demonstrate a required reliability level, which we show later in this thesis.

2.4.2 Random number of tasks

In this section, we consider the case where the number of tasks in a specified period [0,t] in the process after zero-failure testing is a random quantity, say M, with probability $P(M = u)$ for nonnegative integers u. Throughout this thesis we assume that M is independent of the test information $(n, 0)$, which implies that $P(M = u|(n, 0)) = P(M = u)$.

We are interested in a specified failure-free period (FFP) in the process after testing, again assuming that the test revealed no failures in n tasks tested. This probability follows from the theorem of total probability, and is

$$P(FFP|(n, 0)) = \sum_{u=0}^{\infty} P(0|u, (n, 0))P(M = u). \tag{2.12}$$

The convexity of $g(m) = p(0|m, (n, 0))$, as a function of m (Section 2.4.1), leads to an interesting result, as Jensen's inequality for convex functions (see e.g. Hardy, *et*

al. [28] for further details on Jensen's inequality and convexity) gives

$$P(FFP|(n,0)) = E(g(M)) \geq g(E(M)). \tag{2.13}$$

This implies that, if we restrict attention to probability distributions for M with expected value $E(M) = m$, then we have

$$P(FFP|E(M) = m, (n,0)) \geq P(0|m,(n,0)). \tag{2.14}$$

This is possibly surprising, as it implies that given any probability distribution for M with $E(M) = m$, the probability of a specified failure-free period after testing, after n tests without failures, is at least as large as the probability of zero failures in a known number of tasks, m. Hence the required number of successful tests will be no more, in case of such an M, than required for the process with m tasks, which will be illustrated later in Chapters 3, 4 and 5. Therefore, one could consider the exact knowledge of the value of the random quantity M to be disadvantageous, we illustrate this in Example 2.4.1. This worst-case scenario, which is a feature of the deterministic case, also occurs in [14], but based on different foundations.

For the case where the system has to perform $k \geq 1$ independent types of tasks, the corresponding probability of FFP is

$$P(FFP|(\underline{n},\underline{0})) = \prod_{i=1}^{k} P(FFP_i|(n_i,0)), \tag{2.15}$$

where

$$P(FFP_i|(n_i,0)) = \sum_{u=0}^{\infty} P(0|u,(n_i,0))P(M_i = u). \tag{2.16}$$

This factorisation of the joint probability is only allowed due to the assumed independence of the different M_i, where the assumed independence of M_i and the test information is explicitly used in the application of the theorem of total probability above. If one doubts that the system deals with the different tasks independently, in the sense that one would expect a positive correlation between different M_i's, then this product form provides a kind of 'worst-case scenario', in the sense that to reach a particular level of reliability, the required number of successful tests per type will never be less than when positive correlations between the M_i would be taken into account. The convexity argument used above in this section, can be used

again for each of these $P(FFP_i|(n_i, 0))$, to prove that the deterministic case, with known m_i, requires at least as many successful tests as are required for any random M_i with expected value $E(M_i) = m_i$. Hence, also with multiple types of tasks the deterministic case will require at least as many zero-failure tests as corresponding random cases (see e.g. Example 3.2.1).

The Poisson case

Suppose that the tasks in the process after testing arrive according to a homogeneous Poisson process with known arrival rate $\lambda > 0$, and that a reliability requirement of a minimum probability of a failure-free period of length $t > 0$ is used. Let $\mu = \lambda t$. Then, the random number of tasks the system has to deal with during this period, say M, has a Poisson distribution with expected value $E(M) = \mu$, we denote this by $M \sim \text{Po}(\mu)$. Using (2.9), the probability of such a failure-free period, after testing n tasks revealed zero failures, when restricting attention to integer $\alpha > 0$, for any $\beta \geq 0$, is

$$\begin{aligned} P(FFP|M \sim \text{Po}(\mu), (n, 0)) &= \sum_{u=0}^{\infty} P(0|u, (n, 0))P(u|\mu) \\ &= \sum_{u=0}^{\infty} \prod_{j=1}^{\alpha} \frac{j+\beta+n-1}{j+\beta+n+u-1} e^{-\mu} \frac{\mu^u}{u!}. \end{aligned} \quad (2.17)$$

For general values of $\alpha > 0$, (2.8) leads to

$$P(FFP|M \sim \text{Po}(\mu), (n, 0)) = \sum_{u=0}^{\infty} \prod_{j=1}^{u} \frac{j+\beta+n-1}{j+\alpha+\beta+n-1} e^{-\mu} \frac{\mu^u}{u!}. \quad (2.18)$$

The following example illustrates these probabilities, and the influence of prior knowledge on reliability, in terms of the probability of zero failures in the process, after testing has revealed zero failures. We consider both the deterministic case and the Poisson case.

Example 2.4.1

Suppose that a system has performed $n = 20$ tasks in a test without failure. Table 2.1 gives the probabilities of zero failures for a variety of values for α and β in (1) $m = 5$ tasks, and (2) a random number of tasks, M, that has a Poisson distribution

with expected value $E(M) = \mu = \lambda t = 5$ within the period $[0, t]$. This example demonstrates the strong influence of changes to the value of α, and the far weaker influence of the value of β. Table 2.1 also illustrates the result in (2.14) i.e. that the system reliability for a process with a random M, with $E(M) = m$, is at least as large as the system reliability for a process with a known number of tasks m after n tests without failures. Moreover, although we do not have an analytic proof of the following statement, the study of many examples suggests that the difference between these probabilities for the deterministic and the Poisson cases is very small.

α	β	$P(0\|m = 5, (20, 0))$	$P(FFP\|M \sim \text{Po}(5), (20, 0))$
1	0	0.8000	0.8063
1	1	0.8077	0.8136
1	2	0.8148	0.8203
1	5	0.8333	0.8379
1	10	0.8571	0.8606
1	100	0.9600	0.9603
2	0	0.6462	0.6608
3	0	0.5265	0.5497
0.5	0	0.8933	0.8960
0.1	0	0.9775	0.9779
0.01	0	0.9977	0.9978

Table 2.1: $P(0|m = 5, (20, 0))$ and $P(FFP|M \sim \text{Po}(5), (20, 0))$ for varying α, β.

To compute the infinite sums in (2.17) and (2.18), we have used an approximation by cutting off at the point where the difference between two consecutive partial sums does not exceed 10^{-20}. As, for large u, the terms in the sum decrease very rapidly, this gives a good approximation to the infinite sum.

2.5 Reliability prediction in terms of the expected number of failures

Reliability can also be considered in terms of the expected number of failures in the process after testing which will be the focus of Chapter 4. This is relevant if the failures are not catastrophic, in the sense that failures do not stop the use of the system in the process, but may incur costs. Suppose that a system has to perform $k \geq 1$ types of tasks, which we again assume to be fully independent.

Let D_i be the number of failures in m_i tasks of type i, in a process of $k \geq 1$ independent types of tasks. As $D_i|m_i, \theta_i, (n_i, 0) \sim \text{Binomial}(m_i, \theta_i)$, by the conditional independence assumption, and assuming a $\text{Beta}(\alpha_i, \beta_i)$ prior distribution for θ_i, we have

$$\begin{aligned} E(D_i|m_i,(n_i,0)) &= E_{\theta_i}(E(D_i|m_i,\theta_i,(n_i,0))) = E_{\theta_i}(m_i\ \theta_i|(n_i,0)) \\ &= m_i \int_0^1 \theta_i\ p(\theta_i|(n_i,0))d\theta_i = m_i\ \frac{B(\alpha_i+1,\beta_i+n_i)}{B(\alpha_i,\beta_i+n_i)} \\ &= \frac{\alpha_i m_i}{\alpha_i+\beta_i+n_i}. \end{aligned} \tag{2.19}$$

Let $D = \sum_{i=1}^{k} D_i$ be the total number of failures. Then

$$E(D|\underline{m},(\underline{n},\underline{0})) = \sum_{i=1}^{k} E(D_i|m_i,(n_i,0)) = \sum_{i=1}^{k} \frac{\alpha_i m_i}{\alpha_i+\beta_i+n_i}. \tag{2.20}$$

Clearly, $E(D|\underline{m},(\underline{n},\underline{0}))$ is increasing in k, which is logical as the more types of tasks the process is required to perform successfully, the more failures one would expect to encounter. For a random number M_i tasks of type i, with $E(M_i) = m_i$, after zero-failure testing, the expected number of failures of type i is also equal to (2.19).

Suppose that tasks arrive in the process according to a Poisson process over the period $[0, t]$ after testing, $M|\lambda \sim \text{Po}(\lambda t)$, with expected value $E(M|\lambda) = \lambda t$, where we denote the total number of failures in the random process again by D, then $D|\theta, \lambda, (n, 0) \sim \text{Po}(\theta\lambda t)$ [48]. For $k \geq 1$ types of fully independent tasks, each arriving as a Poisson process with $E(M_i|\lambda_i) = \lambda_i t$, with $t > 0$ the length of the period over which the process is considered, the expected total number of failures is

$$E(D|\underline{\lambda},(\underline{n},\underline{0})) = t\sum_{i=1}^{k} \frac{\alpha_i\lambda_i}{\alpha_i+\beta_i+n_i}. \tag{2.21}$$

In Chapter 4, we study Bayesian reliability demonstration in terms of the expected number of failures after testing, and we particularly discuss the Poisson case, with known arrival rates, to study the effect of the length of time t over which the process is considered.

Chapter 3

Reliability demonstration for failure-free periods

3.1 Introduction

In this chapter we study the required number of tasks to be tested in a system, in order to demonstrate a level of reliability with regard to the system's use in a process after testing. Reliability is expressed in terms of the probability of failure-free periods after zero-failure testing. Throughout this chapter we use the setting that is introduced in Chapter 2, mainly Section 2.4.

The system has to perform multiple fully independent types of tasks after testing, which is also assumed to be independent of the process. We focus on the important question of how many tasks of each type should be tested, explicitly assuming that these tests lead to zero failures, to achieve a certain required level of reliability for the process in which the system is used after testing. For the process after testing, we consider both the cases that the numbers of failures are deterministic and random, including tasks arriving according to a Poisson processes. It turns out that the deterministic case is worst in the sense that it requires most tasks to be tested. We consider such reliability demonstration for a single type of task, as well as for multiple types of tasks to be performed by one system. We also consider the situation where tests of different types of tasks may have different costs, aiming at minimal expected total costs, assuming that failure in the process would be catastrophic, in

the sense that failures stop the use of the system in the process and may incur a very high cost. Generally, these inferences are very sensitive to the choice of prior distribution, so one must be very careful with interpretation of non-informativeness of priors.

In Section 3.2 we consider a single type of task, which allows us to introduce the basic concepts used in later sections and to study the effect of the prior distribution on required test numbers. We also show (and illustrate by examples) the fact that the deterministic case is a worst-case scenario with regard to the number of tests needed, using results from Section 2.4.2. Section 3.3 generalizes this approach to a system that has to perform multiple types of tasks. In Section 3.4 we include costs in a related analysis, assuming that failure of the system during the process after testing is catastrophic, in the sense that the process would be discontinued, with all consequences taken into account via a (probably large) cost figure which we assume to be known. The model in Section 3.4 allows for the testing of different types of tasks to have different costs, and allows such tasks to be required in the process in different numbers or to arrive at different rates. We study the effect of these characteristics of the different types of tasks on the optimal test numbers. The problem in Section 3.3 is actually a special case of the more general problem formulation in Section 3.4, but has sufficient interesting features to be studied separately. We end the chapter with some concluding remarks in Section 3.5.

3.2 A single type of task

In this section we restrict attention to a single type of task, and we consider the following basic situation. Suppose we must determine the number n of tests to perform, in order to satisfy a reliability requirement in terms of probability of zero failures in the process after testing. Here, as we do throughout this thesis, we explicitly assume that testing reveals no failures (see Section 2.2). We first consider the deterministic case for the process after testing, meaning that we know the number of tasks that need to be performed after testing: we denote this number by m. In a standard Bayesian setting (Section 2.3), the posterior distribution of the unknown

θ, with a Beta(α, β) prior distribution, after n zero-failure tests is Beta$(\alpha, \beta + n)$, where $\alpha > 0,\ \beta \geq 0$, the choice of which we discuss later.

Section 2.4 shows that the posterior predictive probability (2.7) for the reliability requirement in terms of zero failures in m tasks after testing, where testing revealed zero failures can be written by two expressions: (2.8) for the general values of $\alpha > 0$ and $\beta \geq 0$, and (2.9) when restricting attention to integer values $\alpha > 0$ and $\beta \geq 0$. Section 2.4 also shows that $P(0|m, (n, 0))$ is decreasing and convex as a function of m, and increasing as a function of n.

For positive integer α, and p, the required reliability level, close to 1, if n is large compared to α, then from (2.9) the minimal required value for $z = \beta + n$, assuming again that the system functions successfully for all tasks tested, is close to the solution of

$$(\frac{z}{z+m})^{\alpha} = p.$$

Therefore, this required value of z is close to

$$\frac{p^{1/\alpha}}{1-p^{1/\alpha}} m \approx \frac{p}{1-p} \alpha m, \tag{3.1}$$

which follows from the fact that (for positive integer α and p close to 1)

$$\lim_{q \uparrow 1} \frac{q^{1/\alpha}(1-q)}{(1-q^{1/\alpha})q} = \alpha, \tag{3.2}$$

which can be derived via de l'Hopital's theorem,

$$\begin{aligned} \lim_{q \uparrow 1} \frac{q^{1/\alpha}(1-q)}{(1-q^{1/\alpha})q} &= \lim_{q \uparrow 1} \frac{\frac{\partial q^{1/\alpha}(1-q)}{\partial q}}{\frac{\partial (1-q^{1/\alpha})q}{\partial q}} = \lim_{q \uparrow 1} \frac{\frac{1}{\alpha} q^{\frac{1}{\alpha}-1}(1-q) - q^{\frac{1}{\alpha}}}{-\frac{1}{\alpha} q^{\frac{1}{\alpha}} + (1-q^{\frac{1}{\alpha}})} \\ &= \frac{\frac{1}{\alpha} - 1 - \frac{1}{\alpha}}{1 - 1 - \frac{1}{\alpha}} = \alpha. \end{aligned}$$

For example, if we take $p = 0.95$, then taking α equal to two will lead to a required z of just more than twice the size than that which would be required for α equal to one (e.g. see Table 3.1). Indeed, in this setting, small differences in α can lead to large differences in the inference of interest, namely the required test size n. This might be explained, intuitively, in the sense that α represents the imaginary

number of failures reflecting prior judgements, and hence it takes relatively a lot more tests without failures to get convinced of the reliability of the system if we increase α. Generally, one can only assume that a small value of n is sufficient to demonstrate reliability with a reasonably large p, by either assuming a large value for β, so in effect assuming prior information as if one had already seen many tests without failure, or assuming a very small value of α. For any n, one can find a value of α such that, for any $p \in (0,1)$ and any m, this number n satisfies the reliability requirement. Hence, one can always directly influence the required n via the choice of α, which puts more emphasis on appropriate choices for α irrespective of a predetermined notion of acceptably small n.

Let us now consider the situation that m is not known in advance, but let the number of tasks to be completed, in a specified period in the process after testing, be a random quantity, M, with probability $P(M = m)$ for nonnegative integers m, where M is independent of the test information. Now, we are interested in finding the minimal number of successful tests to achieve a required level of reliability p, in terms of the probability of a FFP of a specified length in the process after testing. We showed in Section 2.4.2, from the convexity of $p(0|m,(n,0))$ as a function of m, and using Jensen's inequality for convex functions, that

$$P(FFP|E(M) = m,(n,0)) \geq P(0|m,(n,0)),$$

which implies that the deterministic case is a worst-case scenario with regard to the number of tests needed, in the sense that for any probability distribution for M, with $E(M) = m$, the number of zero-failure tests is not more than the required number of zero-failure tests in a process of a known number of tasks, m (see Section 2.4.2 for details).

In the following examples, we illustrate the above results in determining the minimal test sizes needed to meet reliability requirements, for the deterministic case and for the situation in which we assume a Poisson distribution for M.

Example 3.2.1

Let the reliability requirement be that the probability of a failure-free process is at least $p = 0.95$. First consider the deterministic case with $m = 5$ tasks in the process,

and take $\alpha = 1$ and $\beta = 0$. Equating (2.9) to 0.95 yields $n = 95$ tasks without revealing a failure. Secondly, let us consider what happens when the number of tasks in the process is a random quantity, M, with probability distribution $P(M = 0) = 0.9$ and $P(M = 50) = 0.1$, hence with $E(M) = 5$. In this case, equating (2.16) to 0.95 yields that we need 50 tasks tested without revealing a failure, which is indeed far fewer than for the corresponding deterministic case. This effect becomes even clearer if we had set $p = 0.9$ for the same situation, as the deterministic $m = 5$ would require a test of 45 successful tasks, but for M with the above distribution the requirement is trivially satisfied without any tests needed.

Example 3.2.2

Suppose that the tasks in the process after testing arrive as a homogeneous Poisson process with expected value $E(M) = \mu t = 10$, and that a reliability requirement is used in terms of a minimum probability of a failure-free period of length $t > 0$. Hence, as shown in Section 2.4.2, the probability of such a failure-free period, after testing n tasks revealed zero failures, is given by (2.17) for integer $\alpha > 0$ and by (2.18) for general values of $\alpha > 0$. This probability can be used to determine the minimum number of tasks to be tested, without revealing any failures, to satisfy a reliability requirement $P(FFP|(n, 0)) \geq p$ for any specified $p \in (0, 1)$. We denote this minimum test size for the Poisson case by $\tilde{n}$, and illustrate this procedure in Table 3.1.

For comparison, Table 3.1 also gives the minimum number of tasks, n, to be tested to achieve the same reliability requirement for the deterministic case with $m = 10$. We see that the required test sizes for the Poisson case are very close to the corresponding numbers for the deterministic case. As such, the Poisson case appears to be nearly as bad, in terms of required test sizes, as the deterministic case. It is perhaps surprising that the consequences, in terms of required test size, of these two quite extreme cases are so similar, but for practical reasons it is quite convenient as it implies that one does not have to worry too much about the actual distribution of M, as long as one has a good idea of its expected value. The most robust method then is to use the deterministic case, setting m equal to $E(M)$, with the added benefit that one hardly does any unnecessary tests if the actual process

is a homogeneous Poisson process. All values in Table 3.1 relate to $\beta = 0$. As both these criteria actually give a minimal number for $\beta + n$, values of both $\tilde{n}$ and n for other values of β follow immediately.

From this example we can conclude that such high reliability can only be achieved by testing many tasks, all leading to zero failures, or, alternatively, assuming quite a high value for β to represent prior beliefs. In particular when choosing α greater than one, many tests are needed to satisfy the reliability requirement.

p	α	$\tilde{n}$	n
0.95	1	190	191
	2	383	385
	3	577	579
0.99	1	990	991
	2	1983	1985
	3	2977	2979
0.995	1	1990	1991
	2	3983	3985
	3	5977	5979
0.999	1	9990	9991
	2	19983	19985
	3	29977	29979

Table 3.1: $\tilde{n}$ and n such that $P(FFP|M \sim \text{Po}(10), (\tilde{n}, 0)) \geq p$ and $P(0|m = 10, (n, 0)) \geq p$, with a Beta$(\alpha, 0)$ prior.

One could also use an α between zero and one, which would reduce the required test size. For example, if we set $p = 0.95$ then, for $\alpha = 0.5$ the minimal number of tests required will be $\tilde{n} = 93$ tasks, whereas for $\alpha = 0.1$, we need only $\tilde{n} = 16$ to achieve this level of reliability requirement, $p = 0.95$. However, it may often be preferable not to rely too strongly on optimistic prior judgements, hence not to choose α less than one nor to choose β large.

3.3 Multiple types of tasks

In this section we generalize the approach from the previous section to a system that has to perform multiple fully independent types of tasks after testing. Our goal in this section is to find the minimum total test effort required to meet a reliability criterion, in the sense of achieving a minimum required probability of a failure-free period in the process after testing. This is a special case of the problem studied in Section 3.4, where costs of testing are explicitly taken into account and where testing different types of tasks can have different costs. The special case considered in this section, which allows us to present and analyse important results, is relevant in situations where the costs involved in testing are relatively insignificant, and where one explicitly aims at low probability of failure for the process after testing. Throughout this section, we assume that the system must deal with $k \geq 2$ types of tasks, and we use notation as introduced in Section 2.4.1 for a process with multiple types of tasks. We should emphasize that all random quantities related to different types of tasks are assumed to be independent throughout our analysis.

Assuming a Beta(α_i, β_i) prior distribution of θ_i, for a task of type i, the posterior predictive joint probability of zero failures in m_i tasks of type i after zero-failure testing, for all $i = 1, \ldots, k$, is given by (2.10) for general values of $\alpha_i > 0$ and $\beta_i \geq 0$, and by (2.11) for integer values of $\alpha_i > 0$.

An interesting and intuitively logical result holds for situations with, so-to-say, exchangeable types of tasks, in the sense that $\alpha_i = \alpha > 0$ and $m_i = m > 0$ for all i. In such cases, the optimum real-valued z_i's, $z_i = \beta_i + n_i \geq 1$, are such that $|z_i - z_l| < 2$ for all $i, l \in \{1, \ldots, k\}$, where optimality is with regard to minimising the sum of the z_i's in order to achieve a minimal required value for $P(0|\underline{m}, (\underline{n}, \underline{0}))$. Even more, under a restriction $\sum_{i=1}^{k} z_i = kz$, for integer z, this probability $P(0|\underline{m}, (\underline{n}, \underline{0}))$ is maximised by taking $z_i = z$ for all i. Of course, these properties mean that effectively the same amount of information for each type of task is required in such situations. Mathematical justification of these properties is as follows. For clarity,

let us write $P(0|\underline{m}, (\underline{n}, \underline{0}))$ as

$$f(z_1, \ldots, z_k) = \prod_{i=1}^{k} \prod_{j=1}^{m} \frac{j + z_i - 1}{j + \alpha + z_i - 1}.$$

This function has the following two properties:

(a) f is maximised over all $\underline{z}$ with $\sum_{i=1}^{k} z_i = kz$, for an integer m, by taking $z_i = z$ for all i.

(b) For a given $p \in (0, 1)$, the requirement $f(z_1, \ldots, z_k) \geq p$ is achieved, for z_i with minimal $\sum_{i=1}^{k} z_i$, by a $\underline{z}$ with $|z_i - z_l| < 2$ for all $i, l \in \{1, \ldots, k\}$.

To prove property (a), first consider the case $k = 2$. It is straightforward to show that $f(z_1, z_2) = \prod_{j=1}^{m} \left[\left(\frac{j + z_1 - 1}{j + \alpha + z_1 - 1} \right) \left(\frac{j + z_2 - 1}{j + \alpha + z_2 - 1} \right) \right]$, with non-negative z_1 and z_2, and constraint $z_1 + z_2 = 2z$, for any non-negative integer z, is strictly maximised by $z_1 = z_2 = z$. The symmetric form of this function now implies that, for general $k \geq 2$, this property holds for any two z_i, z_l with $1 \leq i < l \leq k$. For $k \geq 3$, the product form of $f(z_1, \ldots, z_k)$ together with this result for the case $k = 2$, implies that, for any values of $z_3, \ldots, z_k$ and $z_1 \neq z_2$, with $z_1 + z_2 = 2z$, we have that $f(z, z, z_3, \ldots, z_k) > f(z_1, z_2, z_3, \ldots, z_k)$, and by symmetry the same applies again for any two z_i, z_l. This is sufficient for property (a) to hold, as for any $\underline{z}$ with $\sum_{i=1}^{k} z_i = kz$ one can increase the value of $f(z_1, \ldots, z_k)$ by such pairwise changes, until all $z_i = z$ (note that, at such intermediate steps towards the optimal $\underline{z}$, the z_i do not have to be integers).

Property (b) follows from reasoning similar to that used for property (a), as it is again possible to restrict attention to $k = 2$ and use the symmetry and the product form to get the general result. For property (b), however, we do not restrict attention to integer z_i or z. By the fact that $\left(\frac{j + z_1 - 1}{j + \alpha + z_1 - 1} \right) \left(\frac{j + 2z - z_1 - 1}{j + \alpha + 2z - z_1 - 1} \right) > 0, \forall j = 1, \ldots, m$, is concave as function of z_1 for given non-negative z, with maximum at $z_1 = z$, and by symmetry, $f(z_1, 2z - z_1)$ is increasing on $[0, z)$ and decreasing on $(z, 2z]$, as function of z_1, with maximum at $z_1 = z$. This implies that if $z_1 \geq z_2 + 2$, then $f(z_1 - 1, z_2 + 1) > f(z_1, z_2)$, and applying this pairwise argument repeatedly leads to property (b).

For the special case where $\alpha_i = \alpha$, with α a positive integer, $z_i = z$ and $m_i = m$

for all i, we can again easily find an approximation for the required value of z to meet a reliability requirement $P(0|\underline{m}, (\underline{n}, \underline{0})) \geq p$, with p close to 1, if z is large compared to α, assuming again that the system functions successfully for all tasks tested. From (2.11), and using the same logic used in the previous section, the minimal required value for z is close to

$$\frac{p^{1/(k\alpha)}}{1-p^{1/(k\alpha)}}m \approx \frac{p}{1-p}k\alpha m. \tag{3.3}$$

For the case with $\alpha = 1$ and $\beta_i = 0$, for all $i = 1, \ldots, k$, the expression in (3.3) gives the exact value of the optimal (real-valued) n_i's (see Section 3.4).

The result in (3.3) has an important implication when comparing two possibilities for such reliability demonstration for a system:

(A) one considers all possible tasks as exchangeable, so one does not wish to distinguish tasks of different types;

(B) one separates the possible tasks into $k \geq 2$ different types.

Suppose that, for case (B), one wishes to demonstrate reliability for the situation that m tasks are required for each type, so a total of km tasks, and for case (A) the total number of tasks (all of one type) considered for the process after testing is also km. Assume that p is close to 1, and that 'automatic' non-informative choice of hyperparameters for the Beta prior distributions in each case is used, say $\alpha > 0$ and $\beta \geq 0$ in case (A), and $\alpha_i = \alpha$ and $\beta_i = \beta$ for all $i = 1, \ldots, k$ in case (B). Moreover, for simplicity, assume that $\beta = 0$ has been chosen (but the same applies for other values of β as well). Then, (3.1) implies that, for case (A), we would need a total number of successful tests (with no failures observed) of approximately $\frac{p}{1-p}k\alpha m$. For case (B), from (3.3) we would need approximately the same number of successful tests for each type of task, so we would need approximately k times as many tests, hence approximately $\frac{p}{1-p}k^2\alpha m$ in total (as always without any failures) to meet the same reliability requirement. This is easily explained via the interpretation that the required number of tests without failures (per type of task) depends on how much evidence will effectively cancel out the imaginary number of failures α, and for case (B) we have effectively assumed $k\alpha$ such prior failures. This

problem is easily overcome by changing the parameters α_i used per type of task in case (B) to $\alpha_i = \alpha/k$, in which case the total number of successful tests required becomes approximately the same for cases (A) and (B). This is illustrated in Cases (a) (b) and (f) in Example 3.3.2. We regard this as a serious argument against 'automatically assumed non-informative prior distributions', which is in line with our repeated observation that these inferences are very sensitive to the choice of α.

If one wishes to choose a non-informative prior in such situations, we would support $\alpha = 1$ and $\beta = 0$ in case of a single type of task, and consistent with that we would also support the choice $\alpha_i = 1/k$ and $\beta_i = 0$, for all $i = 1, \ldots, k$, in case of k different types of tasks. Of course, one could equally well defend the use of any positive constant times these α's, e.g. taking $\alpha = k$ in case (A) and $\alpha_i = 1$ in case (B).

If, in case (B), the m_i's are not all the same, then we would also end up with a larger total number of tests required than for the corresponding case (A), if we would take all α_i's in case (B) the same as the α in case (A). Examples we computed suggest that, in this situation, taking the α_i's in case (B) such that they are proportional to the corresponding m_i's, and sum up to α (so $\alpha_i = \frac{m_i}{\sum_{i=1}^k m_i}\alpha$), will lead to a total number of required successful tests in case (B) which is nearly identical to the number required in case (A), e.g. see Cases (d) and (h) in Example 3.3.2. We have not achieved a full analytical justification for this.

For the case with $\alpha_i = \alpha$ and $\beta_i = \beta$, for all $i = 1, \ldots, k$, and a total number of tasks required in the process equal to $\sum_{i=1}^k m_i = km$, the largest total number of successful tests is required in the case $m_i = m$, for all $i = 1, \ldots, k$. Intuitively, we can explain this as follows. Compared to the situation where all m_i are equal, in which case we test equal numbers n_i, if, say, m_1 is reduced by the same number as m_2 is increased, then the overall reliability will be reached via a slightly larger probability of zero failures for tasks of type 1 than for tasks of type 2, which, due to the manner in which the optimal n_i depend on these probabilities of zero failures per type, then implies that we would still have to perform some extra tests of type 2, but far fewer of type 1 than was the case if m_1 and m_2 had been equal, thus reaching the same overall reliability with a saving on the total number of tests, e.g.

see Cases (b), (d) and (e) in Example 3.3.2.

The situation with $\alpha_i = 1$ and $\beta_i = 0$, for all $i = 1, \ldots, k$, but allowing different m_i for each type of task, is of interest as, for this choice of prior distributions, we can actually get close to an analytical solution for the optimisation problem in this section. This is presented in more detail for the more general case in Section 3.4, showing that in this situation the k-dimensional optimisation problem can be reduced to a 1-dimensional search problem, hence simplifying required computational effort, in particular for larger values of k. With these prior distributions, the optimal numbers of successful tests for each type, n_i, are approximately proportional to $\sqrt{m_i}$. For example, if we have $k = 2$ types of tasks in such a case, with $m_1 = 25$ and $m_2 = 100$, it would be optimal to test approximately twice as many tasks of type 2 than tasks of type 1 (of course, the total number of tests needed would depend on the reliability requirement p), see Cases (d) and (e) in Example 3.3.2.

With random quantities M_i representing the numbers of tasks to be completed in a specified period in the process after testing, the convexity argument used in Section 2.4.2 showed that the deterministic case, with known m_i, requires at least as many successful tests as are required for any random M_i with expected value $E(M_i) = m_i$, for each of these $P(FFP_i|(n_i, 0))$. When restricting attention to probability distributions with $E(M) = m$, one can always find a distribution which guarantees a failure-free period with probability p arbitrarily fixed in the range $0 < p < 1$, even without any tests to be performed, so with $n = 0$. This can be achieved by taking any probability distribution with $P(M = 0)$ greater than or equal to p, where the residual probability mass $1 - p$ can be put at such values that $E(M) = m$, in which case clearly the probability of a failure-free period exceeds p.

We illustrate the optimal test numbers, according to the problem criterion considered in this section, via two examples. First, a small example, to give a clear idea of how the optimal numbers of tests of different types of tasks are related to each other. Then a detailed example to illustrate all the cases discussed in this section.

Example 3.3.1

For $\underline{m} = (1, 2, 4, 9)$, $\alpha_i = 1, \beta_i = 0, \ \forall i = 1, \ldots, 4$, and for different required p, we can

aim to minimise the total number of tests, assuming they reveal no failures, such that the resulting predictive probability of zero failures in the process is at least p. The corresponding optimal test numbers are given in Table 3.2.

p	n_1	n_2	n_3	n_4
0.90	70	98	139	207
0.95	144	204	287	429
0.99	737	1042	1474	2209
0.995	1479	2091	2956	4433
0.999	9457	10553	13943	21505

Table 3.2: Minimal test numbers required for different p.

Table 3.2 illustrates that very high reliability, according to the problem criterion in this section, can only be achieved by very many zero-failure tests. All cases in Table 3.2 illustrate that the optimal n_i's are approximately proportional to $\sqrt{m_i}$. For example, it would be optimal to test approximately three times as many tasks of type 4 than tasks of type 1. For this setting, if tasks are assumed to arrive according to Poisson processes, with their expected numbers equal to the m_i used above, then the numbers of zero-failure tests required are indeed nearly identical to those in Table 3.2. If we increase the m_i, then the required test numbers increase by about the same factor, for example optimal testing for $\underline{m} = (10, 20, 40, 90)$, and $p = 0.90$, is achieved by tasking $\underline{n} = (699, 985, 1387, 2067)$. Of course, the required test numbers decrease substantially if we take a smaller value for the α_i. This is illustrated, for example in Table 3.1 and in the example below, Example 3.3.2, e.g. comparing Cases (b) with (f).

Example 3.3.2

In this example, we illustrate some of the results discussed in this section, namely the implication of (3.3) when comparing two possibilities for such reliability demonstration for a system (A), where all tasks are of one type, and for a system (B), where we distinguish between tasks of different types performed by the system. We performed calculations for a wider range of values of p than reported here, and for more combinations of values for the variables in this problem, all of which supported

the above results and insights similarly.

Table 3.3 gives the minimal required integer-valued numbers of tests in the form of a k-vector $\underline{n}$, for several cases under the zero-failure assumption. We assume that the total (expected) number of tasks that the system needs to deal with in the process after testing is 100, and the values β_i are equal to 0. All the cases reported are for the deterministic situation with regard to the required number of tasks to be dealt with in the process after testing, except for Cases (i)-(l) where the Poisson case is considered, with the expected number of tasks in the process denoted by λ_i for tasks of type $i = 1, \ldots, k$, or just λ in the situation of a single type of tasks. We consider the following cases:

(a) $k = 1$, $m = 100$, $\alpha = 1$
(b) $k = 2$, $m_1 = m_2 = 50$, $\alpha_1 = \alpha_2 = 1$
(c) $k = 4$, $m_i = 25$ and $\alpha_i = 1$ for $i = 1, \ldots, 4$
(d) $k = 2$, $m_1 = 25$, $m_2 = 75$, $\alpha_1 = \alpha_2 = 1$
(e) $k = 2$, $m_1 = 10$, $m_2 = 90$, $\alpha_1 = \alpha_2 = 1$
(f) $k = 2$, $m_1 = m_2 = 50$, $\alpha_1 = \alpha_2 = 0.5$
(g) $k = 4$, $m_i = 25$ and $\alpha_i = 0.25$ for $i = 1, \ldots, 4$
(h) $k = 2$, $m_1 = 25$, $m_2 = 75$, $\alpha_1 = 0.25$, $\alpha_2 = 0.75$
(i) $k = 2$, $m_1 = 10$, $m_2 = 90$, $\alpha_1 = 0.10$, $\alpha_2 = 0.90$
(j) $k = 1$, Poisson, $\lambda = 100$, $\alpha = 1$
(k) $k = 2$, Poisson, $\lambda_1 = \lambda_2 = 50$, $\alpha_1 = \alpha_2 = 1$
(l) $k = 2$, Poisson, $\lambda_1 = 25$, $\lambda_2 = 75$, $\alpha_1 = \alpha_2 = 1$

Cases (a), (b) and (c) illustrate that, by dividing the total number of tasks in the process, m, into equal numbers $m_i = m/k$ for k types, the number of tasks that need to be tested *per type* remains close to the total number needed if all tasks were assumed to be of one type, as long as we take the 'automatic' choice $\alpha_i = 1$ for all types. However, if we use $\alpha_i = \alpha/k$ together with $m_i = m/k$, with α the hyperparameter used in the case of all tasks being of a single type, then we need in total (about) the same number of tests as in the case with a single type, which is illustrated by comparing Cases (f) and (g) to Cases (a), (b) and (c). Cases (d)

Case	p	$\underline{n}$	$\sum_{i=1}^{k} n_i$
(a)	0.90	900	900
	0.95	1900	1900
	0.99	9900	9900
(b)	0.90	$(924, 925)$	1849
	0.95	$(1925, 1925)$	3850
(c)	0.90	$(936, 937, 937, 937)$	3747
	0.95	$(1937, 1937, 1937, 1938)$	7749
(d)	0.95	$(1319, 2270)$	3589
(e)	0.95	$(775, 2295)$	3070
(f)	0.90	$(450, 451)$	901
	0.95	$(950, 951)$	1901
(g)	0.90	$(225, 225, 226, 226)$	902
	0.95	$(475, 475, 476, 476)$	1902
(h)	0.90	$(226, 675)$	901
	0.95	$(476, 1425)$	1901
(i)	0.95	$(190, 1711)$	1901
(j)	0.90	900	900
	0.95	1900	1900
(k)	0.90	$(923, 924)$	1847
	0.95	$(1924, 1924)$	3848
(l)	0.95	$(1318, 2269)$	3587

Table 3.3: Minimal test numbers required in Cases (a)-(l).

and (e), when compared to Case (b), illustrate that $m_i = m/k$ represents the worst case (in terms of total number of tests needed) when using k types of tasks, and dividing the m tasks in the process over these k types. Cases (h) and (i), when compared to Cases (d) and (a), illustrate our suggestion that even when the m_i's are not equal, taking the α_i's proportional to the m_i's and such that they sum up to α, leads to (about) the same total number of tests required as for the case with only a single type. We got the same results in other examples, but have not managed a theoretical proof.

Cases (j), (k) and (l), when compared to Cases (a), (b) and (d), illustrate that the Poisson case is (nearly) as bad as the deterministic situation. Finally, Cases (d)

and (e), in which the m_i-values vary, and with $\alpha_i = 1$, illustrate that the minimal required number of tests per type is, for such cases, approximately proportional to the $\sqrt{m_i}$, as e.g. for Case (e) we have that $\sqrt{90}/\sqrt{10} = 3$ and $2295/775 = 2.961$.

3.4 Considering costs

So far in this chapter, we have not yet considered costs of testing, which are of particular interest for determining suitable test sizes for multiple types of tasks if these costs differ per type of task. Suppose that the cost of testing is linear in the number of tasks tested per type, and let $c_i > 0$ be the cost of testing a task of type i, for $i = 1, \ldots, k$. Then we may wish to minimise the total testing costs

$$\sum_{i=1}^{k} n_i c_i, \tag{3.4}$$

again assuming that all tests reveal no failures, such that a minimal required reliability level $p \in (0, 1)$ is achieved, that is $P(0|\underline{m}, (\underline{n}, \underline{0})) \geq p$ if $\underline{m}$ is known, and $P(FFP|(\underline{n}, \underline{0})) \geq p$ in case of random $\underline{M}$.

We can generalize this optimisation criterion by also including cost of failure during the process. Let $C \geq 0$ be the total costs incurred by a failure in the process, explicitly assuming that any such failure is catastrophic in the sense that functioning of the system in the process is ended on occurrence of any failure. Typically, such cost C may include considerations on possibly lost life, or environmental disaster, or lost trust in business relations, *etc.* For the sake of this analysis, we assume C to be a fixed constant, but we acknowledge that it may be difficult in practice to assign it a precise value. For this generalized situation, our objective is to minimise the total expected cost of testing and process failure,

$$EC(\underline{n}, \underline{c}, \underline{m}, C) = \sum_{i=1}^{k} n_i c_i + [1 - P(0|\underline{m}, (\underline{n}, \underline{0}))]C. \tag{3.5}$$

We can include the constraint $P(0|\underline{m}, (\underline{n}, \underline{0})) \geq p$, leading to a k-dimensional constrained optimisation problem, which needs to be solved numerically. Of course, in case of random $\underline{M}$ we again change the probability of zero failures in the process to the appropriate term, $P(FFP|E(\underline{M}) = \underline{m}, (\underline{n}, \underline{0}))$. The situation studied in Section

3.3 is the special case of this constrained problem with $C = 0$ and $c_i = 1$ for all $i = 1, \ldots, k$.

For the special case with Beta prior distributions with hyperparameters $\alpha_i = 1$ and $\beta_i = 0$, for all $i = 1, \ldots, k$, we can achieve an analytical result. The problem then becomes minimisation of

$$\sum_{i=1}^{k} n_i c_i + \left(1 - \prod_{i=1}^{k} \frac{n_i}{n_i + m_i}\right) C, \tag{3.6}$$

subject to

$$\prod_{i=1}^{k} \frac{n_i}{n_i + m_i} \geq p. \tag{3.7}$$

Of course, the solution should consist of positive integers n_i, but we study this problem as if real-valued numbers of zero-failure tests, say n_i^*, for all i, are allowed; the actual optimal solution, due to convexity, will then be found by comparing the value of this function for all integer k-vectors $\underline{n}$, which are 'neighbours' to the real-valued optimal solution, to choose the ones that give the minimum total expected costs and which still satisfy the constraint (3.7). The well-known Kuhn-Tucker (KT) conditions for this constrained optimisation problem are (see e.g. Luenberger [34] for the relevant mathematical theory of optimisation):

$$c_i - (C + \mu) \frac{m_i}{(n_i + m_i)^2} \prod_{l \neq i} \frac{n_l}{n_l + m_l} = 0, \tag{3.8}$$

for $i = 1, \ldots, k$, and either the constraint is active or the Lagrange multiplier $\mu = 0$. Note here that, if $C = 0$ (as in Section 3.3), the constraint will necessarily be active, so the latter term never disappears. Let us take $i, j \in \{1, \ldots, k\}$, with $i \neq j$, and let

$$D = (C + \mu) \prod_{l \neq i,j} \frac{n_l}{n_l + m_l}, \tag{3.9}$$

where we define the product term as one in the case $k = 2$. Then the KT conditions become

$$D \frac{m_i}{(n_i + m_i)^2} \frac{n_j}{n_j + m_j} = c_i, \tag{3.10}$$

$$D \frac{m_j}{(n_j + m_j)^2} \frac{n_i}{n_i + m_i} = c_j. \tag{3.11}$$

So that

$$\frac{n_j(n_j + m_j)}{n_i(n_i + m_i)} = \frac{c_i}{c_j} \frac{m_j}{m_i}. \tag{3.12}$$

As this holds for all i, j, we can introduce a constant $q > 0$ such that $n_i(n_i + m_i) = q\frac{m_i}{c_i}$, and consequently

$$n_i^* = \frac{1}{2}\left(-m_i + \sqrt{m_i^2 + 4q\frac{m_i}{c_i}}\right). \tag{3.13}$$

If we define the function $Q(q)$ by

$$Q(q) = \prod_{i=1}^{k}\left[\frac{-m_i + \sqrt{m_i^2 + 4q\frac{m_i}{c_i}}}{m_i + \sqrt{m_i^2 + 4q\frac{m_i}{c_i}}}\right], \tag{3.14}$$

then the constraint (3.7) is $Q(q) \geq p$. It is easy to show that $Q(\cdot)$ is a continuous and strictly increasing function of q, with $Q(0) = 0$ and $Q(q) \to 1$ for $q \to \infty$. This implies that (i) if the constraint is active at the optimal solution, we compute the value q^* for which $Q(q^*) = p$, which is a one-dimensional search problem, and (ii) if the optimum is achieved for a solution $\underline{n}$ at which the constraint is not active, the components of $\underline{n}$ as function of q, are given by (3.13), for some $q \in (q^*, \infty)$. So again, we can search for the solution by varying only the value of q.

Typically, if p is close to 1, q^* will be very large compared to the values m_i. Consequently, from (3.13), n_i is approximately equal to $\sqrt{\frac{qm_i}{c_i}}$, with $q = q^*$ if the constraint is active at the solution, and $q > q^*$ if not. Hence, in such situations, the ratio between two optimal numbers to be tested, n_i^*/n_j^*, is approximately equal to

$$\frac{\sqrt{\frac{m_i}{c_i}}}{\sqrt{\frac{m_j}{c_j}}}. \tag{3.15}$$

For the special case with $m_i = m$ and $c_i = c$ for all $i = 1, \ldots, k$, and the constraint active at the optimal solution, then from solving the equation $Q(q) = p$ which in this case is

$$\left[\frac{-m + \sqrt{m^2 + 4q\frac{m}{c}}}{m + \sqrt{m^2 + 4q\frac{m}{c}}}\right]^k = p, \tag{3.16}$$

it yields

$$q^* = cm\frac{p^{1/k}}{(1 - p^{1/k})^2}, \tag{3.17}$$

leading to optimal test numbers

$$n_i^* = m\left(\frac{p^{1/k}}{1 - p^{1/k}}\right), \tag{3.18}$$

which agrees with (3.3) for $\alpha_i = 1$, for all $i = 1, \ldots, k$.

Although it appears that c plays no role in this solution, its value relative to C determines whether or not the constraint will be active at the optimal solution. This method reduces this k-dimensional constrained optimisation problem to a relatively straightforward 1-dimensional search problem. Apart from significantly reducing the computational effort required to solve this problem, this also provides the following equation for optimal test numbers of types i and j:

$$\frac{n_j(n_j + m_j)}{n_i(n_i + m_i)} = \frac{c_i m_j}{c_j m_i},$$

for all $i, j \in \{1, \ldots, k\}$, where of course the equality can only be achieved for real-valued n_i^* and n_j^*, so will hold approximately for integer numbers of tests. This also implies that, for p close to 1, the optimal test numbers are approximately proportional to

$$\sqrt{\frac{m_i}{c_i}}, \tag{3.19}$$

where the optimal total number of tests $\sum_{i=1}^{k} n_i^*$, of course, depends on p and C. This result gives clear insight into the way that the m_i and the c_i influence the optimal numbers of tasks of each type to be tested, for this particular choice of hyperparameters $\alpha_i = 1$ and $\beta_i = 0$. It turns out that, in this situation, replacing the known numbers m_i again by Poisson distributed random quantities M_i; with $E(M_i) = m_i$, seems again to lead to approximately the same optimal results.

Although numerical examples showed us that such a relation also approximately tends to hold for other values of α_i and β_i, we have not been able to prove this more generally due to the complexity of the probability of zero failures involved. This is illustrated in Example 3.4.2.

It is worth mentioning here that for the special case with $C = 0$ and $\alpha_i = 1$, for $i = 1, \ldots, k$, we achieved a conjectured solution in a closed form for the real-valued $z_i = n_i + \beta_i$ which is,

$$\frac{pm_i\sqrt{c_i} + \sum_{j \neq i}^{k} \sqrt{pm_i m_j c_j}}{(1-p)\sqrt{c_i}}.$$

This solution, which we could not prove for general k due to the complexity of the probability of zero failures involved, gave precise answers to all the numerical

examples that were explored. For $k = 2$ and $k = 3$, this conjectured solution can be obtained by using the first-order KT conditions, although this is somewhat tedious. Moreover, it is easy to see that, if true, this conjecture would agree with the reliability constraint (3.7), which for this case is an active constraint.

Example 3.4.1

In this example we illustrate the method discussed in this section, both for the deterministic (Cases (1)-(3)) and the Poisson cases (Case (4)). Throughout, we take all the $\alpha_i = 1$ and $\beta_i = 0$. The results are presented in Table 3.4. As our interest is explicitly in the required test numbers, we do not report the associated optimal total expected costs. We consider the following cases:

(1) $k = 3$, $\underline{m} = (1, 3, 6)$, $\underline{c} = (20, 50, 50)$, $C = 10,000$

(2) $k = 3$, $\underline{m} = (1, 3, 6)$, $\underline{c} = (20, 50, 50)$, $C = 1,000,000$

(3) $k = 4$, $\underline{m} = (1, 2, 4, 9)$, $\underline{c} = (1, 5, 1, 5)$, $C = 0$

(4) $k = 4$, Poisson, $\underline{\lambda} = (1, 2, 4, 9)$, $\underline{c} = (1, 5, 1, 5)$, $C = 0$

Case	p	$\underline{n}$
(1)	0.90	(71, 78, 109)
	0.95	(148, 161, 227)
	0.99	(758, 828, 1170)
(2)	0.90	(219, 239, 337)
	0.95	(219, 239, 337)
	0.99	(758, 828, 1170)
(3)	0.90	(122, 76, 242, 160)
	0.95	(250, 158, 500, 332)
(4)	0.90	(121, 75, 241, 159)
	0.95	(249, 157, 499, 331)

Table 3.4: Optimal test numbers required in Cases (1)-(4).

In the cases presented in Table 3.4, the 'probability of zero failures' constraint (3.7) is active in all situations, except in Case (2) for $p = 0.90$ and $p = 0.95$; in these two situations that probability, at the optimum $\underline{n}$, is actually equal to 0.966. In relation to Cases (1) and (2), we also calculated the optimum solutions for

$C = 100,000$, which gave precisely the same solutions as Case (1), as the constraint was still active for this value of C, for all these values of p. In any situation, for all values of C such that the constraint is active, the optimal solution is identical. Once the constraint is not active anymore, due to a relatively large value of C, the optimum starts to change with changing C.

All cases in this example illustrate the presented approximate relationship in (3.19) between the optimal n_i, m_i and c_i, as can be confirmed easily. Cases (3) and (4) allow comparison between the deterministic and Poisson cases, again the difference between these situations is very small, with the deterministic case requiring most tasks to be tested.

Example 3.4.2

In this example we illustrate the optimal zero-failure tests for the Cases (1), (3) and (4) stated in Example 3.4.1, but for different values of α_i and β_i. Table 3.5 illustrates the fact that the relationship between n_i, m_i and c_i in (3.19) also appears to hold for values of α_i and β_i other than $\alpha_i = 1$ and $\beta_i = 0$, for all $i = 1, \ldots, k$. Moreover, it illustrates that, in our setting, where we assume no failures in the tests, the choice of β_i do not have much impact on n_i. For example, comparing the first column in Table 3.5 with Table 3.4 and the second column with the third column in Table 3.5, shows that increasing β_i by 1 decreases n_i by 1, in order to demonstrate the required level of reliability. Table 3.5 also illustrates the great impact that the choice of α_i has on n_i, since the reliability demonstration requires n_i tests without failures to effectively counter the prior information of α_i 'imaginary test failures'.

Case	p	$\underline{n}$			
		$\alpha_i = 1,\ \beta_i = 1$	$\alpha_i = 0.5,\ \beta_i = 0$	$\alpha_i = 0.5,\ \beta_i = 1$	$\alpha_i = 2,\ \beta_i = 0$
(1)	0.90	(70, 77, 108)	(34, 38, 54)	(33, 37, 53)	(143, 156, 221)
	0.95	(147, 160, 226)	(75, 80, 112)	(74, 79, 111)	(297, 323, 456)
	0.99	(757, 827, 1169)	(380, 413, 584)	(379, 412, 583)	(1515, 1657, 2343)
(2)	0.90	(218, 238, 336)	(156, 170, 239)	(155, 169, 238)	(308, 336, 474)
	0.95	(218, 238, 336)	(156, 170, 239)	(155, 169, 238)	(308, 336, 474)
	0.99	(757, 827, 1169)	(380, 413, 584)	(379, 412, 583)	(1515, 1657, 2343)
(3)	0.90	(121, 75, 241, 159)	(61, 38, 119, 78)	(60, 37, 118, 77)	(243, 153, 486, 323)
	0.95	(249, 157, 499, 331)	(125, 79, 248, 164)	(124, 78, 247, 163)	(501, 316, 1002, 668)
(4)	0.90	(120, 74, 240, 158)	(61, 37, 121, 77)	(60, 36, 120, 76)	(242, 152, 484, 321)
	0.95	(248, 156, 498, 330)	(125, 78, 250, 163)	(124, 77, 249, 162)	(500, 314, 1001, 667)

Table 3.5: Optimal test numbers required in Cases (a)-(c) for different values of α_i and $\beta_i = 1$.

3.5 Concluding remarks

In this chapter, we analysed optimal test numbers for a single type of task, and for multiple types of tasks (with full independence between different types), including possible costs of testing and process failure for the latter situation. Under the assumption that testing reveals zero failures, our interest was explicitly in a failure-free period, of a given length of time, for the process after testing. We studied the Bayesian approach to this problem, using conjugate Beta(α_i, β_i) prior distributions for the probability of a failure for type $i = 1, \ldots, k$. Special attention was given to the effect of the choices of the hyperparameters α_i and β_i on the required numbers of successful tests. Throughout, the emphasis has been on reliability *demonstration*, which implies that we assume that the test results must be strong enough to predict high reliability with very little information added via the prior distribution. For reliability demonstration, we favour a small value of α_i, and $\beta_i = 0$. For testing a single type of task, we advocate the choice $\alpha = 1$. If there are $k \geq 2$ types, then using this same value for all α_i will lead to more tests required, in the worst case, in terms of the number of tests, about k times as many as when all tasks where

assumed to be of the same type. From this perspective, taking the α_i's such that they sum up to 1, but are in the same proportions as the m_i's, with m_i the required number of tasks of type i to be dealt with in the process after testing, gets us back to about the same total number of tests needed as when all tasks are assumed to be of a single type. This emphasizes possible problems with an 'automatic' choice of non-informative prior, which is often advocated as an advantage of Bayesian statistics. So, we conclude that careful attention is required to the choice of the prior distribution in case of reliability demonstration with zero failures in the tests, in particular because the required test numbers are very sensitive to the choice of the α_i's.

For the setting in this chapter, we have identified a few 'worst case' scenarios in the sense of requiring the largest number of successful tests in order to achieve a specified reliability level. These are: (1) Deterministic numbers m_i of tasks needed to be dealt with in the process after testing, are worse than random numbers M_i with $E(M_i) = m_i$, when expressing reliability in terms of probability of zero failures in the process. Poisson distributions for M_i are almost equally bad as the deterministic case. (2) Distinguishing tasks into more different types (assuming independence between the types) is worse than regarding them all as a single type, *if* one uses the same 'automatically chosen' values $\alpha_i = \alpha$ for all types as one would use for a single type. In this same situation of 'automatically chosen' α_i's, we also have that: (3) If in total mk tasks appear in the process, with k types of tasks, most testing is needed if $m_i = m$ for all $i = 1, \ldots, k$. The disadvantages mentioned in these insights (2) and (3) disappear if we adapt the α_i's by taking them proportional to the m_i's, with the sum of the α_i's remaining at the value one would take in case of a single type.

In the next chapter we report on a study of reliability demonstration from similar perspectives as in this chapter, but where we assume that process failures are not catastrophic, in the sense that they do incur a cost but allow continued use of the system in the process. There, we study how optimal test numbers, again assuming no failures in testing, depend on the costs involved, and we also include constraints on testing opportunities in terms of budget and time available for testing. Expand-

ing the approach presented here to include constraints on testing would also be of possible interest, we leave this as a topic for future research.

Chapter 4

Reliability demonstration for non-catastrophic failures

4.1 Introduction

In this chapter we introduce optimal testing of a technical system in order to demonstrate reliability with regard to its use in a process after testing, where the system has to function for different types of tasks, which we assume to be independent. We assume that process failures are not catastrophic in the sense that they allow continued use of the system in the process, but do incur a cost. Throughout this chapter, reliability demonstration is formulated in terms of minimisation of the total expected costs of testing and process failures. It is assumed that such costs are linear both in the numbers of tasks tested per type and in the expected number of failures per type in the process. It is also assumed that the system's functioning remains unchanged during the period over which the process is considered, in the sense that it does not suffer from wear-out and is not repaired. We explicitly assume that testing reveals zero failures. The optimal numbers of tasks to be tested are derived by minimisation of a cost function, taking into account the costs of testing and of failures in the process after testing. We focus on study of the optimal numbers of tests for different types of tasks, depending on the numbers of tasks in the process and the costs involved. We briefly compare the results of this chapter with optimal test numbers in the setting presented in Chapter 3, where the optimality criterion

was more suitable for catastrophic failures. For these two different optimality criteria, the dependence of the optimal numbers to be tested for different types of tasks, on the costs of testing per type, and on the number of tasks per type in the process after testing, turns out to be very similar.

The focus is explicitly on the system's required performance, during a specified period after testing, with the number of tasks that the system has to perform during that period either random quantities or known in advance. We study how optimal test numbers, again assuming no failures in testing, depend on the costs involved. We also include constraints on testing opportunities in terms of budget and time available for testing.

Throughout this chapter, we use the setting introduced in Chapter 2, and specifically in Section 2.5. We particularly consider tasks that arrive as Poisson processes with known arrival rates. We are interested in the required numbers of tasks, of each type, to be tested, aiming at minimisation of the total expected costs of testing and process failures, subject to both testing budget and time constraints.

In Section 4.2, optimal testing is considered in the sense of minimal total expected costs, with constraints on budget and time for testing. In Section 4.3, we present the main results, with a brief discussion, when the reliability demonstration problem is formulated in terms of minimisation of the total expected number of failures subject to a constraint on the total number of tests allowed before the start of the process. We consider the problem in this section as a special case of Section 4.2, however, we believe that it is of specific interest in providing an optimal test (ignoring costs) towards an interesting given reliability requirement, namely the maximum expected number of failures allowed in the process after testing. In Section 4.4 some main results on optimal test numbers per type of task according to the approaches in Chapters 3 and 4 are compared. This leads to a useful conclusion that, for both optimality criteria, the optimal numbers of tasks per type to be tested depend similarly on the expected number of tasks in the process and on the different costs of testing per type. Some concluding remarks are given in Section 4.5.

4.2 Minimising total expected costs

In this section we consider the optimal number of tasks to be tested, assuming that the system will only be released for use in the process after testing, if testing revealed no failures, for a system of $k \geq 1$ independent types of tasks. We aim at minimisation of the total expected costs, that is the costs of testing and the expected costs of failures in the process during the period of length $t > 0$ after testing, under constraints on budget and time for testing. We assume that the costs of testing are linear in the number of tests per type of task, and that failure in the process is not catastrophic, in the sense that the process continues after failure as before, yet such failure does incur a (possibly high) cost. In this section, we will particularly consider tasks that arrive as Poisson processes with known arrival rates after testing, but the same general results hold for other processes since we only take the expected number of failures in the process into account. Poisson processes have the advantage that the effect of the length $t > 0$ of the process considered, on the optimal numbers to be tested, can be studied explicitly.

For tasks of type i, for $i = 1, \ldots, k$, let n_i be the number of tasks to be tested, and let $\lambda_i > 0$ be the arrival rate in the process after testing, which is assumed to be known. The cost of testing one task of type i is denoted by $c_i > 0$, and the time needed for one such test is $t_i > 0$. A failure in the process, for a task of type i, costs $f_i > 0$, and the total budget and time available for testing are $B > 0$ and $T > 0$, respectively. The total expected number of failures in such a process during a period of length t immediately after testing, is presented in (2.21).

The optimisation problem considered here is minimisation, by choice of n_i, for $i = 1, \ldots, k$, of the total expected costs of failures in the process and the cost of testing,

$$t \sum_{i=1}^{k} \frac{f_i \alpha_i \lambda_i}{\alpha_i + \beta_i + n_i} + \sum_{i=1}^{k} n_i c_i, \tag{4.1}$$

subject to the cost and the time constraints on testing,

$$\sum_{i=1}^{k} n_i c_i \leq B, \tag{4.2}$$

$$\sum_{i=1}^{k} n_i t_i \leq T, \tag{4.3}$$

We analyse this problem as if n_i is not restricted to be an integer. The actual solution is then easily found by considering all 'nearest' integers n_i which satisfy the constraints, because the function to be minimised is convex, which is easily confirmed as its Hessian matrix is positive definite with respect to n_i, for $i = 1, \ldots, k$.

This convexity, together with the linearity of the two constraints as functions of n_i, implies that the necessary first-order Kuhn-Tucker conditions for the solution of a constrained minimisation problem are actually sufficient conditions [34]. As our main interest is in the testing requirements, we do not consider the corresponding values of the object function in detail. However, as this function is fairly flat around the real-valued optimum, rounding to a 'nearest' integer solution would not have much effect on the total expected costs. The solution of this optimisation problem depends on whether or not there are active constraints, and is derived via the Kuhn-Tucker conditions. The first-order necessary conditions are

$$\frac{-t f_i \alpha_i \lambda_i}{(\alpha_i + \beta_i + n_i)^2} \; + \; c_i \; + \; \mu_c c_i \; + \; \mu_t t_i = 0, \quad i = 1, \ldots, k, \tag{4.4}$$

$$\mu_c \left(\sum_{i=1}^{k} n_i c_i - B \right) = 0, \tag{4.5}$$

$$\mu_t \left(\sum_{i=1}^{k} n_i t_i - T \right) = 0, \tag{4.6}$$

$$\mu_c \geq 0, \text{and } \mu_t \geq 0,$$

where, μ_c and μ_t are the Lagrange multipliers [34], and of course, we also restrict to $n_i \geq 0$. We present and discuss the solutions to the four possible combinations of active and inactive constraints (4.2) and (4.3). The real-valued optimal n_i will be denoted by n_i^*, with additional superscripts to indicate the corresponding active constraints. When rounding n_i^* to an integer-valued solution, the constraints must still be satisfied. However, for the integer-valued solution a small amount of the budget or the time may be left unused, depending on which of the constraints are active. These results are illustrated, and discussed in more detail, in Example 4.2.1.

1. Cost and time constraints active ($\mu_c, \mu_t \geq 0$):

If both constraints, (4.2) and (4.3), are active, then the set of equations (4.4), (4.5) and (4.6) becomes

$$\begin{aligned} \frac{-tf_i\alpha_i\lambda_i}{(\alpha_i+\beta_i+n_i)^2} \; + \; c_i \; + \; \mu_c c_i \; + \; \mu_t t_i \;\; &= \;\; 0, \qquad i = 1, \ldots, k, \\ \sum_{i=1}^{k} n_i c_i - B \;\; &= \;\; 0, \\ \sum_{i=1}^{k} n_i t_i - T \;\; &= \;\; 0, \\ \mu_c \geq 0, \;\; \mu_t \geq 0. & \end{aligned}$$

By solving these equations, the optimal real-valued n_i, which we denote by n_i^{*ct}, for $i = 1, \ldots, k$, are functions of Lagrange multipliers μ_c and μ_t, corresponding to cost and time constraints, respectively

$$n_i^{*ct} = \sqrt{\frac{tf_i\alpha_i\lambda_i}{(\mu_c+1)c_i + \mu_t t_i}} - (\alpha_i + \beta_i), \tag{4.7}$$

where μ_c and μ_t can be derived from the equations

$$\sum_{j=1}^{k} c_j \sqrt{\frac{f_j\alpha_j\lambda_j}{(\mu_c+1)c_j + \mu_t t_j}} = \frac{B + \sum_{j=1}^{k}(\alpha_j+\beta_j)c_j}{\sqrt{t}} \tag{4.8}$$

and

$$\sum_{j=1}^{k} t_j \sqrt{\frac{f_j\alpha_j\lambda_j}{(\mu_c+1)c_j + \mu_t t_j}} = \frac{T + \sum_{j=1}^{k}(\alpha_j+\beta_j)t_j}{\sqrt{t}}. \tag{4.9}$$

The solutions n_i^{*ct} depend on t in a way that may not be directly clear from these equations, since the Lagrange multipliers μ_c and μ_t also depend on t. These solutions will vary only a little with increasing t. The only change is due to the presence of the term '+1' in the denominators $(\mu_c+1)c_j + \mu_t t_j$ under the square-roots in these equations, as this prevents the μ_c and μ_t corresponding to the optimal solution to be linear in t, and therefore they remain dependent on the individual values c_j and t_j. However, for reasonably large t this minor effect becomes neglectable, and the real-valued solutions n_i^{*ct} hardly change anymore with increasing t, hence changes in the integer solutions become very unlikely. For Case 3, with only the time constraint active, we will see that a similar effect occurs, but stronger than in this case.

The solutions for situations with one or two inactive constraints are derived by setting the corresponding Lagrange multipliers equal to zero, and deleting the equations related to the inactive constraints.

2. Cost constraint active, time constraint inactive:

If the cost constraint (4.2) is active but the time constraint(4.3) is inactive, so we use the entire test budget but there is sufficient time available for testing, then from (4.7) and (4.8), with $\mu_c \geq 0$ and $\mu_t = 0$, the real-valued solution to our optimisation problem is

$$n_i^{*c} = \sqrt{\frac{f_i \alpha_i \lambda_i}{c_i}} \left[\frac{B + \sum_{j=1}^{k} (\alpha_j + \beta_j) c_j}{\sum_{j=1}^{k} \sqrt{f_j \alpha_j \lambda_j c_j}} \right] - (\alpha_i + \beta_i), \tag{4.10}$$

with corresponding Lagrange multiplier

$$\mu_c = t \left[\frac{\sum_{j=1}^{k} \sqrt{f_j \alpha_j \lambda_j c_j}}{B + \sum_{j=1}^{k} (\alpha_j + \beta_j) c_j} \right]^2 - 1.$$

This optimal n_i^{*c} does not depend on t, but the Lagrange multiplier corresponding to this solution does depend on t, implying that the benefit of relaxing the active constraint (testing budget) a little, on the total expected costs, increases with t.

3. Cost constraint inactive, time constraint active:

If the cost constraint (4.2) is inactive but the time constraint(4.3) is active, so optimal testing is restricted by the time available but not by the test budget, then $\mu_c = 0$ and $\mu_t \geq 0$, and from (4.7) and (4.9), the real-valued solution to this optimisation problem is

$$n_i^{*t} = \sqrt{\frac{t f_i \alpha_i \lambda_i}{c_i + \mu_t t_i}} - (\alpha_i + \beta_i), \tag{4.11}$$

and the Lagrange multiplier μ_t can be derived from the equation

$$\sum_{j=1}^{k} t_j \sqrt{\frac{f_j \alpha_j \lambda_j}{c_j + \mu_t t_j}} = \frac{T + \sum_{j=1}^{k} (\alpha_j + \beta_j) t_j}{\sqrt{t}}. \tag{4.12}$$

This solution varies again as a function of t in the following manner, where we assume that on increasing t the cost constraint remains inactive. The Lagrange multiplier μ_t is uniquely determined by (4.12), and is an increasing function of

t. So, for smaller values of t for which this case applies, n_i^{*t} is relatively more influenced by its approximate inverse proportionality to $\sqrt{c_i}$. Whereas for larger t, n_i^{*t} is relatively more influenced by its approximate inverse proportionality to $\sqrt{t_i}$. This implies that for smaller t we should test more of the tasks which are cheap to test, while for large t we test more of the tasks which take less time to test, and costs of testing become less relevant with increasing t, which is intuitively logical due to the influence of the testing costs on the total expected costs. Of course, such a shift from cheap tests to short tests, for increasing t, is only intuitively clear when the cost constraint remains inactive. This is a similar, but stronger, effect as we briefly discussed in Case 1, where both constraints active. This might be clearer when the time constraint becomes active (keeping the cost constraint inactive) at $t = \left[\frac{T+\sum_{j=1}^{k}(\alpha_j+\beta_j)t_j}{\sum_{j=1}^{k} t_j\sqrt{f_j\alpha_j\lambda_j/c_j}}\right]^2$, with $\mu_t = 0$, (see (4.15) in Case 4) for which there will be a change from the case with no active constraint (Case 4) to this case. At this value of t, the optimal real-valued n_i^{*t} becomes equal to (4.17), which is more influenced by its approximate inverse proportionality to $\sqrt{c_i}$ than to $\sqrt{t_i}$.

If t increases to infinity, with the time constraint remaining active and the cost constraint remaining inactive, then the limiting solution can be found by minimising the total expected costs due to process failures, without taking the testing costs $\sum_{i=1}^{k} n_i c_i$ into account, as the expected costs due to failures increases to infinity (as a technical detail we should remark that the optimal solution converges as it becomes independent of t in the limiting situation). The optimal test numbers in this limiting situation are

$$n_i^{\infty t} = \sqrt{\frac{f_i\alpha_i\lambda_i}{t_i}}\left[\frac{T+\sum_{j=1}^{k}(\alpha_j+\beta_j)t_j}{\sum_{j=1}^{k}\sqrt{f_j\alpha_j\lambda_j t_j}}\right] - (\alpha_i+\beta_i), \tag{4.13}$$

so $n_i^{\infty t}$ is approximately inversely proportional to $\sqrt{t_i}$. The Lagrange multiplier corresponding to $n_i^{\infty t}$ is infinite. For large but finite t, in this case, the Lagrange multiplier is approximately

$$\mu_t \approx t\left[\frac{\sum_{j=1}^{k}\sqrt{f_j\alpha_j\lambda_j t_j}}{T+\sum_{j=1}^{k}(\alpha_j+\beta_j)t_j}\right]^2,$$

which can also be simply derived from (4.12), with $c_j = 0$, for $j = 1, \ldots, k$.

4. No active constraints:

If both constraints are inactive, the budget and time constraints do not affect optimal testing any more, i.e. $\mu_c = \mu_t = 0$ and the equations related to both constraints, (4.5) and (4.6) are deleted. Hence, the k equations of (4.4), or simply substituting $\mu_c = \mu_t = 0$ in (4.7), gives the real-valued solution to this optimisation problem as

$$n_i^* = \sqrt{\frac{t f_i \alpha_i \lambda_i}{c_i}} - (\alpha_i + \beta_i). \tag{4.14}$$

This situation occurs for small values of t, when both constraints are inactive, so if

$$t \leq \min \left\{ \left[\frac{B + \sum_{j=1}^{k} (\alpha_j + \beta_j) c_j}{\sum_{j=1}^{k} \sqrt{f_j \alpha_j \lambda_j c_j}} \right]^2, \left[\frac{T + \sum_{j=1}^{k} (\alpha_j + \beta_j) t_j}{\sum_{j=1}^{k} t_j \sqrt{f_j \alpha_j \lambda_j / c_j}} \right]^2 \right\}. \tag{4.15}$$

If the process after testing is only considered over a relatively short period, the optimal solution may not use the entire test budget and time, which is intuitively logical. However, this does not appear in the more traditional approaches to reliability demonstration, referred to in Chapter 1, as there the reliability or cost targets are not formulated predictively, but only consider estimates of summaries (e.g. quantiles) of underlying probability distributions, the interpretation and use of which are often hard to understand.

Intuitively, for larger t one would expect to test more as long as the two constraints are still satisfied (testing budget is still not totally used and the allowed testing time has not elapsed yet). Therefore, by increasing time t to reach the value $\left[\frac{B + \sum_{j=1}^{k} (\alpha_j + \beta_j) c_j}{\sum_{j=1}^{k} \sqrt{f_j \alpha_j \lambda_j c_j}} \right]^2$, when this is the minimum in (4.15), where the whole budget is used, $\mu_c = 0$ but the allowed testing time has not elapsed yet, then we better test the closest integer combination to ($\forall\ i = 1, 2, \ldots, k$)

$$n_i^{*\underline{c}} = \sqrt{\frac{f_i \alpha_i \lambda_i}{c_i}} \left[\frac{B + \sum_{j=1}^{k} (\alpha_j + \beta_j) c_j}{\sum_{j=1}^{k} \sqrt{f_j \alpha_j \lambda_j c_j}} \right] - (\alpha_i + \beta_i), \tag{4.16}$$

which is equal to n_i^{*c} as given by (4.10).

On the other hand, by increasing time t to reach the value $\left[\frac{T + \sum_{j=1}^{k} (\alpha_j + \beta_j) t_j}{\sum_{j=1}^{k} t_j \sqrt{f_j \alpha_j \lambda_j / c_j}} \right]^2$, where this is now the minimum in (4.15), where the total testing time has elapsed before finishing the available testing budget and $\mu_t = 0$, then it is optimal to test

$$n_i^{*\underline{t}} = \sqrt{\frac{f_i \alpha_i \lambda_i}{c_i}} \left[\frac{T + \sum_{j=1}^{k} (\alpha_j + \beta_j) t_j}{\sum_{j=1}^{k} t_j \sqrt{f_j \alpha_j \lambda_j / c_j}} \right] - (\alpha_i + \beta_i). \tag{4.17}$$

Logically, increasing time t could lead to the situation where the two constraints are active and equal, which is not very likely to happen, then

$$t = \left[\frac{B + \sum_{j=1}^{k}(\alpha_j + \beta_j)c_j}{\sum_{j=1}^{k}\sqrt{f_j\alpha_j\lambda_j c_j}}\right]^2 = \left[\frac{T + \sum_{j=1}^{k}(\alpha_j + \beta_j)t_j}{\sum_{j=1}^{k} t_j\sqrt{f_j\alpha_j\lambda_j/c_j}}\right]^2, \tag{4.18}$$

and $\mu_c = \mu_t = 0$. Therefore, (4.16) and (4.17) become identical, hence a closed form of n_i^{*ct} would be derived.

To solve an optimisation problem as considered here, we should check which constraints are active via the condition on t shown in Case 4, or just find the optima for all four cases above, and compare the corresponding expected costs to find the solution. In all situations, we will need to round off the real-valued solutions to integer-valued solutions, where we should be careful that the constraints must be satisfied. For an integer-valued solution, a part of the test budget may be left unused, even if the cost constraint is active for the corresponding real-valued solution, but this amount not used will be smaller than the maximum of the c_i, $i = 1, \ldots, k$, and a similar fact holds of course for the t_i if the time constraint is active at the real-valued solution. For example, if c_1 is very large compared to the other c_i, the cost constraint can be active only in the sense of not allowing an unconstrained optimal value for n_1, but allowing all other n_i to take their unconstrained optimal values.

If the tasks in the process do not arrive as Poisson processes, the analysis in this section remains effectively the same, as we only use the expected numbers of tasks in the process. Of course, throughout the analysis, $t\lambda_i$ would be replaced by the expected number of tasks of type i in the process during a period of length t immediately after testing. However, the dependence of the optimal solution on t might become far more complex than in the analysis above, in particular of course if the tasks arrive as non-homogeneous Poisson processes [2], with arrival rates not constant over time.

All the optimal n_i^*'s, and therefore also the corresponding integer-valued solutions, imply that these optimal numbers of tests are in constant ratio to each other, independent of t, and are approximately proportional to $\sqrt{f_i}$, $\sqrt{\alpha_i}$, $\sqrt{\lambda_i}$, and to $\left(\sqrt{c_i}\right)^{-1}$, and, if the time constraint is active, to $\left(\sqrt{t_i}\right)^{-1}$. The optimal n_i's are af-

fected by the choice of β_i such that, if we increase β_i by 1, the optimal n_i decreases (approximately, in the case where only the cost constraint is active) by 1. Clearly, the optimal n_i are much more sensitive to the value of α_i than to the value of β_i.

Example 4.2.1

Suppose we have $k = 4$ fully independent types of tasks arriving according to Poisson processes during a period of length t after testing. The relevant parameters are given in Table 4.1.

Type i	λ_i	c_i	t_i	f_i
1	1	1	1	10
2	2	5	1	10
3	4	1	4	5
4	9	5	4	1

Table 4.1: Parameters for 4 types of tasks.

Throughout this example, we use Beta prior distributions for the failure probability of type i with $\alpha_i = 1$ and $\beta_i = 0$, for all $i = 1, \ldots, 4$. Table 4.2 gives the optimal numbers to be tested, depending on the length t of the process after testing, with $B = 1000$ and $T = 500$.

t	n_1	n_2	n_3	n_4	$\sum n_i c_i$	$\sum n_i t_i$
10	9	5	13	3	62	78
100	31	19	44	12	230	274
324	56	35	79	23	425	499
325	56	35	79	23	425	499
700	67	49	67	29	524	500
5,000	71	85	53	33	714	500
100,000	70	98	50	33	775	500

Table 4.2: Optimal test numbers with $B = 1000$ and $T = 500$.

The time constraint becomes active at $t = 322.09$, in the sense that it affects the real-valued solution to this optimisation problem. The first integer value of t for which the integer-valued solution to this optimisation problem is affected by the active time constraint is $t = 325$, for which the unconstrained integer-valued solution would be $\underline{n} = (56, 35, 80, 23)$. For the limiting case $t \to \infty$, the cost constraint remains inactive, and the real-valued optimal limiting solution, from (4.13), is $\underline{n}^{\infty t} = (70.43, 100.02, 49.51, 32.88)$. This illustrates that, with only the time constraint active, increasing t leads to a shift towards testing more tasks with small t_i-values, as the influence of the costs of testing on the total expected costs decreases. This allows in particular more tasks of type 2 to be tested, for which the corresponding failure is relatively expensive, and which arrive at a rate that is twice the arrival rate of tasks of type 1.

If we take the same parameters as above, but set $T = 2000$, then the cost constraint becomes active at $t = 1728.33$, and the time constraint remains inactive for all t. In this case, the optimal real-valued solution n_i^{*c}, given by (4.10), remains constant as function of t when the cost constraint is active, and the corresponding integer-valued solution also remains constant, namely $\underline{n} = (130, 82, 185, 55)$. Hence, with the cost constraint active, we tend to test relatively more of the tasks that are cheap to test, i.e. tasks of types 1 and 3.

Example 4.2.2

Suppose that we have the same information considered in Example 4.2.1, but with $B = 425$ instead. Table 4.3 illustrates that, for values of T and B such that both constraints are active, the real-valued solutions change very little (only in a few decimals), while the integer-valued solutions tend not to change over all the t values for which the constraints are active. In the real-valued optimisation problem, both constraints are active for $t \geq 323$. However, for the corresponding integer-valued solutions, which remain constant, namely $\underline{n} = (56, 35, 79, 23)$, for $t \geq 320$, the test budget is fully used while one unit of time remains unused.

t	n_1^*	n_2^*	n_3^*	n_4^*	$\sum n_i^* c_i$	$\sum n_i^* t_i$
10	9.001	5.325	13.142	3.243	64.983	79.866
100	30.623	19.000	43.721	12.416	231.424	274.171
300	53.772	33.641	76.460	22.238	409.627	482.205
320	55.568	34.777	79.001	23.000	423.454	498.349
323	55.774	34.910	79.242	23.087	425.001	500.000
100,000	55.770	34.910	79.243	23.087	424.998	500.000

Table 4.3: Optimal real-valued numbers of tests, with $B = 425$ and $T = 500$.

4.3 Minimising total expected number of failures

The general problem formulation in Section 4.2 has several special cases which could be of practical interest. For example, if one or both of the constraints are not relevant, clearly this can be taken into account by taking B or T very large, reducing the possible combinations of active constraints that need to be considered. An interesting special case occurs when one sets all $f_i = 1$, $c_i = 0$, $t_i = 1$ and $T = n_{max}$, in which case the above problem represents optimal testing in the sense of minimising the total expected number of failures in the process, during a period of length t after testing, such that the total number of tests should not exceed n_{max}. Clearly, for this situation, the optimal solution will be achieved with the constraint active, as we still assume that no failures are discovered during testing, and hence from (4.11) and (4.12) the corresponding real-valued solution is

$$n_i^{*n} = \sqrt{\alpha_i \lambda_i} \left[\frac{n_{max} + \sum_{j=1}^{k} (\alpha_j + \beta_j)}{\sum_{j=1}^{k} \sqrt{\alpha_j \lambda_j}} \right] - (\alpha_i + \beta_i), \tag{4.19}$$

with Lagrange multiplier

$$\mu = t \left[\frac{\sum_{j=1}^{k} \sqrt{\alpha_j \lambda_j}}{n_{max} + \sum_{j=1}^{k} (\alpha_j + \beta_j)} \right]^2. \tag{4.20}$$

Substituting these real-valued n_i^{*n}, which indeed sum up to n_{max}, in (2.21), the corresponding expected number of failures is

$$t \frac{[\sum_{i=1}^{k} \sqrt{\alpha_i \lambda_i}]^2}{\sum_{i=1}^{k} (\alpha_i + \beta_i) + n_{max}}. \tag{4.21}$$

Hence, even though the n_i^{*n} vary per type of task, due to the values of the λ_i, α_i and β_i, this minimum expected number of failures in the process only depends on n_{max}, so not on the particular values n_i^{*n}. This is convenient for determining the total number of tests (without failures) required to get the expected number of failures in the process, over a period of length t, below a chosen target value by (4.21). For $i, j = 1, \ldots, k$,

$$\frac{n_i^{*n} + \alpha_i + \beta_i}{n_j^{*n} + \alpha_j + \beta_j} = \frac{\sqrt{\alpha_i \lambda_i}}{\sqrt{\alpha_j \lambda_j}},$$

which, again implies that the optimal n_i are in constant ratio to each other, independent of t, and are approximately proportional to $\sqrt{\alpha_i}$ and to $\sqrt{\lambda_i}$.

If one allows one more test, the optimal integer-valued n_i's all remain the same except for one, which is increased by one. This can be justified as follows. Suppose that the optimal real-valued solution of the number of tests of type i is $n_i^{'}$, given that we are allowed to test $n_{max} + 1$ tasks in total so, $\sum_{i=1}^{k} n_i^{'} = n_{max} + 1$. Then from (4.19) yields

$$\begin{aligned} n_i^{'} &= \sqrt{\alpha_i \lambda_i} \left[\frac{(n_{max} + 1) + \sum_{j=1}^{k} (\alpha_j + \beta_j)}{\sum_{j=1}^{k} \sqrt{\alpha_j \lambda_j}} \right] - (\alpha_i + \beta_i) \\ &= \sqrt{\alpha_i \lambda_i} \left[\frac{n_{max} + \sum_{j=1}^{k} (\alpha_j + \beta_j)}{\sum_{j=1}^{k} \sqrt{\alpha_j \lambda_j}} \right] - (\alpha_i + \beta_i) + \frac{\sqrt{\alpha_i \lambda_i}}{\sum_{j=1}^{k} \sqrt{\alpha_j \lambda_j}} \\ &= n_i^{*n} + \frac{\sqrt{\alpha_i \lambda_i}}{\sum_{j=1}^{k} \sqrt{\alpha_j \lambda_j}}, \quad i = 1, \ldots, k. \end{aligned}$$

By the same convexity argument as before, $\dfrac{\sqrt{\alpha_i \lambda_i}}{\sum_{j=1}^{k} \sqrt{\alpha_j \lambda_j}} \leq 1$, the integer-valued of $n_i^{'}$ (for all $i = 1, \ldots, k$) is either n_i or $n_i + 1$, given that $\sum_{i=1}^{k} n_i^{'} = n_{max} + 1$. Hence, the optimal numbers of tests should be one of the following combinations: $(n_1 + 1,\ n_2,\ n_3,\ \ldots,\ n_k)$, $(n_1,\ n_2 + 1,\ n_3, \ldots,\ n_k), \ldots$, or $(n_1,\ n_2,\ n_3,\ \ldots,\ n_k + 1)$. This is intuitively logical, and it has advantages if it is not clear from the start how many tests are allowed in total, and it can be useful for computation of the optimal numbers of tasks of different types to be tested.

Example 4.3.1

Suppose that we have a process similar to the one considered in Example 4.2.1,

but we do not consider costs and times of testing. So, the relevant parameters are $\underline{\lambda} = (1, 2, 4, 9)$, $\alpha_i = 1$, and $\beta_i = 0$, for all $i = 1, \ldots, k$. Suppose now that we are allowed to test in total $n_{max} = 50$ tasks. The optimal real-valued numbers of tests are $\underline{n}^* = (6.2835, 9.3004, 13.5661, 20.8500)$, regardless to the value of t (as these n_i^* are independent of t). As the numbers of tasks that should be tested are non-negative integers and the function for the expected number of failures is convex, we should look at all the combinations of integers that surround the optimal real-valued solution. In this example, the optimal integer solution that satisfies the corresponding constraint $\sum_{i=1}^{4} n_i \leq 50$, is $\underline{n} = (6, 9, 14, 21)$, which leads to a minimal expected number of failures equal to 10.1861 for a process with $t = 10$. Of course, if we had considered $t = 100$ instead, the minimal expected number of failures would be 10 times that for $t = 10$, namely 101.861, which follows from (4.21).

Suppose now that we are allowed to test one more extra task, so $n_{max} = 51$, and with $t = 10$, then according to our analysis it is optimal to test one extra task of the second type, so $\underline{n} = (6, 10, 14, 21)$, as this gives the minimum expected number of failures, 10.0043, from all other possible integer combinations.

If, the expected number of failures in the process is required not to exceed 5 during the length of time $t = 10$, then, from (4.21), at least 106 tests in total are required, and the optimal solution is then $\underline{n} = (14, 20, 29, 43)$.

4.4 Comparison with failure-free periods

In Chapter 3, we presented a similar setting for zero-failure reliability demonstration as in Section 4.2, but assuming that failures in the process after testing are catastrophic, in the sense that the system would stop functioning for all types of tasks on occurrence of the first failure, at a (probably large) fixed cost. The optimisation problem in Chapter 3 is explicitly formulated in terms of the predictive probability of a failure-free period in the process after testing. In that setting, it turns out that most tests are required if the number of tasks in the process after testing is deterministic, for each type of task. However, in many examples we calculated, the case where the numbers of tasks of each type in the process after testing have Pois-

son distributions, required almost the same numbers of tests without failures. In Chapter 3, we mostly restricted to Beta prior distributions with $\alpha_i = 1$ and $\beta_i = 0$ for the probability of a failure of type i, and we considered minimisation of the total expected costs of testing and process failure, similar to the approach in Section 4.2, but without constraints on the budget and time available for testing. A main conclusion reached in Chapter 3, with tasks of different types arriving according to independent Poisson processes, and presented in the same notation as used in Section 4.2, is that the optimal required number of tests of tasks of type i is approximately proportional to $\sqrt{\lambda_i/c_i}$, see (3.19), where the proportionality constant depends on the cost of process failure and the length t of the period during which the process is considered.

Comparing the optimal test numbers when focusing on failure-free periods, Chapter 3, with our results in Section 4.2, we can conclude that the same approximate proportionality results hold for the optimal number of tests per type of task, when considered as a function of λ_i and c_i. We consider this a useful result, in particular because when speaking about 'reliability demonstration' one may not always have a clear optimality criterion in mind. The results in Section 4.2 and in Chapter 3 imply that, when using Beta prior distributions with $\alpha_i = 1$ and $\beta_i = 0$ for all types i, and assuming that testing time is not a major restriction, and that failures in the process are either catastrophic or lead to fairly similar costs for different types of tasks, testing n_i tasks of type i, with

$$n_i \propto \sqrt{\frac{\lambda_i}{c_i}},$$

will be very reasonable. This will not be too far away from an optimal test scheme according to the different optimality criteria considered in these two chapters, which we think are both natural and attractive, as they are explicitly formulated in terms of minimal costs considering the actual test effort and expected failure costs in the process. The proportionality constant for these n_i depends of course on the actual criterion used and on the possible constraints on testing. Of course, we do not claim that testing in such proportions as functions of the λ_i and c_i is close to optimal for any possible optimality criterion. But our results suggest that, when explicitly focusing on the system's functioning in the process after testing, and when assuming

that testing reveals zero failures, testing in these proportions will be sensible. This is, for example, also useful if at the start of testing one does not yet know precisely the total number of tests one can actually perform.

4.5 Concluding remarks

Throughout our study of Bayesian reliability demonstration, with the explicit assumption that the system will only be used after testing if no failures occur during testing, we have restricted attention to Beta priors, which simplifies computations and enables interpretation of the hyperparameters. One could use any prior distribution to reflect prior knowledge, and use powerful computational methods (e.g. MCMC [22]) for the final inferences. However, as we are aiming at optimising test numbers, this could easily lead to an enormous computational burden. We think it is more important to focus on the choice of appropriate hyperparameters for the Beta priors. According to the optimality criteria in this chapter and in Chapter 3, the β_i only influence the optimal test numbers a little, in the sense that the sums of the required n_i and β_i remain approximately constant, so indeed the choice of β_i can be regarded as if you have already seen β_i successful tests of tasks of type i before the actual tests take place. Hence, from the perspective of reliability demonstration, we feel that it is appropriate to advocate the use of the value $\beta_i = 0$. The α_i play a far more important role. Effectively, one can interpret this as 'having seen α_i tests failing', and then demanding sufficiently many tests without failure such that, on the basis of the combined information represented by this prior and test information, the resulting confidence in the system's reliability is high.

From a Bayesian statistical perspective, it may seem natural to propose the use of very small values of α_i (possibly in combination with large values of β_i), to reflect prior beliefs that the system will be highly reliable, which would also be a natural requirement to justify explicit restriction to test results without failures. In our methods, the optimal numbers of tasks to test, per type of task, depend strongly on the choice of the α_i, where in particular the sum of these α_i's influences any measure of predicted system reliability after testing, whereas the individual values

α_i influence the optimal values n_i. From the perspective of reliability demonstration, we are in favour of the value $\alpha_i = 1$ for each type i, unless, for example, one has particular information that suggests that the system is more reliable with regard to tasks of type 1 than with regard to tasks of type 2, in which case one may wish to use $\alpha_1 < \alpha_2$. This may occur, for example, if tasks of type 2 are part of newly added system functionality, whereas the system has not been changed with regard to tasks of type 1 (note that one may still want to test tasks of type 1 to ensure that adding new functionality has not affected the system's further functioning).

Chapter 5

Reliability demonstration for systems with redundancy

5.1 Introduction

Systems with built in redundancy can provide high availability, reliability, and safety. Therefore, they are used in critical processes that require maximum operational readiness and system availability. In other words, they can be required when high reliability is essential, such as in security or safety systems, communications, computer systems (data storage in particular), process controls *etc.* In this chapter, we consider redundancy where reliability prediction is only considered at the system level. The references [38–40, 42] discuss reliability at system, subsystem and component levels. Components here are regarded as either functional or not (pass or fail components) [30, 39, 40], and we only consider systems consisting of exchangeable components.

We consider a system that has to perform tasks of $k \geq 1$ different types, dealing with tasks of different types fully independently. The system consists of $y \geq 1$ exchangeable components, which function independently in the system for each type of task. For example, we can think of a safety system for a large plant, with components being detectors at several sites within the plant, where each detector should be able to detect a variety of possible risks. Another possible setting for such systems is in critical data management, where components could be copies of data bases stored

at different locations, and communicating through an overall software system.

The main interest in this chapter is to find the optimal number of zero-failure tests in order to achieve a required reliability level for such systems, where reliability demonstration is only considered at the component level. This study considers x-out-of-y systems, including the two extreme cases of parallel and series systems, which includes those presented in Chapters 2 to 4 where $x = y = 1$, which are, in this chapter, referred to as unit-systems.

The Bayesian approach with a Binomial model and Beta prior distribution for θ_i, for all types of tasks $i = 1, \ldots, k$, is used again, as in the previous chapters of this thesis.

We assume, again, that the system does not wear-out or improve over time, and that it performs tasks of one type exchangeably (so not identically), so that failures of tasks of the same type can be represented by exchangeable Bernoulli random quantities, enabling a simple Bayesian model. We assume further, as in Chapter 3, that process failures are catastrophic in the sense that failures stop the use of the system in the process and might incur a high cost. We assume that such a cost is a fixed constant, but we acknowledge that it may be difficult in practice to assign it a precise value, making sensitivity studies with regard to this cost figure useful. The focus is on the following two important questions: How many zero-failure tests should be performed, and how many components, y ($\geq x$), should be installed in the system, for known required x, to demonstrate a certain required level of reliability for the process after testing. We also take costs into account to achieve a certain reliability level, p, again assuming that all tests reveal no failures. The required reliability level for the process after testing is in terms of a minimal probability p that the system will perform, without any failures, all the required m_i tasks for types $i = 1, \ldots, k$, after testing, or, for the random case, that the system's performance will be failure-free in the process during a specified period. We assume that costs for testing are linear in the number of tasks tested per type, and we are particularly interested in the effect of these cost parameters, and the required numbers of tasks m_i in the process, on the optimal numbers of tasks to be tested per type.

The same notation as introduced earlier in this thesis is used throughout this

chapter, with some additional straightforward notation. In Section 5.2, prediction of the reliability of such systems is discussed. In Section 5.3, reliability demonstration is studied for a single task, to allow clear analysis and a deep insight. In Section 5.4, m tasks of a single type are considered for reliability demonstration. In Section 5.5, reliability demonstration is studied when the system has to perform multiple types of tasks. In Section 5.6, costs for testing and failures are also taken into account. Both the deterministic and the random cases are considered, in the later case the focus is again on the Poisson case. In Section 5.7, we focus on how many tests should be performed and how many components $y \geq x$ should be used, providing redundancy, to minimise expected total costs while meeting a specific reliability requirement. We end with some concluding remarks for this chapter, as well as for the whole thesis, in Section 5.8, and some suggestions for related future research in Section 5.9.

5.2 Reliability prediction

Let $\theta \in [0, 1]$ be the parameter that can be interpreted as the unknown probability of a component failure, where components are exchangeable, when the system performs a task. We denote the x-out-of-y system reliability, which is the probability of performing one task successfully, by $r(x, y|\theta)$, using the Binomial distribution

$$r(x, y|\theta) = \sum_{j=x}^{y} \binom{y}{j} (1-\theta)^j \theta^{y-j}, \tag{5.1}$$

which is decreasing in x. Hence for series systems, with $x = y$, the system reliability,

$$r(y, y|\theta) = (1-\theta)^y. \tag{5.2}$$

This is can be interpreted as the probability of zero failures in y tasks for a unit-system as studied in Chapter 3. This is indeed logical if we think of the number of identical components as the number of tasks in a process of a single type of task. For parallel systems, $x = 1$,

$$r(1, y|\theta) = \sum_{j=1}^{y} \binom{y}{j} (1-\theta)^j \theta^{y-j} = 1 - \theta^y, \tag{5.3}$$

which can be interpreted as the probability of zero failures for a unit-system with failure probability equal to θ^y.

Clearly, $r(x, y|\theta)$ is increasing in y. Using the posterior distribution (2.6) of θ, the unconditional system reliability after zero-failure testing is

$$\begin{aligned} r(x, y|(n, 0)) &= \int_0^1 r(x, y|\theta)\ p(\theta|(n, 0))\ d\theta \\ &= \int_0^1 \sum_{j=x}^{y} \binom{y}{j} (1-\theta)^j\ \theta^{y-j}\ \frac{\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{B(\alpha, \beta+n)}\ d\theta \\ &= \frac{\sum_{j=x}^{y} \binom{y}{j} B(\alpha+y-j, \beta+n+j)}{B(\alpha, \beta+n)} \\ &= \frac{\Gamma(\alpha+\beta+n) \sum_{j=x}^{y} \binom{y}{j} \Gamma(\alpha+y-j)\Gamma(\beta+n+j)}{\Gamma(\alpha+\beta+n+y)\Gamma(\alpha)\Gamma(\beta+n)} \\ &= \frac{\sum_{j=x}^{y} \binom{y}{j} \prod_{l=1}^{y-j} (\alpha+l-1) \prod_{h=1}^{j} (\beta+n+h-1)}{\prod_{i=1}^{y} (\alpha+\beta+n+i-1)}, \end{aligned} \tag{5.4}$$

where we define $\prod_{\varnothing} = 1$ (as we do throughout this thesis). Interchanging the integral and the summation in the third equality is justified since all converges uniformly [35]. As in Chapter 2, β appears in the form $\beta + n$, so an increase in β will mean that the optimal or required n is reduced by the same number, hence we again advocate the choice $\beta = 0$. However, (5.4) is decreasing as function of α and is highly sensitive to its choice, as effectively the reliability demonstration requires n tests without failures to counter the prior information of α 'imaginary test failures', which of course agrees with the analysis we presented in previous chapters.

Obviously, (5.4) is a decreasing function of x. As $r(x, y|\theta)$ is increasing in y and $p(\theta|(n, 0))$ is independent of y, $r(x, y|(n, 0))$ is increasing in y, confirming that adding more redundant components to the system increases its reliability. Moreover, (5.4) is strictly increasing function of n, which is intuitively correct and can be justified by

$$\begin{aligned} r(x, y|(n+1, 0)) - r(x, y|(n, 0)) &= \sum_{j=x}^{y} \binom{y}{j} \prod_{l=1}^{y-j} (\alpha+l-1) \prod_{h=1}^{j-1} (z+h)[j(\alpha+z) - zy] \\ &= \frac{\Gamma(y-x+\alpha+1)\Gamma(y+1)\Gamma(z+x)}{\Gamma(x)\Gamma(z+1)\Gamma(y-x+1)\Gamma(\alpha)} > 0, \end{aligned} \tag{5.5}$$

where $z = \beta + n$. See Appendix A.1 for the proof of (5.5).

For a series system (y-out-of-y system), the unconditional system reliability, for any value of $\alpha > 0$ and $\beta \geq 0$, is

$$r(y, y|(n, 0)) = \prod_{j=1}^{y} \frac{\beta+n+j-1}{\alpha+\beta+n+j-1}, \tag{5.6}$$

which is a decreasing function of α. If we restrict attention to integer values $\alpha > 0$ and $\beta \geq 0$, then a second expression for this reliability is

$$r(y, y|(n, 0)) = \prod_{j=1}^{\alpha} \frac{\beta + n + j - 1}{y + \beta + n + j - 1}. \tag{5.7}$$

Clearly, $r(x, y|(n, 0)) \geq r(y, y|(n, 0))$, which can be confirmed by comparing the right hand sides of (5.6) with (5.4), which is intuitively correct as the more redundant components are in the system, the more reliable this system gets.

For a parallel system (1-out-of-y system), the unconditional system reliability, where $r(1, y|\theta) = 1 - \theta^y$, is

$$\begin{aligned} r(1, y|(n, 0)) &= \int_0^1 [1 - \theta^y] \, \frac{\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{B(\alpha, \beta + n)} \, d\theta \\ &= 1 - \frac{\Gamma(\alpha + y)\Gamma(\alpha + \beta + n)}{\Gamma(\alpha + \beta + n + y)\Gamma(\alpha)} \\ &= 1 - \prod_{j=1}^{y} \frac{\alpha + j - 1}{\alpha + \beta + n + j - 1}, \end{aligned} \tag{5.8}$$

which is an increasing function of y and n. Obviously, parallel system reliability, $r(1, y|(n, 0))$, can also be derived directly from (5.4) by substituting x by 1, which then confirms the fact that $r(1, y|(n, 0)) \geq r(x, y|(n, 0))$. And of course, $r(1, y|(n, 0)) \geq r(y, y|(n, 0))$ which is also intuitively correct, as the smaller x is, the more reliable the system gets.

If the system has to perform $m \geq 0$ tasks, assuming that each component fails in functioning any task with probability equal to θ, then the probability that the x-out-of-y system performs m tasks successfully is

$$r(x, y|m, \theta) = [\sum_{j=x}^{y} \binom{y}{j} (1 - \theta)^j \theta^{y-j}]^m, \tag{5.9}$$

which is a decreasing function of m. Similarly, the unconditional x-out-of-y system reliability, after n zero-failure tests, is

$$\begin{aligned} r(x, y|m, (n, 0)) &= \int_0^1 r(x, y|m, \theta) \; p(\theta|(n, 0)) \; d\theta \\ &= \int_0^1 [\sum_{j=x}^{y} \binom{y}{j} (1 - \theta)^j \; \theta^{y-j}]^m \; \frac{\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{B(\alpha, \beta + n)} \, d\theta, \end{aligned} \tag{5.10}$$

which is a decreasing function of m. Due to the power m over the sum in (5.10), which prevents exchanging the orders of the sum and the integral, we could not simplify this equation any further. Let us denote the right-hand side (RHS) of (5.10), as function of m, by $g(m)$. As $g(m) \geq 0$ is decreasing, and

$$g(m) - \quad g(m+1) = \\ \int_0^1 [\sum_{j=x}^{y} \binom{y}{j} (1-\theta)^j \theta^{y-j}]^m \frac{\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{B(\alpha, \beta+n)} [1 - \sum_{j=x}^{y} \binom{y}{j} (1-\theta)^j \theta^{y-j}] d\theta, \quad (5.11)$$

is decreasing in m, it follows that $g(m)$ is a convex function of m.

From (5.10), we have for a series system,

$$\begin{aligned} r(y, y|m, (n, 0)) &= \int_0^1 r(y, y|m, \theta)\ p(\theta|(n, 0))\ d\theta \\ &= \int_0^1 (1-\theta)^{ym}\ \frac{\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{B(\alpha, \beta+n)}\ d\theta \\ &= \prod_{j=1}^{my} \frac{\beta+n+j-1}{\alpha+\beta+n+j-1}, \end{aligned} \quad (5.12)$$

for any value of α, and

$$r(y, y|m, (n, 0)) = \prod_{j=1}^{\alpha} \frac{\beta+n+j-1}{ym+\beta+n+j-1}, \quad (5.13)$$

for integer values of α.

Using $(a+b)^n = \sum_{r=0}^{n} \binom{n}{r} a^{n-r} b^r$, we get for a parallel system,

$$\begin{aligned} r(1, y|m, (n, 0)) &= \int_0^1 (1-\theta^y)^m\ \frac{\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{B(\alpha, \beta+n)}\ d\theta \\ &= \int_0^1 \sum_{r=0}^{m} \binom{m}{r} (-\theta^y)^r\ \frac{\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{B(\alpha, \beta+n)}\ d\theta \\ &= \frac{1}{B(\alpha, \beta+n)} \sum_{r=0}^{m} \binom{m}{r} (-1)^r \int_0^1 \theta^{yr+\alpha-1}(1-\theta)^{\beta+n-1}\ d\theta \\ &= \sum_{r=0}^{m} \binom{m}{r} (-1)^r \frac{B(yr+\alpha, \beta+n)}{B(\alpha, \beta+n)} \\ &= \sum_{r=0}^{m} \binom{m}{r} (-1)^r \prod_{j=1}^{yr} \frac{\alpha+j-1}{\alpha+\beta+n+j-1}. \end{aligned} \quad (5.14)$$

Let us now consider the case of random M, with probability distribution $P(M = u)$ for nonnegative integers u. The same analysis as in Section 2.4.2, using Jensen's

inequality for convex functions, yields that the x-out-of-y system reliability for M tasks in the process after zero-failure testing, is at least as large as the x-out-of-y system reliability for a known number m tasks after n tests without failures, when we restrict attention to probability distributions for M with expected value $E(M) = m$. So, again, one could consider the exact knowledge of the value of the random quantity M, to be disadvantageous.

For the situation in which we assume a Poisson distribution for M, as occurs if in the process after testing tasks arrive according to a homogeneous Poisson process with expected value $E(M) = \lambda t = \mu$, the corresponding x-out-of-y system reliability, after n zero-failure tests, is

$$\begin{aligned} r(x,y|M \sim \text{Po}(\mu),(n,0)) &= \sum_{u=0}^{\infty} r(x,y|u,(n,0)) e^{-\mu}\frac{\mu^u}{u!} \\ &= \sum_{u=0}^{\infty}\int_0^1 [\sum_{j=x}^{y}\binom{y}{j}(1-\theta)^j\ \theta^{y-j}]^u\ \frac{\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{B(\alpha,\beta+n)} e^{-\mu}\frac{\mu^u}{u!}\ d\theta \\ &= \int_0^1 \sum_{u=0}^{\infty}[\sum_{j=x}^{y}\binom{y}{j}(1-\theta)^j\ \theta^{y-j}]^u \frac{\mu^u}{u!}e^{-\mu}\ \ \frac{\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{B(\alpha,\beta+n)}\ d\theta \\ &= \int_0^1 e^{-\mu[1-\sum_{j=x}^{y}\binom{y}{j}(1-\theta)^j\ \theta^{y-j}]} \frac{\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{B(\alpha,\beta+n)} d\theta, \end{aligned} \tag{5.15}$$

where again, the interchanging of the integral and the summation is justified as before. Clearly, (5.15) is decreasing as function of μ and x, and increasing as function of y.

In the following example, we illustrate the above probabilities, for the deterministic and for the Poisson cases, showing the influence of prior knowledge on the reliability function.

Example 5.2.1

We consider 1-out-of-2 system, when $n = 20$ tests on exchangeable components did not reveal any failure. Table 5.1 gives the 1-out-of-2 system reliabilities for: (1) $m = 5$ future tasks; (2) a random M that has a Poisson distribution with expected value $E(M) = \mu = \lambda t = 5$ within the period of length t.

As in Example 2.4.1, this example demonstrates the influence of changes to the value of α, and the far weaker influence the value of β has. Table 5.1 illustrates

that system reliability for a process with a random M, with $E(M) = m$, is at least as large as the system reliability in a known number of tasks, m. It also shows that these reliabilities are again very similar for the Poisson and the deterministic cases. Comparing Table 5.1 with Table 2.1 confirms the fact that systems with redundancy are more reliable than unit-systems. Moreover, comparing these two tables illustrates that systems with redundancy are less sensitive to the choice of α than unit-systems, which is intuitively logical as installing more components in the system reduces the effect of what we call the number of imaginary test failures α. For example, changing the values of α from 1 to 2 (or from 1 to 3), for $\beta = 0$, decreases the reliability of the corresponding system with redundancy by 0.0349 (0.0787). While the corresponding reduction of reliability of the unit-system was much larger, namely 0.1538 (0.2735).

α	β	$r(1,2\|m=5,(20,0))$	$r(1,2\|M\sim \text{Po}(5),(20,0))$
1	0	0.9793	0.9795
1	1	0.9810	0.9812
1	2	0.9825	0.9827
1	5	0.9862	0.9863
1	10	0.9901	0.9902
1	100	0.9993	0.9993
2	0	0.9444	0.9452
3	0	0.9006	0.9025
0.5	0	0.9918	0.9918

Table 5.1: $r(1,2|m=5,(20,0))$ and $r(1,2|M\sim \text{Po}(5),(20,0))$ for varying α,β.

In the following sections, we will show that fewer zero-failure tests are required for systems with redundancy than for unit-systems, in order to demonstrate the same level of reliability.

5.3 Reliability demonstration for a single task

Our main interest in this section is to determine the minimal number of zero-failure tests required to demonstrate a specific level of reliability, say p, before the system

is used in a process. We consider a single task, $m = 1$, which allows us to introduce the basic concepts used in later sections.

From (5.4), which is an increasing function of n, the minimal required real-valued number of zero-failure tests, say n^*, assuming again that the system functions successfully for all tasks tested, can be found numerically by solving the following equation.

$$\frac{\sum_{j=x}^{y} \binom{y}{j} \prod_{l=1}^{y-j} (\alpha + l - 1) \prod_{h=1}^{j} (\beta + n + h - 1)}{\prod_{z=1}^{y} (\alpha + \beta + n + z - 1)} = p. \tag{5.16}$$

Of course, the optimal number of tests is the closest integer that is larger than or equal to n^*, i.e. $n = \lceil n^* \rceil$.

As before, we advocate the use of $\beta = 0$, while α again has a greater effect on $r(x, y|(n, 0))$, and hence on the minimal test size required for $r(x, y|(n, 0)) \geq p$, which is illustrated in the following example.

Example 5.3.1

Table 5.2 shows the optimal number of zero-failure tests of a 6-out-of-8 system, to achieve a required reliability level of p, for different values of α with $\beta = 0$. The optimal number of tests increases significantly with the increasing values of p and α.

	α			
p	0.001	0.5	1	1.5
0.90	1	5	8	12
0.95	1	8	12	17
0.97	1	10	16	21
0.99	1	17	26	34
0.995	1	22	34	45
0.999	2	42	63	83

Table 5.2: Minimal test numbers for a 6-out-of-8 system.

Notice that for $\alpha = 0.001$ there will be no need to do any testing if $\beta \geq 1$, in order to achieve any $p \leq 0.995$.

If n is large compared to α, for a positive integer α, with $\beta = 0$ and p close to 1, then similar to results in Section 3.2, (5.7) can be approximated by $[n/(n+y)]^{\alpha}$, hence the minimal required number of tests to meet a reliability requirement in a series system, say n_{ser}, is close to

$$\frac{p^{1/(\alpha)}}{1-p^{1/(\alpha)}}y \approx \frac{p}{1-p}\alpha y. \tag{5.17}$$

For example, for $p = 0.95$, taking α equal to two will lead to a required n_{ser} of just more than twice the size than would be required for $\alpha = 1$, e.g. see Example 5.4.1. For the case of $\alpha = 1$ and $\beta = 0$, the RHS of (5.17) gives the exact value of the optimal real-valued n^*.

For a parallel system and any value of α, with $\beta = 0$, from (5.8) the minimal required number of tests, say n_{par}, to meet a reliability requirement $r(1, y|(n, 0)) \geq p$, is the minimal integer value for which

$$\prod_{j=1}^{y} \frac{\alpha + j - 1}{\alpha + n + j - 1} \geq 1 - p, \tag{5.18}$$

which is the corresponding system unreliability. The LHS of (5.18) is decreasing in y and n, so for larger y, fewer tests are required in order to achieve a specific reliability level, p.

Again, as discussed in Sections 3.5 and 4.5, from the perspective of reliability demonstration of a highly reliable system (restricted to zero-failure tests), the use of a very small value of α to reflect prior beliefs of how reliable the system is, may be appropriate. From the point of view of reliability demonstration, we are again in favour of $\alpha = 1$. For $\beta = 0$ and $\alpha = 1$, (5.4) yields

$$r(x, y|(n, 0)) = \frac{ny!}{(n+y)!}\sum_{j=x}^{y}\frac{(n+j-1)!}{j!} = 1 - \frac{y!(n+x-1)!}{(x-1)!(n+y)!}, \tag{5.19}$$

the proof of which is given in Appendix A.2. Hence, the minimal number of zero-failure tests, n, required to demonstrate a level of reliability p, can be obtained from

$$\frac{(n+x-1)!}{(n+y)!} = \frac{(1-p)(x-1)!}{y!}, \tag{5.20}$$

which leads to

$$\prod_{j=x}^{y}(n+j) = \frac{y!}{(1-p)(x-1)!}. \tag{5.21}$$

From (5.21), the real-valued solution of optimal tests for a parallel system satisfies

$$\prod_{j=1}^{y}(n+j) = \frac{y!}{(1-p)}, \tag{5.22}$$

and for a series system is

$$\frac{yp}{1-p}, \tag{5.23}$$

These results are illustrated in the example below.

Example 5.3.2

Table 5.3 presents n_{par}, n_{ser}, and n of a 6-out of-8 system for Beta$(1,0)$ prior distribution and for different reliability levels. We see that the required number of zero-failure tests increases as a function of p and as a function of x. So indeed, using more redundant components in the system decreases the required number of zero-failure tests to achieve a specific level of reliability. For example, a very high number of tests, $n_{ser} = 7992$, is required to achieve $p = 0.999$, whereas only 5 zero-failure tests will be needed to achieve the same reliability level for a parallel system, $x = 1$.

p	n_{par}	n	n_{ser}
0.90	2	8	72
0.95	2	12	152
0.97	2	16	259
0.99	3	26	792
0.995	4	34	1592
0.999	5	63	7992
0.9995	6	81	15992

Table 5.3: Minimal n, n_{par}, and n_{ser} for $y = 8$ and for different p.

5.4 Reliability demonstration for multiple tasks of one type

Now we are interested in finding the minimal number of zero-failure tests, for a process with m tasks that the system is required to perform successfully after testing,

and with a Beta(α, β) prior distribution for θ. The optimal number of tests, n, for an x-out-of-y system to achieve a certain reliability level p, is the minimal integer value that is larger than or equal to the real-valued for which

$$r(x, y|m, (n, 0)) = \int_0^1 [\sum_{j=x}^{y} \binom{y}{j} (1-\theta)^j \ \theta^{y-j}]^m \ \frac{\theta^{\alpha-1}(1-\theta)^{\beta+n-1}}{B(\alpha, \beta+n)} \ d\theta = p, \quad (5.24)$$

which can be solved numerically. As $r(x, y|m, (n, 0))$ is a decreasing function of m and an increasing function of n, then if m increases, one requires an increased n in order to demonstrate the reliability level p. In other words, the more tasks the system is required to perform successfully after testing, the more zero-failure tests are needed to achieve the required reliability level.

From (5.12), the minimal tests for a series system, n_{ser}, is the closest integer value that is larger than or equal to the real-valued solution of the following equations

$$\prod_{j=1}^{my} \frac{\beta + n + j - 1}{\alpha + \beta + n + j - 1} = p, \quad (5.25)$$

for any value of α, and (5.13) yields

$$\prod_{j=1}^{\alpha} \frac{\beta + n + j - 1}{ym + \beta + n + j - 1} = p, \quad (5.26)$$

for integer values of α. If n is large compared to a positive integer α, with $\beta = 0$ and p close to 1 then, using the same justifications used in Section 3.2, n_{ser} is close to

$$\frac{p^{1/(\alpha)}}{1 - p^{1/(\alpha)}} ym \approx \frac{p}{1-p} \alpha ym. \quad (5.27)$$

For the case with $\alpha = 1$, (5.27) gives the exact value of the optimal n_{ser}, as shown in Section 3.3 and also illustrated in Example 5.4.1.

The minimal required number of zero-failure tests of a parallel system, n_{par}, using (5.14), is the minimal integer value larger than or equal to the real-valued n^* for which

$$\sum_{r=0}^{m} \binom{m}{r} (-1)^r \prod_{j=1}^{n} \frac{\alpha + j - 1}{\alpha + yr + j - 1} = p. \quad (5.28)$$

Example 5.4.1

Table 5.4 presents minimal numbers of zero-failure tests for parallel and series systems, n_{par} and n_{ser} respectively. It also presents the minimal numbers of zero-failure

tests n for 6-out-of-8 and 6-out-of-9 systems. Again, we consider the Beta$(1, 0)$ prior distribution. Systems are required to function m tasks successfully to achieve $p = 0.95$. We see that the required number of tests increases as function of m, and as function of x, and decreases as function of y. Table 5.4 also shows that n_{ser} increases by about the same factor as m and (or) y increase, which agrees with the results in Section 3.2.

	y =8			y =9		
m	n_{par}	n	n_{ser}	n_{par}	n	n_{ser}
1	2	12	152	2	9	171
2	2	17	304	2	11	342
3	3	20	456	2	13	513
4	3	22	608	2	14	684
6	3	26	912	3	16	1026
9	3	30	1368	3	18	1539
10	3	31	1520	3	19	1710
100	5	73	15200	4	37	17100

Table 5.4: Minimal n_{par}, n, and n_{ser} for $y = 8$ and $y = 9$ and for different m.

If we take $\alpha = 2$, then for $m = 1$ we need $n_{ser} = 308$ tests without failures to demonstrate reliability 0.95 for the 8-out-of-8 system, which is about twice the optimal number of tests for $\alpha = 1$, which agrees with (5.27).

5.5 Reliability demonstration for multiple independent types of tasks

Suppose that an x-out-of-y system has to perform $k \geq 1$ independent types of tasks, with m_i tasks of type i, for $i = 1, 2, \ldots, k$. The probability of performing all of these tasks without failures, after n_i zero-failure tests of type i, is

$$r(x, y|\underline{m}, (\underline{n}, \underline{0})) = \prod_{i=1}^{k} \int_0^1 [\sum_{j=x}^{y} \binom{y}{j} (1-\theta_i)^j \ {\theta_i}^{y-j}]^{m_i} \ \frac{{\theta_i}^{\alpha_i - 1}(1-\theta_i)^{\beta_i + n_i - 1}}{B(\alpha_i, \beta_i + n_i)} \ d\theta_i. \tag{5.29}$$

The minimal required number of tests, n_i^*, for such a system to demonstrate a reliability level p, is the smallest such that

$$r(x, y|\underline{m}, (\underline{n}, \underline{0})) \geq p. \tag{5.30}$$

Again as the β_i's appear only in the form $\beta_i + n_i$, we suggest to use $\beta_i = 0$.

For a series system, the optimal number of tests of type i, $n_{ser,i}$, the smallest integer n_i such that

$$\prod_{i=1}^{k} \prod_{j=1}^{m_i y} \frac{\beta_i + n_i + j - 1}{\alpha_i + \beta_i + n_i + j - 1} \geq p, \tag{5.31}$$

for any value of α_i, and

$$\prod_{i=1}^{k} \prod_{j=1}^{\alpha_i} \frac{\beta_i + n_i + j - 1}{y m_i + \beta_i + n_i + j - 1} \geq p, \tag{5.32}$$

for integer values of α_i. Again for positive integer α_i, $\beta_i = 0$, and p close to 1, and if $n_{ser,i}$ is large compared to α_i, then using same justification as in the previous section, $n_{ser,i}$ is close to (see Section 3.3 for details)

$$\frac{p^{1/(k\alpha_i)}}{1 - p^{1/(k\alpha_i)}} y m_i \approx \frac{p}{1-p} \alpha_i k y m_i. \tag{5.33}$$

Similarly, the minimal number of tests, without failures, of type i for a parallel system, $n_{par,i}$, with $\beta_i = 0$, using (5.14), is the smallest integer n_i such that

$$\prod_{i=1}^{k} \sum_{r=0}^{m_i} \binom{m_i}{r} (-1)^r \prod_{j=1}^{n_i} \frac{\alpha_i + j - 1}{\alpha_i + yr + j - 1} \geq p. \tag{5.34}$$

Example 5.5.1

To illustrate the results in this section and compare them with results of Section 3.3, we consider the same situation considered in Example 3.3.2. Table 5.5, and Table 5.6 for big systems, give the minimal required numbers of zero-failure tests for systems with redundancy, with $\beta_i = 0$ for all $i = 1, \ldots, k$, for the following cases:

(a) $k = 1, m = 100, \alpha = 1$

(b) $k = 2, m_1 = m_2 = 50, \alpha_1 = \alpha_2 = 1$

(c) $k = 4, m_i = 25, \alpha_i = 1$ for $i = 1, \ldots, 4$

(d) $k = 2$, $m_1 = 25$, $m_2 = 75$, $\alpha_1 = \alpha_2 = 1$

(e) $k = 2$, $m_1 = 10$, $m_2 = 90$, $\alpha_1 = \alpha_2 = 1$

(f) $k = 2$, $m_1 = m_2 = 50$, $\alpha_1 = \alpha_2 = 0.5$

(g) $k = 4$, $m_i = 25$ and $\alpha_i = 0.25$ for $i = 1, \ldots, 4$

(h) $k = 2$, $m_1 = 25$, $m_2 = 75$, $\alpha_1 = 0.25$, $\alpha_2 = 0.75$

(i) $k = 2$, $m_1 = 10$, $m_2 = 90$, $\alpha_1 = 0.10$, $\alpha_2 = 0.90$

(j) $k = 1$, Poisson, $\mu = 100$, $\alpha = 1$

(k) $k = 2$, Poisson, $\mu_1 = \mu_2 = 50$, $\alpha_1 = \alpha_2 = 1$

(l) $k = 2$, Poisson, $\mu_1 = 25$, $\mu_2 = 75$, $\alpha_1 = \alpha_2 = 1$

Case	p	1-out-of-2		1-out-of-3		2-out-of-3	
		$\underline{n}$	$\sum_{i=1}^{k} n_i$	$\underline{n}$	$\sum_{i=1}^{k} n_i$	$\underline{n}$	$\sum_{i=1}^{k} n_i$
(a)	0.90	38	38	14	14	66	66
	0.95	58	58	19	19	100	100
	0.99	138	138	37	37	239	239
(b)	0.95	(60, 59)	119	(20, 20)	40	(103, 103)	206
	0.99	(139, 139)	278	(37, 37)	74	(241, 240)	481
(c)	0.95	(60, 60, 60, 61)	241	(20, 20, 20, 21)	81	(104, 104, 104, 105)	417
	0.99	(139, 139, 139, 140)	557	(37, 37, 37, 37)	148	(241, 241, 241, 242)	965
(d)	0.95	(46, 67)	113	(16, 22)	38	(80, 116)	196
(e)	0.95	(33, 68)	101	(13, 22)	35	(57, 118)	175
(f)	0.95	(35, 36)	71	(13, 13)	26	(61, 61)	122
	0.99	(84, 85)	169	(24, 25)	49	(146, 146)	292
(g)	0.95	(23, 23, 23, 22)	91	(9, 9, 9, 9)	36	(39, 39, 39, 39)	156
	0.99	(55, 55, 55, 55)	220	(18, 18, 18, 17)	71	(95, 95, 95, 96)	381
(h)	0.90	(13, 32)	45	(6, 12)	18	(23, 55)	78
	0.95	(20, 49)	69	(8, 17)	25	(35, 84)	119
	0.99	(50, 116)	166	(16, 32)	48	(86, 201)	287
(i)	0.95	(12, 71)	83	(5, 22)	27	(21, 124)	145
(j)	0.90	38	38	14	14	66	66
	0.95	58	58	19	19	100	100
	0.99	138	138	37	37	239	239
(k)	0.95	(59, 60)	119	(20, 20)	40	(103, 103)	206
	0.99	(139, 139)	278	(37, 37)	74	(241, 240)	481
(l)	0.95	(46, 67)	113	(16, 22)	38	(80, 116)	196

Table 5.5: Minimal test numbers required for different x-out-of-y systems and for different cases, Cases (a)-(l).

Case	p	5-out-of-8		6-out-of-8		7-out-of-8	
		$\underline{n}$	$\sum_{i=1}^{k} n_i$	$\underline{n}$	$\sum_{i=1}^{k} n_i$	$\underline{n}$	$\sum_{i=1}^{k} n_i$
(a)	0.95	32	32	73	73	305	305
	0.99	56	56	140	140	731	731
(b)	0.95	(33, 34)	67	(76, 76)	152	(314, 314)	628
	0.99	(56, 57)	113	(141, 141)	282	(735, 735)	1470
(c)	0.95	(35, 35, 35, 34)	139	(78, 78, 78, 77)	311	(319, 319, 318, 318)	1274
	0.99	(57, 57, 57, 57)	228	(142, 142, 142, 141)	567	(737, 737, 737, 737)	2948
(d)	0.95	(29, 36)	65	(63, 83)	149	(246, 352)	598
(e)	0.95	(23, 37)	60	(48, 85)	133	(174, 359)	
(f)	0.95	(22, 22)	44	(48, 48)	96	(185, 186)	371
	0.99	(39, 39)	78	(93, 94)	187	(446446)	892
(g)	0.95	(16, 16, 16, 17)	65	(34, 34, 34, 33)	135	(119, 119, 119, 118)	475
	0.99	(29, 29, 30, 30)	118	(67, 67, 67, 66)	267	(291, 291, 291, 292)	1165
(h)	0.95	(14, 29)	43	(29, 64)	93	(106, 257)	363
	0.99	(26, 50)	76	(59, 123)	182	(261, 615)	876
(i)	0.95	(9, 37)	46	(19, 87)	106	(65, 394)	459
(j)	0.95	32	32	73	73	305	305
	0.99	56	56	139	139	731	731
(k)	0.95	(33, 34)	67	(76, 76)	152	(313, 314)	627
	0.99	(56, 57)	113	(141, 141)	282	(734, 735)	1469
(l)	0.95	(29, 36)	65	(63, 83)	146	(245, 352)	597

Table 5.6: Minimal test numbers required for different x-out-of-y systems, for Cases (a)-(l).

Tables 5.5 and 5.6 in comparison with Table 3.3, illustrates the following results. Systems with built-in redundancy need far fewer tests than unit-systems. It also shows that systems with redundancy and unit-systems share the following results. Dividing the total number of tasks in the process, m, into equal numbers $m_i = m/k$ for k types, the number of tasks that need to be tested *per type* remains close to the total number needed if all tasks were assumed to be of one type, as long as we take the 'automatic' choice $\alpha_i = 1$ for all types (e.g. Cases (a), (b) and (c)). Moreover, cases with $m_i = m/k$ represent the worst case in terms of total number of tests needed (e.g. Cases (d) and (e), when compared to Case (b)).

Comparing Cases (f) and (g) with Cases (a), (b) and (c) illustrates the fact that using $\alpha_i = \alpha/k$ together with $m_i = m/k$, with α the hyperparameter used in the case of all tasks being of a single type, in total fewer zero-failure tests are required than for cases with $\alpha_i = \alpha$. However, as shown in Table 3.3, the total required

zero-failure tests for unit-systems becomes (about) the *same* number of required tests as in the case with a single type of tasks in the process, which we do not see in systems with redundancy. This is due to the far stronger influence of α_i on n_i for unit-systems than it is for systems with redundancy (see Section 5.2).

Moreover, Cases (h) and (i), when compared to Cases (d) and (a) in Tables 5.5 and 5.6 in comparison with Table 3.3, suggest that even when the m_i's are not equal, taking the α_i's proportional to the m_i's and such that they sum up to α, leads to a total number of zero-failure tests that is close to the number required in the case of all tasks being of a single type. Again, the reduction in the total number of zero-failure tests for unit-systems here is more than it is for systems with redundancy, which is also due to the greater effect of α_i on n_i for unit-systems than it is for systems with redundancy.

Tables 5.5 and 5.6 also illustrate that the Poisson case is nearly as bad as the deterministic situation, in terms of minimal numbers of zero-failure tests needed, (e.g. Cases (j), (k) and (l), in comparison to Cases (a), (b) and (d)).

5.6 Reliability demonstration considering costs

In this section, we generalise the results of Section 3.4, as we are interested in determining the optimal number of zero-failure tests, taking into consideration costs of testing and process failures, for an x-out-of-y system that performs $k \geq 1$ independent types of tasks . Let $c_i > 0$ be the cost of each test of type i, for $i = 1, 2, \ldots, k$, and $C \geq 0$ be the cost of the process failures, where the functioning of the system in the process is ended on occurrence of any failure.

The objective is to minimise the total expected cost of testing and process failure, $(\forall i = 1, 2, \ldots, k)$

$$EC(\underline{n}, \underline{c}, \underline{m}, C) = \sum_{i=1}^{k} n_i c_i + [1 - r(x, y|\underline{m}, (\underline{n}, \underline{0}))]C.$$

Although minimisation of these expected costs already takes the probability of zero failures in the process into account, one may wish to include the constraint $r(x, y|\underline{m}, (\underline{n}, \underline{0})) \geq p$. Unfortunately, we could not achieve an analytical solution

due to the complexity of the corresponding reliability function (5.29). The situation studied in Section 5.5 is the special case of this constrained problem with $C = 0$ and $c_i = 1$, for all $i = 1, 2, \ldots, k$.

For these x-out-of-y systems, the deterministic case is again the worst-case scenario, in precisely the same sense as discussed at Section 2.4.2 for unit systems.

Example 5.6.1

In this example we compare the optimal numbers of zero-failure tests, taking costs into consideration, for systems with built-in redundancy, with the optimal numbers of tests for corresponding unit-systems. We consider the four cases stated in Example 3.4.1, for both the deterministic (Cases (1)-(3)) and the Poisson cases (Case (4)), with all $\alpha_i = 1$ and $\beta_i = 0$.

In Cases (3) and (4), $C = 0$ implies that we just aim at minimal costs of zero-failure testing required to meet the reliability constraint. This gives an attractive problem formulation in situations where one is not willing or able to provide a meaningful value related to a failure in the process after testing.

Table 5.7 gives the optimal test numbers for some x-out-of-y systems with additional superscript '$\blacktriangleleft$' to indicate that the constraint $r(x, y|\underline{m}, (\underline{n}, \underline{0})) \geq p$ is not active.

Case(s)	p	$\underline{n}$			
		1-out-of-2	1-out-of-3	2-out-of-3	3-out-of-3
(1)	0.90	$(11, 11, 14)$	$(7, 8, 10)^{\blacktriangleleft}$	$(18, 19, 23)$	$(215, 233, 327)$
	0.95	$(16, 16, 20)$	$(7, 8, 10)^{\blacktriangleleft}$	$(26, 28, 35)$	$(446, 483, 680)$
	0.99	$(37, 38, 48)$	$(15, 15, 17)$	$(62, 66, 84)$	$(2145, 2501, 3548)$
(2)	0.90	$(24, 25, 32)^{\blacktriangleleft}$	$(14, 15, 18)^{\blacktriangleleft}$	$(82, 87, 110)^{\blacktriangleleft}$	$(374, 407, 573)^{\blacktriangleleft}$
	0.95	$(24, 25, 32)^{\blacktriangleleft}$	$(14, 15, 18)^{\blacktriangleleft}$	$(82, 87, 110)^{\blacktriangleleft}$	$(446, 483, 680)$
	0.99	$(37, 38, 48)$	$(14, 15, 18)^{\blacktriangleleft}$	$(82, 87, 110)^{\blacktriangleleft}$	$(2145, 2501, 3548)$
(3)	0.90	$(14, 10, 23, 16)$	$(7, 5, 10, 8)$	$(25, 17, 40, 28)$	$(365, 229, 728, 478)$
	0.95	$(21, 15, 33, 24)$	$(9, 7, 13, 11)$	$(36, 25, 58, 43)$	$(734, 473, 1517, 997)$
(4)	0.90	$(14, 10, 22, 16)$	$(7, 5, 10, 8)$	$(24, 17, 39, 28)$	$(362, 226, 724, 476)$
	0.95	$(21, 15, 33, 24)$	$(9, 7, 13, 11)$	$(36, 25, 58, 43)$	$(733, 470, 1516, 993)$

Table 5.7: Optimal test numbers for x-out-of-y systems.

Table 3.4, in Example 3.4.1, and Table 5.7 illustrate the logical facts that the required number of zero-failure tests increases as function of x, and decreases as function of y, for such systems. It is clear that redundancy can greatly reduce the required numbers of zero-failure tests. If the reliability constraint is active in the optimal solution, then the optimal test numbers do not depend on the precise value of C, see for example Tables 3.4 and 5.7 (3-out-of-3 system), Cases (1) and (2) for $p = 0.99$. And if the constraint is inactive at the optimal solution, then the precise value of p does not influence the (effectively unconstrained) optimum, see for example the 1-out-of-3 system, Case (2) for all reported values of p. Cases (3) and (4) in Table 5.7 enable comparison between the deterministic and Poisson cases, showing that the differences in the required test numbers are very small.

All cases in Table 5.7, and all other examples that we have calculated, have suggested an approximate relation between the optimal n_i for x-out-of-y systems, and m_i and c_i, which generalizes the suggested approximate relation (3.19) for unit-systems. It appears to be the case that n_i is approximately proportional to $\sqrt[y-x+2]{\frac{m_i}{c_i}}$. Unfortunately, we have not been able to prove this analytically, due to the complexity of the x-out-of-y system reliability, (5.29).

5.7 Optimal testing and redundancy level

Increasing built-in redundancy in a system makes the system more reliable. Hence, it is interesting to explore the possibility of reducing the required zero-failure test effort by increasing y, for fixed x, in x-out-of-y systems. As installing additional components in the system may be expensive, we can include such costs in our expected total cost function, as presented in Section 5.6, and then again minimise the resulting expected total costs, with the possible constraint on the system's predicted reliability in the process after testing. Such minimisation would involve choosing the optimal combination of the required numbers of zero-failure tests for each type of task, and the number of components y in the system, where $y \geq x$ is required for system functionality. We assume that x is given by required system functionality,

or other factors such as safety requirements.

Let us assume that the cost of each of the y components in the system is Q, and that the setting of Section 5.6 applies again. We can express the total expected costs as

$$EC(\underline{n}, \underline{c}, \underline{m}, y, Q, C) = yQ + \sum_{i=1}^{k} n_i c_i + [1 - r(x, y|\underline{m}, (\underline{n}, \underline{0}))]C. \tag{5.35}$$

We can again include the reliability constraint $r(x, y \,|\, m, (n, 0)) \geq p$, leading to a constrained optimisation problem that needs to be solved numerically. We illustrate this combined optimisation of y and the test numbers n_i in Examples 5.7.1 and 5.7.2.

Example 5.7.1

Let us consider the method presented above for a 2-out-of-y system, so at least 2 components that must function to ensure system functionality. Again, we use Beta prior distributions with $\alpha_i = 1$ and $\beta_i = 0$ for all types of tasks. In this example, we focus only on a single type of task in this example, simplifying notation in the obvious way, using n instead of n_1, *etc.* Tables 5.8, 5.9 and 5.10 present optimal numbers of zero-failure tests, n, and components, y, in an x-out-of-y system, for given x, corresponding to minimal total expected costs (5.35), under the reliability constraint with $p = 0.95$ in Tables 5.8 and 5.9, and with $p = 0.99$ in Table 5.10. These tables illustrate that the optimal y and n tend to increase in m. However, once m has increased sufficiently to cause the integer-valued optimal y to increase, the corresponding optimal n will decrease due to the fact that higher built-in redundancy level requires fewer zero-failure tests in this constrained optimisation problem.

Table 5.8, where $c = 1$, illustrates that if the cost Q per component increases, the optimal solution tends to be achieved for a smaller number of components and more tests. However, for a small increase in Q and with the reliability constraint remaining inactive, the optimal combination of n and y does not change (e.g. Cases with $m = 2$ and $m = 10$ with Q changes from 150 to 300 in Tables 5.8, 5.9 and 5.10). Table 5.8 also illustrates that increasing process failure costs C requires an increased number of components or more zero-failure tests, to minimise the total expected costs. For example, it is optimal to test 131 tasks (without failures) and to

have 3 components installed in the system for $C = 100,000$ whereas, it is optimal to test 70 tasks with 4 components in the system when $C = 200,000$ instead, in order to achieve the same reliability level, $p = 0.95$ with $m = 2$ and $Q = 150$.

	$C = 10,000$						$C = 100,000$		$C = 200,000$	
	$Q = 150$		$Q = 300$		$Q = 3000$		$Q = 150$		$Q = 150$	
m	n	y	n	y	n	y	n	y	n	y
1	47	3	139	2	139	2	104	3	131	3
2	59	3	59	3	196	2	131	3	70	4
10	103	3	103	3	427	2	89	4	106	4
100	88	4	222	3	221	3	160	4	192	4
150	98	4	253	3	253	3	178	4	104	5

Table 5.8: Optimal n and y for $x = 2$, $p = 0.95$ and $c = 1$.

Table 5.9 presents solutions to the above constrained optimisation problem, with the cost of process failure fixed at $C = 10,000$. It clearly illustrates that an increase in the cost of testing leads to fewer zero-failure tests to be performed, and more components installed where necessary to meet the reliability requirement. For example, comparing optimums n and y for $C = 10,000$, $Q = 150$ and $c = 1$, in Table 5.8 with the corresponding optimums in Table 5.9 for $c = 2$ illustrates that increasing c, namely from 1 to 2, decreases n with no change required on the optimal y for the values of m considered. However, for large c it might be more effective to test fewer tasks, with larger y, such as shown in Table 5.9 for the case where $Q = 150$ and $c = 24$ in comparison with, for example the case where $Q = 150$ and $c = 6$.

Table 5.10 presents the optimal n and y for $p = 0.99$, for a few situations also presented in Tables 5.8 or 5.9. Comparison of the results in these tables shows that increasing p can lead to an increase in the optimal n, which of course only occurs if the reliability constraint is active for $p = 0.99$, but for the same reason it can also lead to an increase of the optimal number of components, which simultaneously allows a smaller number of zero-failure tests. For example, in Tables 5.9 and 5.10 with $Q = 300$ and $c = 6$, the optimal n and y for $m = 1$, where the reliability constraint is not active, do not change for both $p = 0.95$ and $p = 0.99$, but for $m = 10$, where the reliability constraint is active for $p = 0.99$, the optimal n and y

	$c=2$				$c=6$				$c=24$			
	$Q=150$		$Q=300$		$Q=150$		$Q=300$		$Q=150$		$Q=300$	
m	n	y	n	y	n	y	n	y	n	y	n	y
1	37	3	98	2	25	3	25	3	10	4	15	3
2	47	3	47	3	31	3	31	3	12	4	19	3
10	81	3	81	3	29	4	54	3	14	5	19	4
100	73	4	73	4	33	5	53	4	17	6	23	5
150	81	4	81	4	36	5	59	4	19	6	25	5

Table 5.9: Optimal n and y for $x = 2$, $p = 0.95$ and $C = 10,000$.

change with p.

	$c=1$				$c=6$	
	$Q=150$		$Q=300$		$Q=300$	
m	n	y	n	y	n	y
1	47	3	198	2	25	3
2	59	3	59	3	32	3
10	103	3	103	3	29	4
100	88	4	239	3	58	4
150	98	4	294	3	67	4

Table 5.10: Optimal n and y for $x = 2$, $p = 0.99$ and $C = 10,000$.

We want to emphasize that one should be careful in determining the optimal integer-valued solution to this constrained optimisation problem. For example, for $m = 10$, $c = 6$ and $Q = 300$ in Table 5.10, the minimum total costs for real-valued n^* and y occurred at $n = 74.0041$ and $y = 3$, but the integer-valued optimal solution is at $y = 4$ and $n = 29$, with total expected costs 1444.29, while choosing $y = 3$ and $n = 75$ leads to expected costs 1447.50, and the values $y = 3$ and $n = 74$ would not satisfy the reliability requirement.

Example 5.7.2

In this example we illustrate the optimal $\underline{n}$ and y, considering multiple independent types of tasks required to be dealt with in the process after testing. Table 5.11 presents the optimal $\underline{n}$ and y for $x = 2$ components required for system functionality,

for different cost Q per component installed. Again using Beta prior distributions with $\alpha_i = 1$ and $\beta_i = 0$ for all types of tasks, and $p = 0.95$, we consider the following cases:

(1) $k = 3$, $\underline{m} = (1, 3, 6)$, $\underline{c} = (1, 1, 1)$, $C = 10,000$

(2) $k = 3$, $\underline{m} = (1, 3, 6)$, $\underline{c} = (20, 50, 50)$, $C = 10,000$

(3) $k = 3$, $\underline{m} = (1, 3, 6)$, $\underline{c} = (20, 50, 50)$, $C = 1,000,000$

	$\underline{n},\ y$		
Case(s)	$Q = 300$	$Q = 1,000$	$Q = 3,000$
(1)	$(47, 68, 86),\ 3$	$(47, 68, 86),\ 3$	$(201, 347, 489),\ 3$
(2)	$(8, 9, 10),\ 5$	$(12, 12, 15),\ 4$	$(26, 28, 35),\ 3$
(3)	$(14, 14, 16),\ 8$	$(20, 21, 23),\ 6$	$(41, 43, 51),\ 4$

Table 5.11: Optimal n_i, for all i, and y for $x = 2$ and $p = 0.95$.

Comparing Cases (2) and (3) in Table 5.11 with Cases (1) and (2) in Table 3.4, where a unit-system was considered, illustrates that higher built-in redundancy (larger y) requires fewer zero-failure tests. Cases (1) and (2) in Table 5.11, illustrate the fact that increasing testing costs c_i per test of type i, reduces the optimal number of zero-failure tests n_i, and maybe increases the optimal y. For example, for the reliability requirement with $p = 0.95$ and for $Q = 1,000$, this leads to the optimal solution of installing $y = 4$ components and $\underline{n} = (12, 12, 15)$ zero-failure tests. If we increase the cost per component to $Q = 3,000$ in the same setting, the optimum solution is $y = 3$ components with $\underline{n} = (26, 28, 35)$ tests. If we consider a 2-out-of-2 system for this case, the optimal zero-failure test numbers are $(296, 322, 454)$. This illustrates again that the option of building in redundancy can greatly reduce the test requirements and corresponding expected costs.

Moreover, Table 5.11 illustrates that for larger cost Q per component installed , it might be optimal to use fewer components, with more zero-failure tests to demonstrate the same level of reliability. Comparing Case (2) and (3) illustrates that increasing process failure cost C requires an increased number of components y

and/or an increased number of zero-failure tests $\underline{n}$ to minimise the total costs and to demonstrate the required reliability level.

Of course, if the reliability constraint is active, with higher required reliability level p, one needs to install more components in the system or perform more zero-failure tests to demonstrate this required level of reliability. For example, in Case (1) with $Q = 300$, the optimal solution to demonstrate $p = 0.99$ is to use the same number of components, namely $y = 3$, but more zero-failure tests, namely $\underline{n} = (48, 70, 89)$ in order to achieve this level of reliability. For $p = 0.95$, as presented in Table 5.11, the constraint is not active at the optimal solution, with reliability equal to 0.9894, whereas for $p = 0.99$ it is active.

5.8 Concluding remarks

In this chapter we studied Bayesian reliability demonstration for systems with built-in redundancy, which we denoted by x-out-of-y systems, including the special cases of parallel and series systems. we only considered testing at the 'component' level. The same settings as introduced in Chapter 2 were considered. The methods presented in this chapter fit in the general Bayesian reliability demonstration framework discussed in the previous chapters where unit-systems ($x = y = 1$) were considered. We only considered reliability demonstration in terms of failure-free periods after testing. Components were regarded as either functioning or not (pass or fail), and assumed to be independent and exchangeable. As in Chapter 3, we assumed that process failures are catastrophic and may incur a very high cost. We also studied the optimal combinations of the number of zero-failure tests, and the number of components to be used, to minimise expected total costs while meeting a specified reliability requirement. In practice, testing opportunities to demonstrate high reliability may be restricted, for example due to budget and time constraints. Such practical restrictions can often easily be translated into constraints in the optimisation problems presented in this chapter, as in Chapter 4, without causing too many difficulties for the numerical computation of optimal test numbers and redundancy levels. Clearly, if testing is restricted due to time, optimal solutions would probably

involve installing more components in the system, whereas budget constraints may lead to more testing if components are relatively expensive. Results in this chapter are in line with the results achieved in Chapters 3 and 4. As redundant components are meant to be used to increase systems reliability, this chapter is a natural extension of the study in Chapters 2 to 4 where a *unit*-system was considered.

Throughout this thesis, we focused explicitly on reliability demonstration expressed in terms of failure-free periods, and the expected number of failures in the process after testing. We assumed that testing can be performed separately, in the sense that testing does not consist of actually observing the same process for a period of time, hence we can indeed choose the required test numbers. Most methods presented in the literature so far, see Section 1.3, have focused on characteristics such as 'mean time to failure', which are unobservable, and for related inferences one often ends up with fewer required tests than in our approach.

Throughout, where highly reliable systems are considered, we explicitly assumed that testing reveals zero failures. If a failure happens to occur during testing, the system may need to be redesigned, and testing may have to be reconsidered all over again. For the process after testing, we considered both the cases that the number of failures in the process are deterministic and random, including tasks arriving according to a Poisson processes. It turns out that the deterministic case is the worst case in the sense that it requires most tasks to be tested, and that the Poisson case is almost as bad as the deterministic case. We also found that the actual optimal test numbers, according to the different optimality criteria considered, are quite insensitive to the actual criterion used for as far as the dependence on the m_i (or λ_i for the random case) and c_i are considered, which does not occur on the work on reliability demonstration, presented in the literature. This study suggested an approximate relation between the optimal n_i, for generally x-out-of-y systems including the unit-systems, and m_i and c_i. It appears to be the case that the optimal number of zero-failure tests for tasks of type i is approximately proportional to $(\frac{m_i}{c_i})^{\frac{1}{y-x+2}}$, which generalizes the approximate relation for unit-systems (3.19), discussed in Sections 3.4 and 4.4. Unfortunately, we have not been able to prove this, and the additional interesting results on comparisons for the optimal test numbers

according to the different optimality criteria mentioned above, analytically, due to the complexity of the general expressions when focusing on probabilities of failure-free periods as considered in chapters 3 and 5.

Throughout, we have restricted attention to Beta priors, which simplifies computations and enables interpretation of the hyperparameters. One could use any prior distribution if one feels it would better reflect prior knowledge, and use powerful computational methods (e.g. MCMC) for the final inferences. It is clear that, in our formulations with zero-failure testing, the β_i only influence the required test numbers a little, in the sense that the sums of the required n_i and β_i remain approximately constant, so indeed the choice of β_i immediately translates as assuming you have already seen β_i successful tests of tasks of type i. Hence, from the perspective of reliability demonstration, we feel it is appropriate to advocate the use of the value $\beta_i = 0$. The α_i play a far more important role. Effectively, one can interpret this as 'having seen α_i tests failing', and then demanding sufficiently many (n_i) tests without failure such that, on the basis of the combined information represented by this prior and test information, the reliability requirement is met. In situations of high reliability, one would naturally wish to use very small values of α_i, and a key problem is that the total required number of tests depends very strongly on α_i. From the perspective of reliability demonstration, we are in favour of the value $\alpha_i = 1$, which however may lead to many tests being required, and as such can be considered as conservative. This choice of hyperparameters coincides with results in [14], but from a different foundational perspective, and was also advocated by Hartigan [29] for cautious Bayesian inferences in similar settings. Generally, these inferences are very sensitive to the choice of prior distribution, so one must be very careful with interpretation of non-informativeness of priors.

As the more established methods for reliability demonstration, as presented in the literature do not take any aspects of the process after testing explicitly into account, we have not compared our results with optimal testing according to those methods. We believe that our Bayesian predictive approach is easier to interpret, of course it would be valuable to compare different methods for reliability demonstration in a detailed study, which we regard as an interesting topic for future research.

5.9 Further suggestions

Our approach, for which restriction to tests without failures was crucial, can be extended in many ways, leading to interesting and challenging research topics. For example, if one could allow failures in testing, sensitivity with regard to hyperparameters would probably decrease, but one would have to take prior inferences on probabilities of such failures into account. Even more important would be consideration of retesting and of possible repair during the process considered. For example, Tal *et al.* [58] introduce a testing policy depending on test, analyse and fix. In other words, consideration of activities to improve the system's functioning when a failure is observed during testing, which would also lead to far more complex analyses. Guida and Pulcini [27] considered the effectiveness of design modification by using what they call an improvement factor which measures the ratio of the average unreliability of the new product, at a given time, to the corresponding average of the past product.

One could also consider dependence of multiple types of tasks, which could be modelled via Bayesian graphical models [60], where inclusion of positive correlations, e.g. due to some shared functionality for all tasks, leads to reduced test numbers (as an extreme case, this allows consideration of 'one-test-tests-all' failure modes). Again, this inclusion would also require far more complex analyses. Moreover, allowing more general processes after testing, e.g. nonhomogeneous Poisson processes where the arrival rates depend on the performance of the system, different cost structures, e.g. testing costs not linear in the number of tests, and indeterminacy about the process, which could be modelled via imprecise probabilities [13], increase the practicality of our approach.

The assumption of exchangeable components in Chapter 5, that all deal simultaneously with each task, needs to be generalized to enhance applicability of this method. For example, Bayesian reliability demonstration should be studied for more complex systems [39, 40] with different components, and subsystems dealing with particular tasks. In principle, if the system structures are known, this is relatively straightforward due to the direct predictive formulation of the reliability functions in our approach, and the zero-failure assumption for determining required test num-

bers. There is an interesting added complexity if tests can also be performed at system or subsystem level, as zero failures at that level may not imply that all components functioned successfully. Conceptually, the Bayesian statistical framework has no difficulty in dealing with such data. For example, in our approach for x-out-of-y systems, the likelihood function can incorporate a successful test at system level, which translates to 'at least x components functioned for this particular task'. This would, however, increase computational complexity, and may therefore require the use of more sophisticated computational methods [22].

It is also interesting to study the optimal combinations of numbers of tests, and the minimal required number of components, x, required for the system to function with meeting the specified reliability requirement.

Another way to increase system reliability is via standby systems [1, 9, 10] or so-called 'active redundancy' [44], which maximise the lifetime of the resulting systems. Due to the direct predictive formulation of our reliability targets, we expect it to be relatively straightforward to include such issues in our approach of zero-failure testing for Bayesian reliability demonstration.

We consider such topics very interesting and relevant for future research, and for applicability of such reliability demonstration methods. We consider this thesis as a first step in the direction of practically useful Bayesian predictive reliability demonstration methods, providing basic analyses and insights which we believe will be relevant beyond the simple settings discussed here.

Appendix A

A.1

We proof that $r(x,y|(n+1,0)) - r(x,y|(n,0)) \geq 0$, hence (5.5) is true. So, we need to show

$$\sum_{j=x}^{y} \binom{y}{j} \prod_{l=1}^{y-j} (\alpha + l - 1) \prod_{h=1}^{j-1} (z+h)[j(\alpha+z) - zy] = \frac{\Gamma(y-x+\alpha+1)\Gamma(y+1)\Gamma(z+x)}{\Gamma(x)\Gamma(z+1)\Gamma(y-x+1)\Gamma(\alpha)}. \tag{A.1.1}$$

Proof: The proof follows by backwards induction

1. For $x = y$

$$\begin{aligned} \text{LHS} &= \prod_{h=1}^{y-1} (z+h)[y(\alpha+z) - zy] \\ &= \alpha y \prod_{h=1}^{y-1} (z+h) = \alpha y \frac{\Gamma(z+y)}{\Gamma(z+1)}, \end{aligned} \tag{A.1.2}$$

where we defined $\prod_{\varnothing} = 1$.

$$\begin{aligned} \text{RHS} &= \frac{\Gamma(\alpha+1)\Gamma(y+1)\Gamma(z+y)}{\Gamma(y)\Gamma(z+1)\Gamma(\alpha)} \\ &= \alpha y \frac{\Gamma(z+y)}{\Gamma(z+1)}, \end{aligned} \tag{A.1.3}$$

which is equal to the LHS.

2. Suppose that equation (A.1.1) is true for $x = d + 1$ then, for $x = d$ we find

$$\begin{aligned}
\text{LHS} &= \sum_{j=d}^{y} \binom{y}{j} \prod_{l=1}^{y-j} (\alpha + l - 1) \prod_{h=1}^{j-1} (z + h)[j(\alpha + z) - zy] \\
&= \binom{y}{d} \prod_{l=1}^{y-d} (\alpha + l - 1) \prod_{h=1}^{d-1} (z + h)[d(\alpha + z) - zy] \; + \\
&\qquad \sum_{j=d+1}^{y} \binom{y}{j} \prod_{l=1}^{y-j} (\alpha + l - 1) \prod_{h=1}^{j-1} (z + h)[j(\alpha + z) - zy] \\
&= \frac{\Gamma(y+1)\Gamma(y-d+\alpha)\Gamma(z+d)[d(\alpha+z) - zy]}{\Gamma(d+1)\Gamma(y-d+1)\Gamma(z+1)\Gamma(\alpha)} \; + \\
&\qquad \frac{\Gamma(y-d+\alpha)\Gamma(y+1)\Gamma(z+d+1)}{\Gamma(d+1)\Gamma(z+1)\Gamma(y-d)\Gamma(\alpha)} \\
&= \frac{\Gamma(y+1)\Gamma(y-d+\alpha)\Gamma(z+d)[d(\alpha+z) - zy + (y-d)(z+d)]}{\Gamma(d+1)\Gamma(y-d+1)\Gamma(z+1)\Gamma(\alpha)} \\
&= \frac{\Gamma(y+1)\Gamma(y-d+\alpha)\Gamma(z+d)[d(y-d+\alpha)]}{\Gamma(d+1)\Gamma(y-d+1)\Gamma(z+1)\Gamma(\alpha)} \\
&= \frac{\Gamma(y+1)\Gamma(y-d+\alpha+1)\Gamma(z+d)}{\Gamma(d)\Gamma(y-d+1)\Gamma(z+1)\Gamma(\alpha)}. \qquad \text{(A.1.4)}
\end{aligned}$$

The RHS of (A.1.1) for $x = d$ is

$$\text{RHS} = \frac{\Gamma(y-d+\alpha+1)\Gamma(y+1)\Gamma(z+d)}{\Gamma(d)\Gamma(z+1)\Gamma(y-d+1)\Gamma(\alpha)}, \qquad \text{(A.1.5)}$$

which is equal to the LHS in (A.1.4). Hence, (A.1.1) is true.

A.2

To prove (5.19), we need to prove

$$\sum_{j=x}^{y} \frac{(n+j-1)!}{j!} = \frac{(n+y)!}{ny!} - \frac{(n+x-1)!}{n(x-1)!},$$

which follows by backwards induction.

1. For $x = y$

$$\text{LHS} = \frac{(n+y-1)!}{y!}.$$

$$\text{RHS} = \frac{(n+y)!}{ny!} - \frac{n+y-1}{n(y-1)!} = \frac{(n+y)! - (n+y-1)!y}{ny!} = \frac{(n+y-1)!}{y!},$$

which is equal to the LHS.

2. Suppose that (5.19) is true for $x = b + 1$, then, for $x = b$,

$$\begin{aligned}
\text{LHS} &= \sum_{j=b}^{y} \frac{(n+j-1)!}{j!} = \frac{(n+b-1)!}{b!} + \sum_{j=b+1}^{y} \frac{(n+j-1)!}{j!} \\
&= \frac{(n+b-1)!}{b!} + \frac{(n+y)!}{ny!} - \frac{(n+b)!}{nb!} \\
&= \frac{(n+y)!}{ny!} + \frac{n(n+b-1)! - (n+b)!}{nb!} \\
&= \frac{(n+y)!}{ny!} + \frac{(n+b-1)![n-n-b]}{nb!} \\
&= \frac{(n+y)!}{ny!} - \frac{(n+b-1)!}{n(b-1)!} = \text{RHS}.
\end{aligned}$$

Bibliography

[1] Andrews, J.D. & Moss, T.R. (2002). *Reliability and Risk Assessment*. 2nd Ed. Professional Engineering Publishing, London.

[2] Aven, T. & Jensen, U. (1999). *Stochastic Models in Reliability*. Springer, New York.

[3] Balaban H.S. (1975). Reliability demonstration: purposes, practices, and value. *Proceedings of the annual reliability and maintainability symposium*. 246-248.

[4] Barlow R.E. (2002). Mathematical reliability theory: from the beginning to the present time. *Proceedings of the Third International Conference on Mathematical Methods In Reliability, Methodology And Practice (MMR2002)*, Trondheim, Norway. http://www.math.ntnu.no/mmr2002/papers/

[5] Bedford, T. & Cooke, R. (2001). *Probabilistic Risk Analysis: Foundations and Methods*. Cambridge University Press.

[6] Bernardo J.M. & Smith A.F.M. (1994). *Bayesian Theory*. Wiley, Chichester.

[7] Bertolino, A. & Strigini L. (1996) Predicting software reliability from testing taking into account other knowledge about a program, *Proc. Quality Week'96*, San Francisco, Software Research Institute, San Francisco,May.

[8] Birnbaum, Z.W., Esary, J.D. & Saunders, S.C. (1961). Multi-component systems and their structures and their reliability. *Technometrics*, **3**, 55-77.

[9] Coit, D.W. & Liu, J. (2000). System reliabilility optimization with k-out-of-n subsystems. *International Journal of Reliability, Quality and Safety Engineering*, **7**, 129-142.

[10] Coit, D.W. (2001) Cold-standby redundancy optimization for nonrepairable systems. *IIE Transactions*, **33**, 471-478.

[11] Colombo, A.G. & Constantini, D. (1980), Ground-Hypotheses for Beta distribution as Bayesian prior. *IEEE Transactions On Reliability,* **29**, 17-21.

[12] Colombo, A.G. & Constantini, D. (1981), A rational reconstruction of the Beta distribution. *Statistica*, **41**, 1-10.

[13] Coolen, F.P.A. (2004). On the use of imprecise probabilities in reliability. *Quality and Reliability Engineering International*, **20**, 193-202.

[14] Coolen, F.P.A. & Coolen-Schrijner, P. (2005). Nonparametric predictive reliability demonstration for failure-free periods. *IMA Journal of Management Mathematics*,**16**, 1-11.

[15] Coolen, F.P.A., Coolen-Schrijner, P. & Rahrouh, M. (2004). On Bayesian zero-failure reliability demonstration. *Proceedings of the 5th IMA international Conference on Modelling in Industrial Maintenance and Reliability*, Salford, 69-74.

[16] Coolen, F.P.A., Coolen-Schrijner, P. & Rahrouh, M. (2005a). Bayesian reliability demonstration for failure-free periods. *Reliability Engineering and System Safety*, **88**, 81-91.

[17] Coolen, F.P.A., Coolen-Schrijner, P. & Rahrouh, M. (2005b). Bayesian reliability demonstration with multiple independent tasks. *IMA Journal of Management Mathematics*, to appear.

[18] Coolen, F.P.A., Coolen-Schrijner, P. & Yan, K.J. (2002). Nonparametric predictive inference in reliability. *Reliability Engineering and System Safety*, **78**, 185-193.

[19] Coolen, F.P.A., Goldstein, M. & Munro, M. (2001). Generalized partition testing via Bayes linear methods. *Information and Software Technology*, **43**, 783-793.

[20] Dyer, D. & Chiou, P. (1984). An information theoretic approach to incorporating prior information in Binomial sampling. *Comunications in Statisitics-Theory and Mehtods,* **13**, 2051-2083.

[21] French, S. & Rios Insua, D. (2000). *Statistical Decision Theory*. Arnold, London.

[22] Gamerman, D. (1997). *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference*. Chapman and Hall, London.

[23] Gaver, D.P. & Jacobs, P.A. (1997). Testing or fault-finding for reliability growth: a missile destructive-test example. *Naval Research Logistics*, **44**, 623-637.

[24] Gelman, A., Carlin, J.B., Stern, H.S. & Rubin, D.B. (1995). *Bayesian Data Analysis*. Chapman & Hall, London.

[25] Gradshteyn, I.S. & Ryzhik, I.M. (1980). *Table Of Integrals, Series, And Products; Corrects and Enlarged Edition*. Academic Press, Orlando.

[26] Grohowski, G. & Hausman, W.C. (1976). A Bayesian statistical inference approach to automotive reliability estimation. *Journal of Quality Technology*, **8**, 197-208.

[27] Guida, M. & Pulcini, G. (2002). Automotive reliability inference based on past data and technical knowledge. *Reliability Engineering and System Safety*, **76**, 129-137.

[28] Hardy, G.H., Littlewood, J.E. & Plya, G. (1988). *Inequalities, 2nd ed*. Cambridge, England: Cambridge University Press.

[29] Hartigan, J.A. (1983). *Bayes Theory*. Springer, New York.

[30] Johnson, V.E., Graves, T.L., Hamada, M.S. & Feese, C.S. (2003). A hierarchical model for estimating the reliability of complex systems. *Bayesian Statistics*, **7**, 199-213. Oxford: University press.

[31] Keller-McNulty, S. (2002). Reliability for the 21$^{\text{st}}$ century. *Proceedings of the Third International Conference on Mathematical Methods In Reliability, Methodology And Practice (MMR2002)*, Trondheim, Norway. http://www.math.ntnu.no/mmr2002/papers/

[32] Kvam, P.H. & Miller, J.G. (2002). Discrete predictive analysis in probabilistic safety assessment. *Journal of Quality Technology*, **34**, 106-117.

[33] Lee, P.M. (1997). *Bayesian Statistics (2nd ed.).* Arnold, London.

[34] Luenberger, D.G. (1973). *Introduction to Linear And Nonlinear Programming.* Addison-Wesley, Menlo Park.

[35] Marsden, J.D. & Hoffman, M.J. (1999). *Basic Complex Analysis (3rd ed.).* W. H. Freeman, New York.

[36] Mann, N.R., Schafer, R.E. & Singpurwalla, N. D. (1974). *Methods for Statistical Analysis of Reliability and Life Data.* Wiley, New York.

[37] Martz, H.F. & Waller, R.A. (1979). A Bayesian zero-failure (BAZE) reliability demonstration testing procedure. *Journal of Quality Technology*, **11**, 128-138.

[38] Martz, H.F. & Waller, R.A. (1982). *Bayesian Reliability Analysis.* Wiley, New York.

[39] Martz, H.F. & Waller, R.A. (1990). Bayesian reliability analysis of complex series/parallel systems of Binomial subsystems and components. *Technometrics; American Statisitical Association and the American Society for Quality Control*, **32**, 407-416.

[40] Martz, H.F., Waller, R.A. & Fickas, E.T. (1988). Bayesian reliability analysis of series systems of Binomial subsystems and components. *Technometrics;*

American Statisitical Association and the American Society for Quality Controle, **30**, 143-154.

[41] Mastran, D. V. (1976). Incorporating component and system test data into the same assessment: a Bayesian approach. *Operations Research*, **24**, 491-499.

[42] Mastran, D. V. & Singpurwalla, N. D. (1978). A Bayesian estimation of the reliability of coherent structures. *Operations Research*, **26**, 663-672.

[43] Meeker, W.Q. & Escobar, L.A. (1998). *Statistical Methods for Reliability Data*. Wiley, New York.

[44] Mi, J. (1999). Optimal active redundancy allocation in k-out-of-n system. *Journal of Applied Probability*, **36**, 927-933.

[45] *Military Handbook 781A: Reliability Test Methods, Plans and Environments*, DoD, Washington D.C., (1996).

[46] Moustafa, M.S. (2001). Availability of K-out-of-N:G systems with Exponential failures and general repairs. *Economic Quality Control*, **16**, 75-82.

[47] Myers, G. (1979). *The Art of Software Testing*, Wiley, New York, N.Y.

[48] Parzen, E. (1962). *Stochastic Processes.* Holdden-Day, INC. F & S Probability & Statistics. San Francisco.

[49] Percy, D.F. (2002). Bayesian enhanced strategic decision making for reliability. *European Journal of Operational Research*, **139**, 133-145.

[50] Rai, G. (1968). Compound distributions. *Defence Science JNL*, **18**, 21-24.

[51] Rahrouh, M., Coolen, F.P.A. & Coolen-Schrijner, P. (2005). Bayesian reliability demonstration for systems with redundancy. *Proceedings of the 16th Advances in Reliability Technology Symposium*, Loughborough University, Loughborogh, 277-288.

[52] Redmill, F. (1999). Why systems go up in smoke. *The Computer Bulletin*, 26-28.

[53] Sandoh, H. (1991). Reliability demonstration testing for software. *IEEE Transactions on Reliability*, **40**, 117-119.

[54] Springer, M. D. & Thompson, W.E. (1966). Bayesian confidence limits for the product of N Binomial parameters. *Biometrika*, **3**, 611-613

[55] Springer, M. D. & Thompson, W.E. (1969). Bayesian confidence limits for system reliability. *in Proceedings of the 1969 Annual Reliability and Maintainability Symposium*. New York: Institute of Electrical and Electronics Engineers, 515-523.

[56] Tal, O., Bendell, A. & McCollin, C. (2000). Short communication: a comparison of methods for calculating the duration of software reliability demonstration testing, particularly for safety-critical systems. *Quality and Reliability Engineering International*, **16**, 59-62.

[57] Tal, O., McCollin, C. & Bendell, T. (2001). Reliability demonstration for safety-critical systems. *IEEE Transactions on Reliability*, **50**, 194-203.

[58] Tal, O., McCollin, C. & Bendell, A. (2002). An optimal statistical testing policy for software reliability demonstration of safety-critical systems. *European Journal of Operational Research*, **137**, 544-557.

[59] Weiler, H. (1965). The use of Incomplete Beta functions for prior distributions in Binomial sampling. *Technometics*, **7**, 335-347.

[60] Wooff, D.A., Goldstein, M. & Coolen, F.P.A. (2002). Bayesian graphical models for software testing. *IEEE Transactions on Software Engineering*, **28**, 510-525.