Florian Mussner BSc

Aggregation to system reliability based on success run assumptions of component/failure mode tests

MASTERARBEIT

zur Erlangung des akademischen Grades
Diplom-Ingenieur

Masterstudium Technische Mathematik: Operations Research und Statistik

eingereicht an der

Technischen Universität Graz

Betreuer:
Univ.-Prof. Mag.rer.nat. Dr.rer.nat. Siegfried Hörmann
Institut für Statistik

Dipl.-Ing. Esther Lichtenegger
Uptime Engineering

Graz, August 2018

DANKSAGUNG

Diese Arbeit entstand in Kooperation mit UPTIME ENGINEERING in Graz, Österreich. An dieser Stelle möchte ich mich beim gesamten Team für die offene Aufnahme bedanken.

Einen besonderen Dank gilt meinen Betreuern Univ.-Prof. Mag.rer.nat. Dr.rer.nat. Siegfried Hörmann vom Institut Statistik an der Technischen Universität Graz und an Dipl.-Ing. Esther Lichtenegger von UPTIME ENGINEERING für die Unterstützung und den hilfreichen Anregungen, sowie für die konstruktive Kritik bei der Erstellung dieser Arbeit.

Mein größter Dank gilt meiner Familie, insbesondere meinen Eltern, welche mir das Studium ermöglicht und mich in all meinen Lebensentscheidungen immer wieder unterstützt haben.

Abschließend möchte ich mich bei meinen Freunden bedanken, die mich auf diesem Weg begleitet haben. Allen voran Sarah Baierl, welche mich immer wieder motiviert hat und mir stets Rückhalt gegeben hat.

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz,am ....

(Unterschrift)

ABSTRACT

The main focus of this thesis is deriving system reliability from component reliabilities. This thesis is realized in cooperation with UPTIME ENGINEERING GmbH.

A lot of papers in reliability engineering focus on reducing sample size or integrating former knowledge for reliability calculations. There are many approaches, often using maximum likelihood estimators or mean square estimators for parameter estimation. Those methods can not be used if the reliability demonstration testing is conducted with zero failures. As Uptime Engineering is focusing on assessing validation programs with zero failure assumption during early steps of the planning phase, it is only possible to use methodologies to calculate lower confidence limits of reliability for a certain confidence level for single components and certain failure modes.

The second chapter of this thesis is going to introduce the basics of reliability engineering. Afterwards there is a short literature review on elected subjects of reliability engineering. The fourth chapter explains the functionality of Uptime LOCATE$^{\text{TM}}$. It is a software application developed for a product validation process leading to reliability demonstration. As the current methods of Uptime LOCATE$^{\text{TM}}$ are insufficient for the proposed method of calculating system reliability, a different method to calculate reliability for components using existing structures is proposed in the fifth chapter. Using this reliability calculations, a way to optimize given test programs with Matlab is then discussed in the sixth chapter. The thesis concludes with a recap and possible future topics.

KURZFASSUNG

Der Fokus dieser Arbeit ist die Berechnung von Systemzuverlässigkeit anhand von Komponentenzuverlässigkeit. Diese Masterarbeit wurde durchgeführt in Kooperation mit UPTIME ENGINEERING GmbH.

Viele Abhandlungen in der Zuverlässigkeitstechnik konzentrieren sich auf die Reduktion von Stichprobengrößen oder das Integrieren von vorhandenen Wissen. Viele Ansätze verwenden dabei Maximum Likelihood oder Mean Square Schätzer, um die Parameter einer Lebensdauerverteilung zu schätzen. Diese Methoden können jedoch nicht verwendet werden, wenn es sich bei den Daten um ausfallsfreie Daten handelt. UPTIME ENGINEERING GmbH spezialisiert sich auf die Bewertung von Validierungsprogrammen. Diese sind oft schon in frühen Planungsphasen vorhanden und da wir nicht in der Lage sind, Ausfallzeiten genau vorherzusagen, wird bei der Validierungsplanung angenommen, dass keine Fehler während der Tests auftreten. Deshalb müssen Methoden verwendet werden, welche untere Konfidenzschranken für die Zuverlässigkeit von Komponenten berechnen.

Zuerst werden in dieser Arbeit die Grundlagen der Zuverlässigkeitsanalyse erklärt. Nach einer kurzen Literaturdurchsicht wird im vierten Kapitel die Funktionalität von der Software Uptime LOCATE$^{\text{TM}}$ erklärt und anhand eines Beispiels verdeutlicht. Da die bestehenden Methoden von Uptime LOCATE$^{\text{TM}}$ unzureichend sind für die vorgeschlagene Methode zur Systemzuverlässigkeitsberechnung, wird zunächst eine andere Methode vorgestellt, um Zuverlässigkeit zu berechnen, welche schon vorhandene Strukturen verwendet. Im sechsten Kapitel wird auf die verschiedenen Funktionen von Matlab eingegangen, um ein gegebenes Validierungsprogramm zu optimieren. Zum Abschluss erfolgt eine kurze Zusammenfassung der Arbeit und ein Ausblick auf zukünftige Themen wird vorgestellt.

Contents

List of Tables

List of Figures

List of Symbols

α	confidence level, probability of type II error, consumer's risk
β	shape parameter of the Weibull distribution
β_i	shape parameter for component i
γ	location parameter of the three-parameter Weibull distribution
η	scale parameter of the Weibull distribution
$\underline{\eta}$	lower confidence bound for the estimation of η
μ	Normal distribution mean
σ	Normal distribution deviance
λ	Exponential distribution parameter
Φ	transformation factor
ρ	weighing factor
τ	target type, e.g. survival probability, failure rate, etc.
θ	mean of the exponential distribution
κ	acceleration factor
$\kappa_{i,j}$	acceleration factor of test i for event/component j
$\kappa_{i,j,k}$	acceleration factor of test i for component j and failure mode k
a	parameter of the Beta distribution
a_0	parameter of the Beta distribution, a-priory information
A	constraint matrix
b	parameter of the Beta distribution
b_0	parameter of the Beta distribution, a-priory information
BE_i	base event i
C_i	component i
$d_{i,j,k}$	damage value of test i on component j and failure mode k
$\hat{d}_{i,j,k}$	standardized damage value of test i on component/failure mode j,k
$\hat{d}_{i,j,k}$	standardized damage value with respect to the reference usage
F_j	failure mode j
iq	inequality constraints vector for optimization problem
L_i	leave i of the fault tree, corresponding to BE or UE
LR_i	life-span ratio $\frac{t_i}{t_0}$
n_i	sample size of test i
$n_i^{\max}$	maximum sample size of test i
P_{CL}	confidence level, usually set to 0.9.
r_i	number of risks of component i
$R_{C_i,\mathrm{LCL},(1-\alpha)}(t)$	$(1-\alpha)$ lower confidence limit for reliability of component i at time t
$R_{\mathrm{S,LCL},(1-\alpha)}(t)$	$(1-\alpha)$ lower confidence limit for reliability of the system at time t

R_0	prior knowledge for the reliability (Beyer/Lauster method)
$R_{\text{ac},i}(t)$	actual reliability of component i at time t
$R_{\text{ac,S}}(t)$	actual reliability of the system at time t
$R_i(t)$	reliability of component i at time t
$R_{\text{S}}(t)$	reliability of the system at time t
$R_{\text{tar},i}(t)$	reliability target of component i at time t
$R_{\text{tar,S}}(t)$	reliability target of the system at time t
$\underline{R}_i(t_0\|v)$	demonstrable reliability of C_i at target time t_0 with validation program v
$\underline{R}_{\text{S}}(t_0\|v)$	demonstrable reliability of S at target time t_0 with validation program v
$\hat{R}_i(t_0\|v)$	projected reliability of C_i at target time t_0 with validation program v
$\hat{R}_{\text{S}}(t_0\|v)$	projected reliability of S at target time t_0 with validation program v
$\tilde{R}_i(t_0\|v)$	combined reliability of C_i at target time t_0 with validation program v
$\tilde{R}_{\text{S}}(t_0\|v)$	combined reliability of S at target time t_0 with validation program v
S	system
t_0	reliability target reference value
t_i	time of test i
$t_i^{\max}$	maximum test duration for a single test
$\bar{t}_i^{\max}$	maximum test duration for all tests of $\mathcal{T}_i$
$t^{\max}$	maximum overall test duration
T_i	test i
$\tilde{t}_{i,j,k}$	equivalent test duration of test i for component/failure-mode j, k
u_0	unit of the reference value of reliability target, e.g. hours, cycles, distance
u_R	unit of reliability target type τ
UE_i	undeveloped event i
v	validation program with times $t_1, t_2, \ldots$
V	validation program with tests $\mathcal{T}_1, \mathcal{T}_2, \ldots$
X	number of failures
X_i	number of failures in component i
X_{S}	number of failures in the system

Chapter 1

Introduction

Reliability engineering is an essential component of a good Product Lifecycle Management program. It consists of the application of engineering principles and techniques throughout a product life cycle. The goal is the evaluation of reliability of a product and determining areas for improvement. Although it is not possible to detect and eliminate all failures of a product, another goal of reliability engineering is to identify the most likely failures and trying to find solutions to reduce corrective measures.

With increasing product complexity and warranty periods, the use of reliability engineering is becoming more important. It can help to reduce the costs of corrective actions by locating possible defects in early steps of development. As most companies only have limited budget and testing time, extensive testing of the finished product is not viable. One possibility to reduce those costs is to test not only the whole system, but single components, as they are cheaper to produce and can reveal failures that would also occur in the system tests.

In addition to diagnostics and prognostics activities, the Uptime Engineering approach to reliability engineering addresses the creation, evaluation and optimization of product validation programs. For this, several fields of expertise have to be combined effectively. Product validation requires the knowledge of the expected reliability, the average customers usage behavior to get a reference profile, the technical expertise about the product as a system consisting of hierarchically structured assemblies and components, as well as its failure modes and mechanisms. Especially the physics of failure (PoF) need to be understood and modeled for the evaluation of damage models.

Uptime LOCATE$^{\mathrm{TM}}$ is a software designed for the optimization of validation programs. As corrective measures are cheaper in earlier steps of production, Uptime Engineering is starting their validation during the concept phase.

Uptime Engineering does not calculate the reliability of a product, but evaluates the quality of a validation program with various component and system tests. Those validation programs are planned long before production. Instead of real life data, the validation program consists of planned tests which might be subject to changes by changing duration or amount of tested objects. As it is almost impossible to predict failures, all tests are expected to have zero failures, a so called Success Run. This assumption complicates the process of calculating the lifetime of a product and therefore we can only evaluate the quality of a validation program.

1.1 Problem

We are able to make a statement about the failure performance of the tested product, only when many products of a test run fail . The usual methods for parameter estimation of the reliability functions like maximum likelihood or mean square estimator can only be applied if there are failures in the test data. Having no such failures, we are only able to calculate lower confidence limits. Thus the methods for calculating reliability in early stages of development bear insecurities.

Endurance tests are mostly used to predict the reliability of a product. With a large sample size and real conditions we can reduce the insecurities of our calculations. Due to financial and material restrictions, it is often not possible to get these requirements. Thus we have to develop methods to consider additional information, like expert knowledge or knowledge of forerunner projects. Currently the software Uptime LOCATE$^{\text{TM}}$ uses a method to include this additional knowledge and calculate lower confidence bounds for the reliability of component/failure-modes by adapting the parameters of a Beta distribution. The software is able to calculate the demonstrable reliability on a component/failure-mode level, but at this stage there is no method to aggregate the system reliability. Especially management holds interest for an easy way to compare validation programs on their reliability performance for the whole system.

1.2 Target of the thesis

Target of this master thesis is developing a methodology to aggregate system reliability based on component reliability. At the end, there should be one indicator measuring the quality of a validation program. Additionally, we should be able to optimize a given validation program by means of maximizing system reliability. The existing structures of Uptime LOCATE$^{\text{TM}}$ should be included in those calculations.

1.3 Structure of the thesis

The second chapter introduces the basic definitions of reliability engineering, i.e. the used distributions and methods to describe systems. The third chapter gives a short overview about typical methods used in reliability engineering and focuses on the method by A. Krolo and M. Kemmner. Chapter 4 describes the functionality of the software Uptime LOCATE$^{\text{TM}}$, the current structure and methods used to calculate reliability, accompanied by an example. As the methods used are insufficient for the method proposed in chapter 5, a different method to calculate reliability is proposed and in chapter 6 used to optimize a validation program by means of increasing system reliability.

Chapter 2

Definitions

Adapted from the works of Bertsche and Lechner (2004), Meeker and Escobar (1998), McCool (2012), Hamada et al. (2008) and Rinne (2009), the basics of statistic and reliability engineering are described in the following chapter.

2.1 Reliability

The ISO defines reliability as "[...] the ability of an item to perform a required function, under given environmental and operating conditions and for a stated period of time." (ISO, 1986)

The boundary conditions have to be defined precisely. The item is the observed system, e.g. a car or wind turbine. As for the required performance, we have to define when our product does not provide its intended function anymore and has a failure. The time interval has to be expressed in appropriate units u_0 depending on the item, e.g. calendar time, operating cycles, distance, etc. The given conditions describe the reference profile, i.e. usage of the customer, the environmental conditions, maintenance and stress levels.

There are several ways to characterize lifetime: failure probability, survival probability and hazard rate.

Definition 2.1. *Let T be a continuous random variable with cumulative distribution function (cdf) $F(t)$ on the interval $[0, \infty)$. $F(t) = P(T \leq t)$ is the **failure probability**, the probability that an object does not survive the time t. Alternatively, $F(t)$ can be interpreted as the ratio of units in the population that will fail before time t.*

Definition 2.2. *The probability density function (pdf) for a continuous random variable T is defined as $f(t) = dF(t)/dt$. In reliability engineering, it is often assumed that the lifetime density of a component can be described with a Weibull distribution.*

Definition 2.3. *The probability that an object survives beyond time t is called **survival***

probability *or* ***reliability*** $R(t)$ *and is defined as:*

$$R(t) = P(T > t) = \int_t^\infty f(u)du = 1 - F(t).$$

The reliability of a product is depending on the given conditions, e.g. environmental impact or different load conditions.

Definition 2.4. *The* ***hazard function****, also known as* ***hazard rate****, is defined by*

$$h(t) = \lim_{\Delta t \to 0} \frac{P(t < T < t + \Delta t | t < T)}{\Delta t} = \frac{f(t)}{R(t)}.$$

The hazard rate expresses the probability to fail in the next small time interval, given survival to time t.

The hazard rate is not only used to describe fatigue, but can also be used to describe early failure and random failure. It is possible to describe the failure performance of a component or machine. The typical trend of this curve, the so called bathtub curve, can be seen in Figure 2.1. This curve can be separated in three parts: the first section describes early failures, the second one describes random failures and the third one wearout failures.

The first section is characterized by a decreasing hazard rate. The risk for a product to fail decreases. Those failures are mostly caused by assembly error, material defect, faulty craftsmanship or a severe design flaw. Those quality related defects are often described as infant mortality.

During much of the useful life of a product, the hazard rate can be seen as constant, as failures are caused by random external influences, like shock, handling error, etc.

In the section of wearout and fatigue failures, we have an increasing hazard rate. Those failures are caused by wear, fatigue fracture, deterioration, etc.

Each of those section has different failure causes and we need appropriate actions to increase the reliability of our product. Many reliability studies focus on either the first or last section of this curve. In this thesis we focus on the third section, as we assume that all failures in our models are caused by wearout failures and the quality errors are already eliminated and not of our concern.

Definition 2.5. *The* $\mathbf{B_x}$ ***lifetime*** *is a measurement for the point of time by which $x\%$ of the population of our product has failed. In practice the most used are* $\mathbf{B_1}$*,* $\mathbf{B_{10}}$ *and* $\mathbf{B_{90}}$ *lifetime values to describe the degree of reliability of our product.*

2.2 Censoring

If we have the exact failure time for every member of our sample, we call it a complete or uncensored sample. Due to financial or time limits, it is common to have censored data

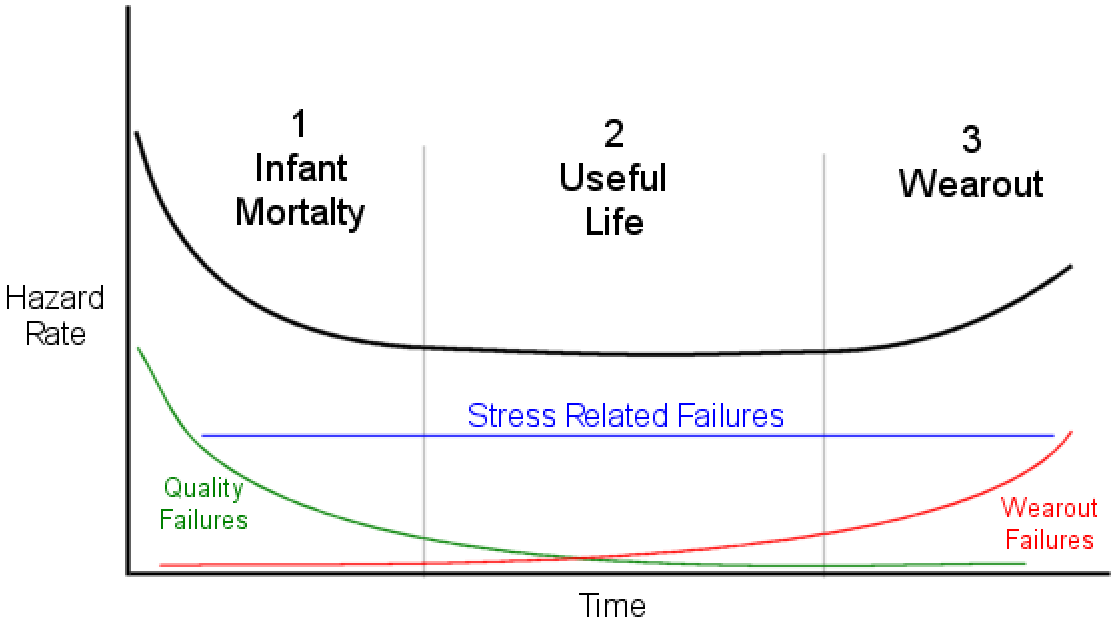


Figure 2.1: Bathtub curve hazard rate. Image downloaded from
http://www.reliabilityanalytics.com/blog/2011/08/31/bathtub-curve/ in June 2018

in reliability data analysis. This censoring restricts the ability to observe failure times exactly. We distinguish between left, right and interval censoring.

Left-censored observations occur if the unit has failed at the time of its first inspection. It is only known, that the unit failed before the inspection and we have an upper bound for the failure time.

Right censoring is common in reliability data analysis. This lower bound is often due to tests being terminated prior to failure for some units. Therefore we do not get a failure time. As this thesis assumes a Success Run, all of our data is going to be right censored.

Interval censoring occurs when a life test is periodically observed, e.g. once every hour. A failure is only known to have occurred during the last time the unit was observed, but no exact time is known.

Censoring of data may be random or by design. Planned censoring is distinguished by the condition to stop the tests. Type I censoring restricts the testing time. After a set time, the tests are terminated, even if there is no failure yet. Type II censoring has a predetermined number of failures: after a set amount of units have failed, the tests are terminated. The latter is not of our interest, as we assume that all our tests are terminated at a fixed time and with no failures.

2.3 Statistical Distributions

In this chapter, the most typical lifetime distributions used in reliability engineering are being examined. The best known distribution is the normal distribution, however it is hardly used in reliability theory. Bertsche and Lechner (2004) state that for a Weibull distribution with shape parameter $\beta = 3.5$ we have approximately a normal distribution. The exponential distribution is often used for electrical engineering, whereas in engineering, the Weibull distribution is mostly used. Thus our main focus lies on the Weibull distribution. The Beta distribution is used in Uptime LOCATE$^{\text{TM}}$ to calculate the projected reliability, as can be seen in Section 4.5.

2.3.1 Normal distribution

The most important distribution in applied statistics is the normal, or Gaussian distribution. It is a two-parameter distribution with the density function

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

where μ is the mean or expectation of the distribution and σ is the standard deviation. For the failure probability, reliability and hazard rate we have no closed form, so for the failure probability we have to integrate the density function

$$F(t) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{t} e^{-\frac{(\tau-\mu)^2}{2\sigma^2}} d\tau$$

and calculate the reliability and hazard rate respectively.
Figure 2.2 shows the failure functions for the normal distribution. The normal distribution starts at $t = -\infty$. As failure times only can have positive values, we can use the normal distribution only if failures in the negative time area can be neglected.

2.3.2 Exponential-distribution

The density function of the exponential distribution is monotone decreasing based on a inverse exponential function, see Figure 2.3. The distribution has only one parameter λ. For the relevant failure functions we have

$$\begin{aligned}
f(t) &= \lambda e^{-\lambda t} \\
F(t) &= 1 - e^{-\lambda t} \\
R(t) &= e^{-\lambda t} \\
h(t) &= \lambda.
\end{aligned}$$

The exponential distribution is defined by its constant failure rate λ. Thus the exponential distribution can be used to describe random failures, those occurring in the second section in Figure 2.1. There are only few cases where we can use the exponential distribution to describe the failure performance.

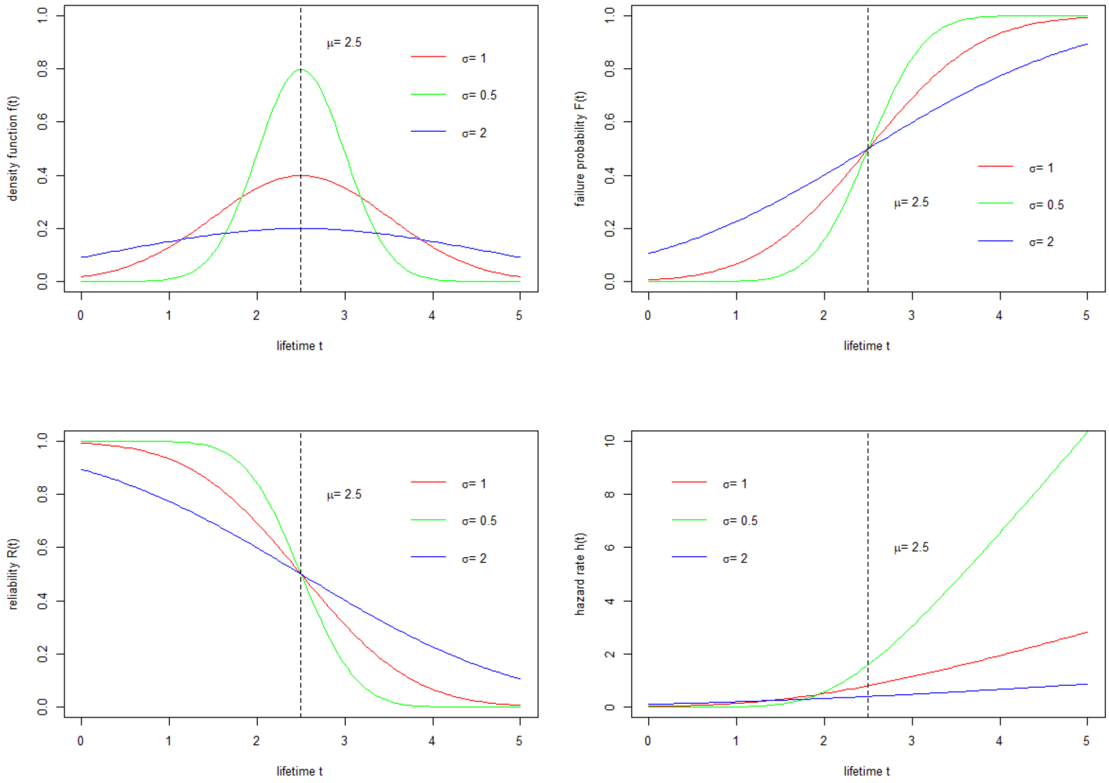


Figure 2.2: Normal distribution failure functions for different σ and a fixed $\mu = 2.5$.

2.3.3 Weibull-distribution

The Weibull distribution can be used to describe different failure performances. This can be seen in Figure 2.4. For the Weibull distribution we have to distinguish between the two-parameter and three-parameter Weibull distribution.

Two parameter Weibull

The Weibull distribution always consists of the shape parameter β and scale parameter η. Varying β causes the graph of the density function to change its form, as can be seen in Figure 2.4. The second parameter η is also called characteristic life and has the same unit as the abscissa. It has the special property that for $t = \eta$ we have $F(t) \approx 63,2\%$ and $R(t) \approx 36,8\%$. In Figure 2.5 we can see the Weibull density function for different values of the scale parameter with a fixed shape parameter of $\beta = 2$. The two parameter Weibull

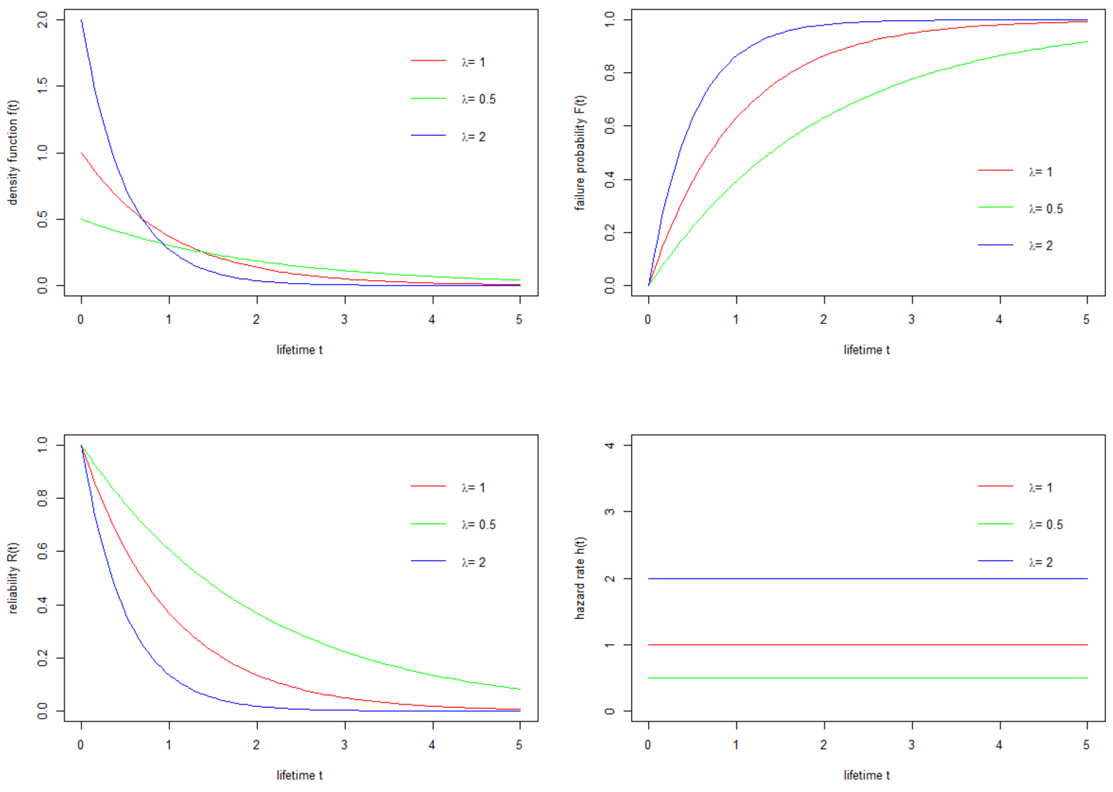


Figure 2.3: Exponential distribution failure functions for different λ.

distribution can be described with:

$$\begin{aligned} R(t) &= e^{-\left(\frac{t}{\eta}\right)^{\beta}} \\ F(t) &= 1 - e^{-\left(\frac{t}{\eta}\right)^{\beta}} \\ f(t) &= \frac{dF(t)}{dt} = \frac{\beta}{\eta}\left(\frac{t}{\eta}\right)^{\beta-1} e^{-\left(\frac{t}{\eta}\right)^{\beta}} \\ h(t) &= \frac{f(t)}{R(t)} = \frac{\beta}{\eta}\left(\frac{t}{\eta}\right)^{\beta-1}. \end{aligned} \tag{2.1}$$

Three parameter Weibull

Additionally to the two parameters mentioned before, the three parameter Weibull distribution has a location parameter γ. Changing γ when the other parameters are held constant will result in a parallel movement of the density curve over the x-axis, as can be seen in Figure 2.6. Thus γ is also called shift parameter. For the two parameter Weibull distribution we have $\gamma = 0$.

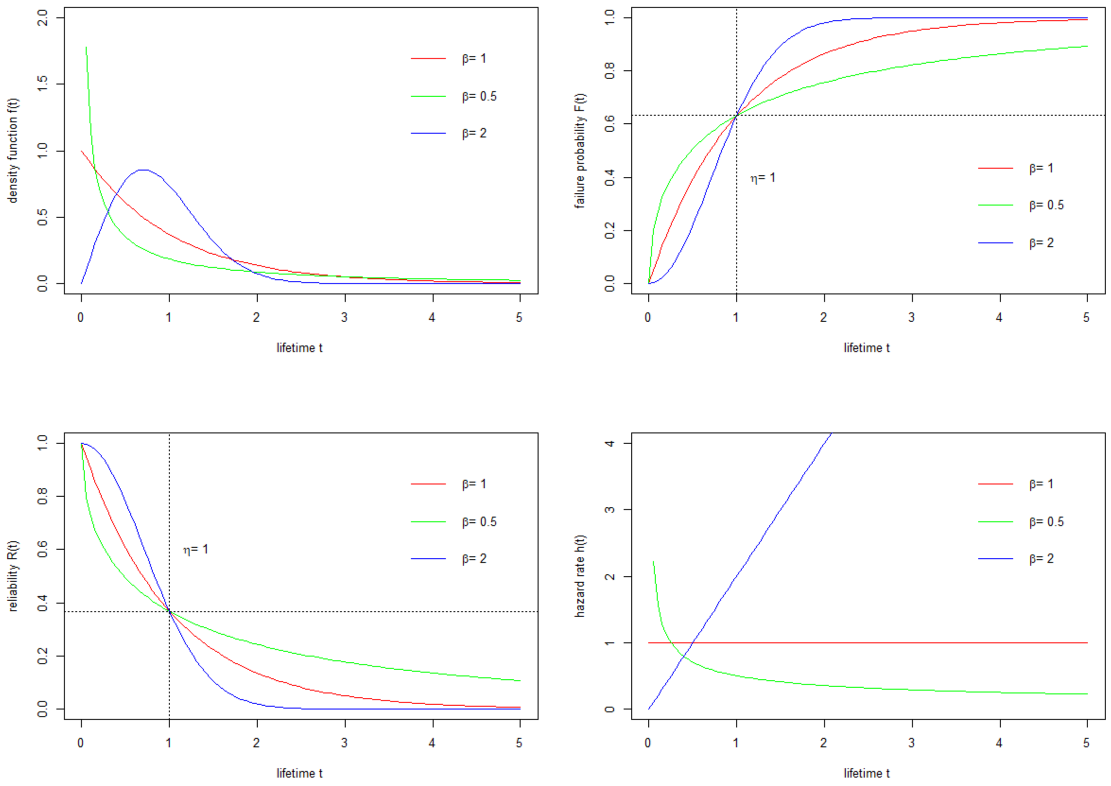


Figure 2.4: Weibull distribution failure functions with a fixed characteristic life $\eta = 1$, location parameter $\gamma = 0$ and different shape parameters β.

We have the following formula for the tree parameter Weibull distribution:

$$\begin{aligned}
R(t) &= e^{-\left(\frac{t-\gamma}{\eta-\gamma}\right)^{\beta}} \\
F(t) &= 1 - e^{-\left(\frac{t-\gamma}{\eta-\gamma}\right)^{\beta}} \\
f(t) &= \frac{dF(t)}{dt} = \frac{\beta}{\eta-\gamma}\left(\frac{t-\gamma}{\eta-\gamma}\right)^{\beta-1} e^{-\left(\frac{t-\gamma}{\eta-\gamma}\right)^{\beta}} \\
h(t) &= \frac{f(t)}{R(t)} = \frac{\beta}{\eta-\gamma}\left(\frac{t-\gamma}{\eta-\gamma}\right)^{\beta-1}.
\end{aligned} \tag{2.2}$$

Depending on the shape parameter β, we are able to describe all three sections of the bathtub curve. For $\beta < 1$, we have a decreasing hazard rate. Having a large failure probability at the beginning, the failure rate decreases over time. We can use this to describe the infant mortality. If the shape parameter $\beta = 1$, the Weibull distribution is simplified to be the exponential distribution. For $\beta > 1$, the hazard rate is increasing, thus we are able to describe products which are subject to wear and fatigue.

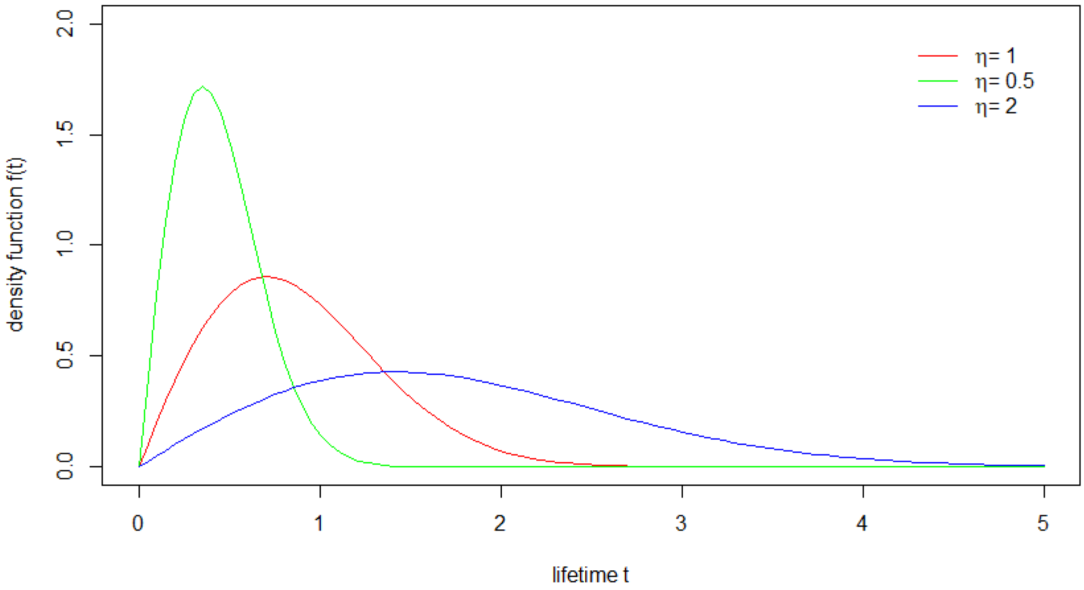


Figure 2.5: Two-parameter Weibull density functions with a fixed shape parameter $\beta = 2$ and different scale parameters η.

Weibull paper

The failure probabilities in Figure 2.4 have a sigmoid curve progression. With a special Weibull probability paper we are able to plot the failure probability $F(t)$ of the two-parameter Weibull distribution as a straight line, see Figure 2.7. We are then able to illustrate the failure performance in a graphical way. The transformation of the curve to the straight line is due to scaling the abscissa and ordinate. The abscissa is logarithmic scaled, while the ordinate uses a double logarithmic scale

$$\begin{aligned} x &= \ln t \\ y &= \ln(-\ln(1 - F(t))) = \ln(-\ln(R(t))). \end{aligned}$$

2.3.4 Beta-distribution

The Beta distribution qualifies for describing continuous random variables in the interval $[0, 1]$. We do not need it to describe a lifetime distribution, but it is used to determine the reliability for a fixed point of time. In Section 3.3 we need the Beta distribution, as well as for the projected reliability calculation in Chapter 4.

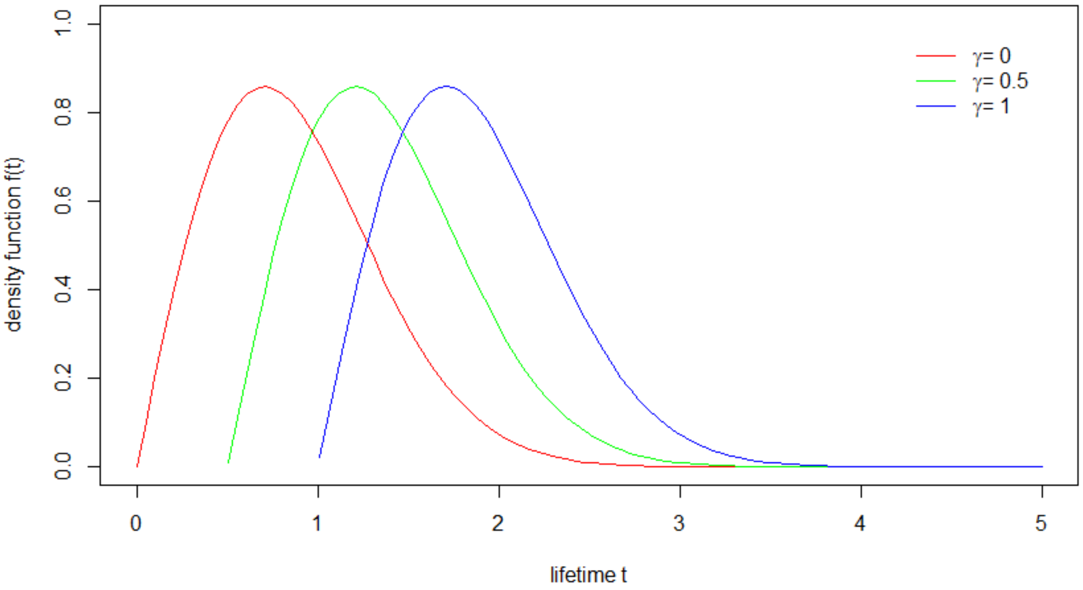


Figure 2.6: Three-parameter Weibull density functions with a fixed shape parameter $\beta = 2$, scale parameter $\eta = 1$ and different location parameter γ.

The density function of the Beta distribution is defined as:

$$f(x) = \begin{cases} \frac{1}{\beta(a,b)} x^{a-1}(1-x)^{b-1} & 0 \leq x \leq 1;\ a,b > 0 \\ 0 & \text{else} \end{cases} \tag{2.3}$$

with the parameter a, b and the beta function $\beta(a, b)$:

$$\beta(a,b) = \int_0^1 x^{a-1}(1-x)^{b-1} \mathrm{d}x, a, b > 0,$$

which can be written subject to the gamma function Γ:

$$\beta(a,b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}.$$

For $n \in \mathbb{N}$ we have $\Gamma(n+1) = n!$. For identical values of a and b, we have symmetrical density with respect to $x = 0.5$. Figure 2.8 shows the density function of the Beta distribution for varying parameters. The expected value of the Beta distribution is

$$E(X) = \frac{a}{a+b}.$$

Another useful characteristic of the two parameter Beta distribution is its relationship to the binomial distribution.

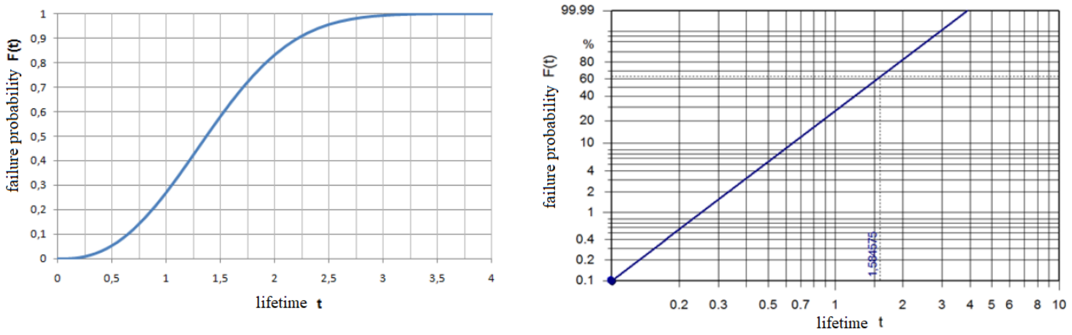


Figure 2.7: Failure probability F(t) for $\beta = 2.5$ and $\eta \approx 1.58$: normal display and in Weibull probability paper, (Jordan, 2011).

It might often be necessary to calculate the distribution function of the binomial distribution

$$F(k|p,n) = \sum_{i=0}^{k} B(i|p,n)$$

where $B(i|p,n)$ is the binomial probability mass function with p the success probability and k the exact number of successes in n trials. We then have the following relationship between the binomial and Beta distribution:

$$\sum_{i=0}^{k} \binom{n}{i} p^i (1-p)^{n-i} = \frac{n!}{(n-k-1)!k!} \int_0^{1-p} u^{n-k-1}(1-u)^k du \tag{2.4}$$

as n and k are integer. We use this relationship to calculate confidence levels.

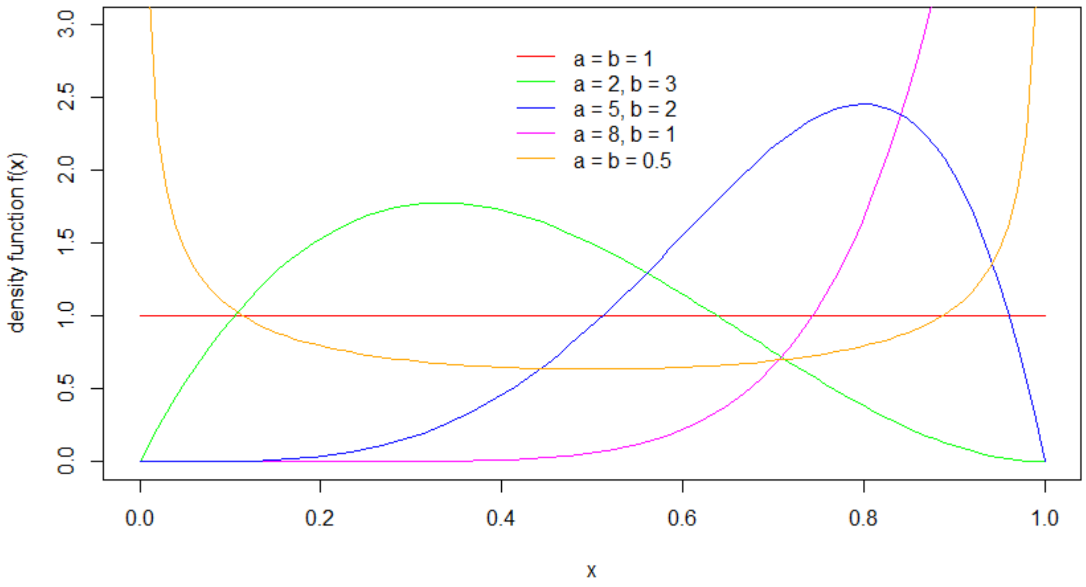


Figure 2.8: Beta density function for different parameters.

2.4 Producer/Consumer Risk

In reliability demonstration testing, there are two errors that can occur: either a product/system is denied, although it has sufficient reliability $R_{\text{ac,S}}(t_0)$ at time t_0, or a product is approved, although it does not reach its reliability target $R_{\text{tar,S}}(t_0)$. As overachievement of reliability targets is not considered a problem, we define the single-sided hypothesis as follows:

$$\begin{aligned} H_0 &: R_{\text{ac,S}}(t_0) \geq R_{\text{tar,S}}(t_0) \\ H_1 &: R_{\text{ac,S}}(t_0) < R_{\text{tar,S}}(t_0). \end{aligned}$$

The following errors can occur when making a decision based on test data or statistics.

- type I error: H_0 is rejected although it is true - producer's risk
- type II error: H_0 is not rejected, although it is false - consumer's risk

In reliability engineering, a type II error is more critical, since the real value of the reliability of the population is lower than what is stated by H_0. The type II error is often called consumer's risk, since it corresponds to the acceptance of a product that does not fulfill its reliability target. This can lead to high warranty costs and can damage the reputation of a company.

On the other hand, the type I error corresponds to rejection of a product that already reaches its reliability target. Although this often leads to higher development costs trying to improve the reliability of the product, it is still more favorable.

Reducing both errors leads to extensive testing and thus high costs. Therefore in practice it is typical trying to minimize the type II error. With certain methods, which are being described later, it is possible to compute lower $(1-\alpha)$ confidence limits of the reliability $R_{\mathrm{LCL},(1-\alpha)}$ for available test data. This means that when the result of a statistic acceptance test leads to not rejecting H_0, we can be confident with a probability of $(1-\alpha)$ that the real value of reliability is $R_{\mathrm{LCL},(1-\alpha)}$ or greater.

2.5 Confidence Level

We often have to deal with small sample sizes in practice and still have to give a statement about the product. The test evaluation only gives us information about the researched specimen, taking another sample results in a different outcome, even if the sample size and testing conditions stay the same. We are not interested in the samples, but in the basic population. Especially when dealing with small sample sizes, the results of the sample can differ quite a lot from the real behavior. We therefore introduce a confidence level.

Analyzing samples ideally results in a behavior that can be approximated with a mathematical probability distribution. The parameters of the distributions are then only estimators, as we are not able to get the real parameters.

For each estimator, we can specify an upper and lower limit, such that with a certain probability the real parameter of the basic population is within that interval. We call this a confidence interval. The confidence interval is always specified with a certain probability. A 90% confidence interval implies that a parameter is with a probability of 90% within that random interval and with a probability of 10% outside. We often call the probability in connection with the confidence interval as confidence level P_{CL}.

Having a Weibull lifetime distribution, we could give confidence intervals for the two parameters β and η. In practice, we do not give the confidence intervals for the parameters, but rather a confidence interval for the failure probability $F(t)$ for each point of time t. A detailed description can be found in Bertsche and Lechner (2004), we are only going to discuss the main idea.

Having a life test, we get the failure times for our system or component. For the evaluation with the probability paper, we only have the values for the abscissa, but not for the ordinate. We have to assign each failure at time t a failure probability $F(t)$. Let us assume we have a sample with $n = 30$ sample size. As a result we have 30 different random independent identical distributed failure times $T_1, T_2, \ldots, T_{30}$ with T_i being from a continuous distribution with the cdf F_T. The order statistics $T_{(1)}, T_{(2)}, \ldots, T_{(30)}$ are then also random variables, defined by sorting the values of $T_1, \ldots, T_{30}$ in increasing order, so

$$T_{(1)} \leq T_{(2)} \leq \cdots \leq T_{(30)}.$$

This ordered values are called rank values. The index corresponds to the rank. After the first rank value, $1/n$ of the sample has failed, after the second one $2/n$ etc. We could therefore estimate F with $\hat{F}$ such that we assign the i-th rank value a failure probability of $\hat{F}(T_{(i)}) = i/n$. Then $\hat{F}$ is the empirical distribution function. For a given t we may set $\hat{F}(t) = I/n$, where $I \leq t < I + 1$.

Note that we only considered the failure times of a single sample. Taking another sample with the same volume we get different failure times. The failure time of a rank value fluctuates within a range. A rank value can therefore be seen as a random variable, which can be assigned a distribution. By a well known result we can obtain the probability density function of the i^{th} order statistic as follows:

Theorem 1. *Let $T_1, T_2, \ldots, T_n$ independent and identical distributed random variables from a continuous distribution with cumulative distribution function F and probability density function f. Let*

$$T_{(1)} \leq T_{(2)} \leq \cdots \leq T_{(n)}$$

be the order statistics. Then, the probability density function of the i^{th} order statistic $T_{(i)}$ is:

$$f_{T_{(i)}}(y) = \frac{n!}{(i-1)!(n-i)!} F(y)^{i-1} f(y) [1 - F(y)]^{n-i}. \tag{2.5}$$

Here we assume that we know the distribution of the failure times, i.e. $f(y)$ and $F(y)$.

Usually we do not know the distribution of the failure times, but we want to identify those failure functions. The wanted failure probability of the failure times should have values between 0 and 1. It is easy to see that for $U_1, \ldots, U_n \overset{iid}{\sim} \text{Uniform}(0,1)$ we have

$$(F(T_{(1)}), \ldots, F(T_{(n)})) \overset{d}{=} (U_{(1)}, \ldots, U_{(n)}).$$

This ensures that the rank values are equally assigned the failure probability between 0 and 1. Using the transformation for (2.5) we get the density function for the failure probabilities of the rank values

$$f_{U_{(i)}}(u) = \frac{n!}{(i-1)!(n-i)!} u^{i-1} (1-u)^{n-i}.$$

This equates to a Beta distribution with the variable u and the parameters $a = i$ and $b = n - i + 1$.

Assessing failure times, we try to assign each failure time a singe failure probability and then fitting a line through the points in the Weibull paper. We therefore have to choose an adequate value from the spread of the failure probability. We can either choose the following: mean, median or mode. The value of those can be calculated with $f_{U_{(i)}}(u)$ or the Beta-distribution respectively as

$$\begin{aligned} \text{mean}(U_{(i)}) &= \frac{i}{n+1}; \\ \text{median}(U_{(i)}) &\approx \frac{i-0.3}{n+0.4}; \\ \text{mode}(U_{(i)}) &= \frac{i-1}{n-1}. \end{aligned} \tag{2.6}$$

For the median there is no closed form, (2.6) is therefore only an approximation. For the estimation of the failure probability $F(T_{(i)})$ at failure time $T_{(i)}$ the most common used are the mean or median. Thus we assign the failure time $T_{(i)}$ the failure probability

$$\begin{aligned}\hat{F}(T_{(i)}) &= \frac{i}{n+1} && \text{using the mean,}\\ \hat{F}(T_{(i)}) &\approx \frac{i-0.3}{n+0.4} && \text{using median.}\end{aligned}$$

Figure 2.9 illustrates an example with a sample size of $n = 30$ and different failure times $T_{(i)}$, $i = 1, ..., 30$. For $i = 25$ we get a median $\hat{F}(T_{(25)}) = 81.3\%$. Ideally we are able to plot a Weibull line trough the values of $(T_{(i)}, \hat{F}(T_{(i)}))$ as in Figure 2.9.

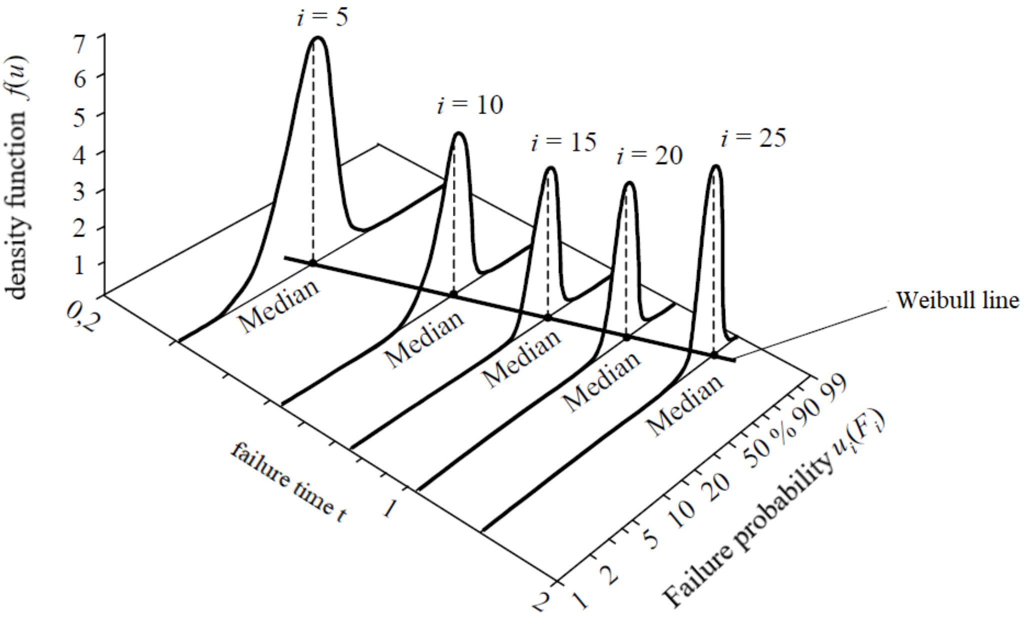


Figure 2.9: Density function of the rank failure probability with median values and the Weibull line, (Bertsche and Lechner, 2004).

At these positions $T_{(i)}$ we can also easily calculate the probability that $F(T_{(i)})$ is in between prespecified bounds (x_l, x_u). We have

$$P_{\text{CL}} = P(x_l \leq F(T_{(i)}) \leq x_u) = \int_{x_l}^{x_u} \frac{1}{\beta(i, n-i+1)} u^{i-1}(1-u)^{n-i} du.$$

There is no exact formula for the quantiles. Choosing

$$T_{(I)} \leq t < T_{(I+1)}$$

we define the confidence level for $F(t)$ with respect to the bounds x_l and x_u as

$$P_{\text{CL}} = \int_{x_l}^{x_u} \frac{1}{\beta(I, n-I+1)} u^{I-1}(1-u)^{n-I} du.$$

In most applications in reliability engineering we only have one-sided confidence intervals. Determining the confidence level, we have to differ between reliability or failure probability. If we require a certain reliability of our product, this value is the minimal value we are going to accept, so we do not want to undercut this value with a certain guarantee. On the other hand, demanding a failure probability for our product, this value is the maximum failure probability. Having a lesser failure probability is desirable, but we do not want to exceed this maximum value. Having set those values we are now able to calculate the confidence level.
For the failure probability, having a maximum allowed failure rate F_u, we can calculate the confidence level for $F(t)$ with $T_{(I)} \leq t < T_{(I+1)}$ by

$$P_{\mathrm{CL}} = P(0 \leq F(t) \leq F_u) = \int_0^{F_u} f_{U_{(I)}}(u)du.$$

For the reliability, we set the value R_l as being the reliability we want to obtain. With $R(t) = 1 - F(t)$ we can calculate the confidence level as follows:

$$P_{\mathrm{CL}} = P(R_l \leq R(t)) \leq 1) = \int_{R_l}^{1} \frac{1}{\beta(n-I+1, I)} s^{n-I}(1-s)^{I-1}\mathrm{d}s$$

with $T_{(I)} \leq t < T_{(I+1)}$.

Here we fix the failure probability, or rather the reliability and then calculate the confidence level. In practice, we often set a confidence level and we want to know the minimum reliability or maximum failure probability, so we use numerical methods to calculate the bounds R_l or F_u respectively when having a fixed confidence level P_{CL}. Section 2.8 describes the method of calculating the minimum reliability when assuming zero failures and setting a confidence level.

2.6 System reliability

A system is a collection of components combined to perform a given task. To determine whether a system is working or not, we have to understand the structural property of the system and the state of its components (e.g. working or not working). In this chapter we introduce the reliability block model diagrams and fault trees to visualize simple system structures and calculate system reliability.

2.6.1 Reliability Block Diagrams

In reliability engineering, we often model systems graphically. One of the most commonly used system representations in a risk and reliability analysis is the reliability block diagram. For this we have to make some assumptions:

- the system is not repairable, i.e. the first failure of the system ends the system life.
- the components can only have one of two conditions: "working" and "failed"
- the components are independent, i.e. the failure performance of one component does not influence the failure performance of another component.

Figure 2.10 shows how components are illustrated. A *series system* with n components is a system that is working if and only if all of its components are functioning. Figure 2.11 shows the representation of a series system with n components. The system is working if between entry (a) and exit (b) there is a connection in which all components are faultless. The failure of just one component leads to a failure of the system.

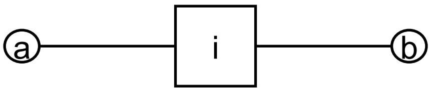


Figure 2.10: Component i in a reliability block diagram.

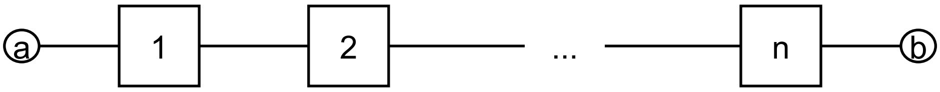


Figure 2.11: A system with n components in series.

Let $R_i(t)$ be the reliability of component i at time t and $R_\mathrm{S}(t)$ the reliability of our system. Then we get

$$R_\mathrm{S}(t) = R_1(t) \cdot R_2(t) \cdot \ldots \cdot R_n(t) = \prod_{i=1}^{n} R_i(t).$$

From this we can deduce the following: the reliability of a system is always smaller or equal the minimum reliability of our components, i.e. $R_\mathrm{S}(t) \leq \min_i R_i(t)$. Figure 2.12 shows the relationship between system reliability $R_{\mathrm{S}}(t)$ and individual component reliability $R_i(t)$ with $R_i(t) = R_j(t)$, $\forall i, j \in (1, ..., n)$, for different numbers n of components in series. We

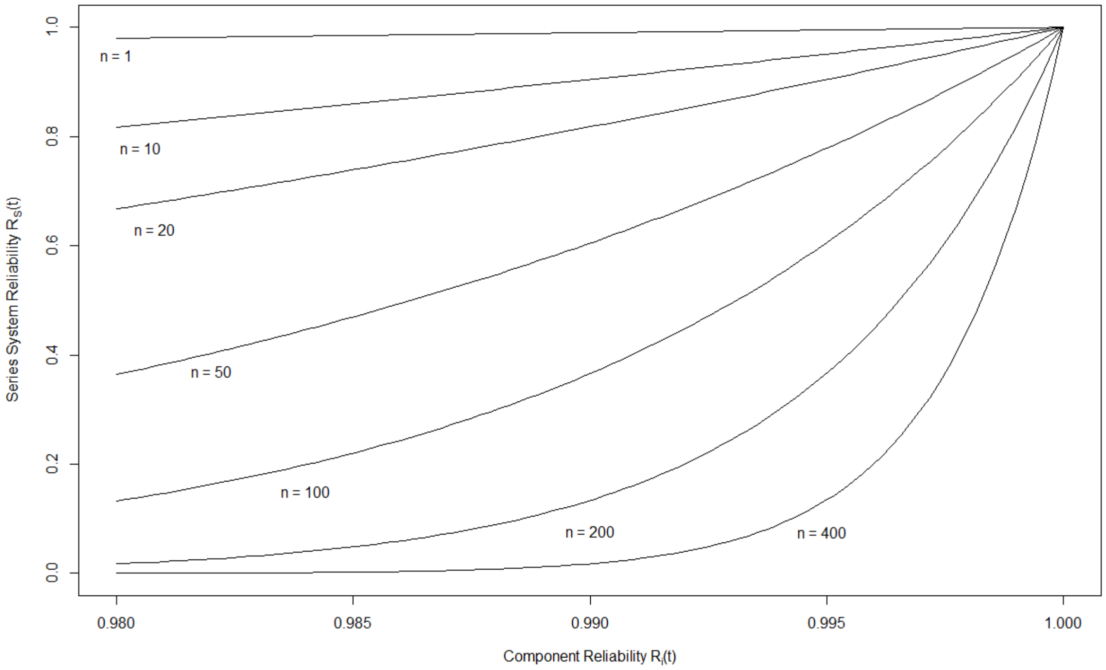


Figure 2.12: Reliability of a series system with n independent components with same component reliability (Meeker and Escobar, 1998).

can see that with a growing number of components, the reliability of those has to be high to get a high system reliability.

On the contrary, a *parallel system* is working as long as one of its n components is functioning. Figure 2.13 shows the representation of a parallel system with n components. The system is working if between entry (a) and exit (b) there is a at least one connection, i.e. one component is working. The failure of all components leads to a failure of the system. Again, let $R_i(t)$ be the reliability of component i at time t and $R_{\mathrm{S}}(t)$ the reliability of our system. Then we get

$$\begin{aligned} R_{\mathrm{S}}(t) &= 1 - (1 - R_1(t)) \cdot (1 - R_2(t)) \cdot ... \cdot (1 - R_n(t)) \\ &= 1 - \prod_{i=1}^{n} (1 - R_i(t)). \end{aligned}$$

The reliability $R_{\mathrm{S}}(t)$ of such a system is at least as high as the highest component reliability $R_i(t)$, i.e. $R_{\mathrm{S}}(t) \geq \max_i R_i(t)$. There is also the possibility of mixed reliability block diagrams, as seen in Figure 2.14.
Most systems in engineering are having a series system structure, as the integration of redundancies is costly (Bertsche and Lechner, 2004).

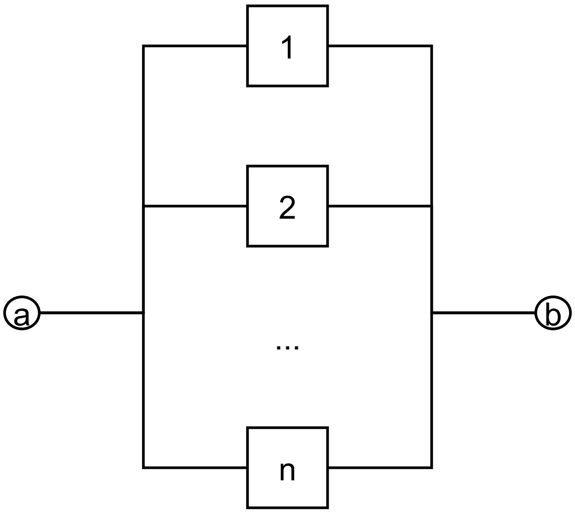


Figure 2.13: Parallel system with n components.

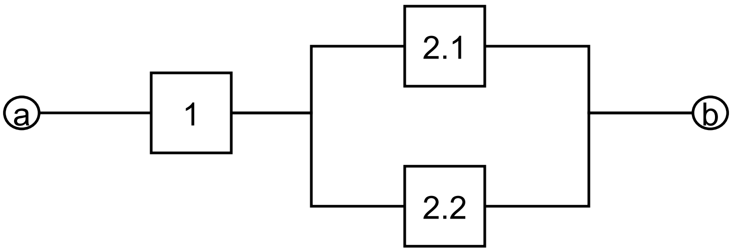


Figure 2.14: Combination of series and parallel structure.

2.6.2 FTA (Fault tree analysis)

Another commonly used method for system representation in risk and reliability engineering is the fault tree analysis (FTA). It is used to determine internal or external causation, which are reasons for a critical event, typically a failure of our system. This logic diagram illustrates how states of the systems components are linked with the state of the system. We use logic gates as graphical symbols, see Figure 2.15, to represent the connections between the components and the system. We only discuss the most basic logic gates as can be seen in Hamada et al. (2008). A *basic event* is an initiating failure. It requires no further breakup. Fault trees describe the functioning of a system to the decomposition of its basic events. Those basic events are mostly component failures. If we consider a basic event insignificant or we have a lack of information to further develop it, we define it as an *undeveloped event*. An *intermediate event* is a fault that arises when one or more of the previous faults have occurred. Logic gates are used to connect the intermediate events with its antecedent faults. Most commonly used are the AND gate and the OR gate. The intermediate event is the output of the gate, the previous events are the inputs. They are either undeveloped, basic or intermediate events. With an AND gate, the output occurs only when all of the inputs fail. With an OR gate, one fault in the inputs leads to a fault in the output.

When a fault tree only contains AND and OR gates, we can use the former mentioned reliability block diagram as a equivalent description. In Figure 2.16 we can see the relationship between FTA and reliability block diagrams.
As we consider only serial systems in our analysis, we omit the logic gates for further illustrations of our fault tree.

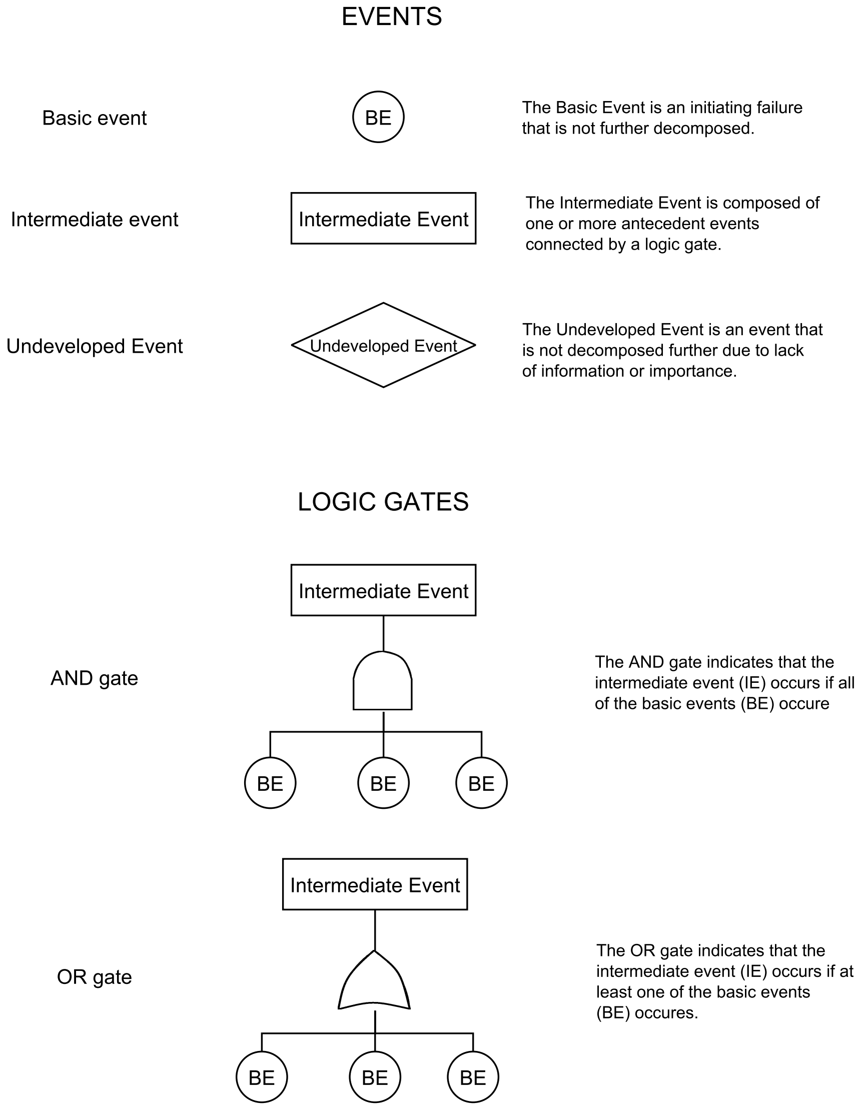

EVENTS
Basic event
BE
The Basic Event is an initiating failure that is not further decomposed.
Intermediate event
Intermediate Event
The Intermediate Event is composed of one or more antecedent events connected by a logic gate.
Undeveloped Event
Undeveloped Event
The Undeveloped Event is an event that is not decomposed further due to lack of information or importance.
LOGIC GATES
Intermediate Event
AND gate
The AND gate indicates that the intermediate event (IE) occurs if all of the basic events (BE) occure
BE
BE
BE
Intermediate Event
OR gate
The OR gate indicates that the intermediate event (IE) occurs if at least one of the basic events (BE) occures.
BE
BE
BE

Figure 2.15: Some common fault tree symbols, (Hamada et al., 2008).

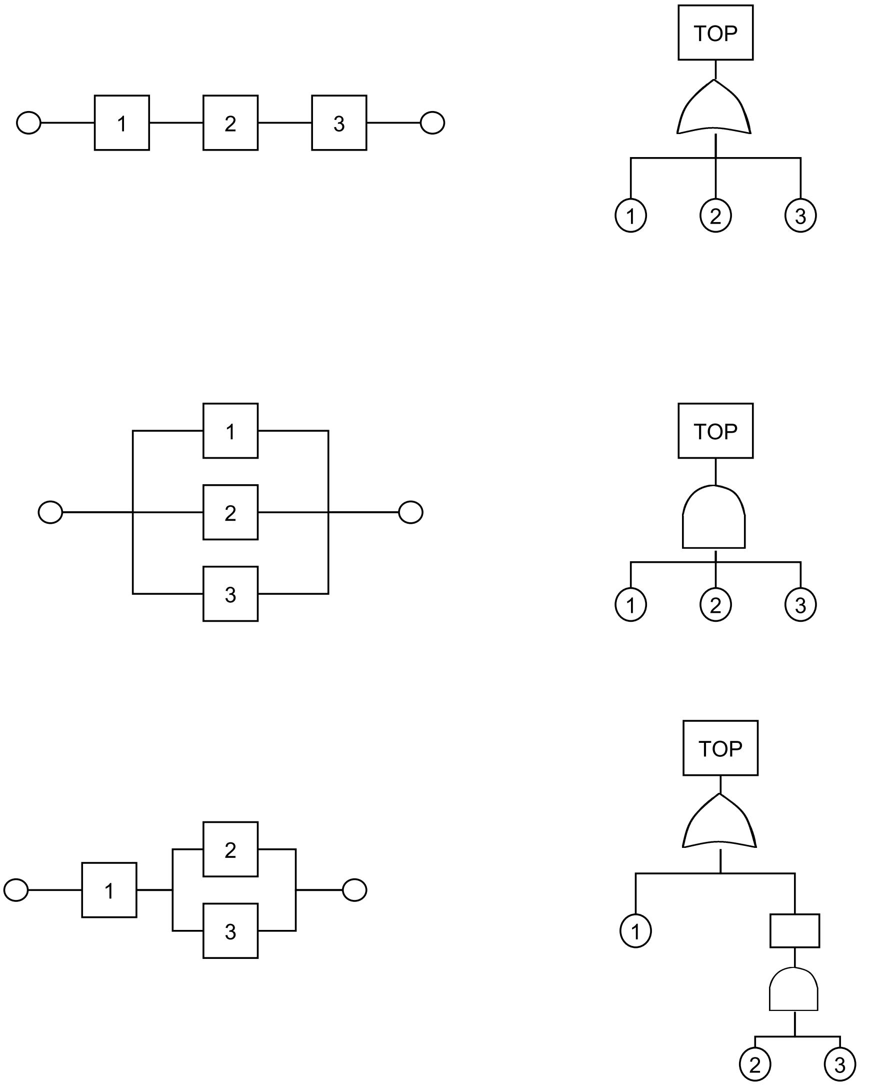


Figure 2.16: Relationship between reliability block diagrams and fault trees, (Hamada et al., 2008).

2.7 Accelerated Lifetime Testing

There are several procedures to shorten the testing time for our reliability calculations. By increasing the load and using physics of failure models, one is able to convert this accelerated lifetime to lifetime under normal load. This is only possible under the assumption, that increasing the load leads to no change in the failure mode. Typical methods are the Step-Stress method and HALT (Highly Accelerated Life Testing) explained in Bertsche and Lechner (2004). A practical approach is by collecting load collectives in road trials, determining damaging events and make a projection to the wanted lifetime. Those stresses are then reproduced in bench tests with increased load to get an accelerated test. Due to the increased load, there is a higher deterioration compared to the real life usage and therefore our components or system have decreased lifetimes. The ratio between the lifetime under "normal" conditions t_{norm} to the lifetime in the accelerated tests t_κ is called acceleration factor κ and is calculated by

$$\kappa = \frac{t_{norm}}{t_\kappa}.$$

It is also possible to calculate the acceleration factors by comparing damaging events under normal conditions with damaging events under increased load, as with a zero failure assumption we do not know the lifetime under increased load. This method is explained in Chapter 4.

2.8 Success Run

We assume that we have a sample of size n and we test each specimen for a time t. We have a random number of failures X with $0 \leq X \leq n$. The confidence level can be described by calculating the integral of a Beta distribution with certain limits as seen in Section 2.5. We calculate the confidence level depending on the reliability $R(t)$ at time t as

$$\begin{aligned} P_{\mathrm{CL}} &= \int_{R(t)}^{1} \frac{1}{\beta(n-I+1, I)} s^{n-I}(1-s)^{I-1} \mathrm{d}s \\ &= \int_{R(t)}^{1} \frac{1}{\beta(n-X, X+1)} s^{n-X-1}(1-s)^{X} \mathrm{d}s. \end{aligned}$$

We have n for the sample size and I for the rank. As the rank is calculated depending on the number of failures X, we have $I = X + 1$. This is due to the assumption that while testing n specimen for a duration of t and having X failures, the next failure is bound to happen exactly after our test ending, so we get $T_{(I+1)} = t$. This is a worst case assumption. As for the value P_{CL}, we can interpret it as follows: with a confidence level of $100 \cdot P_{\mathrm{CL}}\%$, our product has a minimum reliability of $R(t)$. We can now either fix our minimum reliability R_l we want to reach and calculate the confidence level by setting $R(t) = R_l$, or we fix the confidence level and determine the corresponding minimum reliability. The latter is of our interest.

We are also able to specify the confidence level with a binomial distribution with

$$P_{\mathrm{CL}} = 1 - \sum_{i=0}^{X} \binom{n}{i} R(t)^{n-i}(1-R(t))^{i}. \tag{2.7}$$

This is due to the relationship between the Beta and the binomial distribution.

Having no failure at all, we have $X = 0$ and we get

$$P_{\mathrm{CL}} = 1 - R(t)^n.$$

As we fix a confidence level P_{CL} and we want to determine the reliability at time t, we reorder the equation as

$$R(t) = (1 - P_{\mathrm{CL}})^{1/n}, \tag{2.8}$$

where P_{CL} is the wanted confidence level. This equation is often called "Success Run" and gives us the relationship between the confidence level P_{CL}, the reliability $R(t)$ at time t and the sample size n when having zero failures. Having two of the variables given, we are able to calculate the third value.

For a Weibull distribution we have the reliability probability

$$R(t) = e^{-\left(\frac{t}{\eta}\right)^{\beta}}.$$

Until now we assumed that all tests have the duration t. If we are testing until a time $t_p \neq t$ for all our tests, we have $R(t_p) = \exp(-(t_p/\eta)^{\beta})$. Calculating the ratio we get

$$\frac{\ln(R(t_p))}{\ln(R(t))} = \left(\frac{t_p}{t}\right)^{\beta} = \mathrm{LR}^{\beta}.$$

The ratio of testing time t_p and required time t

$$\mathrm{LR} = \frac{t_p}{t}$$

is called life-span ratio LR. Applying the life-span ration to (2.8) we get

$$R(t) = (1 - P_R)^{\frac{1}{\mathrm{LR}^{\beta} \cdot n}}.$$

This reliability can be seen as a guaranteed minimum reliability with confidence level P_{CL}.

$$R_{\min}(t) = (1 - P_{\mathrm{CL}})^{\frac{1}{\mathrm{LR}^{\beta} \cdot n}}.$$

An increase of testing time t_p with constant reliability $R(t)$ and confidence level P_{CL} results in a decrease of the sample size n and vice versa. Converting the equation we get an equation for the required sample size or required testing time:

$$\begin{aligned} n &= \frac{1}{\mathrm{LR}^{\beta}} \left(\frac{\ln(1 - P_{\mathrm{CL}})}{\ln(R(t))}\right) \\ \mathrm{LR} &= \left(\frac{1}{n} \left(\frac{\ln(1 - P_{\mathrm{CL}})}{\ln(R(t))}\right)\right)^{1/\beta}. \end{aligned}$$

If we have varying testing times of our sample size, VDA (2016) gives us a formula for calculating the minimum guaranteed reliability as

$$R_{\min}(t) = (1 - P_{\mathrm{CL}})^{\left(\sum\limits_{i=1}^{k} \mathrm{LR}_i^{\beta} n_i\right)^{-1}} \tag{2.9}$$

with k being the number of different testing times $t_1, t_2, ..., t_k$ and n_i the number of tests with testing time t_i. One method to reduce the testing time is by using accelerated stress conditions. The acceleration factor describes the ratio between the lifetime of the product under normal stress t_{norm} and the lifetime under higher load t_{κ} and is defined by

$$\kappa = \frac{t_{norm}}{t_{\kappa}}.$$

Using this acceleration factor in the equation (2.9) we get

$$R_{\min}(t) = (1 - P_{\mathrm{CL}})^{\left(\sum\limits_{i=1}^{k} n_i (\kappa_i \mathrm{LR}_i)^{\beta}\right)^{-1}} \tag{2.10}$$

This equation can be adapted to handle failures, but as we assume in the planning phase that there are no failures, we will not discuss it. For further information see VDA (2016).

Taking into account previous knowledge

We are able to take previous knowledge into account using the Beyer/Lauster method Beyer and Lauster (1990). This additional knowledge can originate from expert knowledge or predecessor models and is expressed by the value R_0 that is valid for a confidence level of 63, 2%. The expected minimum reliability is then

$$R_{\min}(t) = (1 - P_{\mathrm{CL}})^{\frac{1}{\mathrm{LR}^{\beta} \cdot n + 1/\ln(1/R_0)}}.$$

Expanding this equation by adding the acceleration factor and different testing times we get

$$R_{\min}(t) = (1 - P_{\mathrm{CL}})^{\left(\sum\limits_{i=1}^{k} n_i (\kappa_i \mathrm{LR}_i)^{\beta} + 1/\ln(1/R_0)\right)^{-1}}. \tag{2.11}$$

We can add a weighting factor ρ to be able to weight the influence of our additional knowledge

$$R_{\min}(t) = (1 - P_{\mathrm{CL}})^{\left(\sum\limits_{i=1}^{k} n_i (\kappa_i \mathrm{LR}_i)^{\beta} + \rho/\ln(1/R_0)\right)^{-1}}. \tag{2.12}$$

The previous information factor ρ lies between 0 and 1. $\rho = 0$ signifies that no additional information is being accounted for, whereas for $\rho = 1$, we are sure that the information is completely transferable to our problem. In this thesis we are going to assume that we can fully use the additional knowledge, so $\rho = 1$.

Chapter 3

Literature

In this chapter, a few methods being used in reliability engineering are being discussed. Main focus will be on the WeiBayes method by Nelson (1985), as this method is being used in Uptime LOCATE$^{\text{TM}}$, the results from Kemmner (2012), based on the WeiBayes approach and the method by Krolo (2004), which gives a different approach to calculate reliability.

3.1 WeiBayes

With a lack of failure data, it is not possible to use the typical methods to estimate the parameters of a Weibull distribution, like maximum likelihood or mean square estimator. Therefore Nelson (1985) introduced a way to estimate confidence limits for the scale parameter η when the shape parameter β has a known value or can be estimated and we only have few or no failure data. In practice, such a value of β can be approximated from experience or similar data. Thus it is possible to estimate the parameters needed for a two-parameter Weibull distribution. Before introducing the WeiBayes method, we are going to describe the derivation with the exponential distribution.

Exponential distribution: likelihood of censored data

Let $t_1, \ldots, t_n$ be the n testing times that we have set. In case of a Success Run, all those times are reached. If there is a failure in test j, we get a random variable $T_j < t_j$ as the exact failure time. Let X be the random variable which describes the number of failures. We define $Y_j = \min\{t_j, T_j\}$. So in case of a failure, $Y_j = T_j$, else $Y_j = t_j$.

For now let us assume that we have no censored data, so we get the random failure times $T_1, \ldots, T_n \overset{iid}{\sim} \text{Exp}(\lambda)$ with rate $\lambda > 0$ unknown. Let $f(t; \lambda) = \lambda e^{-\lambda t}$ be the density function, $F(t; \lambda) = 1 - e^{-\lambda t}$ be the cumulative distribution or failure probability function and $R(t; \lambda) = e^{-\lambda t}$ the reliability or survival probability. The likelihood function is then given by

$$L(\lambda | T_1, \ldots, T_n) = \prod_{i=1}^{n} f(T_i; \lambda)$$

and the log-likelihood function becomes

$$l(\lambda|T_1,\ldots,T_n) = n\log\lambda - \lambda\sum_{i=1}^{n} T_i.$$

The maximum likelihood estimator of λ is then

$$\hat{\lambda} = \frac{n}{\sum_{i=1}^{n} T_i}.$$

We can then conclude that the maximum likelihood estimator of the mean θ is then the sample mean

$$\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n} T_i.$$

In Ross (2009) we have a detailed explanation for the confidence interval for the mean of the exponential distribution. We have the following relationship:

$$\frac{2}{\theta}\sum_{i=1}^{n} T_i \sim \chi^2_{2n} \tag{3.1}$$

Hence, a $100(1-\alpha)\%$ confidence interval for θ is

$$\theta \in \left(\frac{2\sum_{i=1}^{n} T_i}{\chi^2_{1-\alpha/2;2n}}, \frac{2\sum_{i=1}^{n} T_i}{\chi^2_{\alpha/2;2n}}\right)$$

where $\chi^2_{(\alpha,n)}$ is the α-quantile of a χ^2- variable with n degrees of freedom.

Let us now include censored data. We assume that X is the number of failures. Wlog those failures occur in the first X tests. So our resulting testing times are $T_1, T_2, \ldots, T_X$ with random failure times and $t_{X+1},\ldots,t_n$ as censored data. For those values we just know that the failure time is greater than the observed value. The likelihood is the probability of the observed data, for the censored observations that is given by $P(T_j > t_j) = R(t_j;\lambda)$, so the full likelihood function is given by

$$L(\lambda|Y_1,\ldots,Y_n) = \prod_{i=1}^{X} f(T_i;\lambda)\cdot\prod_{j=X+1}^{n} R(t_j;\lambda).$$

The log-likelihood function then becomes

$$l(\lambda|Y_1,\ldots,Y_n) = X\log\lambda - \lambda(T_1+\cdots+T_X+t_{X+1}+\cdots+t_n).$$

This has the same form as the log-likelihood for the usual case without censored data, except for the first term which has $X\log\lambda$ in place of $n\log\lambda$. Solving for the maximum likelihood estimator of λ or θ we get

$$\hat{\lambda} = \frac{X}{\sum_{i=1}^{n} Y_i},$$
$$\hat{\theta} = \frac{\sum_{i=1}^{n} Y_i}{X}.$$

For certain special types of life tests it is possible to obtain exact confidence intervals for these estimators, see Lawless (1982). In a Type II censored test, the life tests terminates when the x^{th} failure occurs, where x is prespecified. The data consists then of the x smallest order statistics $T_{(1)} < \cdots < T_{(x)}$ and for the censored data we have the time $T_{(x)}$. For the log-likelihood we have then

$$l(\lambda|Y_1, \ldots, Y_n) = x \log \lambda - \lambda(\sum_{i=1}^{x} T_{(i)} + (n-x)T_{(x)})$$

and $\hat{\theta} = \sum_{i=1}^{n} Y_{(i)}/x$. We can then show that under these conditions

$$T = \sum_{i=1}^{x} T_{(i)} + (n-x)T_{(x)}$$

has a distribution given by $2T/\theta \sim \chi^2_{2r}$, see Lawless (1982). Hence an exact $100(1-\alpha)\%$ confidence interval for θ can be computed from

$$[\underline{\theta}, \bar{\theta}] = \left[\frac{2\sum_{i=1}^{n} Y_i}{\chi^2_{(1-\alpha/2;2x)}}, \frac{2\sum_{i=1}^{n} Y_i}{\chi^2_{(\alpha/2;2x)}} \right].$$

With time censoring, this method still provides an useful approximation.

If all observations are censored, we get $X = 0$ and the log-likelihood becomes

$$l(\theta|Y_1, \ldots, Y_n) = -\frac{n}{\theta} \sum_{i=1}^{n} t_i$$

which is linear increasing in θ. The maximum would then be $\theta = \infty$, which is not defined. We can only conclude that the MLE does not exist. However it is still possible to obtain a lower confidence bound for θ. If there are no failures in a life test with $t_1, \ldots, t_n$ being the censored times, a conservative $100(1-\alpha)\%$ lower confidence bound on θ is

$$\underline{\theta} = \frac{2\sum_{i=1}^{n} t_i}{\chi^2_{(1-\alpha;2)}} = \frac{\sum_{i=1}^{n} t_i}{-\ln(\alpha)}.$$

The bound is based on the fact that under the exponential failure time distribution, with immediate replacement of failed units, the number of failures observed in a life test with a fixed total time on test has a Poisson distribution (Meeker and Escobar, 1998).

WeiBayes

The estimates and confidence limits for the scale parameter η are based on the following relationship between the Weibull and exponential distributions. Suppose that a product has a Weibull distribution with shape parameter β and scale parameter η. Let T be the time to failure of this product. Then $Y = T^\beta$ has an exponential distribution with mean

$$\theta = \eta^\beta.$$

So we have

$$T \sim \text{Weibull}(\beta, \eta) \quad iff \quad T^{\beta} \sim \text{Exp}(\frac{1}{\eta^{\beta}}).$$

With this relationship we can derive the standard methods for analyzing censored data from an exponential distribution to get estimates for the scale parameter η and also confidence limits in the case of having zero failures.

Let $Y_1, \ldots, Y_n \sim \text{Weibull}(\beta, \eta)$, $Y_j = \min\{t_j, T_j\}$, with the X times being failure times (random) and $(n - X)$ censored times. The mean of the corresponding exponential distribution hast MLE

$$\hat{\theta} = \frac{\sum_{i=1}^{n} Y_i^{\beta}}{X}.$$

Thus follows

$$\hat{\eta} = \left[\sum_{i=1}^{n} \frac{Y_i^{\beta}}{X} \right]^{\frac{1}{\beta}} \tag{3.2}$$

With this function we are able to consider failure times and censored testing times to get a Weibull distribution. If there are no failures, $X = 0$ and and $\eta = \infty$. This gives us no real information, just that the true η is probably greater than the running time. There are now two approaches, when having a Success Run.
Either we calculate a lower $100(1-\alpha)\%$ confidence limit. A lower $100(1-\alpha)\%$ confidence limit for the Weibull scale parameter is

$$\underline{\eta} = \begin{cases} \hat{\eta}\{2X/\chi^2_{(1-\alpha;2X+2)}\}^{1/\beta} & \text{for} \qquad X \geq 1, \\ \{2\sum_{i=1}^{n} t_i^{\beta}/\chi^2_{(1-\alpha;2X+2)}\}^{1/\beta} & \text{for} \qquad X = 0, \end{cases}$$

where $\chi^2_{(\alpha,n)}$ is the α-quantile of a χ^2- variable with n degrees of freedom. As $\chi^2_{(1-\alpha;2)} = -2\ln(\alpha)$, we can in case of a Success Run simplify the second equation to

$$\underline{\eta} = \left(\frac{\sum_{i=1}^{n} t_i^{\beta}}{-\ln(\alpha)} \right)^{1/\beta}. \tag{3.3}$$

The confidence level is usually set to either 90%, 95% or 99%. It should be mentioned, that the confidence level with few failures is quite conservative and often far from the real value. Nelson Nelson (1985) therefore suggests that using the 50% confidence level might be more reasonable.

Using this estimator of η we get the lower confidence limit for the reliability for the target time t_0 as

$$\underline{R}(t_0) = e^{-\left(\frac{t_0}{\underline{\eta}}\right)^{\beta}} = e^{\left(\frac{t_0^{\beta} \ln \alpha}{\sum_{i=1}^{n} t_i^{\beta}}\right)} = \alpha^{\left(\frac{t_0^{\beta}}{\sum_{i=1}^{n} t_i^{\beta}}\right)}. \tag{3.4}$$

If instead of the testing times $t_1, ..., t_n$ we use the equivalent test durations $\hat{t_i}$, i.e. the testing times multiplied with the acceleration factors, and the connection between $P_{\text{CL}} = (1-\alpha)$, we can transform (3.4) to (2.10).

The second approach is assuming that the first failure is soon to happen and we set $x = 1$. This value can then be chosen as a lower confidence limit. We consider a "worst case" assumption and choosing this bound is therefore not critical. This bound is roughly a 63.2% lower confidence bound, as $-\log(0.632) \approx 1$ and (3.3) becomes the same as (3.2) with $X = 1$.

Taking into account previous knowledge

If we have additional knowledge about the shape parameter η_R, for example due to expert knowledge or lifetime calculations of similar products, it is possible to link these informations with the real failure data. Equation (3.5) below is from Abernethy (2006) and results as an extension of (3.2) by including the known characteristic lifetime η_R and weighting factor ρ. We get a conservative estimator of η with

$$\tilde{\eta} = \left[\frac{\rho \cdot \eta_R^\beta}{\rho + X} + \left(\sum_{i=1}^{N} \frac{t_i^\beta}{\rho + X} \right) \right]^{1/\beta}. \tag{3.5}$$

This equation offers the possibility to add with ρ and η_R lifetime calculations of similar products or expert knowledge. This additional knowledge raises the level of reliability for a fixed time point t_0. We use the parameter ρ to weight the influence of the knowledge η_R in respect to the real data. For the choice of ρ in practice we need to have additional expert knowledge.

3.2 Results from Kemmner

Kemmner (2012) focuses on finding lower confidence bounds for the reliability of systems consisting of Weibull components, when having zero failures. For single items he uses the methodology by Nelson (1985) to calculate the lower confidence limits of reliability. As there is no comparable method for systems, Kemmner formulates a theorem for the lower confidence limit of system reliability for a systems consisting of components which follow a Weibull distribution. The theorem allows for different shape parameters for components and is even usable if each component has been exposed to different test durations. In this section we are going to introduce the necessary definitions and review the proof for the theorem. The proof is firstly conducted for systems with two components and then extended towards the general case of n components. In the last subsection we discuss the proposed optimization of reliability demonstration testing plans and the drawbacks of the proposed method.

3.2.1 Framework

The methodology is only applicable when reliability testing is conducted with zero failures.

Let i denote the component ($i = 1, \ldots, n$) and j a test ($j = 1, \ldots, m$). The respective test time of test j is t_j. We denote with $\kappa_{i,j}$ the acceleration factor related to the nominal load under normal operating conditions for component i in test j. An acceleration factor of $\kappa_{p,l} = 0$ indicates that the test l has no influence on component p. We define with X_i the number of failures in component i and X_S the number of failures in the system.

If a sample of m items is tested with duration $t_j, j = 1, \ldots, m$, the probability of zero failures in all m tests for component i is the product of the reliability at the individual test durations:

$$P(X_i = 0) = \prod_{j=1}^{m} e^{-\left(\frac{\kappa_{i,j} \cdot t_j}{\eta_i}\right)^{\beta_i}} = e^{-\frac{\sum_{j=1}^{m}(\kappa_{i,j} \cdot t_j)^{\beta_i}}{\eta_i^{\beta_i}}}.$$

The equivalent single test duration TW_i for component i is defined as

$$TW_i = \sqrt[\beta_i]{\sum_{j=1}^{m} (\kappa_{i,j} \cdot t_j)^{\beta_i}}$$

as a shortcut. It represents the accumulation of all test durations for a given item i.

Wanting a $100(1 - \alpha)\%$ confidence limit, we can calculate the lower confidence limit of the scale parameter η_i of component i by adding the acceleration factor to (3.3) and simplifying with the single test duration TW_i:

$$\underline{\eta_i} = \left(\frac{TW_i^{\beta_i}}{-\ln(\alpha)}\right)^{1/\beta_i}.$$

For our reliability bound of component i we have then:

$$R_{C_i,\mathrm{LCL},(1-\alpha)}(t_0) = e^{\left(\frac{t_0^{\beta_i} \ln(\alpha)}{TW_i^{\beta_i}}\right)} = \alpha^{\left(\frac{t_0^{\beta_i}}{TW_i^{\beta_i}}\right)}. \tag{3.6}$$

Before introducing the theorem, we need to have certain assumptions, see Kemmner (2012):

- Zero failures during testing.
- All components follow a Weibull distribution with known shape parameter β_i, known from historical data or estimated by experts. The failure behavior of all components is assumed to be mutually independent.
- Countermeasures for issues do not affect other components.
- The acceleration factors $\kappa_{i,j}$ are known.
- If a sub-system of the system is a copy from another already tested system, then the respective test durations (TW_i) for this respective sub-system can be converted to the considered system.
- Failures with a prototype specific root cause will be ignored.
- If the design of a component has changed, then the former testing times before that change will be ignored.

- The design level of the system that will be finally released had no failures, since all were fixed.

There are some conservative assumptions, but for simplicity they can be used. Note that in our problem we can use the same assumptions, as most of them are already made within the current calculations in Uptime LOCATE$^{\text{TM}}$.

Having all the necessary definitions we can finally introduce the theorem:

Theorem 2 (Kemmner (2012)). *A system consisting of n Weibull components with shape parameter β_i that is exposed to the Weibull equivalent single test duration TW_i with zero failures occurring has the following $(1-\alpha)$ lower confidence limit of system reliability $R_{\text{S;LCL;}(1-\alpha)}$ at duration t_0:*

$$R_{\text{S;LCL;}(1-\alpha)}(t_0) = \min_i R_{C_i\text{;LCL;}(1-\alpha)}(t_0) = \min_i \left[\alpha^{\left(\frac{t_0^{\beta_i}}{TW_i^{\beta_i}}\right)} \right]$$

The equation above means that the lower confidence limit of system reliability is identical to the lowest lower confidence level of component reliability calculated according to (3.6).

Depending on the different shape parameters β_i, the limiting component for the calculation of the lower system reliability bound can change at different points in time t. We can be sure with probability $(1-\alpha)$ that the real reliability is higher than $R_{\text{S;LCL;}(1-\alpha)}$. The advantages of the proposed method are being able to making statements for the lower confidence limit reliability for a system with different shape parameters β_i of its components i, as well as having a much higher system reliability than multiplying the limits of each component when having a serial system.

3.2.2 Proof of the theorem

Proof for n=2

Let $R_{\text{ac},i}$ be the actual reliability of component i and $\eta_{\text{ac},i}$ the respective Weibull scale parameter. So the actual reliability of component i at duration t_0 is given as

$$R_{\text{ac},i}(t_0) = e^{-\left(\frac{t_0}{\eta_{\text{ac},i}}\right)^{\beta_i}}.$$

For the serial system with two components we then get the system reliability at duration t_0 with

$$R_{\text{ac,S}}(t_0) = \prod_{i=1}^{2} e^{-\left(\frac{t_0}{\eta_{\text{ac},i}}\right)} = e^{-\left[\left(\frac{t_0}{\eta_{\text{ac},1}}\right)^{\beta_1} + \left(\frac{t_0}{\eta_{\text{ac},2}}\right)^{\beta_2}\right]}.$$

The probability of zero failures occurring in component i with having $j=1,...,m$ tests is

$$P(X_i = 0) = \prod_{j=1}^{m} e^{-\left(\frac{\kappa_{i,j}\cdot t_{i,j}}{\eta_{\text{ac},i}}\right)^{\beta_i}} = e^{-\left(\frac{TW_i}{\eta_{\text{ac},i}}\right)^{\beta_i}}$$

and again we get for the system:

$$P(X_{\mathrm{S}} = 0) = e^{-\left[\left(\frac{TW_1}{\eta_{\mathrm{ac},1}}\right)^{\beta_1} + \left(\frac{TW_2}{\eta_{\mathrm{ac},2}}\right)^{\beta_2}\right]}.$$

We now have to verify the following statement: The probability of zero failures during our tests is less or equal α for the case that the actual system reliability $R_{\mathrm{ac,S}}(t_0)$ is less or equal to $R_{\mathrm{S,LCL},(1-\alpha)}(t_0)$. This is exactly the type II error, as having zero failures during our tests results in accepting H_0, even tough the real reliability might be lower than the lower confidence limit. This type II error should be at most α.

It is sufficient to reduce it to the case of equality, as having an increased system reliability also increases the probability of having zero failures.

We now have:

$$P(X_{\mathrm{S}} = 0) = e^{-\left[\left(\frac{TW_1}{\eta_{\mathrm{ac},1}}\right)^{\beta_1} + \left(\frac{TW_2}{\eta_{\mathrm{ac},2}}\right)^{\beta_2}\right]} \leq \alpha. \tag{3.7}$$

This defines that the probability of a Success Run is less or equal than α. The second part of the statement gives us the relationship for the actual system reliability and the lower confidence limit:

$$R_{\mathrm{ac,S}}(t_0) = e^{-\left[\left(\frac{t_0}{\eta_{\mathrm{ac},1}}\right)^{\beta_1} + \left(\frac{t_0}{\eta_{\mathrm{ac},2}}\right)^{\beta_2}\right]} = R_{\mathrm{S,LCL},(1-\alpha)}(t_0). \tag{3.8}$$

According to the theorem we have:

$$R_{\mathrm{S,LCL},(1-\alpha)}(t_0) = \min_i \left[\alpha^{\left(\frac{t_0^{\beta_i}}{TW_i^{\beta_i}}\right)}\right].$$

Let us assume wlog that $i = 1$ is the limiting component that determines the value of the lower confidence limit of the system reliability. Considering $0 < \alpha < 1$ this leads to:

$$\frac{t_0^{\beta_1}}{TW_1^{\beta_1}} \geq \frac{t_0^{\beta_2}}{TW_2^{\beta_2}}. \tag{3.9}$$

The lower confidence limit of the system with $i = 1$ as limiting component is then

$$R_{\mathrm{S,LCL},(1-\alpha)}(t_0) = \alpha^{\left(\frac{t_0^{\beta_1}}{TW_1^{\beta_1}}\right)}.$$

Substituting into (3.8):

$$e^{-\left[\left(\frac{t_0}{\eta_{\mathrm{ac},1}}\right)^{\beta_1} + \left(\frac{t_0}{\eta_{\mathrm{ac},2}}\right)^{\beta_2}\right]} = \alpha^{\left(\frac{t_0^{\beta_1}}{TW_1^{\beta_1}}\right)}.$$

Solving for α we get:

$$\alpha = e^{-\left[\left(\frac{t_0}{\eta_{\mathrm{ac},1}}\right)^{\beta_1} + \left(\frac{t_0}{\eta_{\mathrm{ac},2}}\right)^{\beta_2}\right]\frac{TW_1^{\beta_1}}{t_0^{\beta_1}}} = e^{-\frac{TW_1^{\beta_1}}{\eta_{\mathrm{ac},1}^{\beta_1}} - \frac{TW_1^{\beta_1} \cdot t_0^{\beta_2}}{t_0^{\beta_1} \cdot \eta_{\mathrm{ac},2}^{\beta_2}}}.$$

Now substituting α in (3.7):

$$e^{-\left[\left(\frac{TW_1}{\eta_{\mathrm{ac},1}}\right)^{\beta_1}+\left(\frac{TW_2}{\eta_{\mathrm{ac},2}}\right)^{\beta_2}\right]} \leq \alpha = e^{-\frac{TW_1^{\beta_1}}{\eta_{\mathrm{ac},1}^{\beta_1}}-\frac{TW_1^{\beta_1}\cdot t_0^{\beta_2}}{t_0^{\beta_1}\cdot\eta_{\mathrm{ac},2}^{\beta_2}}} .$$

Taking the log on both sides:

$$\begin{aligned} -\left(\frac{TW_1}{\eta_{\mathrm{ac},1}}\right)^{\beta_1} - \left(\frac{TW_2}{\eta_{\mathrm{ac},2}}\right)^{\beta_2} &\leq -\frac{TW_1^{\beta_1}}{\eta_{\mathrm{ac},1}^{\beta_1}} - \frac{TW_1^{\beta_1}\cdot t_0^{\beta_2}}{t_0^{\beta_1}\cdot\eta_{\mathrm{ac},2}^{\beta_2}} \\ -\frac{TW_2^{\beta_2}}{\eta_{\mathrm{ac},2}^{\beta_2}} &\leq -\frac{TW_1^{\beta_1}\cdot t_0^{\beta_2}}{t_0^{\beta_1}\cdot\eta_{\mathrm{ac},2}^{\beta_2}} \\ -\left(\frac{TW_2}{t_0}\right)^{\beta_2} &\leq -\left(\frac{TW_1}{t_0}\right)^{\beta_1} . \end{aligned}$$

The last inequality is equivalent to inequality (3.9) and therefore fulfilled when component $i = 1$ is the limiting component. The proof for component $i = 2$ being the limiting factor is analog. We now have the proof for a system consisting of two components.

General proof for n

We recall the actual reliability for component i and respective scale parameter $\eta_{\mathrm{ac},i}$ at time t_0 is defined as:

$$R_{\mathrm{ac},i}(t_0) = e^{-\left(\frac{t_0}{\eta_{\mathrm{ac},i}}\right)^{\beta_i}} .$$

The reliability of the serial system of Weibull components is then given as:

$$R_{\mathrm{ac,S}}(t_0) = \prod_{i=1}^{n} e^{-\left(\frac{t_0}{\eta_{\mathrm{ac},i}}\right)^{\beta_i}} = e^{-\sum_{i=1}^{n}\left(\frac{t_0}{\eta_{\mathrm{ac},i}}\right)} .$$

The probability of zero failures occurring in component i with having $j = 1, \ldots, m$ tests is

$$P(X_i = 0) = \prod_{j=1}^{m} e^{-\left(\frac{\kappa_{i,j}\cdot t_{i,j}}{\eta_{\mathrm{ac},i}}\right)^{\beta_i}} = e^{-\left(\frac{TW_i}{\eta_{\mathrm{ac},i}}\right)^{\beta_i}}$$

and again we get for the probability of the system having zero failures:

$$P(X_{\mathrm{S}} = 0) = \prod_{i=1}^{n}\prod_{j=1}^{m} e^{-\left(\frac{\kappa_{i,j}\cdot t_{i,j}}{\eta_{\mathrm{ac},i}}\right)^{\beta_i}} = \prod_{i=1}^{n} e^{-\left(\frac{TW_i}{\eta_{\mathrm{ac},i}}\right)^{\beta_i}} .$$

To prove the theorem for general n we have to verify the following statement:
The probability of zero failures occurring during testing is less or equal than α for the case that the actual value of the system reliability $R_{\mathrm{ac,S}}(t_0)$ is less or equal to $R_{\mathrm{S,LCL},(1-\alpha)}(t_0)$. Again, it is sufficient to verify this for the case of equality. The first part is described with the following equation:

$$P(X_{\mathrm{S}} = 0) = \prod_{i=1}^{n} e^{-\left(\frac{TW_i}{\eta_{\mathrm{ac},i}}\right)^{\beta_i}} = e^{-\sum_{i=1}^{n}\left(\frac{TW_i}{\eta_{\mathrm{ac},i}}\right)^{\beta_i}} \leq \alpha \tag{3.10}$$

and for the second statement:

$$R_{\text{ac,S}}(t_0) = e^{-\sum_{i=1}^{n}\left(\frac{t_0}{\eta_{\text{ac},i}}\right)} = R_{\text{S,LCL},(1-\alpha)}(t_0).$$

In accordance with the theorem, we have

$$R_{\text{S,LCL},(1-\alpha)}(t_0) = \min_i \left[\alpha^{\left(\frac{t_0^{\beta_i}}{TW_i^{\beta_i}}\right)}\right].$$

We then have

$$e^{-\sum_{i=1}^{n}\left(\frac{t_0}{\eta_{\text{ac},i}}\right)} = \min_i \left[\alpha^{\left(\frac{t_0^{\beta_i}}{TW_i^{\beta_i}}\right)}\right].$$

Let us define Y_i as: $Y_i = \left(\frac{TW_i}{t_0}\right)^{\beta_i}$ with $Y_i > 0$.
High values of Y_i show us that the equivalent single test duration TW_i is large in comparison to the time t_0. Substituting Y_i we get

$$e^{-\sum_{i=1}^{n}\left(\frac{t_0}{\eta_{\text{ac},i}}\right)} = \min_i \left[\alpha^{(1/Y_i)}\right].$$

Having α defined $0 < \alpha < 1$ we can conclude that $\alpha^{(1/Y_i)}$ is smaller if the values of Y_i gets smaller. We denote as $Y_{\min} = \min_i Y_i$ and we get:

$$e^{-\sum_{i=1}^{n}\left(\frac{t_0}{\eta_{\text{ac},i}}\right)} = \alpha^{(1/Y_{\min})}.$$

Solving for α:

$$\alpha = e^{-\sum_{i=1}^{n}\left[Y_{\min}\left(\frac{t_0}{\eta_{\text{ac},i}}\right)^{\beta_i}\right]}.$$

Substituting this equation into (3.10) gives

$$e^{-\sum_{i=1}^{n}\left(\frac{TW_i}{\eta_{\text{ac},i}}\right)^{\beta_i}} \leq e^{-\sum_{i=1}^{n}\left[Y_{\min}\left(\frac{t_0}{\eta_{\text{ac},i}}\right)^{\beta_i}\right]}.$$

Now substituting $TW_i^{\beta_i} = t_0^{\beta_i} \cdot Y_i$ yields:

$$e^{-\sum_{i=1}^{n}\left[Y_i\left(\frac{t_0}{\eta_{\text{ac},i}}\right)^{\beta_i}\right]} \leq e^{-\sum_{i=1}^{n}\left[Y_{\min}\left(\frac{t_0}{\eta_{\text{ac},i}}\right)^{\beta_i}\right]}.$$

Taking the log we get

$$\sum_{i=1}^{n}\left[Y_i\left(\frac{t_0}{\eta_{\text{ac},i}}\right)^{\beta_i}\right] \geq \sum_{i=1}^{n}\left[Y_{\min}\left(\frac{t_0}{\eta_{\text{ac},i}}\right)^{\beta_i}\right] \text{ with } t_0, \eta_{\text{ac},i}, \beta_i > 0.$$

This statement is fulfilled, if for each component i we have:

$$Y_i \left(\frac{t_0}{\eta_{\mathrm{ac},i}} \right)^{\beta_i} \geq Y_{\min} \left(\frac{t_0}{\eta_{\mathrm{ac},i}} \right)^{\beta_i}.$$

As $\left(\frac{t_0}{\eta_{\mathrm{ac},i}} \right)^{\beta_i} \geq 0$ for $t_0, \eta_{\mathrm{ac},i}, \beta_i > 0$ we can simplify this equation to

$$Y_i \geq Y_{\min} \quad \forall i.$$

This inequality is fulfilled for all components i per definition. This concludes the proof of the theorem.

3.2.3 Optimizing system reliability

Kemmner also proposes a highly nonlinear optimization program to maximize the system reliability. Objective function is maximizing $R_{\mathrm{S,LCL},(1-\alpha)}$ with certain constraints, one being that the system lower confidence bound is the minimal component lower confidence bound. The additional constraints are budget constraints, time limitation and limiting integer sample size. Although we do not have information about the costs, the approach would still be viable for our problem. In Chapter 5 we discuss reasons for not choosing this method.

3.3 Method by Krolo

The method was developed by A. Krolo in the context of her PhD thesis. It is based on Bayes' theorem which states the following:

Theorem 3 (Bayes' Theorem)**.** *Let A and B two events with a-priori probability $P(A)$ und $P(B) > 0$. We can then calculate the conditional probability for event A, given that event B already happened, as follows:*

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

This theorem is used to include previous knowledge for the planning of reliability tests. It combines known information with the data of the current test.

We want to know the reliability $R(t)$ at time t. We now fix the time t and consider $R = R(t)$. Knowing the reliability function of our system or component, we would be able to calculate the value of R. As we mostly do not know the reliability functions, R is unknown between 0 and 1. But we can often estimate a rough value for R or restrict the interval to an area with a higher probability, so that R is likely within that interval. Knowledge about R can then be expressed by the probability for a reliability value, i.e. a density function $f^R(s)$. By assuming a prior density function $f^R_{\mathrm{prior}}(s)$ from experience and updating this function with information from new test data, we get a updated a-posteriori density function $f^R_{\mathrm{post}}(s)$. This additional information due to tests is given by the probability for the number of failures occurring during the tests. With

E_x we characterize the event of x units failing. We are able to describe this event by the probability of a single unit failing using the binomial-distribution. Combining the a-priori density with the results of the current test run gives us an a-posteriori density, which is an improved estimation of the real value of the random variable. The a-posteriori density can be described using Bayes' theorem as follows:

$$f^R_{\text{post}}(s|E_x) = \frac{P(E_x|R = s) f^R_{\text{prior}}(s)}{\int_0^1 P(E_x|R = s) f^R_{\text{prior}}(s)\mathrm{d}s}. \tag{3.11}$$

As mentioned, $f^R_{\text{prior}}(s)$ describes the a-priori density. The probability $P(E_x|R = s)$ equates to the conditional probability and specifies the current test data, given by the binomial-distribution:

$$P(E_x|R = s) = \binom{n}{x} s^{n-x}(1 - s)^x. \tag{3.12}$$

The variable x represents the number of failures in the test, n is the sample size. If there are no failures during testing, we get a Success Run and (3.12) simplifies to

$$P(E_0|R = s) = s^n.$$

This equation can then be adapted to consider a life-span ratio and acceleration factors, for the exact description see Krolo (2004). To get the confidence level P_{CL} of the a-posteriori density we have to integrate (3.11) from the reliability target $R_{\text{tar}}(t)$ to 1:

$$P_{\text{CL}} = \int_{R_{\text{tar}}(t)}^1 f^R_{\text{post}}(s|E_x)ds = \frac{\int_{R_{\text{tar}}(t_0)}^1 P(E_x|R = s) f^R_{\text{prior}}(s)\mathrm{d}s}{\int_0^1 P(E_x|R = s) f^R_{\text{prior}}(s)\mathrm{d}s}.$$

If there is no additional prior information before the reliability testing, we can still assume, that the reliability of the tested specimen is a uniform distributed random variable, see Stange (1977). The density function of the uniform distribution is $f(R) = 1$. For the a-priori confidence level we get

$$P_{\text{CL0}} = 1 - R_{\text{tar}}(t_0)$$

For a $\mathbf{B}_{10}$ target we already get a a-priori confidence level of $P_{\text{CL0}} = 10\%$. So just assuming a uniform distribution for the density function, we can, with a confidence level of 10% say, that 90% of our specimen still are functioning at time t.

Using the uniform distribution as a-priori density and (3.12) for the conditional probability, we get the a-posteriori density of a Beta distribution with

$$f^R_{\text{post}}(s|E_x) = \frac{s^{n-x}(1 - s)^x}{\int_0^1 s^{n-x}(1 - s)^x\mathrm{d}s} = \frac{s^{n-x}(1 - s)^x}{\beta(n - x + 1, x + 1)}.$$

Integrating this function from $R_{\text{tar}}(t)$ to 1 returns the level of confidence P_{CL}. It can be shown that assuming a uniform distribution, the sample size needed reduces by one (Bertsche and Lechner, 2004), (Krolo, 2004).

Having more information about the product, we can often assume to have a Beta distribution as a-priori density. The spread of the failure probability can be described by a two-parameter Beta distribution in accordance with Bertsche and Lechner (2004). The a-priori density is then given by (2.3). We give the parameters a and b the indices 0 to mark the prior information

$$f^R_{\text{prior}}(s) = \frac{1}{\beta(a_0, b_0)} s^{a_0-1}(1-s)^{b_0-1}.$$

Omitting a life-span ratio, we assume that the testing duration is the same as the target time t. Using Bayes' formula for the a-posteriori density we get a Beta distribution:

$$f^R_{\text{post}}(s|E_x) = \frac{s^{a_0-1+n-x}(1-s)^{b_0-1+x}}{\int_0^1 s^{a_0-1+n-x}(1-s)^{b_0-1+x}\mathrm{d}s} = \frac{s^{a_0-1+n-x}(1-s)^{b_0-1+x}}{\beta(a_0+n-x, b_0+x)}. \tag{3.13}$$

We can see that the parameters of the a-posteriori density is a sum of the parameters of the a-priori density and the sample size, as well as the number of failures. Thus without using Bayes theorem, the a-posteriori density can be easily determined. Having a Beta distribution as an a-priori density, we get a Beta distribution as an a-posteriori density with the parameters

$$\begin{aligned} a &= a_0 + n - x \\ b &= b_0 + x. \end{aligned}$$

Assuming that b_0 is integer, we get for the confidence level due to the connection between Beta and binomial distribution:

$$P_{\text{CL}} = 1 - \sum_{i=0}^{b_0+x-1} \binom{a_0+n+b_0-1}{i} R_{\text{tar}}(t_0)^{a_0+n+b_0-1-i}(1-R_{\text{tar}}(t_0))^i.$$

If b_0 is not integer, we have to calculate the confidence level by integrating (3.13).

3.3.1 Transformation factor

As the Bayesian approach assumes identical conditions, we always have to critically consider the portability of the information. Krolo (2004) thus introduces a transformation factor. If there are slightly different conditions, i.e. testing conditions, or modernization of a component, we are not able to use the prior information unrestricted. We thus have to determine a transformation factor, such that we are able to partially use the prior knowledge. The transformation factor $\Phi \in [0,1]$. The factor is then used by multiplying it with the prior information when calculating the parameter of the Beta distribution

$$\begin{aligned} &\Phi \cdot a_0 \\ &\Phi \cdot (b_0 - 1) + 1. \end{aligned}$$

A transformation factor of $\Phi = 1$ implies, that the prior information can be fully transferred to the new situation, as the parameters of the Beta distribution are fully incorporated in the calculations. For $\Phi = 0$, we omit all prior information.

Usually, it will hold that $0 < \Phi < 1$. Due to the different conditions in the prior knowledge, i.e. gained trough test runs of former products, and the new product, we cannot fully integrate the full former knowledge and we have to weight that knowledge. Hitziger (2007) discusses methods to determine the transformation factor. Next we are going to discuss the choice of the parameters a_0 and b_0 depending on the number of failures.

3.3.2 Defining the a-priory density

As described before, we know that the confidence interval of a sample can be described with the Beta distribution. The parameters of the distribution are dependent on the sample size n and the rank i. Thus we define the Beta density function on the random variable reliability as

$$f^R_{\text{prior}}(s) = \frac{1}{\beta(a,b)} s^{a-1}(1-s)^{b-1} \quad \text{with } a = n - i + 1 \text{ and } b = i. \tag{3.14}$$

The density function of the reliability is independent from the failure distribution of the sample, i.e. we are only interested in the sample size and number of failures. Having enough failures, we are able to fit the failure data with a Weibull distribution.

Many failures

We consider a test without acceleration factors. The failure probability for a target time t_0 is then given as

$$F(t_0) = 1 - e^{-\left(\frac{t_0}{\eta}\right)^\beta}$$

This value equates to the median of the Beta distribution, approximated with (2.6). Equalize those equations, we get a dependency from i on the Weibull parameters, the reliability target time t_0 and the sample size:

$$1 - e^{-\left(\frac{t_0}{\eta}\right)^\beta} = \frac{i - 0.3}{n + 0.4} \rightarrow i = (n + 0.4)\left[1 - e^{-\left(\frac{t_0}{\eta}\right)^\beta}\right] + 0.3.$$

With this we get for the parameter of the Beta distribution in accordance with (3.14)

$$\begin{aligned} a_0 &= n - (n + 0.4)\left[1 - e^{-\left(\frac{t_0}{\eta}\right)^\beta}\right] + 0.7 \\ b_0 &= (n + 0.4)\left[1 - e^{-\left(\frac{t_0}{\eta}\right)^\beta}\right] + 0.3. \end{aligned}$$

Again, we are able to include acceleration factors κ in our calculations, for further information see Krolo (2004).

Few failures

If there are only a few failures in the tests, it can be critical to describe the results with help of a Weibull distribution. Krolo (2004) therefore analyses the exact failure times of the specimen. The parameter of the Beta distribution regarding the a-priori density are given as

$$a_0 = n - \left(\sum_k i_k - 1\right)$$
$$b_0 = \left(\sum_k i_k - 1\right) + 1.$$

Here n is the sample size and i_k the rank of the kth failure time t_k. The rank is calculated with

$$i_k = 1.4\left(1 - 0.5^{\frac{1}{\left(\frac{t_k}{t_0}\right)^\beta}}\right) + 0.3.$$

This holds under the assumption that there are no two failures at the exact same time. In practice this assumption can be made without problems.

Having a Success Run, Krolo assumes that the first failure is bound to happen shortly after the end of the test, so we get the rank $i = 1$. Thus for the choice of our a-priory density parameters we get

$$a_0 = n \quad \text{and} \quad b_0 = 1$$

3.3.3 Calculating the a-posteriori density

If we have the parameters of the a-priory density, we can calculate the a-posteriori parameters as:

$$a = \sum_{i=1}^{p} \Phi_i a_{0i} + n$$
$$b = \sum_{i=1}^{p} \Phi_i (b_{0i} - 1) + 1$$

where p is the number of a-priory distributions with respective parameters a_i and b_i and n as the needed sample size for the reliability demonstration. The transformation factor Φ_i weights the portability of the a-priory information.

With the method of Krolo we get a confidence level of P_{CL} by integrating the a-posteriori density

$$P_{\text{CL}} = \int_{R_{\text{tar}}(t_0)}^{1} \frac{1}{\beta(a,b)} s^{a-1}(1-s)^{b-1}\text{d}s.$$

With a reliability target $R_{\text{tar}}(t_0)$ and a given confidence level P_{CL} we are able to calculate the necessary sample size depending on the transformation factor and the prior knowledge.

Chapter 4

Uptime LOCATE

In this chapter, the current functionality of the software Uptime LOCATE$^{\text{TM}}$ is being explained and illustrated with the help of an example.

4.1 Scope

At the beginning of a project, we have to define our reliability targets and operating conditions. Given a system S, we first have to define its required function and possible failure modes. For that, knowledge about the environmental and operating conditions is needed. The failure modes with their respective defect components can be best described with a fault tree. Having the system S, we want to identify all components that are prone to failure. For these components we further develop our fault tree. At the end we get a tree structure with all necessary components and their respective failure mode. Note that we assume that a failure in any component leads to a failure of the system, thus we omit the use of the logic gate in our representation. Figure 4.1 gives us the fault tree of our example "hydraulic system". The basic events are failure modes, a complete list can be found in Table 4.1.

Having the structure and failure analysis of our system, we need to define the reliability targets for our system. One system can have multiple targets, e.g. warranty target and lifetime target. The reliability target is a quintuple $(\tau, u_R, R_{\text{tar,S}}(t_0), t_0, u_0)$. Here τ is the target type and has to be either survival probability or failure rate. It is possible to convert from one type to another. The unit used for our reliability type u_R is either unit-less for failure rate and %, ‰ or ppm (parts per million) for survival probability. In this thesis we are only considering survival probability when talking about the reliability target. $R_{\text{tar,S}}(t_0)$ is our target value, e.g. 90% for our reference value t_0. The reference unit u_0 can be either a duration in $[h]$ and $[a]$, operating cycles, but also distance $[km]$ or $[mi]$. For our example we have two reliability targets: warranty target (failure rate, $[-], 0.20, 3600, [h]$) and lifetime target (survival probability, $[\%], 90, 10000, [h]$). So the latter means: After 10000 hours running time, at least 90% of our hydraulic systems still works. As exact statements about the lifetime of a product are not possible, we have to define an statistical error probability α. Per default, we set $\alpha = 10\%$.

After defining the reliability target and fault tree, we have to specify the operating conditions: how is the operator using our system and under which conditions. In practice

cavitation	Pitting of material surface due to repeated impact of collapsing cavitation bubbles
wear, lubricated	Loss of material in contact zone due to failure of hydrodynamic lubricant to fully seperate surfaces
HCF	Crack initiation due to load cycling and hence accumulation of material defects
wear, dry	Loss of material in contact zone due to relative movement of surfaces under applied pressure
therm. aging	Hardening, embrittlement, softening, plastic strain due to thermally activated diffusion in solid state materials
seizure	Loss of freedom of movement, due to thermal/mechanical deformation, loss of lubricant
corrosion	Material loss, deposition, surface modification due to aggressive agents
leakage	Loss of fluid
fretting wear	Wear due to micro-slip at stick-slip boundaries in contact zone
fretting fatigue	Crack initiation due to multi-axial stress variation at stick-slip boundaries
LCF	Crack initiation and propagation due to cyclic plastic strain
TMF	Crack initiation due to cyclic thermal loading
abrasive wear	Local abrasion of material
adhesive wear	Local material transfer, microwelding
chem. aging	Changes of surface properties due to exposure in chemically active atmosphere
creep	Irreversible strain due to temperature or stress
deposition	Deposition of substances on (functional) surfaces
distortion	Irreversible deformation of components
drift	Temporal drift of sensitivity or zero point of sensors
el. chem. corrosion	corrosion driven by difference in chemical potential
erosion	Abrasion by impact of particles in fluids
freezing	Crackformation or blocking
hot gas corrosion	corrosion due to liquid oxides at high temperatures
therm. overload	Change of surface properties due to incipent melting or phase transformation etc.

Table 4.1: Examples of failure modes.

there can be multiple reference profiles, i.e. a hydraulic system in a truck has a different load compared to the hydraulic system in a car. As a start we are going to assume that our system is used by only one operator, so we know the exact load and environmental conditions our system is exposed to.

4.2 Risk Filter

The system risk filter is a feature of Uptime LOCATE$^{\text{TM}}$. It consists of 12 standardized risk categories. For every component of our system, the risk is evaluated by engineers. For a component with zero risks, we can assume that the component has no risk and is therefore not in the focus of the evaluation. With adding risks, it is being indicated that the component is more prone to failure and therefore we need more extensive testing. Figure 4.2 shows an excerpt of the risk filter of the axial piston unit in our hydraulic system. The risks are inherited bottom up, i.e. a risk in a component leads to the risk in every parent component and the system.

Given the reliability target of our system $R_{\text{tar,S}}(t_0)$, the fault tree and the risk filter, Uptime LOCATE$^{\text{TM}}$ derives reliability targets $R_{\text{tar},C_i}(t_0)$ for the components C_i. The allocation is such that components with more risks have a "lower" survival probability target compared to components with less risks. In Figure 4.3 we can see the allocation of our lifetime reliability target 90% to component reliability targets. As components can have multiple failure modes, the component reliability target $R_{\text{tar},C_i}(t_0)$ is then further derived to a component/failure-mode reliability target $R_{\text{tar},C_i;F_j}(t_0)$ for component C_i with failure mode F_j by using a concentration factor. The exact calculations are not discussed in this thesis. Table 4.2 shows the reliability target for our component/failure-modes in the example.

Component/Failure Mode	Reliability Target survival prob. in %
Cylinder: abr. wear	99.50367
Cylinder: TMF	99.50367
Piston: HCF	99.50367
Piston: abr. wear	99.50367
Retraction Unit: abr. wear	98.71467
Retraction Plate: abr. wear	98.87117
Retraction Plate: HCF	98.87117
Actuator: abr. wear	99.41328
Actuator: Leakage	99.41328
Connect. Plate: Leakage	99.52374
Connect. Plate: HCF	99.52374
Cont. Piston abr. wear	99.39455

Table 4.2: Reliability targets of component/failure-modes in surv. probability.

In Figure 4.4 we can see the risk filter for our example "hydraulic system".

4.3 Damage calculation and equivalent test duration

Using PoF models, Uptime LOCATE$^{\text{TM}}$ is able to calculate damage values from a given test $\mathcal{T}_i$ with time t_i. A test consists of multiple time series with measurements on various parts of the system. We can either have system tests, mostly with real life environmental and load conditions, or bench tests, which are often carried out with increased load on parts of the systems, i.e. subsystems. Examples of measured parameters are temperatures, rotational speed, torque, etc. For each component/failure-mode that is influenced by this test (e.g. testing the engine gives us no information about failure modes in the drive train), Uptime LOCATE$^{\text{TM}}$ calculates the damage values. Let C_{jk} be failure mode k on component j, which is stressed by the test $\mathcal{T}_i$. Then we get a damage value $d_{i,j,k}$. This value is the accumulated load the component is exposed to in the test; dividing it by the testing time t_i we get a standardized damage value that can be compared with other damage values with different running time, so

$$\hat{d}_{i,j,k} = \frac{d_{i,j,k}}{t_i}.$$

Calculating the ratio with the standardized damage value $\hat{d}_{\text{ref},j,k}$ of our reference profile Ref with time t_{ref}, we are able to calculate the acceleration factor $\kappa_{i,j,k}$ by

$$\kappa_{i,j,k} = \frac{\hat{d}_{i,j,k}}{\hat{d}_{\text{ref},j,k}}.$$

Figure 4.5 is an example for calculating the acceleration factor. The reference profile Ref with testing time t_{ref} is driving with a speed of $60\frac{km}{h}$ and having 8 damaging events within 4 hours on the component/failure-mode C_{jk}. So $d_{\text{ref},j,k} = 8\,dmg$ and $\hat{d}_{\text{ref},j,k} = 2\,dmg/h$. On the other hand, our test $\mathcal{T}_i$ with a duration $t_i = 500$ has $\hat{d}_{i,j,k} = 6\,dmg/h$. The test i puts three times higher stress on the component for the respective failure mode compared to the reference cycle. We get an acceleration factor $\kappa_{i,j,k} = 3$. So the test duration of $500\,h$ in test i is equivalent to the reference operating time of $1500\,h$, or $90.000\,km$. Note that the damage values can be based on classified data and integrals instead of event sums, although the calculation principle is always the same. We define the equivalent test duration of test $\mathcal{T}_i$ with running time t_i on component/failure-mode C_{jk} as

$$\tilde{t}_{i,j,k} = \kappa_{i,j,k} \cdot t_i.$$

Those equivalent test duration are then needed to calculate the demonstrable reliability using WeiBayes. In our example, there are 4 tests that can be conducted on our system or its components. Table 4.3 shows the acceleration factors κ for different component/failure-modes and the 4 different tests. Note that the fourth test is only conducted on a single component, whereas the other three are conducted on the whole system. Having a testing time of $t_1 = 1500\,h$, $t_2 = 300\,h$, $t_3 = 2000\,h$ and $t_4 = 20\,h$ we can calculate the equivalent test durations seen in Table 4.4.

4.4 Demonstrable Reliability

Just using the information gain from the tests, Uptime LOCATE$^{\text{TM}}$ uses the WeiBayes method to calculate the demonstrable reliability. To simplify the following calculations we choose a fixed component/failure-mode C_{jk} and omit the indices j, k. Given β (per default set to $\beta = 2$ if no other information is known), the confidence level $(1 - \alpha) = 0.9$ and equivalent testing times for n tests $\tilde{t}_1, \tilde{t}_2, ..., \tilde{t}_n$, Uptime LOCATE$^{\text{TM}}$ uses (3.3) to calculate a lower confidence level for η and then calculates the component/failure-mode reliability $\underline{R}(t_0)$. In our example, the reliability target time is $t_0 = 10000\,h$. With this we

	T_1: Dynamic	T_2: Overload	T_3: Pr. Fluc.	T_4: Puls. rig.
Cylinder: abr. wear	1.004520056	2.219037778	1.359476994	
Cylinder: TMF	1.181817967	1.181817967	0.909090744	
Piston: HCF	421.7813773	1091.039001	2.994374595	
Piston: abr. wear	1.640949944	4.951122305	0.971803427	
Retr. Unit: abr. wear	1.800696137	2.258832947	1.194733201	
Retr. Plate: abr. wear	1.800696137	2.258832947	1.194733201	
Retr. Plate: HCF	421.7813773	1091.039001	2.994374595	
Actuator: abr. wear	1.017672814	0.100857565	6.721835582	
Actuator: Leckage	1.011158549	0.049423544	6.073217267	
Con. Plate: Leckage	1.625715708	0	7.439104832	0
Con. Plate: HCF	1.48378705	65.38011692	7.89814318	15.0847124
Cont. Piston: abr. wear	1.534081909	1.936646995	1.318645139	

Table 4.3: Acceleration factors for component/failure-modes and different tests.

	T_1	T_2	T_3	T_4
Cylinder: abr. wear	1506.78	665.71	2718.95	0
Cylinder: TMF	1772.73	354.55	1818.18	0
Piston: HCF	632672.07	327311.70	5988.75	0
Piston: abr. wear	2461.42	1485.34	1943.61	0
Retraction Unit: abr. wear	2701.04	677.65	2389.47	0
Retraction Plate: abr. wear	2701.04	677.65	2389.47	0
Retraction Plate: HCF	632672.07	327311.70	5988.75	0
Actuator: abr. wear	1526.51	30.26	13443.67	0
Actuator: Leckage	1516.74	14.83	12146.43	0
Connection Plate: Leckage	2438.57	0	14878.21	0
Connection Plate: HCF	2225.68	19614.04	15796.29	301.69
Cont. Piston: abr. wear	2301.12	580.99	2637.29	0

Table 4.4: Equivalent test duration for hydraulic system tests in hours.

can calculate the demonstrable reliability for our example, as can be seen in Table 4.5. Note that for only two component/failure-modes we reach the component/failure-mode reliability target, for a lot of component/failure-modes our demonstrable reliability is $\leq 1\%$. These calculations use only the prior information about knowing or estimating the shape parameter β of the Weibull distribution. As mentioned before, we only get a conservative lower confidence bound for the reliability. In practice, the actual reliability is much higher than the demonstrable reliability. Previous knowledge like from forerunner projects or risk filter are not included. To account for this additional input, Uptime LOCATE$^{\text{TM}}$ is using a Beta probability distribution which is parameterized according to previous knowledge.

	$\underline{R}_{j,k}(10000)$ in %
Cylinder: abr. wear	1.273E-8
Cylinder: TMF	6.147E-14
Piston: HCF	99.9546
Piston: abr. wear	4.966E-7
Retraction Unit: abr. wear	3.741E-6
Retraction Plate: abr. wear	3.741E-6
Retraction Plate: HCF	99.9546
Actuator: abr. wear	28.4277
Actuator: Leckage	21.5083
Connection Plate: Leckage	36.3135
Connection Plate: HCF	69.7547
Cont. Piston: abr. wear	1.137E-6

Table 4.5: Demonstrable reliability of example hydraulic system.

4.5 Projected Reliability

As seen in Figure 2.8, the Beta distribution is versatile. Depending on the choice of parameters, the density function can have different shapes. As it is typically the case that the mode of our density is greater 0.5, we restrict our choice of parameters. For $a > 1$ the probability density on the left side is starting at 0, which should be the case, as otherwise the purpose of the projection would be void. For the parameter b we have $0 < b \leq 1$. For the exact definition of the parameters, Uptime LOCATE$^{\mathrm{TM}}$ uses the following rules:

- **parameter a:** changes via risk filter in Uptime LOCATE$^{\mathrm{TM}}$
  1. 12 risk categorizes. The more risks are selected, the higher the possible risk to fail.
  2. $a = \exp\left(\frac{MaxRiskValue=12-(RiskValueComponent-1)}{2}\right)$ as a start value.
- **parameter b:** failure rates of forerunner product and component testing
  1. $b = \min(\frac{observedFR}{mean(observedFR)}, b_c)$.
  2. $b_c = 0.5$ if component tests fully cover the failure mode and 1 otherwise, to be discussed with engineers.
  3. observedFR: failure rate for components derived from forerunner projects.
  4. mean(observedFR): mean failure rate calculated over comparable sibling components.

The starting parameter a is then getting adapted. For this, first the demonstrable reliability $\underline{R}(t_0)$ is calculated and then the cumulative distribution function $F(\underline{R}(t_0)|a, b)$ for the Beta distribution. If $F(\underline{R}(t_0)|a, b) > \alpha$, increase parameter a until $F(\underline{R}(t_0)|a, b) \leq \alpha$. The expected value of the Beta distribution is the so called *projected reliability* and is

$$\hat{R}(t_0) = \frac{a}{a+b}.$$

Although the reliability values for this projection were shown to be closer to the real reliability, one downside of this projection is the belated adaption of the parameter a. If the demonstrable reliability is too low, the projected reliability is only depending on the input of the risk filter. Increasing the testing time might have no influence on the reliability calculations. Table 4.6 shows the 10% percentile of the Beta distribution for $b = 0.5$ and $b = 1$. The a is chosen depending on the number of risks, as explained. As the parameter a is only increased if the demonstrable reliability $\underline{R}(t_0)$ is greater than the α-quantile, and $\alpha = 0.1$, our demonstrable reliability has to be greater than the values in the table. Having less than 8 risks already requires a demonstrable reliability over 90% to increase the starting parameter a, less then that, and the demonstrable reliability has no influence on the projected reliability. Note that the minimum value of the projected reliability is 62.25% when having 12 risks and $b = 1$.

# Risks	parameter a	10% percentile of Beta(a,0.5)	10% percentile of Beta(a,1)
0	665.141633	0.997967497	0.99654419
1	403.4287935	0.996650353	0.994308725
2	244.6919323	0.994481176	0.990633994
3	148.4131591	0.990911305	0.984605044
4	90.0171313	0.985043287	0.974744976
5	54.59815003	0.975416899	0.958703637
6	33.11545196	0.959676862	0.932830215
7	20.08553692	0.934083819	0.891688108
8	12.18249396	0.892863512	0.827780247
9	7.389056099	0.827563763	0.732258797
10	4.48168907	0.727145195	0.598232269
11	2.718281828	0.581054688	0.428667068
12	1.648721271	0.390361786	0.247439384

Table 4.6: 10% percentile of the Beta distribution with different b.

Table 4.7 shows the projected reliability $\hat{R}_{j,k}(t_0)$ for component j with failure mode k and the corresponding reliability target. Half of the component/failure-modes reach the reliability target, but for ten of the twelve component/failure-modes, the demonstrable reliability has no influence on the projected reliability. Only for ”Retraction Plate: HCF” and ”Piston: HCF” do we get a high enough demonstrable reliability to change the starting parameter a of the Beta distribution.

	Reliability target $R_{\mathrm{tar},j,k}(t_0)[\%]$	proj. Reliability $\hat{R}_{j,k}(t_0)[\%]$
Cylinder: abr. wear	99.504	99.664
Cylinder: TMF	99.504	99.664
Piston: HCF	99.504	99.983
Piston: abr. wear	99.504	99.664
Retraction Unit: abr. wear	98.715	89.963
Retraction Plate: abr. wear	98.871	81.757
Retraction Plate: HCF	98.871	99.983
Actuator: abr. wear	99.413	99.093
Actuator: Leckage	99.413	98.201
Connection Plate: Leckage	99.524	99.448
Connection Plate: HCF	99.524	99.448
Cont. Piston: abr. wear	99.395	99.395

Table 4.7: Projected reliability of example hydraulic system.

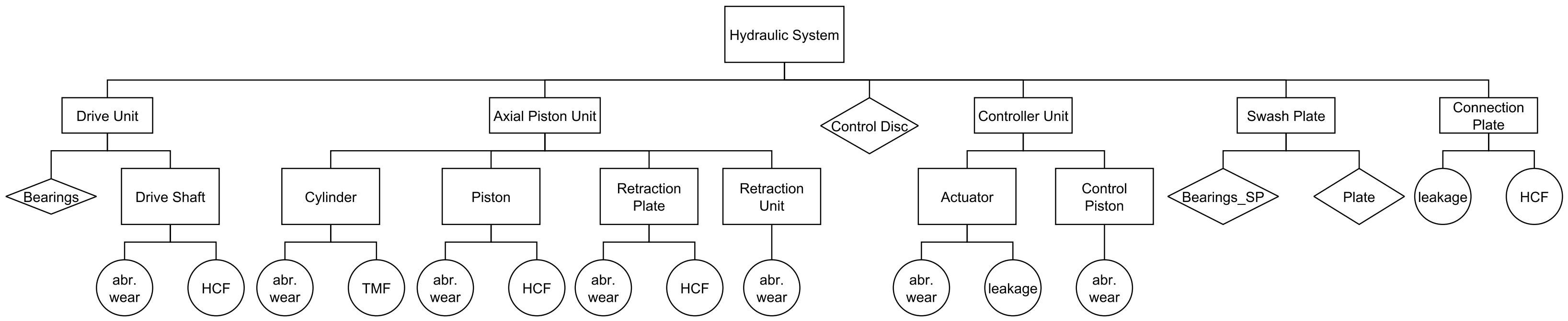
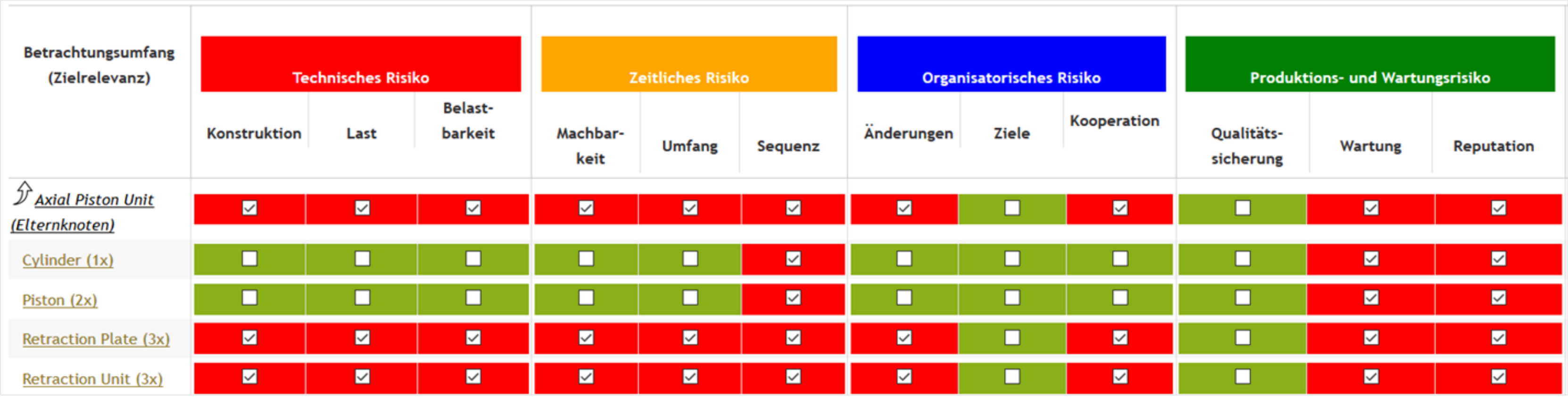
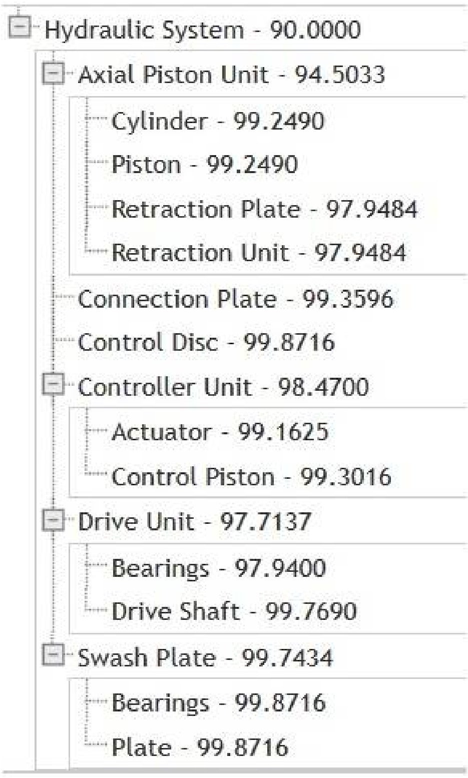


Figure 4.1: Fault tree of ”Hydraulic System”



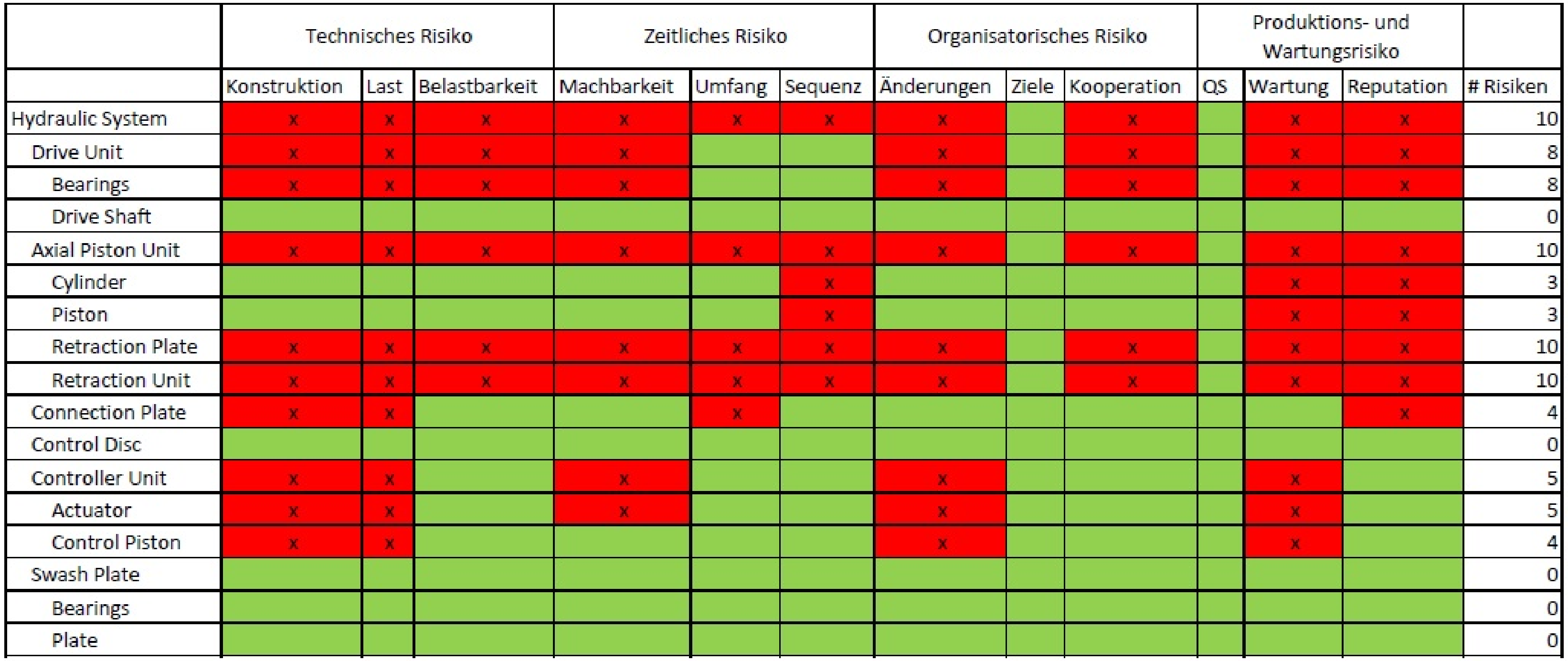
Betrachtungsumfang (Zielrelevanz)	Technisches Risiko			Zeitliches Risiko			Organisatorisches Risiko			Produktions- und Wartungsrisiko		
	Konstruktion	Last	Belast-barkeit	Machbar-keit	Umfang	Sequenz	Änderungen	Ziele	Kooperation	Qualitäts-sicherung	Wartung	Reputation
Axial Piston Unit (Elternknoten)	☑	☑	☑	☑	☑	☑	☑	☐	☑	☐	☑	☑
Cylinder (1x)	☐	☐	☐	☐	☐	☑	☐	☐	☐	☐	☑	☑
Piston (2x)	☐	☐	☐	☐	☐	☑	☐	☐	☐	☐	☑	☑
Retraction Plate (3x)	☑	☑	☑	☑	☑	☑	☑	☐	☑	☐	☑	☑
Retraction Unit (3x)	☑	☑	☑	☑	☑	☑	☑	☐	☑	☐	☑	☑

Figure 4.2: Excerpt from risk filter of ”Hydraulic System”.



- Hydraulic System - 90.0000
  - Axial Piston Unit - 94.5033
    - Cylinder - 99.2490
    - Piston - 99.2490
    - Retraction Plate - 97.9484
    - Retraction Unit - 97.9484
  - Connection Plate - 99.3596
  - Control Disc - 99.8716
  - Controller Unit - 98.4700
    - Actuator - 99.1625
    - Control Piston - 99.3016
  - Drive Unit - 97.7137
    - Bearings - 97.9400
    - Drive Shaft - 99.7690
  - Swash Plate - 99.7434
    - Bearings - 99.8716
    - Plate - 99.8716

Figure 4.3: Risk allocation in survival probability in %.



	Technisches Risiko			Zeitliches Risiko			Organisatorisches Risiko			Produktions- und Wartungsrisiko			
	Konstruktion	Last	Belastbarkeit	Machbarkeit	Umfang	Sequenz	Änderungen	Ziele	Kooperation	QS	Wartung	Reputation	# Risiken
Hydraulic System	x	x	x	x	x	x	x		x		x	x	10
Drive Unit	x	x	x	x			x		x		x	x	8
Bearings	x	x	x	x			x		x		x	x	8
Drive Shaft													0
Axial Piston Unit	x	x	x	x	x	x	x		x		x	x	10
Cylinder						x					x	x	3
Piston						x					x	x	3
Retraction Plate	x	x	x	x	x	x	x		x		x	x	10
Retraction Unit	x	x	x	x	x	x	x		x		x	x	10
Connection Plate	x	x			x							x	4
Control Disc													0
Controller Unit	x	x		x			x				x		5
Actuator	x	x		x			x				x		5
Control Piston	x	x					x				x		4
Swash Plate													0
Bearings													0
Plate													0

Figure 4.4: Risk filter Hydraulic System.

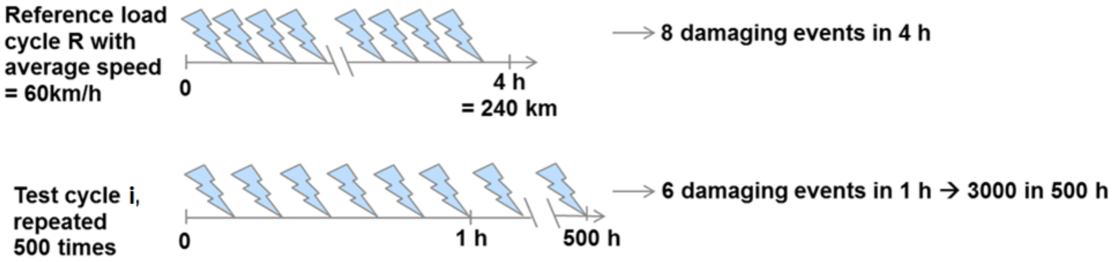

Reference load cycle R with average speed = 60km/h
0
4 h
= 240 km
8 damaging events in 4 h
Test cycle i, repeated 500 times
0
1 h
500 h
6 damaging events in 1 h → 3000 in 500 h

Figure 4.5: Example for calculation of equivalent test duration.

Chapter 5

Aggregation to system reliability

Currently, Uptime LOCATE$^{\mathrm{TM}}$ assess a validation program V consisting of several different tests $\mathcal{T}_1, \mathcal{T}_2, ..., \mathcal{T}_n$ on component/failure-mode level by comparing the demonstrable and projected reliability with the reliability target for every component/failure-mode. Those targets are, as described in Chapter 4, derived top-down from an overall system reliability target along the hierarchy gained by the fault tree. Right now, the assessment terminates with the reliability demonstration on the level of component/failure-mode. The objective is to find a reliability value for the system by aggregating the results from component/failure-mode level bottom-up. The reliability should be available on each level of the system hierarchy, i.e. every component should be assigned a reliability value, and there should be one final indicator measuring the quality of the validation program. Especially management holds interest for an easy way to compare validation programs on their ability to demonstrate a high reliability for the whole system. Additionally an optimization problem should be constructed, so that we are able to optimize a given validation program by distributing the test volume, i.e. number of objects tested or duration of the tests. As the optimization problem is a non-linear optimization problem, we are satisfied with a local optimum for large problems.

5.1 Only having base events in the fault tree

As described in Section 2.6.2, our fault tree can consist of intermediate, undeveloped and base events. For our reliability calculation the leaves of the tree, i.e. base and undeveloped events, are relevant. Let us first assume we only have intermediate and base events, but no undeveloped events. We discuss how to include the undeveloped events later.

As mentioned in Section 2.6.2, our fault tree with only logical gate "OR" can be seen as a series system. The blocks correspond to the base events in the fault tree. Also every base event BE_i in the fault tree is equivalent to a component/failure-mode C_{jk}. Thus it follows, when having n component/failure-modes for the system reliability

$$R_{\mathrm{S}}(t) = \prod_{i=1}^{n} R_{\mathrm{BE}_i}(t),$$

with $R_{\mathrm{BE}_i}(t)$ being the reliability for base event i at time t. Using the demonstrable and projected reliability already calculated in Uptime LOCATE$^{\mathrm{TM}}$, we try to implement this calculation.

Demonstrable Reliability

Calculating the system reliability with our conservative values from the demonstrable reliability, we get an even more conservative value. For our example, we have a system reliability of

$$\underline{R}_{\mathrm{S}}(10000) \approx 9.581\mathrm{e}{-57}\%.$$

when multiplying the demonstrable reliability of the base events having risks and omitting undeveloped events or base events with no risks. This value implies that with the current validation program we are only able to demonstrate a system reliability close to 0. The main problem is multiplying lower confidence bounds, as we know that the system reliability $\underline{R}_{\mathrm{S}}(t_0)$ is always less or equal the minimum component/failure-mode reliability. The resulting value for the lower confidence bound of the system reliability is unrealistic low. In Kemmner (2012) a methodology for calculating the lower confidence limit of the reliability of a system in a serial configuration when zero failures have occurred is being presented. The methodology is discussed in Chapter 3.

Note that Kemmner assumes that the calculation of reliability is on the component level. It is still applicable when expanding it to the component/failure-mode level. With this approach our reliability lower confidence limit of the system is

$$R_{\mathrm{S,LCL},(1-\alpha)}(10000) = \min_i R_{C_i,\mathrm{LCL},(1-\alpha)}(10000) \approx 6.147\mathrm{e}{-14}\%.$$

While the lower confidence limit of the system reliability will be much higher than multiplying the lower confidence limit of each component, there also arise problems with this approach. The optimization model suggested is a highly non-linear mixed-integer program. We have to maximize the minimal lower confidence bound of our components. In practice that can be critical: let us assume we have a component g with failure mode f. Also assume we do not know the PoF model of the component/failure-mode $C_{g,f}$. Thus we cannot have a bench test to get high acceleration factors, and also for system tests with higher load compared to the reference profile we do not know if the component is exposed to more or less stress. Therefore we can only assume, that for system tests with real life usage we get an acceleration factor $\kappa_{g,f} = 1$ and only those testing times can be accounted for calculating the demonstrable reliability. As system tests are often expensive and have restricted test volume, there is a upper bound for the reliability demonstration of $C_{g,f}$. If this reliability $R_{C_{g,f},\mathrm{LCL},(1-\alpha)}(t_0)$ is the minimum component reliability limit of all of our components, we do have a target value of the optimization problem proposed by Kemmner. Still there might be potential to optimize the reliability of the other components. Therefore we are going to calculate the system reliability by multiplying the reliability of the component/failure-modes.

The method by Kemmner also neglects additional knowledge beyond the estimation of the shape parameter β. If we adapt the approach to function with the projected reliability, additional troubles arise.

Projected Reliability

As seen in our example in the former chapter, the projected reliability is highly influenced by the risk filter and sometimes not at all influenced by the demonstrable reliability. In our example we have for the system reliability by multiplying the component/failure mode reliability

$$\hat{R}_{\mathrm{S}}(10000) \approx 69.626\%.$$

Increasing or decreasing testing time can have no influence on the reliability of any component. This makes it even harder to calculate a system reliability with either method, multiplication or by Kemmner. Therefore at first we want to introduce another method to calculate the reliability of a component/failure-mode by using the knowledge of the risk filter to get an a-priory probability R_0 and using the method of Beyer/Lauster as basis for our calculations. We call the calculated reliability *combined reliability*, see Section 5.3.

5.2 Handling undeveloped events

There is no equivalent presentation of undeveloped events from fault tress to reliability block diagrams. We have to distinguish between two cases that can lead to undeveloped problems.

First is a lack of importance, i.e. we have no entry in the risk filter for that component. In this case we can assume that the component reaches its reliability target and we set the reliability of the component to the same level for the demonstrable reliability. This does not cause any problems for calculating the system reliability by multiplication. Using the method by Kemmner also does not lead to problems, as the reliability target of components is always derived from the system reliability target and even if the component corresponding to an undeveloped event is the limiting component, we know that the system surely reaches its target as well.

Using this assumption for the projected reliability can be critical: let us assume we have a component with zero risks and a reliability target of 95%. Now adding up to 7 risks would result in a component reliability at least greater than 95%. So in this case, adding risks would result in an increased reliability, which does not make sense. Thus we cannot use this assumption for the projected reliability.

Having a lack of information is the second case to get an undeveloped event in the fault tree analysis. This is often due to not exactly knowing the PoF model and therefore being unable to calculate the damage in comparison to real life situations. Let us assume that we have a undeveloped event for component C_i. Thus we can only set the acceleration factors to $\kappa_i = 1$ for component i if we have tests in real life conditions, otherwise we can only set $\kappa_i = 0$. This can lead to different issues: If there are no tests under real life conditions, then no test is contributing to the reliability calculations of component i. This would result in a demonstrable reliability $\underline{R}_{C_i}(t) = 0$ and therefore both methods would result in a system reliability of $\underline{R}_{\mathrm{S}}(t) = 0$ and $R_{\mathrm{S,LCL},1-\alpha}(t) = 0$. But even if there are real life tests, those are often expensive and due to a lack of time and financial means restricted. This, in addition to a acceleration factor of $\kappa_j = 1$ for every component (due to real life conditions) results in component i likely being the limiting component, without any means of increasing the reliability. This is especially problematic for the method of Kemmner, as we might be able to increase the reliability of other components with bench-tests.

In case of the projected reliability, some of the problems are not present anymore. If there are no real life tests, we still have a projected reliability of at least $\hat{R}_{C_i}(t) \geq 62.25\%$. Tough the problem still remains, that increasing testing time of some tests, e.g. bench-tests, can lead to no increase in the projected reliability of any component and thus the system reliability. For the combined reliability we want to be able to deal with those problems: any change in testing time should result in a change in reliability for at least one component, and even if there are components with no further information, i.e. undeveloped events, we should have at least a reliability value higher than a fixed value, derived from experience, i.e. we know that the component at least has a minimum reliability.

5.3 Combined Reliability

In this section we are going to introduce a new method to calculate the reliability of components, based on the information in the risk filter and test contributions. First we are going to introduce some basic framework and the ideas for the reliability.

5.3.1 Basic idea

Based on the former chapters, we already figured out some properties which the combined reliability should have

- Including knowledge from the risk filter to get an a-priory reliability with a confidence level of 90%.
- Every change in a test duration should result in a change of reliability.
- Components with no risk have a lower reliability confidence bound of at least the reliability target, i.e. they reach their target.
- Adding an additional risk decreases the lower reliability bound.
- Components with a full risk filter should reach at least a minimum value of reliability, derived from experience.

5.3.2 Calculating a-priory probability

So starting with an empty risk filter and a lower confidence bound with confidence level of 90% at the level of the reliability target, we decrease this bound with every additional risk. Having all risks our component reaches its minimum reliability that we are confident to assign before having additional information. A conservative value would be setting the minimum to 0%, so having a full risk filter gives us no a-priory information about that component. As this approach is too conservative, instead we use an already existing lower bound, derived from the projected reliability. Having a full risk filter, the starting parameter of the Beta distribution is set to $a = e^{1/2}$. Parameter b is depending on observed failure rates or the ability to create a bench-test to fully cover the failure mode or not. For a detailed description see Chapter 4. As the projected reliability is also a lower confidence bound on the reliability of a component with confidence level of 90%, we can assume that this minimum value is reached regardless of any tests conducted.

For assigning a-priory values for components having less than 12 failures there are multiple ways to distribute them. The first approach would be having a linear decline from the maximum value to the minimum value. Adding a risk always decreases the confidence bound with the same value. The second one is the one that we are going to focus on. Again derived from the projected reliability we assume that with increasing risks, the decline of the confidence bound increases as well. With this we get the following formula for the a-priory lower confidence bound for component/failure mode i with r_i risks:

$$R_{0,i} = minbound_i + (maxbound_i - minbound_i) \cdot \frac{1}{\ln(13)} \cdot \ln(13 - r_i)$$

Having zero risks results in the maximum value, which is set to be the reliability target, having all 12 risks results in the confidence bound being the minimum value, which in our case is defined as

$$minbound_i = \frac{a}{a+b}$$

with $a = \exp(0.5)$ and $b = \min\left(\frac{observedFR}{mean(observedFR)}, b_c\right)$ with $b_c = 0.5$ if component test fully covers the failure mode and $b_c = 1$ if not or if there is no PoF model (so we assume the worst case). Figure 5.1 shows the logarithmic decline of the probability between the maximum and minimum value with increasing number of risks.

Using this method, we can calculate the a-priory probability of our example hydraulic system. Table 5.1 gives us the parameter b_c to calculate the minimum lower bound, the maximum value as the reliability target and the number of risks. Note that although the component "Drive Shaft" has two base events, those can be neglected in this example, as there is no risk, so both component/failure modes reach their reliability target and the component "Drive Shaft" reaches its reliability target as well. We could therefore simplify the fault tree by making "Drive Shaft" into an undeveloped event.

5.3.3 Adding test contributions

Having the a-priory reliability with a confidence level of 90%, next we want to include the information from the time series. As we want to use the Beyer/Lauster method, we have to adapt the formula, as the additional knowledge is valid for a higher confidence level. Thus we have to adapt equation (2.11) to

$$R_{\min}(t) = (1 - P_{\mathrm{CL}})^{\left(\sum_{i=1}^{k} n_i(\kappa_i \mathrm{LR}_i)^{\beta} + 1/\log(1/R_0)\right)^{-1}} . \tag{5.1}$$

by changing the natural logarithm to the common logarithm. R_0 is now our a-priory information. Using the validation program V with testing times $v = (1500, 300, 2000, 20)$, Table 5.2 gives us the combined reliability for our example.

Component/failure i	r_i	b_c	$minbound_i$	$maxbound_i$	$R_{0,i}$
Cylinder: abr. wear	3	0.5	0.767303462	0.99504	0.971745221
Cylinder: TMF	3	0.5	0.767303462	0.99504	0.971745221
Piston: HCF	3	0.5	0.767303462	0.99504	0.971745221
Piston: abr. wear	3	0.5	0.767303462	0.99504	0.971745221
Retraction Unit: abr. wear	10	0.5	0.767303462	0.98715	0.861467547
Retraction Plate: abr. wear	10	1	0.622459331	0.98871	0.779330844
Retraction Plate: HCF	10	0.5	0.767303462	0.98871	0.862135722
Actuator: abr. wear	5	0.5	0.767303462	0.99413	0.951195017
Actuator: Leckage	5	1	0.622459331	0.99413	0.923778118
Connection Plate: Leckage	4	0.5	0.767303462	0.99524	0.962561806
Connection Plate: HCF	4	0.5	0.767303462	0.99524	0.962561806
Cont. Piston: abr. wear	4	0.5	0.767303462	0.99395	0.961456747
Drive Shaft	0	0.5	0.767303462	0.99769	0.99769
Control Disc	0	1	0.622459331	0.99872	0.99872
Bearings_SP	0	1	0.622459331	0.99872	0.99872
Plate	0	1	0.622459331	0.99872	0.99872
Bearings	8	1	0.622459331	0.97940	0.846430164

Table 5.1: A-priory reliability of the component/failure modes and components.

Component/failure i	$R_{\text{tar},i}(t_0)$	$R_{0,i}$	$\tilde{R}_{C_i}(t_0\|v)$
Cylinder: abr. wear	0.99504	0.971745221	0.97178021
Cylinder: TMF	0.99504	0.971745221	0.97176799
Piston: HCF	0.99504	0.971745221	0.99955341
Piston: abr. wear	0.99504	0.971745221	0.97178691
Retraction Unit: abr. wear	0.98715	0.861467547	0.86257871
Retraction Plate: abr. wear	0.98871	0.779330844	0.78212788
Retraction Plate: HCF	0.98871	0.862135722	0.99954772
Actuator: abr. wear	0.99413	0.951195017	0.95301764
Actuator: Leckage	0.99413	0.923778118	0.92737836
Connection Plate: Leckage	0.99524	0.962561806	0.963896
Connection Plate: HCF	0.99524	0.962561806	0.96608644
Cont. Piston: abr. wear	0.99395	0.961456747	0.96153778
Drive Shaft	0.99769	0.99769	0.99769
Control Disc	0.99872	0.99872	0.99872
Bearings_SP	0.99872	0.99872	0.99872
Plate	0.99872	0.99872	0.99872
Bearings	0.97940	0.846430164	0.84643016

Table 5.2: Combined reliability of the component/failure modes and components

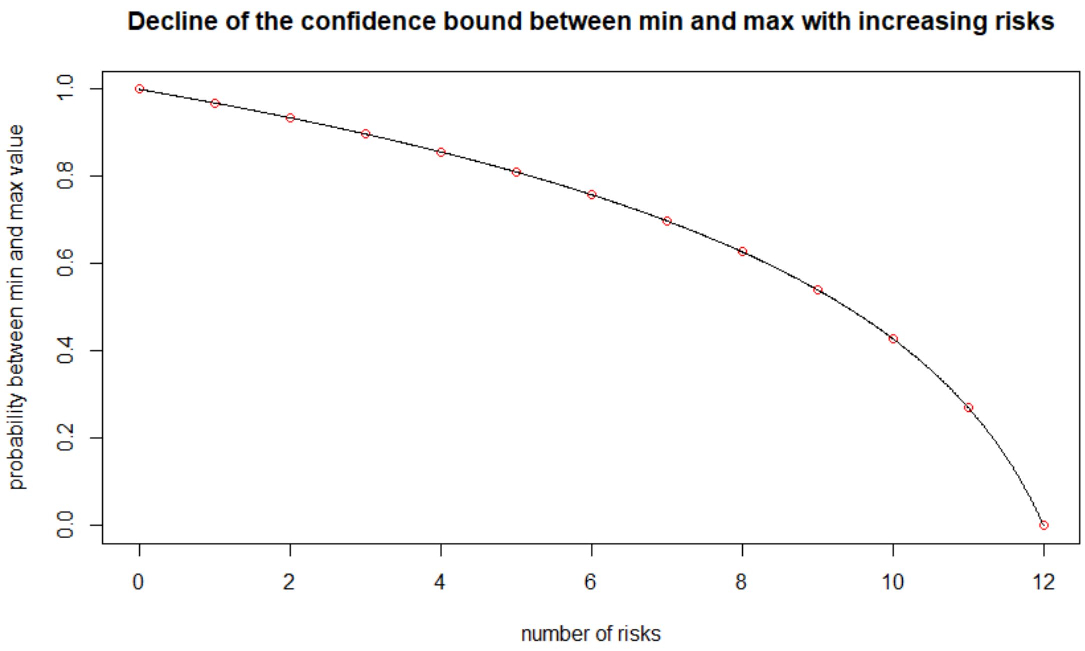


Figure 5.1: Decline between max = 1 and min = 0 with increasing risks

Calculating the system reliability with this new method, we get the following values for considering only the base events with risks

$$\tilde{R}_\text{S}(t_0|v, \text{only base events}) \approx 48.9506\%$$

and with including the undeveloped events as well

$$\tilde{R}_\text{S}(t_0|v, \text{all events}) \approx 41.1790\%.$$

5.4 Comparison of the three methods

Although it has already been mentioned before, we are going to discuss the pros and cons of the three different methods to calculate system reliability.

The advantage of the demonstrable reliability is hardly needing to know anything about the product. The only additional knowledge is included by selecting a shape parameter for the Weibull distribution, although if there is no knowledge, this parameter can also be estimated. The calculated values are often conservative, so although we are calculating $(1 - \alpha)$ lower confidence limits, the probability of the real reliability being lower than that bound is in most cases way less than α. Multiplying those conservative lower bounds results in an even more conservative system reliability bound. If there is even one component having hardly any test contribution, i.e. no bench-tests or low acceleration factors, we get an unrealistic system reliability. Additionally there might be problems dealing with undeveloped events. Having an undeveloped event because of no importance, i.e. no risk, is no problem, as we can safely assume that the component reaches its reliability target. But undeveloped events from having no information about the failure mode are harder to handle: We can only assume that tests under real life conditions contribute to an acceleration factor of $\kappa = 1$, if there are no such tests we can only assume to have a component reliability of 0% and thus having problems for the calculations of the system reliability.

The projected reliability is able to handle this problem: including knowledge about the failure modes by means of the risk filter, results in being able to estimate lower confidence bounds even for undeveloped events with risks. Tough as mentioned before, different problems arise: we cannot assume that undeveloped events with no risks reach their reliability target, as it can lead to lower reliability bounds compared to having a few risks. On the other hand, not assuming this can result in a component not reaching its reliability target, although it has no risk. Additionally, as the 10% percentiles of the Beta distribution are quite high, see Table 4.6, it can often happen that increasing or decreasing testing times has no changes in the reliability of any component. Only trough extensive testing or bench-tests with high acceleration factors do we get high enough demonstrable reliability values to adapt the parameters of the Beta distribution of the projected reliability. Another flaw is the minimum value of the projected reliability. Having a full risk filter, we still assume that the component reaches a reliability of at least 62.2%. This assumption may hold for quite a lot of products, as often those products are refined for many years and the knowledge about functionality, failure modes and weak points are extensive. Still, for the evaluation of completely new products, e.g. electric motors, assuming a lower confidence bound of at least 62.2% with confidence level of 90% can be critical.

While we choose the minimum reliability of the combined reliability to be of the same level, we can simply adapt the minimum value of single components if needed. For example, analyzing a electric car, a lot of components are staying the same compared to a car with a combustion engine. The fault trees are similar for a lot of components/failure modes. For these we can use the default calculations. Only for new components or component/failure modes which are influenced by the electric engine can we adapt the minimum value, i.e. in the worst case set it to be close to 0%.

There are also additional benefits of the combined reliability compared to the projected reliability: calculating the projected reliability for the four components with no risks in our example results in a projected reliability of $\hat{R} = 99.8499\%$. Although having no risk at all, three of the components (Control Disc, Bearings_SP and Plate) do not reach their reliability target. For the projected reliability we cannot assume that components with no risks reach their target, as mentioned before. Thus it is more reasonable to omit this assumption for the projected reliability.

The combined reliability on the other hand is flexible: Although we fixed the calculation of the minimum value, other approaches might be possible. Also with using the advanced formula from Beyer/Lauster (2.12) with the weighing factor ρ, we are able to regulate the influence of the a-priory knowledge. In this thesis we assume that $\rho = 1$, research on the choice of ρ can be analyzed in the future.

Component/failure i	$R_{\text{tar},i}(t_0)$	$\underline{R}_i(t_0\|v)$	$\hat{R}_i(t_0\|v)$	$\tilde{R}_i(t_0\|v)$
Cylinder: abr. wear	0.99504	1.27395E-10	0.99664	0.97178021
Cylinder: TMF	0.99504	6.1471E-16	0.99664	0.97176799
Piston: HCF	0.99504	0.99954634	0.99983	0.99955341
Piston: abr. wear	0.99504	4.96637E-09	0.99664	0.97178691
Retraction Unit: abr. wear	0.98715	3.74116E-08	0.89963	0.86257871
Retraction Plate: abr. wear	0.98871	3.74116E-08	0.81757	0.78212788
Retraction Plate: HCF	0.98871	0.99954634	0.99983	0.99954772
Actuator: abr. wear	0.99413	0.284276712	0.99093	0.95301764
Actuator: Leckage	0.99413	0.215083083	0.98201	0.92737836
Connection Plate: Leckage	0.99524	0.363134758	0.99448	0.963896
Connection Plate: HCF	0.99524	0.697546975	0.99448	0.96608644
Cont. Piston: abr. wear	0.99395	1.13747E-08	0.99395	0.96153778
System Rel.: only BE	0.90	9.581E-59	0.69626	0.48951
Drive Shaft	0.99769	0.99769	0.99925	0.99769
Control Disc	0.99872	0.99872	0.99850	0.99872
Bearings_SP	0.99872	0.99872	0.99850	0.99872
Plate	0.99872	0.99872	0.99850	0.99872
Bearings	0.97940	0	0.92414	0.84643016
complete System Rel.	0.90	0	0.64007	0.41180

Table 5.3: Comparing the component and system reliability of the three methods.

To sum it up, the combined reliability uses the same knowledge base as the other two methods, combines them and removes potential faults in those methods. The calculations can easily be adapted to deal with special cases. Compared to the other two methods it is more simple to optimize the system reliability, as the former methods lead to some problems, i.e. no change in reliability in case of the projected reliability, and no way to deal with undeveloped events and potential low reliabilities with the method of the demonstrable reliability. Table 5.3 gives a comparison of the three methods: first when omitting undeveloped events, then when factoring in the undeveloped methods.

Chapter 6

Optimization of a validation program

6.1 Nonlinear Optimization Problem

In this section, we discuss a method to find a local optimum for a given validation program and its restrictions. We also introduce an additional generic example, which we try to solve, as the example "hydraulic system" has hardly any component tests, and the component test "Pulsation rig Test" has lower acceleration factors compared to the overload test. For the generic example we want to create an acceleration matrix K which is more balanced, has tests under real life conditions and bench tests with high acceleration factors. We still use the same fault tree and risk filter tough.

For the optimization program we have given the following information:

- A validation program V with k different tests $\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_k$.
- The tests of the validation program have test durations $v = (t_1, t_2, \ldots, t_k)$ and corresponding sample sizes $n = (n_1, n_2, \ldots, n_k)$.
- The tests of $\mathcal{T}_i$ are restricted with a maximum test duration $t_i^{\max}$ for a single test and a maximum test duration for all tests from $\mathcal{T}_i$ together $\bar{t}_i^{\max}$ and a maximum sample size $n_i^{\max}$
- A maximum testing time for all tests $t^{\max}$
- Having cost as a restriction factor is possible as well, although this information is hardly available. We omit this restriction.
- A fault tree for our system.
- A risk filter for the components with the number of risks r_i.
- A PoF model for every component/failure-mode corresponding to a base event to calculate the acceleration matrix K subject to a reference profile (usage of the customer).

- A Weibull shape parameter β_i for every component/failure-mode and every component corresponding to an undeveloped event.

- Knowledge about testability of component/failure modes, i.e. parameter b_c.

- Possible former failure knowledge. Together with the testability we have our parameter b for every component/failure-mode and if there is no information we set $b = 1$.

- Reliability target $(\tau, u_\mathrm{R}, R_{\mathrm{tar,S}}(t_0), t_0, u_0)$ for the system.

- The number of undeveloped and base events l. As the calculations for both are similar in the combined reliability, we us the common index L for them, as they are the leaves of the fault tree.

With this we are able to formulate the optimization problem:

$$\max_{v,n} \tilde{R}_\mathrm{S}(t_0|v,n) = \max_{v,n} \prod_{i=1}^{l} \tilde{R}_{L_i}(t_0|v,n)$$

subject to

$$\begin{aligned}
&\tilde{R}_{L_i}(t_0|v,n) = \alpha^{\left(\sum\limits_{j=1}^{k} n_j(\kappa_{i,j}\cdot\frac{t_j}{t_0})^{\beta_i} + 1/\log(1/R_{0,i})\right)^{-1}}\\
&t_j \cdot n_j \leq t_j^\mathrm{max}\\
&0 \leq t_j \leq t_j^\mathrm{max}\\
&n_j \leq n_j^\mathrm{max}, n_j \in \mathbb{N}\\
&\sum_{j=1}^{N} t_j n_j \leq t^\mathrm{max}
\end{aligned}$$

In practice it does not make sense to restrict all tests of $\mathcal{T}_i$ to be of equal time. Kemmner (2012) also showed, that this restriction may show worse results compared to allowing different testing times. It is easy to see, that instead of dividing a maximum testing time between multiple tests, we get a higher result for our reliability calculations when trying to maximize the testing time from as many tests as possible and have a test with the remaining testing time, as $\beta > 1$ in all of our calculations. Thus we introduce some changes for our problem: we omit the parameter n_i, instead we view each test as being an individual test. We let then $N = \sum_{j=1}^{k} n_j^\mathrm{max}$ be the maximum number of possible tests. For the acceleration matrix we have to duplicate the column of test $\mathcal{T}_i$, n_i^max times. As for the new testing times, we set for n_i tests of $\mathcal{T}_i$ the starting time to t_i, for the rest we set it to 0. With this we get the new times $\tilde{t_1}, \ldots, \tilde{t}_N$. We now have to add additional restrictions, pairing equal tests and restricting them with a maximum single test duration and a maximum test duration for all of them together. With this we also lose the integer restriction in our problem. As a result we get the following optimization problem:

$$\max_v \tilde{R}_{\mathrm{S}}(t_0|v) = \max_v \prod_{i=1}^{l} \tilde{R}_{L_i}(t_0|v)$$

subject to

$$\begin{aligned}
&\tilde{R}_{L_i}(t_0|v) = \alpha^{\left(\sum_{j=1}^{N}(\kappa_{i,j}\cdot\frac{\tilde{t_j}}{t_0})^{\beta_i}+1/\log(1/R_{0,i})\right)^{-1}} && \\
&0 \le \tilde{t_j} \le t_j^{\max} && \text{for } j = 1,\dots,N \\
&\sum_{k=1}^{N} \tilde{t_k} \le t^{\max} && \\
&\sum_{k\in T_l} \tilde{t_k} \le \bar{t}_p^{\max} && \text{for } p = 1,\dots,k
\end{aligned}$$

With this we only have linear inequality constraints for our optimization problem. We can write the linear inequality constraints in the form $Av \le iq$ with $v = (\tilde{t_1}, \tilde{t_2}, ..., \tilde{t_{\tilde{N}}})^T$ and A being a $(N+1+k)\times N$ matrix and iq being a $(N+1+k)\times 1$ vector with the form

$$A = \begin{pmatrix} & I_{N\times N} & & \\ & & & \\ 1 & 1 & \dots & 1 \\ \underbrace{1\cdots1}_{n_1} & & & 0 \\ & \underbrace{1\cdots1}_{n_2} & & \vdots \\ \vdots & & \ddots & \\ 0 & & & \underbrace{1\cdots1}_{n_k} \end{pmatrix} \qquad iq = \begin{pmatrix} t_1^{\max} \\ \vdots \\ t_N^{\max} \\ t^{\max} \\ \bar{t}_1^{\max} \\ \vdots \\ \bar{t}_k^{\max} \end{pmatrix}$$

We can use the Matlab optimization tool with *fmincon* to solve this problem.

6.2 Hydraulic System: Standard Example

We want to solve the following problem:

$$\max_v \tilde{R}_{\mathrm{S}}(t_0|v) = \prod_{i=1}^{17} \tilde{R}_{L_i}(t_0|v)$$

subject to

$$\begin{aligned}
&0 \le t_1 \le 5000 \\
&0 \le t_2 \le 5000 \\
&0 \le t_3 \le 3000 \\
&0 \le t_4 \le 1000 \\
&\sum_{k=1}^{4} t_k \le 10000
\end{aligned}$$

with starting value of $t_1 = 2000, t_2 = 300, t_3 = 2000, t_4 = 300$. Using the Matlab optimization tool, the resulting solution is $v_1 = (2000, 5000, 3000, 0)$. This result is not surprising: T_4 is a component test with lower acceleration factors compared to test T_2. Using this solution, we get a combined system reliability of $\tilde{R}_S(t_0|v_1) \approx 45.01\%$. Still, this solution is only a local optimum. The validation program V_2 with solution $v_2 = (5000, 5000, 0, 0)$ results in a combined system reliability of $\tilde{R}_S(t_0|v_2) \approx 45.07\%$. So V_2 returns a better system reliability.

6.3 Hydraulic System: generic example

For this generic example, we introduce five tests $\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3, \mathcal{T}_4$ and $\mathcal{T}_5$. The first one is a real test, the other are component tests. For the maximum sample size per test we set $n_1^{\max} = n_5^{\max} = 2, n_2^{\max} = n_4^{\max} = 3, t_3^{\max} = 4$. With this we generate the following acceleration factors, see Table 6.1. For the constraints we get the following matrix A and iq:

$$A = \left(\begin{array}{cccccccccccccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
\hline
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
\hline
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1
\end{array}\right) \quad iq = \begin{pmatrix} 5000 \\ 5000 \\ 2000 \\ 2000 \\ 2000 \\ 3500 \\ 3500 \\ 3500 \\ 3500 \\ 4500 \\ 4500 \\ 4500 \\ 700 \\ 700 \\ 20000 \\ 8000 \\ 5000 \\ 10000 \\ 8000 \\ 1000 \end{pmatrix}$$

Component/failure	$\mathcal{T}_1$		$\mathcal{T}_2$			$\mathcal{T}_3$				$\mathcal{T}_4$			$\mathcal{T}_5$	
test	$\tilde{t}_1$	$\tilde{t}_2$	$\tilde{t}_3$	$\tilde{t}_4$	$\tilde{t}_5$	$\tilde{t}_6$	$\tilde{t}_7$	$\tilde{t}_8$	$\tilde{t}_9$	$\tilde{t}_{10}$	$\tilde{t}_{11}$	$\tilde{t}_{12}$	$\tilde{t}_{13}$	$\tilde{t}_{14}$
Cylinder: abr. wear	1	1	0	0	0	2	2	2	2	0	0	0	0	0
Cylinder: TMF	1	1	0	0	0	1.7	1.7	1.7	1.7	0	0	0	0	0
Piston: HCF	1	1	0	0	0	3.3	3.3	3.3	3.3	0	0	0	0	0
Piston: abr. wear	1	1	0	0	0	3.3	3.3	3.3	3.3	0	0	0	0	0
Retraction Unit: abr. wear	1	1	0	0	0	6	6	6	6	0	0	0	0	0
Retraction Plate: abr. wear	1	1	0	0	0	5	5	5	5	0	0	0	1000	1000
Retraction Plate: HCF	1	1	0	0	0	1.2	1.2	1.2	1.2	0	0	0	800	800
Actuator: abr. wear	1	1	0	0	0	0	0	0	0	10	10	10	0	0
Actuator: Leckage	1	1	0	0	0	0	0	0	0	6	6	6	0	0
Connection Plate: Leckage	1	1	3.5	3.5	3.5	0	0	0	0	1.8	1.8	1.8	0	0
Connection Plate: HCF	1	1	6	6	6	0	0	0	0	1.5	1.5	0.15	0	0
Cont. Piston: abr. wear	1	1	0	0	0	0	0	0	0	4	4	4	0	0
Drive Shaft	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Control Disc	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Bearings_SP	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Plate	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Bearings	1	1	0	0	0	0	0	0	0	0	0	0	0	0

Table 6.1: Acceleration factors of generic example.

As before we use the optimizer tool in Matlab with the function *fmincon*. We use the standard interior point algorithm. As for the subproblem algorithm, there are two possible choices: ldl factorization and CG. We use the same start point

$$v_{start} = (1000, 0, 1000, 0, 0, 1000, 0, 0, 0, 1000, 0, 0, 500, 0) \tag{6.1}$$

and compare the results of both methods. Our $\beta = 2$ for every component and component/failure mode. The other parameters stay the same, i.e. a-priori probability.

6.3.1 Ldl factorization

For $\tilde{R}_{\mathrm{S}}(t_0|v_{start}) \approx 50.78\%$. Using the optimizer tool we get the solution

$$\begin{aligned} &v_1 = (1000, 0, 0, 0, 0, 0, 3333, 3333, 3333, 0, 4500, 3500, 300, 700) \\ &\tilde{R}_{\mathrm{S}}(t_0|v_1) \approx 56.76\%. \end{aligned}$$

As mentioned before, we are able to find a better solution, by changing the time distribution of T_3 by trying to maximize as many tests as possible. We then get the solution

$$\begin{aligned} &v_2 = (1000, 0, 0, 0, 0, 0, 3500, 3500, 3000, 0, 4500, 3500, 300, 700) \\ &\tilde{R}_{\mathrm{S}}(t_0|v_2) \approx 56.77\% \end{aligned}$$

This is possible as all of our exponents $\beta > 1$.

6.3.2 CG

The CG (conjugent gradient) method behaves differently compared to the ldl factorization. Using the same starting point we get the solution

$$\begin{aligned} &v_3 = (0, 0, 1300, 0, 0, 0, 3500, 3000, 3500, 0, 4500, 3500, 700, 0) \\ &\tilde{R}_{\mathrm{S}}(t_0|v_2) \approx 56.78\% \end{aligned}$$

In this case, the CG method gives the best solution. Note that using different starting points, the optimization tool converges to different local optima.

Table 6.2 shows the component/failure reliability when using the different validations programs v_1, v_2 and v_3. In all three scenarios we have the restricting component "bearings". Although v_3 has the highest system reliability, using the method by Kemmner would result in v_1 or v_2 being the better solution, as it has a higher minimal component reliability. Using the approach by Kemmner the optimal solution would be

$$\begin{aligned} &v_4 = (5000, 3000, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 25, 0) \\ &\tilde{R}_{\mathrm{S}}(t_0|v_4) \approx 51.02\%. \end{aligned}$$

With this validation program, "bearings" is the restricting component with a reliability of 84.98%. As we cannot increase the testing time of the tests under real conditions, this validation program v_4 is an optimal solution as per Kemmner. But as we have shown, we are still able to increase the reliability of the other components with the remaining testing time.

Component/failure i	$\tilde{R}_i(t_0\|v_1)$	$\tilde{R}_i(t_0\|v_2)$	$\tilde{R}_i(t_0\|v_3)$
Cylinder: abr. wear	0.9722	0.9722	0.9722
Cylinder: TMF	0.9721	0.9721	0.9721
Piston: HCF	0.973	0.973	0.973
Piston: abr. wear	0.973	0.973	0.973
Retraction Unit: abr. wear	0.9195	0.9197	0.9197
Retraction Plate: abr. wear	0.9996	0.9996	0.9995
Retraction Plate: HCF	0.9994	0.9994	0.9993
Actuator: abr. wear	0.9711	0.9711	0.9711
Actuator: Leckage	0.9451	0.9451	0.9451
Connection Plate: Leckage	0.9632	0.9632	0.9633
Connection Plate: HCF	0.9628	0.9628	0.9632
Cont. Piston: abr. wear	0.9645	0.9645	0.9645
Drive Shaft	0.9977	0.9977	0.9977
Control Disc	0.9987	0.9987	0.9987
Bearings_SP	0.9987	0.9987	0.9987
Plate	0.9987	0.9987	0.9987
Bearings	0.8465	0.8465	0.8464
System S	0.5776	0.5777	0.5778

Table 6.2: Component and component/failure reliability with the three validation programs v_1, v_2, v_3

Chapter 7

Conclusion

The focus of this master thesis is calculating system reliability based on component reliability by using the structures given by Uptime LOCATE$^{\text{TM}}$. Although the calculation by multiplication by itself is simple, there arise problems when using this approach with the existing methods to calculate component reliability, i.e. demonstrable and projected reliability. Thus a new method to calculate a component lower confidence bound for reliability, namely combined reliability, is being introduced and compared with the old methods. While this method is simply implemented, there remain open questions about choosing the weighting factor ρ and the minimum bound of the combined reliability. It might be advisable to check the method with existing data of large reliability systems and compare the values with the real life data. Still, with this method we are able to compare different validation programs and optimize given validation programs. The optimization of a validation program with the former methods is more difficult, especially with the projected reliability. As the optimum is only a local optimum, it might be a good idea to not only choose a given validation program as a starting point, but also random generated starting values.

The constructed optimization problem can also be extended. We can include testing cost as a restriction. Right now we assumed that all test are conducted one after another. In practice we can execute multiple tests at the same time. This parallel testing is more difficult to describe with linear equalities and inequalities. One approach might be constructing an assignment problem.

Future research in this area might also include the possibility to extend the method to deal with failures. Instead of the Success Run assumption, we might have tests with failure times.

Bibliography

Abernethy, R. B. (2006). *The new Weibull handbook.* Abernethy, North Palm Beach.

Bertsche, B. and Lechner, G. (2004). *Zuverlässigkeit im Fahrzeug und Maschinenbau.* Springer, Berlin, Heidelberg.

Beyer, R. and Lauster, E. (1990). Statistische Lebensdauerprüfpläne bei Berücksichtigung von Vorkenntnissen. *QZ*, 35. 93 - 98.

Hamada, M. S., Wilson, A. G., Reese, C. S., and Martz, H. F. (2008). *Bayesian Reliability.* Springer, New York.

Hitziger, T. (2007). *Übertragbarkeit von Vorkenntnissen bei der Zuverlässigkeitstestplanung.* PhD thesis, University of Stuttgart.

Jordan, C. (2011). Auswertung und Planung von Lebensdauertests unter Berücksichtigung von beschleunigenden Faktoren. Master's thesis, University of Technology Graz.

Kemmner, M. W. (2012). *Reliability demonstration of a multi-component Weibull system under zero-failure assumption.* PhD thesis, University of Louisville.

Krolo, A. (2004). *Planung von Zuverlässigkeitstests mit weitreichender Berücksichtigung von Vorkenntnissen.* PhD thesis, University of Stuttgart.

Lawless, J. F. (1982). *Statistical Models and Methods for Lifetime Data.* John Wiley & Sons, New York.

McCool, J. I. (2012). *Using the Weibull Distribution.* John Wiley & Sons, Hoboken, Ney Jersey.

Meeker, W. Q. and Escobar, L. A. (1998). *Statistical Methods for Reliability Data.* John Wiley & Sons, New York.

Nelson, W. (1985). Weibull analysis of reliability data with few or no failures. *Journal of Quality Technology*, 17. 140 - 146.

Rinne, H. (2009). *The Weibull Distribution.* CRC-Press, Florida.

Romer, A. (2014). *Anwendungsspezifischer Zuverlässigkeitsnachweis auf Basis von Lastkollektiven und Vorwissen.* PhD thesis, University of Stuttgart.

Ross, S. M. (2009). *Introduction to Probability and Statistics for Engineers and Scientists.* Academic Press.

Stange, K. (1977). *Schätz- und Testverfahren bei Berücksichtigung von Vorinformationen.* Springer, Berlin.

VDA (2016). *Zuverlässigkeitssicherung bei Automobilherstellern und Lieferanten.* VDA QMC, Berlin.