NTNU

Norwegian University of
Science and Technology

Effect of Safe Failures on the Reliability of Safety Instrumented Systems

Eva Kvam

Master of Science in Physics and Mathematics

Submission date: December 2008

Supervisor: John Sølve Tyssedal, MATH

Co-supervisor: Marvin Rausand, Institutt for produksjons- og kvalitetsteknikk

Ragnar Aarø, Satetec AS

Norwegian University of Science and Technology
Department of Mathematical Sciences

Problem Description

Effect of Safe Failures on the Reliability of Safety Instrumented Systems

Assignment given: 05. August 2008
Supervisor: John Sølve Tyssedal, MATH

Preface

The work with this thesis was carried out in the 10th semester of my Master's degree on Industrial Mathematics at the Norwegian University of Science and Technology. The title of the thesis is *Effect of Safe Failures on the Reliability of Safety Instrumented Systems* and it is written under guidance of professor Marvin Rausand at the Department of Production and Quality Engineering. The supervisor at my own institute was John Sølve Tyssedal whose main task was to ensure that the thesis met the requirements to mathematical contents.

Even though I did not have any prior knowledge about the field safety and reliability before starting working with this master thesis, I have found the subject most interesting, although challenging. It is supposed that the reader have some basic knowledge about reliability of safety systems and is familiar with the textbook *System Reliability Theory: Models, Statistical Methods, and Applications* by Rausand and Høyland [2004].

I will thank Marvin for guiding me through the process of specifying such an interesting topic and for helpful support during the work to this final product. We have had many informative and encouraging conversations during the autumn. I would also like to thank my future colleagues at Safetec and especially Ragnar Aarø, the division manager in System Analysis, for including me in the workplace environment and for helpful assistance.

Trondheim,
August 5, 2008

Abstract

Safety instrumented systems (SISs) are of prime importance to the process industry to avoid catastrophic consequences or even loss of human life. The dangerous situations that any equipment may face should be analysed in order to quantify the associated risk and to choose a design of the SIS that reduces the risk to a tolerable level.

The safe failure fraction (SFF) is a parameter defined in the standards IEC 61508 and IEC 61511, and is used to determine the need for additional channels that can activate the safety function if a failure is present. The standards consider a high SFF as an indicator of a safe design, and by increasing SFF, one may allow a lower redundancy level for a SIS and therefore reduce costs. Safety engineers discuss the suitability of this parameter, and some argue that the negative effects of safe failures on the reliability are so significant that the parameter should not be used.

For a safety shutdown valve installed to prevent overpressure, a safe failure is defined as a spurious closure where the source of high pressure is isolated. This thesis examines the effects of safe failures on the reliability of such systems by using a Markov model. According to IEC 61508 and IEC 61511 the system reliability of a safety shutdown system is measured by the probability of failure on demand (PFD).

From the results it can be concluded that the time needed to restore the system back to initial state after a safe failure does not have a significant effect on PFD. A long restoration time after a safe failure in combination with a high frequency of safe failures is negative with respect to production downtime.

The main contributor to PFD is the long run probability of being in a state where a dangerous undetected (DU) failure is present. DU failures are normally detected by function tests or sometimes upon demand, but they can also be revealed by a spurious closure. This effect is based on the assumption of perfect repair of safe failures, which means that all possible failure modes are detected and the failed items are repaired or replaced after restoration of safe failures. The ability to reveal DU failures is clearly dependent on the frequency of a DU failure and safe failure occurring in the same test interval. This thesis demonstrates that safe failures only have significant effect when the dangerous failure rate is high. Other parameters affect the PFD to a greater extent, and the importance of exact parameter estimation is crucial and more important than the positive effects of safe failures.

The SFF must be close to 100% to have a significant effect on the PFD, and since it is always aimed at minimising the number of dangerous failures, the alternative is to

add safe failures. This is probably not the intent of SFF and is negative with respect to production downtime.

Safe failures does not justify a lower degree of redundancy. On the other hand, the positive effects of safe failures show a satisfactory reason for adopting a longer test interval. This is an optimisation of PFD and can reduce costs or even the frequency of dangerous situations during start-up and shutdown.

This thesis demonstrates that the PFD is not affected by safe failures, and indicates no reason to be in doubt about this parameter as a measure of reliability. The SFF gives hardly any information and the choice of SIS architecture should not be based on SFF alone. An alternative parameter that considers different means of revealing DU failures seems to be a better choice.

Abbreviations

CCF	Common cause failure
DD	Dangerous detected
DOP	Delayed operation
DU	Dangerous undetected
ELU	External leakage of utility medium
EUC	Equipment under control
FMECA	Failure modes effects and criticality analysis
FSC	Fail safe close
FTC	Fail to close
FTO	Fail to open
HFT	Hardware fault tolerance
HIPPS	High integrity pressure protection system
LCP	Leakage in closed position
MTTR	Mean time to restore
PFD	Probability of failure on demand
SD	Safe detected
SFF	Safe failure fraction
SIF	Safety instrumented function
SIL	Safety integrity level
SIS	Safety instrumented system
SPO	Spurious operation
SU	Safe undetected

Contents

1 Introduction **1**

2 Basic concepts and mathematical methods **5**
2.1 Reliability theory 5
2.1.1 Safety instrumented systems 5
2.1.2 Failure classification 6
2.1.3 Safety integrity requirements 6
Quantitative requirements 6
Semi-quantitative requirements 9
Qualitative requirements 10
2.2 Mathematical models 10
2.2.1 Markov modelling 11
2.2.2 Common cause-modelling 12
2.2.3 Reliability block diagram 12
1oo1 system 12
1oo2 system 13

3 Application **17**
3.1 Possible effects of safe failures 17
3.2 1oo1 system 19
3.3 1oo2 system 23

4 Results **27**
4.1 Variation of restoration time of safe failures for a 1oo1 system 27
4.2 The effect of assuming instantaneous restoration from safe state for a 1oo1 system 29
4.3 Variation of dangerous failure rate for a 1oo1 system 29
4.4 Variation of β-factor for a 1oo2 30
4.5 Comparison of PFD for 1oo1 and 1oo2 system 33
4.6 Comparison of PFD calculated by the Markov model and normal probability calculations 33

5 Conclusions **37**

Appendices **41**

A Supplementary theory **43**
A.1 Markov model 43

B R files **47**
B.1 Variation of restoration time of safe failures 47
B.2 The effect of assuming instantaneous restoration from safe state 49
B.3 Variation of dangerous failure rate 51
B.4 Variation of beta factor 53
B.5 Comparison of PFD for 1oo1 and 1oo2 system 55

CHAPTER 1

Introduction

Background

Reliability of safety instrumented systems (SISs) is an important issue for safe plant operation and SIS selection. A SIS comprises sensors, logic solvers and final elements, and a simplified SIS is illustrated in Figure 1.1. IEC 61508 and IEC 61511 are international standards providing a framework for design and implementation of SISs where safety integrity is a fundamental concept. Safety integrity is, according to IEC 61508, part 5, defined as "the probability of a safety related system satisfactorily performing the required safety functions under all the stated conditions within a stated period of time". Safety integrity is classified into four discrete levels (SILs) where both quantitative and qualitative requirements must be met in order to achieve a given SIL. For SISs operating in the low demand mode of operation, i.e., where the frequency of demands is no greater than once a year, the probability of failure on demand (PFD) is used as a quantitative measure [IEC 61508, IEC 61511]. Practical experience has shown that this estimate does not cover all aspects of SIS failures and may be a too optimistic measure. Architectural constraints have been introduced to avoid selecting the SIS design based on PFD alone. The architectural constraints are expressed by the hardware fault tolerance (HFT) which is the number of failures that can be tolerated before the SIS is no longer able to respond adequately upon demand. The HFT is in turn is based on the type of component (A or B), the safe failure fraction (SFF) and the given SIL. The SFF is the proportion of safe failures among all failures of a SIS where a safe failure is either safe with respect to the safety function or detected and repaired immediately after arising. The standards assume that a high SFF indicates safe design and allow for a lower HFT if the SFF is increased.

Today, there is no upper limit for the fraction of safe failures and safe failures must be assumed to have a positive effect on the availability of a SIS. Some researchers claim that the negative effects are more important and question the suitability of SFF [CCPS, 2007, Langeron et al., 2008, Lundteigen and Rausand, 2008a,b].

Itaru Yoshimura and Yoshinobu Sato have recently proposed a paper that has been accepted for publication in IEEE Transactions on Reliability [Yoshimura and Sato, 2008]. The title of the paper is "Safety Achieved by the Safe Failure Fraction (SFF) in IEC

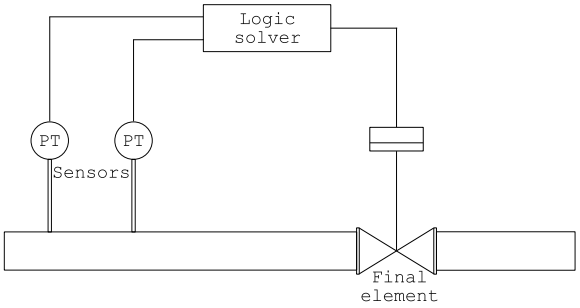

Figure 1.1: Illustration of a SIS [Lundteigen and Rausand, 2008a].

61508" and it examines the effect of safe failures to reduce the possibility of dangerous events and discusses the validity of the SFF constraints in IEC 61508. The Japanese researchers use Markov models to analyse and quantify the effect on a selection of safety systems, and conclude that the application of the SFF constraints to the standard should be put on hold until they are validated. They argue that the effects of SFF on safety are almost negligible, while the negative effects become much stronger.

No firm conclusion of the impact of safe failures has yet been drawn, and this thesis tries to study the relationship between SFF and PFD to gain more insight into the effects of safe failures. A master thesis written by Munkeby [2008] tries to examine these effects, and the thesis at hand attempts to extend his work and go into details with a specific SIS.

Objectives

The main objective of this thesis is to discuss possible effects of safe failures on system reliability. This thesis will give insight into what has been written about the problem and hopefully answer questions like: Will PFD change when safe failures are taken directly into account in the calculation? Can an increased portion of safe failures be a reason for choosing a lower degree of redundancy? Or is it possible to increase the test interval?

The main objective of this thesis is:

To evaluate the effect of safe failures on the safety integrity.

The following objectives have been the guiding principle through the work with this thesis:

1. To identify the positive effects of safe failures on the safety unavailability

2. Incorporate the positive effects of safe failures into a Markov model for different case studies

3. Quantify the PFD by applying realistic parameter values to the model

4. Carry out sensitivity analyses on the parameters

Limitations

This thesis aims at evaluating the positive effect of safe failures. Negative effects as discussed in [Langeron et al., 2008, Lundteigen and Rausand, 2008a,b] are omitted from the analysis, but should be discussed before a final conclusion is drawn.

To clarify the analysis, a specific SIS has been analysed. Only the final element is treated, and in the light of this specification, a detailed description of failure modes and transitions between these are possible to examine.

This thesis focuses on the IEC 61508 approach to quantify the safety integrity, and the scope of this work has thus been limited to only consider random hardware failures. It follows that systematic failures are omitted in this thesis.

Some authors state that safe failures get less attention when collecting data to the OREDA project and uncertainties in the variables will be a limitation of reliable PFD values.

These limitations do not, however, prevent the possibility of reaching the main objective of this thesis. It is still possible to conclude whether or not safe failures have an effect.

Structure of the thesis

The thesis is organised as follows: Chapter 2 gives a brief introduction to the reliability theory related to safe failures and how the safety integrity should be quantified. A detailed procedure for examining the positive effects of safe failures for a HIPPS system is given in Chapter 3. The results are presented in Chapter 4 and in Chapter 5 these results are discussed and a conclusion drawn. Finally, the theory behind Markov modelling is included in appendix A1. It is mathematical proofs not directly necessary to understand the implementation and it is considered reasonable to move it to the Appendix. Appendix B comprises programming code implemented to get numerical results from the models.

CHAPTER 2

Basic concepts and mathematical methods

Some prior knowledge about reliability analysis and mathematical methods required to carry out such analysis is necessary when reading the following report. The reader should be familiar with the textbook System Reliability Theory: Models, Statistical Methods, and Applications [Rausand and Høyland, 2004] or similar publications.

The first section of this chapter gives a brief introduction to the concepts related to SISs and reliability requirements where the focus of attention is safe failures. The last section presents mathematical tools used to evaluate the reliability of a SIS.

2.1 Reliability theory

The theory presented in this section is mainly based on Goble and Cheddie [2005], OLF-070 [2004], Rausand and Høyland [2004].

2.1.1 Safety instrumented systems

SIS is a physical safety system with the purpose of mitigating the risk associated with the so-called equipment under control (EUC). OLF-070 [2004] defines EUC as "a piece of equipment, machinery, part of an offshore installation, or even the entire installation." A simplified SIS is illustrated in Figure 1.1 where the final element is a safety shutdown valve intended to stop the flow if high pressure is detected by the pressure transmitters (PTs). A safety instrumented function (SIF) is a specific function implemented by a SIS which task is to protect the EUC against a single, specific hazard by carrying the system to safe state. One or more SIFs may be implemented in a SIS for a common purpose, e.g., to protect a reactor containing flammable liquid. Two possible SIFs that is implemented is one that protects against high temperature and another SIF that is implemented to protect against high pressure.

IEC 61508 and IEC 61511 require that reliability targets are assigned to each SIF that is implemented into a SIS, and the IEC standards use safety integrity level (SIL) as a measure of reliability. Safety integrity is defined [IEC 61508] as "the probability of a safety related system satisfactorily performing the required safety functions under all the stated conditions within a stated period of time". Each SIF has to fulfil a safety

integrity requirement, where SIL 1 has the lowest level of safety integrity and SIL 4 is the most stringent.

IEC 61508 distinguishes between *hardware safety integrity* and *systematic safety integrity* where both parts must be evaluated according to their respective requirements in order to fulfil a specified SIL. Hardware and systematic safety integrity is defined as [IEC 61508, Part 4] "the safety integrity of the SIS related to random hardware failures and systematic failures", respectively. As seen there is a close relationship between safety integrity and failure classification.

2.1.2 Failure classification

IEC 61508 differentiates between two main categories of failure classification; classification by *cause* or *effect*. A *random hardware failure* is a physical failure occurring at random time, which is due to natural degradation mechanism in the hardware. A *systematic* failure is related in a deterministic way to a certain cause. The error is made during the specification, design, operation or maintenance phase of the safety system and this classification rule is based on the failure cause.

IEC 61508 proposes a failure classification by effect where failures are categorised as *dangerous* or *safe*. A dangerous failure is defined as a failure having a potential to put the safety function in a fail-to-function state. This means that the safety system is not able to respond properly upon a demand. A safe failure, also called a non-dangerous failure, does not put the safety system in a fail-to-function state. It can rather result in an activation of the safety function without any demand present. Both dangerous and safe failures can further be split into *detected* and *undetected*, characterised by its ability or disability to be detected by on-line self-testing[1], respectively. It implies that a detected failure is revealed at the time the failure arises, while an undetected failure discloses oneself only when the SIS is function tested or sometimes upon demand.

Common cause failures (CCF) happen when multiple components fail due to a shared event. Repair and maintenance are often claimed to be the prime causes of CCF because of mis-calibrating and other installation failures. A CCF can also occur when two components are likely to be from the same manufacturer and therefore share the same design flaw or when two components are located at the same place making them vulnerable to the same environmental stresses. Common cause modelling is described in Section 2.2.2.

2.1.3 Safety integrity requirements

To fulfil a specified SIL, it is necessary to meet three different requirements; quantitative, semi-quantitative, and qualitative. The former two are related to hardware safety integrity which is the main topic of this thesis.

Quantitative requirements

To quantify the hardware safety integrity it is necessary to evaluate the ability of a SIF to perform its intended safety functions upon demand. A distinction is made between SISs operating in the low or high mode of operation. Low demand mode of operation means that the frequency of demands for operation is no greater than one per year or twice the test frequency. High demand mode of operation means that demands occur more than once a year or twice the test-interval. This thesis treats a safety valve which

[1]The fraction of failures detected by diagnostic self-tests is called the diagnostic coverage.

is expected not to be activated very often so it belongs to the low demand category. The probability of a SIF failure due to random hardware failures is then calculated as the *average* probability of failure on demand (PFD).

PFD is split into two contributors; $\text{PFD}_{\text{unknown}}$ and $\text{PFD}_{\text{known}}$. $\text{PFD}_{\text{unknown}}$ quantifies the loss of safety due to dangerous undetected (DU) failures occurring during the test period when it is known that the SIF is unavailable. A DU failure is the only failure mode that can prevent the safety system to respond adequately upon demand. $\text{PFD}_{\text{unknown}}(\text{t})$ is the probability that a DU failure has occurred at, or before, time t. If T_{DU} denotes the time until a DU failure, then

$$\text{PFD}_{\text{unknown}}(\text{t}) = \Pr(T_{\text{DU}} \leq t) = F_{\text{DU}}(t) = 1 - R_{\text{DU}}(t),$$

where $R_{\text{DU}}(t)$ is the survivor function with respect to DU failures, or the probability that a DU failure does not occur in the time interval $(0, \tau]$.

In reliability calculations it is the long run average value of PFD and not the time dependent value that is of interest. Each test interval of length τ is supposed to be equal in stochastic sense, hence the equation for PFD, derived from Rausand and Høyland [2004, sec. 10.3] becomes

$$\text{PFD}_{\text{unknown}} = \frac{1}{\tau}\int_0^\tau \text{PFD}_{\text{unknown}}(t)dt = \frac{1}{\tau}\int_0^\tau F_{\text{DU}}(t)dt = 1 - \frac{1}{\tau}\int_0^\tau R_{\text{DU}}(t)dt. \quad (2.1)$$

The following assumptions applies in the derivation of equation 2.1:

1. Testing and repair of components in the system are assumed to be perfect. [2]
2. The time required to test the item is negligible.
3. The restoration times for dangerous detected (DD) and DU failures are negligible.

Under these assumptions the term safety unavailability will have the same meaning as PFD and in order to avoid misinterpretations only the term PFD will be used through the rest of this thesis.

Although assumption 2 and 3 may not influence the PFD calculations, the SIS may be affected by considerable downtime. According to IEC 61508 [part 6, annex B] the contribution from restoration of dangerous failures should be included. During restoration it is known that the SIF is unavailable, and under the assumption that process demands can occur during restoration this is a contributor to $\text{PFD}_{\text{known}}$. The number of dangerous failures that occurs during a test interval of length τ is assumed to follow a Poisson process with parameter λ_{D} and the mean number of dangerous failures is equal to $\lambda_{\text{D}}\tau$. $\text{PFD}_{\text{known}}$, the average duration of restoration during a test interval of length τ, becomes

$$\text{PFD}_{\text{known}} \approx \frac{1}{\tau}\text{MTTR}_{\text{D}}\lambda_{\text{D}}\tau = \text{MTTR}_{\text{D}}\lambda_{\text{D}} \quad (2.2)$$

where MTTR_{D} is the mean time to restore a dangerous failure.

The relationship between SIL and the maximum tolerated failure probability is given in Table 2.1. It is important to notice that PFD requirements are related to the complete SIF which means that a specific quota are assigned to every component in Figure 1.1.

[2] Perfect repair means that all possible failure modes are repaired or replaced and the system is brought back to initial state after a function test.

Table 2.1: Safety integrity levels for safety functions operating in the low demand mode of operation

Safety integrity level	Probability of failure on demand
4	$10^{-5} \leq \text{PFD} \leq 10^{-4}$
3	$10^{-4} \leq \text{PFD} \leq 10^{-3}$
2	$10^{-3} \leq \text{PFD} \leq 10^{-2}$
1	$10^{-2} \leq \text{PFD} \leq 10^{-1}$

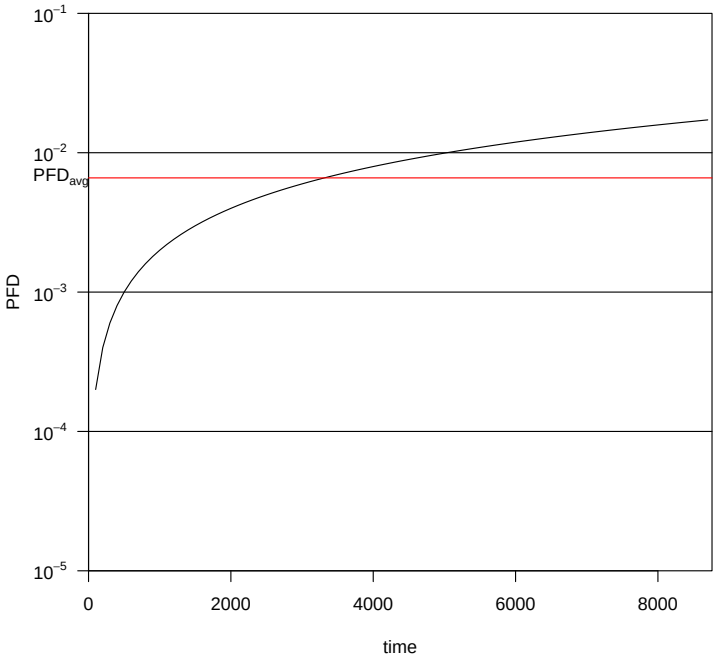

Figure 2.1: PFD, plotted on a logarithmic scale with base 10, during a test interval of 1 year.

Suppose having a SIF application with PFD as presented in Figure 2.1. To get a direct relationship between Table 2.1 and Figure 2.1, the graph is plotted on a logarithmic scale with base 10. This is a common form of PFD(t) and the average PFD during this test interval is equal to $8.7 \cdot 10^{-3}$. From Table 2.1 it is clear that this value corresponds to SIL 2. From Figure 2.1 it appears that most of the time the PFD does not reach SIL 2. This is a problem for discussion because the average PFD seems to be a too optimistic measure. Suppose having an unmanned platform where people arrive only when maintenance and repair activities are necessary. These tasks are most likely to be done during the last part of the test interval when safety integrity is lower than required. The probability of a dangerous situation is too high and in this case the maximum value of PFD seems to be a better choice. A negative response to this choice is the increased economic costs of improving the SIF.

Semi-quantitative requirements

PFD does not take into account all possible failure modes and their causes, and may lead to an optimistic value of the reliability of the system. As a solution to this problem, IEC 61508 and IEC 61511 introduces additional requirements to avoid selecting the SIS architecture based on PFD alone. These requirements are applied either to verify if a given architecture corresponds to a given SIL or to specify the required architecture of a SIF. Architectural constraints on the hardware safety integrity involve four main steps; 1) to classify the subsystem components of a SIF, 2) to calculate the SFF and HFT for each subsystem, 3) to determine the achievable SIL of a subsystem, and 4) to merge these measures in order to calculate the resulting SIL of the SIF.

A subsystem is, in accordance to IEC 61508, Part 2, classified as either type A or type B. A component is classified as type A if it is possible to determine all of its possible failure modes, the behaviour under these fault conditions and if it is possible to find sufficient failure data from field experience. Valves and solenoids are in most cases classified as type A components. Components that does not fulfil these requirements are classified as type B, e.g., logic solvers. IEC 61511 uses a different classification where, in practice, programmable electronic(PE) logic solvers are classified as type B while non-PE-logic solvers may fulfil the criteria of type A. This thesis will consider the IEC 61508 approach as this is commonly used by most oil companies and also in OLF-070 [2004], but more information about the difference between these two classifications are found in the article by van Beurden and Amkreutz [2004].

SFF is the fraction of failures that can be considered as safe and comprises both safe and DD failures. DD failures are considered safe because they are detected and repaired immediately after arising. According to IEC 61508 and IEC 61511, SFF is calculated by the following formula:

$$\mathrm{SFF} = \frac{\lambda_{\mathrm{S}} + \lambda_{\mathrm{DD}}}{\lambda_{\mathrm{S}} + \lambda_{\mathrm{D}}} = \frac{\lambda_{\mathrm{S}} + \lambda_{\mathrm{DD}}}{\lambda_{\mathrm{TOTAL}}}, \tag{2.3}$$

where λ_{S} is the safe failure rate, λ_{DD} is the DD failure rate, and λ_{DU} is the DU failure rate.

There is a great discussion among experts on what the intent of SFF really is, what to consider as a safe failure and which DD failures to include in the calculation. The PDS method [Hauge et al., 2006a] proposes an alternative SFF where non-critical failure rates are excluded. This is done in order to avoid the possibility of increasing non-essential failure rates with the intent of getting a higher SFF. CCPS [2007] makes additional

Table 2.2: Hardware safety integrity: Architectural constraints on type A safety components

Safe failure	Hardware fault tolerance		
fraction	0	1	2
99 - 100%	SIL 3	SIL 4	SIL 4
90 - 99 %	SIL 3	SIL 4	SIL 4
60 - 90 %	SIL 2	SIL 3	SIL 4
0 - 60%	SIL 1	SIL 2	SIL 3

constraints to the SFF by considering only DD failures that automatically lead to a safe state. Langeron et al. [2008] argue that not all safe failures are actually positive for safety. Human errors during repair and restoration may cause a safe failure to evolve into a dangerous failure and people may loose confidence in the SIS if there are frequent alarms. They conclude that a high SFF can not always be considered as an indicator of safe design.

HFT is the second parameter related to architectural constraints. According to IEC 61508, the fault tolerance is defined as "the ability of a functional unit to continue to perform a required function in the presence of faults and errors". In other words, the hardware fault tolerance measures the total number of faults tolerated before the safety system does not function properly. The k-out-of-n structure describes a system that is functioning if and only if at least k of the total n components are functioning, and the HFT of a general koon system is $n - k$.

With reference to the introductory chapter, this thesis will consider only one specific component, a safety valve which is of type A. The HFT table for type A components are shown in Table 2.2.

When the SIL for each subsystem is calculated it remains to determine the resulting SIL for the SIF on the basis of these results. IEC 61508 proposes some simple merging rules where the achievable SIL for subsystems in parallel is equal to the subsystem having the highest SIL plus one level while the achievable SIL for subsystems in series is restricted by the subsystem with the lowest SIL.

Qualitative requirements

Qualitative requirements are related to systematic failures in hardware or software introduced during specification, design, operation or testing. Such failures are, unlike random hardware failures, not quantified because the events leading to them cannot easily be predicted. IEC 61508 rather recommends techniques to avoid and control such failures during design phase. These measures and techniques shall be implemented during the design phase and are graded according to the given SIL requirements.

Since this thesis deals with random hardware failures, it will be a task for further work to go in detail with systematic failures.

2.2 Mathematical models

This section provides a description of mathematical models applicable for system reliability analysis and common cause modelling. The theory is derived from Ross [2003] and Littlewood and Verrall [1973] in addition to Rausand and Høyland [2004].

2.2.1 Markov modelling

The details behind Markov modelling are omitted in the main thesis, but can be found in Appendix A.1. The main advantage of Markov modelling is that it makes it possible to analyse the reliability of systems with dependent components. Detected and undetected failure modes and possible transitions between these states are easily incorporated into a Markov model. For systems with redundant components, the Markov diagram becomes large and complicated and the calculation becomes computationally extensive. In these situations it is often more efficient to use an alternative method such as fault tree, FMECA studies, reliability block diagrams, and so on.

The safety systems considered in this thesis are supposed to fulfil the Markov property and to have stationary transition probabilities. The first assumption is the characteristic property of a Markov process and implies that the future state of the system depends only on its present state and not its past. From the second assumption it follows that transition probabilities are independent of long-term trends and seasonal variations. These assumptions are mathematically expressed in Equation A.1 and A.2, respectively.

The connection between the transition probability matrix and transition rate matrix is given by Equation A.5 and A.6. Since a Markov process is completely characterised by its transition probability matrix it follows that specifying the transition rate matrix does, through this connection, determine the Markov process.

The Markov model can be used to find out what happens when the process has been running for a long time, i.e., to evaluate the limiting probabilities. This is convenient in order to evaluate the reliability of a system expressed by PFD as described in Chapter 2.1.3. It is interpreted as the average or long-run proportion of time the process will be in an unavailable state. The limiting probabilities form a vector, $\mathbf{\Pi} = [\Pi_1, \Pi_2, \ldots, \Pi_r]$, where Π_i equals the long-run proportion of time that the process will be in state i. PFD is computed as the sum of all Π_i's where i is a state where the safety system is not able to respond upon demand. The procedure used to develop a Markov model and compute limiting probabilities for a SIF is as follows:

1. Define possible states of the SIF and give them numbers from 0 up to r where $r+1$ is the total number of possible states.

2. Connect states with transition rates, $a_{ij} \quad \forall \quad i \neq j$, where a_{ij} is the rate of going from state i to j. The diagonal elements, $a_{ii} \quad \forall \quad i = 0, 1, \ldots r$, are found from equation A.6 such that the sum of each row equals 0.

3. Utilise state transition diagram and transition rate matrix, $\mathbb{A}$, where

$$\mathbb{A} = \begin{bmatrix} a_{00} & a_{01} & a_{02} & \ldots & a_{0r} \\ a_{10} & a_{11} & a_{12} & \ldots & a_{1r} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{r0} & a_{r1} & a_{r2} & \ldots & a_{rr} \end{bmatrix}.$$

4. Solve the balance equation A.17 and the normalising equation A.18, i.e., solve the following set of equations:

$$\mathbf{\Pi}\mathbb{A} = \mathbf{0}, \tag{2.4}$$

$$\sum_{j=0}^{r} \Pi_j = 1. \tag{2.5}$$

2.2.2 Common cause-modelling

There is a great deal of disagreement among experts on how to define CCF's and what impact they have on the availability of SIS. As a consequence there exists different models where the most commonly used method today is the β-factor model introduced by Fleming [1974]. The model describes the correlation between independent failures and CCF's in a redundant system. The β-factor model is applicable when the system consists of identically constructed redundant components. The β-factor denotes the fraction of common cause failures among all the failures of a component, i.e.,

$$\beta = \frac{\lambda^{(c)}}{\lambda}, \tag{2.6}$$

where $\lambda^{(c)}$ denotes the failure rate due to an external event whereby all the components of the system fails. β can also be interpreted as the conditional probability that the failure of a component will be shared by all other components of the system, i.e., Equation 2.6 can be rewritten as:

$$\Pr(\text{CCF}|\text{A failure has occurred}) = \beta.$$

A number of methods have been proposed for the assessment of β, either by different criteria or by sound engineering judgement.

In IEC 61508, random hardware failures are supposed to occur independently so only systematic failures contributes to the calculation of CCF's. Qualitatively, they suggest a method to calculate the PFD where the contribution of CCF's are modelled by using the standard β-model [IEC 61508, Part 6, Annex D]. IEC 61508 states that the model may be inadequate for a system with many redundant components. As for systematic failures, they propose qualitative guidelines on how to reduce the possibility of CCF's. They recommend to diversify and separate components to achieve maximal independence and to make staggered testing to reveal possible CCF's before they have had time to affect more than one component.

From definition 2.6 it can be seen that the β-factor model assumes that a certain percentage of all failures are CCFs. Both β-models presented so far have limitations primarily because they do not use different β's for different voting configurations such as 1oo1, 1oo2, 2oo3, and so on. It does not allow for the possibility that more than one, but not all components fail due to a CCF. The PDS method [Hauge et al., 2006a] introduces a configuration factor, C_{MooN}, and sets the β-factor for a MooN system equals βC_{MooN}. Here β is the β-factor which applies for a 1oo2 voting logic.

2.2.3 Reliability block diagram

Reliability block diagrams are often applied to determine the PFD of a SIF. This section provides a description of the application for a 1oo1 and 1oo2 system that are the systems analysed in this thesis.

1oo1 system

A 1oo1 system can be represented by the reliability block diagram in Figure 2.2. This system is operating successfully if it is possible to find a path from the leftmost node to the rightmost node.

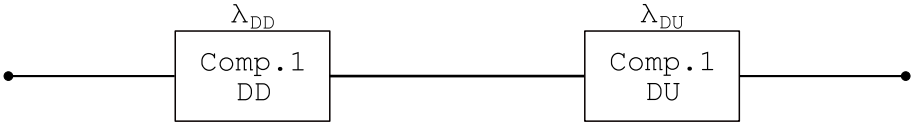

Figure 2.2: Reliability block diagram for a 1oo1 system.

Since T_{DU}, the time until a DU failure, is supposed to be exponentially distributed with parameter λ_{DU}, the survivor function $R_{\mathrm{DU}}(t)$ becomes

$$R_{\mathrm{DU}}(t) = P(T_{\mathrm{DU}} \geq t) = e^{-\lambda_{\mathrm{DU}} t}.$$

The unknown PFD from individual failures is, according to equation 2.1:

$$\begin{aligned} \mathrm{PFD}_{\mathrm{unknown}} &= 1 - \frac{1}{\tau}\int_0^{\tau} e^{-\lambda_{\mathrm{DU}} t} dt \\ &= 1 - \frac{1}{\lambda_{\mathrm{DU}}\tau}\left(1 - e^{-\lambda_{\mathrm{DU}}\tau}\right) \\ &\approx \frac{\lambda_{\mathrm{DU}}\tau}{2}. \end{aligned}$$

The approximation follows from the Maclaurin series expansion of the exponential function[3] and it can be seen that the approximation is always conservative which is important to ensure safe design.

$\mathrm{PFD}_{\mathrm{known}}$ due to repair activities is calculated by using equation 2.2, i.e.,

$$\mathrm{PFD}_{\mathrm{known}} \approx \lambda_{\mathrm{D}}\mathrm{MTTR}_{\mathrm{D}}.$$

The total PFD is the sum of these two contributors, i.e.,

$$\mathrm{PFD}_{\mathrm{tot}} = \mathrm{PFD}_{\mathrm{unknown}} + \mathrm{PFD}_{\mathrm{known}} \approx \underline{\frac{\lambda_{\mathrm{DU}}\tau}{2} + \lambda_{\mathrm{D}}\mathrm{MTTR}_{\mathrm{D}}}. \tag{2.7}$$

1oo2 system

This section looks at a 1oo2 system that can be represented by the reliability block diagram in Figure 2.3. Common cause failures are now introduced because the different components can fail due to a shared event. For this system there are three events that may contribute to $\mathrm{PFD}_{\mathrm{known}}$, and these are:

Event 1: Loss of safety due to individual DU failures, $\mathrm{PFD}^1_{\mathrm{unknown}}$**.** It can be calculated by using the survivor function of the parallel structure shown in Figure 2.3

[3] $1 - e^{-at} = at - \frac{(at)^2}{2!} + \frac{(at)^3}{3!} + \ldots \approx at - \frac{(at)^2}{2!} + \frac{(at)^3}{3!}$. This approximation is commonly used when at is less than 0.1.

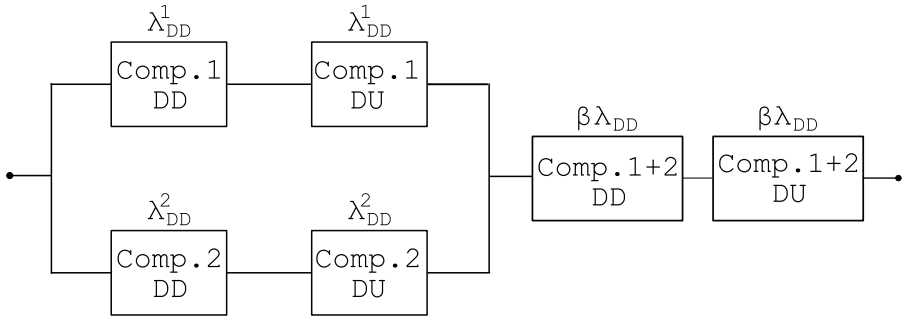

Figure 2.3: Reliability block diagram for a 1oo2 system.

and becomes:

$$\begin{aligned} R_{\text{individual}}(t) &= 1-(1-e^{-(1-\beta)\lambda_{\text{DU}}^{(1)}t})(1-e^{-(1-\beta)\lambda_{\text{DU}}^{(2)}t}) \\ &= 1-\left[1-e^{-(1-\beta)\lambda_{\text{DU}}^{(1)}t}-e^{-(1-\beta)\lambda_{\text{DU}}^{(2)}t}+e^{-2(1-\beta)t(\lambda_{\text{DU}}^{(1)}+\lambda_{\text{DU}}^{(2)})}\right] \\ &= e^{-(1-\beta)\lambda_{\text{DU}}^{(1)}t}+e^{-(1-\beta)\lambda_{\text{DU}}^{(2)}t}-e^{-2(1-\beta)t(\lambda_{\text{DU}}^{(1)}+\lambda_{\text{DU}}^{(2)})}. \end{aligned}$$

Equation 2.1 for individual failures becomes

$$\text{PFD}_{\text{unknown}}^{1}=1-\frac{1}{\tau}\int_0^\tau R_{\text{individual}} \approx \frac{[(1-\beta)\tau]^2}{6}\left[\left(\lambda_{\text{DU}}^{(1)}+\lambda_{\text{DU}}^{(2)}\right)^2-\left(\lambda_{\text{DU}}^{(1)}\right)^2-\left(\lambda_{\text{DU}}^{(2)}\right)^2\right].$$

Event 2: Loss of safety due to common cause failures, $\text{PFD}_{\text{unknown}}^{2}$. Only undetected CCFs contribute to the unknown PFD, and it is necessary to treat only the rightmost component in Figure 2.3. The failure rate due to common cause DU failures can be computed using the geometric mean [Hauge et al., 2006a, Appendix D], i.e., $\lambda_{\text{DU,CC}}=\sqrt{\lambda_{\text{DU}}^{(1)}\cdot\lambda_{\text{DU}}^{(2)}}$. [4] The survivor function $R_{\text{CC,DU}}(t)$ becomes the same as for an individual DU failure in a 1oo1 system and $\text{PFD}_{\text{unknown}}^{2}$ becomes similar to equation 2.7 except that the failure rate is multiplied by β, i.e.,

$$\text{PFD}_{\text{unknown}}^{2} \approx \beta\frac{\lambda_{\text{DU,CC}}\tau}{2} \approx \beta\frac{\sqrt{\lambda_{\text{DU}}^{(1)}\cdot\lambda_{\text{DU}}^{(2)}}\tau}{2}.$$

Event 3: Loss of safety due to degraded operation. When one component has a DD failure the system is supposed to run as a 1oo1 system, and there is a probability that the remaining component will fail DU during restoration of the other

[4] This is not always a good method while it does not take into account the various degrees of coupling between the components.

component and thus contribute to the PFD. This factor is denoted $\mathrm{PFD}^3_{\mathrm{unknown}}$ and becomes

$$\begin{aligned}\mathrm{PFD}^3_{\mathrm{unknown}} &= (1-\beta)\lambda^{(1)}_{\mathrm{DD}}\mathrm{MTTR_D}\frac{\lambda^{(2)}_{\mathrm{DU}}\tau}{2} + (1-\beta)\lambda^{(2)}_{\mathrm{DD}}\mathrm{MTTR_D}\frac{\lambda^{(1)}_{\mathrm{DU}}\tau}{2} \\ &= (1-\beta)\mathrm{MTTR_D}\frac{\tau}{2}\left(\lambda^{(1)}_{\mathrm{DD}}\lambda^{(2)}_{\mathrm{DU}} + \lambda^{(2)}_{\mathrm{DD}}\lambda^{(1)}_{\mathrm{DU}}\right).\end{aligned}$$

The total unknown PFD can be calculated by the probability of the union of these three events, i.e.,

$$\begin{aligned}\mathrm{PFD}_{\mathrm{unknown}} &= \Pr(\text{Event 1} \cup \text{Event 2} \cup \text{Event 3}) \\ &= \Pr(\mathrm{A} \cup \mathrm{B} \cup \mathrm{C}) \\ &= \Pr(\mathrm{A}) + \Pr(\mathrm{B}) + \Pr(\mathrm{C}) \\ &\quad - \Pr(\mathrm{A} \cap \mathrm{B}) - \Pr(\mathrm{A} \cap \mathrm{C}) - \Pr(\mathrm{B} \cap \mathrm{C}) + \Pr(\mathrm{A} \cap \mathrm{B} \cap \mathrm{C}) \\ &\approx \mathrm{PFD}^1_{\mathrm{unknown}} + \mathrm{PFD}^2_{\mathrm{unknown}} + \mathrm{PFD}^3_{\mathrm{unknown}}.\end{aligned}$$

This is an acceptable approximation in most cases because the probabilities of the intersections are so small that they can be neglected. It is important to notice that the approximation is always conservative which is desirable in reliability calculations.

The restoration time due to two individual DD failures are assumed negligible and the contribution from repair activities becomes the same as for a 1oo1 system only multiplied by a factor β.

$$\begin{aligned}\mathrm{PFD}_{\mathrm{known}} &\approx \beta\lambda_{\mathrm{D,CC}}\mathrm{MTTR_D} \\ &\approx \beta\sqrt{\lambda^{(1)}_{\mathrm{D}} \cdot \lambda^{(2)}_{\mathrm{D}}}\mathrm{MTTR_D}.\end{aligned}$$

The total PFD for a 1oo2 system becomes

$$\begin{aligned}\mathrm{PFD}_{\mathrm{total}} &= \mathrm{PFD}_{\mathrm{unknown}} + \mathrm{PFD}_{\mathrm{known}} \\ &\approx \frac{[(1-\beta)\tau]^2}{6}\left[\left(\lambda^{(1)}_{\mathrm{DU}} + \lambda^{(2)}_{\mathrm{DU}}\right)^2 - \left(\lambda^{(1)}_{\mathrm{DU}}\right)^2 - \left(\lambda^{(2)}_{\mathrm{DU}}\right)^2\right] \\ &\quad + \beta\frac{\sqrt{\lambda^{(1)}_{\mathrm{DU}} \cdot \lambda^{(2)}_{\mathrm{DU}}}\tau}{2} \\ &\quad + (1-\beta)\mathrm{MTTR_D}\frac{\tau}{2}\left(\lambda^{(1)}_{\mathrm{DD}}\lambda^{(2)}_{\mathrm{DU}} + \lambda^{(2)}_{\mathrm{DD}}\lambda^{(1)}_{\mathrm{DU}}\right) \\ &\quad + \beta\sqrt{\lambda^{(1)}_{\mathrm{D}} \cdot \lambda^{(2)}_{\mathrm{D}}}\mathrm{MTTR_D}.\end{aligned}$$

CHAPTER 3

Application

This chapter starts with a presentation of what impact safe failures may have on the availability of a SIS. These potential effects are discussed for different models of safety systems and their respective modelling algorithm is presented. The results are presented in Chapter 4.

3.1 Possible effects of safe failures

There are several possible effects of safe failures, but this thesis focuses on the ones that may have positive impact on the availability of a SIS. These are:

1. An increased portion of time spent in safe state reduces the possibility of going to dangerous state.
2. Safe failures can be seen as a function test where DU failures are detected.
3. Safe failures can give assurance that the system functions properly and shorten the expected time the system is unavailable due to a DU failure found by a function test.

Consider a high integrity pressure protection system (HIPPS) that is installed to prevent overpressure by isolating a low pressure rated system for a source of high pressure. The system is also called a production shutdown system. A HIPPS that is designed and built in accordance with IEC 61508 and IEC 61511 is an alternative to the conventional pressure safety valve (PSV) that opens an outlet for the fluid once a set pressure is exceeded.

This section provides an analysis of a specific final element of a HIPPS, a fail-safe-close (FSC) valve including the actuator. If a deviation from the acceptable pressure level is detected, the FSC valve is designed to close and thereby shut down the process. There are several possible failure modes related to a FSC valve, but this thesis pays attention to the ones that may be affected by a safe failure. These failure modes are given in Table 3.1.

Table 3.1: Possible failure modes related to a FSC valve

Failure mode	Abbreviation
Fail to open	FTO
Delayed operation	DOP
Fail to close on demand	FTC
Valve leakage in closed position	LCP
External leakage of utility medium	ELU
Spurious operation	SPO

The following assumptions applies for a FSC valve:

1. The Markov property is assumed to hold.
   This means that the state of the system at the future time step, $(t+1)$, is dependent on the current state, (t), but not the past $(t-n\Delta t$ for $n = 1, 2, \ldots)$. It follows that the failure and restoration rates are assumed to be constant with respect to calendar time. Constant failure rate is valid in what is called the useful life period of an item where failures are supposed to occur randomly as opposed to the burn-in and wear-out period where the failure rate is decreasing and increasing, respectively. These features can be seen from the bathtub curve [Rausand and Høyland, 2004, fig. 2.5], where the failure rate is shown as a function of time. Constant restoration rate is a rough approximation because it is expected that the time left to restore a failure will decrease, and not stay constant, as time goes by. Restoration rates for the possible failures modes are defined later in this section. To simplify the calculations, the failure rates are measured with respect to calendar time and not to operational time.

2. The system is considered working in a low demand mode of operation.
   Safety shutdown systems are not supposed to be activated more frequently than once a year and a FSC valve falls in the low demand category.

3. The system is function tested at regular time intervals of length τ and the system is supposed to be as good as new after each test interval. *This means that all possible failure modes are repaired or the failed item is replaced and the system is brought back to initial state after a function test. It follows that the system has test coverage equal to 100%.*

4. The duration of a test is assumed to be so short compared to τ that it can be neglected.

5. The failure mode SPO is denoted safe detected (SD). It follows that all SD failures result in a spurious closure of the FSC valve given that a FTC failure is not present.
   The presence of FTC failure is explained further in item 9.

6. Several failures in a system are restored simultaneously.

7. The failure mode FTO will not affect PFD since a process shutdown system already is in safe state when the failure is detected. This failure is incorporated into SD state.

8. The failure mode ELU is always supposed to result in a spurious activation and incorporated into SD state.

9. All FTC failures are assumed to be detected by a spurious closure.
The failure mode makes the process unable to get into SD state and the system is brought to DD state. This occurrence depends on a device installed to register that the valve is intended to close even though this is not possible.

10. The leakage resulting from the failure mode LCP is assumed to be so small that it can not be revealed by a spurious closure.

11. All DOP failures are assumed to be detected by a spurious closure.
This property depends on a registration of the time it takes to close the valve.

12. The failure mode DOP will not contribute to PFD since the system is brought to safe state even if a DOP failure is present. This failure is classified as safe undetected (SU), but since it does not result in a spurious closure it is not incorporated into SD state. The extra time needed to shut down the process is assumed negligible.

3.2 1oo1 system

A 1oo1 system can be represented by the reliability block diagram in Figure 2.2. The procedure for establishing Markov models is given in Section 2.2.1 where the first step is to define possible states of the system. The assumptions above reduce the collection of all possible states of a FSC valve to the ones given in Table 3.2.

Table 3.2: Possible states of a FSC valve

State	Property
0	DU_{FTC}
1	DU_{LCP}
2	DD
3	SU_{DOP}
4	SD
5	OK

Transitions between these states and their respective transition rates are given in Figure 3.1.

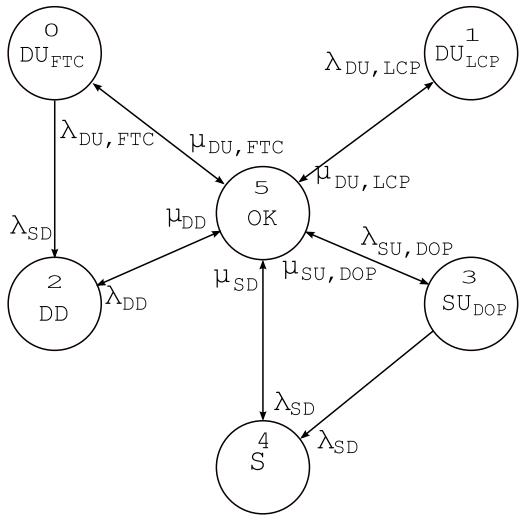

Figure 3.1: State transition diagram for a FSC valve.

The next step in the procedure is to utilise the transition rate matrix which becomes:

$$\mathbb{A} = \begin{bmatrix} a_{00} & 0 & \lambda_{\text{SD}} & 0 & 0 & \mu_{\text{DU,FTC}} \\ 0 & a_{11} & 0 & 0 & 0 & \mu_{\text{DU,LCP}} \\ 0 & 0 & a_{22} & 0 & 0 & \mu_{\text{DD}} \\ 0 & 0 & 0 & a_{33} & \lambda_{\text{SD}} & \mu_{\text{SU,DOP}} \\ 0 & 0 & 0 & 0 & a_{44} & \mu_{\text{SD}} \\ \lambda_{\text{DU,FTC}} & \lambda_{\text{DU,LCP}} & \lambda_{\text{DD}} & \lambda_{\text{SU,DOP}} & \lambda_{\text{SD}} & a_{55} \end{bmatrix}.$$

The last step in the derivation of the PFD is to solve the set of equations given in 2.4 and 2.5 which becomes:

$$\begin{aligned} \lambda_{\text{DU,FTC}}\Pi_5 &= (\mu_{\text{DU,FTC}} + \lambda_{\text{SD}})\Pi_0 \\ \lambda_{\text{DU,LCP}}\Pi_5 &= \mu_{\text{DU,LCP}}\Pi_1 \\ \lambda_{\text{DD}}\Pi_5 &= \mu_{\text{DD}}\Pi_2 - \lambda_{\text{SD}}\Pi_0 \\ \lambda_{\text{SU,DOP}}\Pi_5 &= (\mu_{\text{SU,DOP}} + \lambda_{\text{SD}})\Pi_3 \\ \lambda_{\text{SD}}\Pi_5 &= \mu_{\text{SD}}\Pi_4 - \lambda_{\text{SD}}\Pi_3 \\ \sum_{j=0}^{5}\Pi_j &= 1. \end{aligned}$$

The resulting procedure used to perform the implementation is given in Algorithm 1.

A lot of data has been collected to describe the parameters in the first step, e.g., Hauge et al. [2006b], OREDA [2002]. Testing is supposed to be conducted once a year, i.e., $\tau = 8670$ hours. The parameters k_1, k_2 are the percentage of all dangerous failures that are assumed to be FTC and LCP, respectively, whereas k_3 denote the percentage of all safe failures that are DOP.

Output: PFD values as a function of SFF

1.1 Assign values to the parameters $\tau, C_d, \lambda_{\mathrm{D}}, k_1, k_2, k_3, \mathsf{MTTR}_{\mathrm{SD}},$ and $\mathrm{MTTR}_{\mathrm{D}}$;
1.2 $\lambda_{\mathrm{DD}} \leftarrow \lambda_D C_d$ and $\lambda_{\mathrm{DU}} \leftarrow \lambda_{\mathrm{D}}(1 - C_d)$;
1.3 $\lambda_{\mathrm{DU,FTC}} \leftarrow \lambda_{\mathrm{DU}} k_1$ and $\lambda_{\mathrm{DU,LCP}} \leftarrow \lambda_{\mathrm{DU}} k_2$;
1.4 $\mu_{\mathrm{DD}} \leftarrow \frac{1}{\mathrm{MTTR_D}}$, $\mu_{\mathrm{SD}} \leftarrow [\mathsf{MTTR}_{\mathrm{SD}}]^{-1}$ and $\mu_{\mathrm{DU,LCP}} \leftarrow \frac{1}{\mathrm{MTTR_D}+\tau/2}$;
1.5 $\mathsf{SFF} \leftarrow [\mathsf{SFF}_1, \mathsf{SFF}_2, \ldots, \mathsf{SFF}_m]$;
1.6 **for** $i \leftarrow 1$ **to** *length(SFF)* **do**
1.7 $\quad \lambda_{\mathrm{S}}(i) \leftarrow \frac{\mathsf{SFF}_i \lambda_{\mathrm{D}} - \lambda_{\mathrm{DD}}}{1 - \mathsf{SFF}_i}$;
1.8 $\quad \lambda_{\mathrm{SU,DOP}} \leftarrow \lambda_{\mathrm{S}}(i) k_3$;
1.9 $\quad \lambda_{\mathrm{SD}} \leftarrow \lambda_{\mathrm{S}}(i)(1 - k_3)$;
1.10 $\quad P_{\mathrm{FTC}} \leftarrow \frac{\lambda_{\mathrm{SD}} \lambda_{\mathrm{DU,FTC}} \tau^2}{2}$;
1.11 $\quad P_{\mathrm{DOP}} \leftarrow \frac{\lambda_{\mathrm{SD}} \lambda_{\mathrm{SU,DOP}} \tau^2}{2}$;
1.12 $\quad \mu_{\mathrm{DU,FTC}} \leftarrow \frac{1}{\tau/3 P_{\mathrm{FTC}} + \tau/2(1 - P_{\mathrm{FTC}})}$;
1.13 $\quad \mu_{\mathrm{SU,DOP}} \leftarrow \frac{1}{\tau/3 P_{\mathrm{DOP}} + \tau/2(1 - P_{\mathrm{DOP}})}$;
1.14 $\quad \Pi_5{}^{-1} \leftarrow \frac{\lambda_{\mathrm{DU,FTC}}}{\mu_{\mathrm{DU,FTC}} + \lambda_{\mathrm{SD}}} + \frac{\lambda_{\mathrm{DU,LCP}}}{\mu_{\mathrm{DU,LCP}}} + \frac{\lambda_{\mathrm{DD}}}{\mu_{\mathrm{DD}}} + \frac{\lambda_{\mathrm{SD}}}{\mu_{\mathrm{DD}}} \frac{\lambda_{\mathrm{DU,FTC}}}{\mu_{\mathrm{DU,FTC}} + \lambda_{\mathrm{SD}}} + \frac{\lambda_{\mathrm{SU,DOP}}}{\mu_{\mathrm{SU,DOP}} + \lambda_{\mathrm{SD}}} + \frac{\lambda_{\mathrm{SD}}}{\mu_{\mathrm{SD}}} + \frac{\lambda_{\mathrm{SD}}}{\mu_{\mathrm{SD}}} \frac{\lambda_{\mathrm{SU,DOP}}}{\mu_{\mathrm{SU,DOP}} + \lambda_{\mathrm{SD}}} + 1$;
1.15 $\quad \Pi_4 \leftarrow \left(\frac{\lambda_{\mathrm{SD}}}{\mu_{\mathrm{SD}}} + \frac{\lambda_{\mathrm{SD}}}{\mu_{\mathrm{SD}}} \frac{\lambda_{\mathrm{SU,DOP}}}{\mu_{\mathrm{SU,DOP}} + \lambda_{\mathrm{SD}}} \right) \Pi_5$;
1.16 $\quad \Pi_3 \leftarrow \frac{\lambda_{\mathrm{SU,DOP}}}{\mu_{\mathrm{SU,DOP}} + \lambda_{\mathrm{SD}}} \Pi_5$;
1.17 $\quad \Pi_2 \leftarrow \left(\frac{\lambda_{\mathrm{DD}}}{\mu_{\mathrm{DD}}} + \frac{\lambda_{\mathrm{SD}}}{\mu_{\mathrm{DD}}} \frac{\lambda_{\mathrm{DU,FTC}}}{\mu_{\mathrm{DU,FTC}} + \lambda_{\mathrm{SD}}} \right) \Pi_5$;
1.18 $\quad \Pi_1 \leftarrow \frac{\lambda_{\mathrm{DU,LCP}}}{\mu_{\mathrm{DU,LCP}}} \Pi_5$;
1.19 $\quad \Pi_0 \leftarrow \frac{\lambda_{\mathrm{DU,FTC}}}{\mu_{\mathrm{DU,FTC}} + \lambda_{\mathrm{SD}}} \Pi_5$;
1.20 $\quad \mathsf{PFD}_i \leftarrow \Pi_0 + \Pi_1 + \Pi_2$;
1.21 **end**
1.22 $\mathsf{PFD} \leftarrow [\mathrm{PFD}_i]$;

Algorithm 1: Algorithm for estimating the PFD for a 1oo1 system

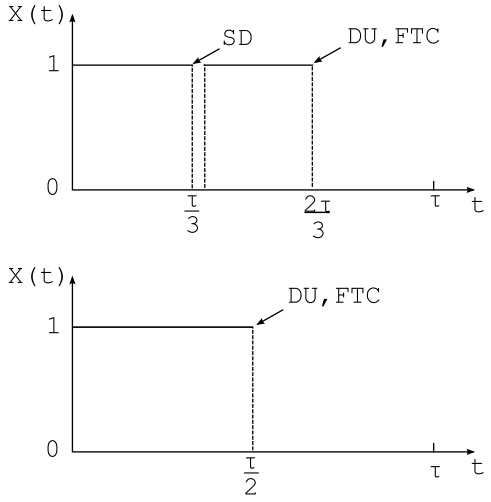

Figure 3.2: The mean behaviour of the state variable $X(t)$ of a system.

Assumption 1 says that the process considered fulfils the Markov property. Since the exponential distribution is the only continuous distribution that models a memoryless process it follows that T_{DD}, the time spent in state 2 before making a transition, is exponentially distributed with parameter μ_{DD}. Thus the expected time spent in state 2 equals $\frac{1}{\mu_{\mathrm{DD}}}$. DD failures are supposed to be repaired immediately after arising, so the expected time spent in state 2 equals the mean time to restore a DD failure, $\mathrm{MTTR}_{\mathrm{DD}}$, which is defined in step 1.1. This relationship is given in the first equation in step 1.4.

The restoration rate for undetected failures is not the same as for detected failures mainly because it is not known when they actually occurred. What is known is that they are revealed and repaired at time τ. In fact, there is a distinction between the restoration rates for the three possible undetected failures. The failure mode FTC and DOP can be revealed by a SD failure which becomes an issue when a SD failure occurs prior to one of these failures within the same test interval. Because of assumption 9 and 11, the system is known to be free from FTC and DOP failures after restoration of a SD failure and the SD failure then has the same properties as a function test.

Suppose that the probability of three or more failures occurring during a test interval is so small that these situations can be neglected. If a $\mathrm{DU}_{\mathrm{FTC}}$ or $\mathrm{SU}_{\mathrm{DOP}}$ failure is detected by the function test, there are two possible scenarios involving no more than two failures in a test interval of length τ. These possible situations can be represented by the function diagrams in Figure 3.2 where the expected state of the system is represented by the binary variable $X(t)$. $X(t) = 0$ means that a $\mathrm{DU}_{\mathrm{FTC}}$ failure is present at time t and $X(t) = 1$ means that no $\mathrm{DU}_{\mathrm{FTC}}$ failure is present at time t and it follows that the safety of the system is maintained. A $\mathrm{DU}_{\mathrm{FTC}}$ failure is used for illustrative purpose, but a $\mathrm{SU}_{\mathrm{DOP}}$ will follow the same arguments.

The upper diagram shows the mean behaviour of the state variable $X(t)$ when a SD failure occurs before a $\mathrm{DU}_{\mathrm{FTC}}$ within the same test interval. The time interval is separated in three periods of equal length, where the SD failure is, on average, supposed to occur at time $\tau/3$ and the $\mathrm{DU}_{\mathrm{FTC}}$ failure at time $2\tau/3$. The expected duration of a $\mathrm{DU}_{\mathrm{FTC}}$ failure before it is detected is $\tau/3$. This partition of the interval is based on the assumption of equal failure rates, i.e., $\lambda_{\mathrm{SD}} = \lambda_{\mathrm{DU,FTC}}$. Different failure rates will

probably change the occurrence time, but the effect is assumed to be so small that it can be neglected.

The lower diagram shows the possibility of a DU_{FTC} failure occurring before a SD failure within the same test interval. Since the possibility of more than two independent failures are assumed negligible, a SD failure cannot occur after a DU_{FTC} failure since it is known that the system is found in failed state at time τ. Suppose that the probability of occurrence of a DU_{FTC} failure is the same for all times within the test interval of length τ. It follows that $T_{DU,FTC}$ is uniformly distributed on the interval $(0, \tau]$ and it follows that the expected duration of a DU_{FTC} failure in this situation is $\tau/2$.

Let T_{DU} and T_{SD} denote the time until a DU_{FTC} failure and SD failure, respectively. The average duration of a DU_{FTC} failure, D, is calculated based on the law of total probability and becomes:

$$\begin{aligned} E[D|X(\tau)=0] &= E\left[D|X(\tau)=0, T_{SD}<T_{DU,FTC}<\tau\right]\Pr\left[T_{SD}<T_{DU,FTC}<\tau\right] \\ &\quad +E\left[D|X(\tau)=0, T_{DU,FTC}<T_{SD}<\tau\right]\left[1-\Pr\left(T_{SD}<T_{DU,FTC}<\tau\right)\right] \\ &= \frac{\tau}{3}\Pr\left(T_{SD}<T_{DU,FTC}<\tau\right)+\frac{\tau}{2}\left[1-\Pr\left(T_{SD}<T_{DU,FTC}<\tau\right)\right], \quad (3.1) \end{aligned}$$

where

$$\begin{aligned} \Pr\left(T_{SD}<T_{DU}<\tau\right) &= \int_{t=0}^{\tau}\Pr\left(T_{SD}<t\right)f_{DU}(t)dt \\ &= \int_{t=0}^{\tau}(1-e^{-\lambda_{SD}t})\lambda_{DU}e^{-\lambda_{DU}t}dt \\ &= 1-e^{-\lambda_{DU}\tau}-\frac{\lambda_{DU}}{\lambda_{SD}+\lambda_{DU}}\left(1-e^{-(\lambda_{SD}+\lambda_{DU})\tau}\right) \\ &\approx \frac{\lambda_{SD}\lambda_{DU}\tau^2}{2}. \end{aligned}$$

The restoration rate from DU_{FTC} is the reciprocal of Equation 3.1, resulting in the equation in Step 1.12. The restoration rate from SU_{DOP} is based on a similar deduction and the result is expressed in Step 1.13. According to assumption 10, DU_{LCP} failures are not detected by a spurious closure. Suppose that the occurrence time is uniformly distributed on the interval, then the expected time spent in state 1 is equal to $MTTR_D + \frac{\tau}{2}$. The resulting restoration rate from DU_{LCP} failures is given in Step 1.4.

It becomes visible that the restoration rate from DU_{FTC}, DU_{LCP}, and SU_{DOP} failures are not constant during the test period and do not satisfy the requirements for a Markov process. The exponential distribution is still an adequate approximation for the purpose of this thesis while it does not affect the limiting probabilities considerably.

SFF is a vector with values ranging from C_d[1], the coverage factor, to 1. λ_S in Step 1.7 is derived from the expression for SSF given in equation 2.3. PFD in Step 1.20 is calculated as the proportion of time spent in dangerous state, i.e., state 0, 1, or 2.

3.3 1oo2 system

The work in the previous section is developed further to treat two identical FSC's connected in series. This system is able to respond adequately upon demand as long as one

[1]SFF equals C_d corresponds to λ_S equals 0.

of the valves is functioning as illustrated by the reliability block diagram in Figure 2.3. The system is supposed to have active functional redundancy which means that both components get the same signal from the logic solver. Common cause failures are taken into consideration since both components may fail as a direct result of a shared cause.

It is necessary to define additional requirements that are specific for a 1oo2 system. These are:

1. The system is made up of two identical components and they are supposed to operate in a common environment.
2. Both components respond in the same manner to a CCF. *It follows that a CCF results in either two safe failures, two DD failures or two DU failures.*
3. Both components are function tested simultaneously.
4. The same β-factor is applied for DU, DD, and safe failures. *This may not be a realistic assumption, but there is little experience on the subject of application of specific β-factors.*
5. The probability of having an undetected failure in one component and a DD failure in the other at the end of the test interval, i.e., at times $n\tau$; $n = 1, 2, 3..$, is assumed negligible.
6. SD state is supposed to be instantaneous, i.e., $\mu_{\text{SD}} \to \infty$.
7. Two independent failures cannot occur simultaneously.
8. A DD failure in one component is repaired without affecting the other component which means that degraded operation is considered.

Possible states of a 1oo2 system is given in Table 3.3:

State 0, 1, and 2 consider the situation where both components fail due to a common cause while state 6, 9, and 11 are due to two individual failures. This separation is done because the restoration rates are different. Since multiple failures are restored simultaneously, the restoration rate from state 0, 1, and 2 are equal to their respective restoration rate computed in the previous section.

It becomes evident that the complexity of a Markov model grows rapidly by adding redundant components. The general view of the Markov diagram becomes difficult to grasp and for a 1oo2 system it is more easy to follow a table of all possible transactions. These are dislayed in Table 3.4.

Table 3.3: Possible states of a 1oo2 system

State	FSC_1	FSC_2
0	$CCSU_{DOP}$	$CCSU_{DOP}$
1	$CCDU_{FTC}$	$CCDU_{FTC}$
2	$CCDU_{LCP}$	$CCDU_{LCP}$
3	SU_{DOP}	DD
4	DU_{FTC}	DD
5	DU_{LCP}	DD
6	SU_{DOP}	SU_{DOP}
7	SU_{DOP}	DU_{FTC}
8	SU_{DOP}	DU_{LCP}
9	DU_{FTC}	DU_{FTC}
10	DU_{FTC}	DU_{LCP}
11	DU_{LCP}	DU_{LCP}
12	DD	DD
13	OK	DD
14	SU_{DOP}	OK
15	DU_{FTC}	OK
16	DU_{LCP}	OK
17	OK	OK

Table 3.4: Possible transitions a 1oo2 system

From	To	Transition rate	Condition
17	16	$2(1-\beta)\lambda_{\mathrm{DU,LCP}}$	A $\mathrm{DU_{LCP}}$ failure occurs in one component.
	15	$2(1-\beta)\lambda_{\mathrm{DU,FTC}}$	A $\mathrm{DU_{FTC}}$ failure occurs in one component.
	14	$2(1-\beta)\lambda_{\mathrm{SU,DOP}}$	A $\mathrm{SU_{DOP}}$ failure occurs in one component.
	13	$\beta\lambda_{\mathrm{DD}}$	A CCDD failure occurs in both components.
	2	$\beta\lambda_{\mathrm{DU,LCP}}$	A $\mathrm{CCDU_{LCP}}$ failure occurs in both components.
	1	$\beta\lambda_{\mathrm{DU,FTC}}$	A $\mathrm{CCDU_{FTC}}$ failure occurs in both components.
	0	$\beta\lambda_{\mathrm{SU,DOP}}$	A $\mathrm{CCSU_{DOP}}$ failure occurs in both components.
16	17	$\mu_{\mathrm{DU,LCP}}$	$\mathrm{DU_{LCP}}$ is restored.
	11	$(1-\beta)\lambda_{\mathrm{DU,LCP}}$	A $\mathrm{DU_{LCP}}$ failure occurs in the faultless component.
	10	$(1-\beta)\lambda_{\mathrm{DU,FTC}}$	A $\mathrm{DU_{FTC}}$ failure occurs in the faultless component.
	8	$(1-\beta)\lambda_{\mathrm{SU,DOP}}$	A $\mathrm{SU_{DOP}}$ failure occurs in the faultless component.
15	17	$\mu_{\mathrm{DU,FTC}}$	$\mathrm{DU_{FTC}}$ is restored.
	13	$(1-\beta)\lambda_{\mathrm{SD}}$	$\mathrm{DU_{FTC}}$ is detected by a SD failure.
	13	$\beta\lambda_{\mathrm{SD}}$	$\mathrm{DU_{FTC}}$ is detected and the faultless component fail SD.
	10	$(1-\beta)\lambda_{\mathrm{DU,LCP}}$	A $\mathrm{DU_{LCP}}$ failure occurs in the faultless component.
	9	$(1-\beta)\lambda_{\mathrm{DU,FTC}}$	A $\mathrm{DU_{FTC}}$ failure occurs in the faultless component.
	7	$(1-\beta)\lambda_{\mathrm{SU,DOP}}$	A $\mathrm{SU_{DOP}}$ failure occurs in the faultless component.
14	17	$\mu_{\mathrm{SU,DOP}}$	$\mathrm{SU_{DOP}}$ is restored.
	17	$(1-\beta)\lambda_{\mathrm{SD}}$	$\mathrm{SU_{DOP}}$ is detected by a SD failure. Because $\mu_{\mathrm{SD}} \to \infty$ the transition for this component is directly to initial state and not to SD state.
	17	$\beta\lambda_{\mathrm{SD}}$	$\mathrm{SU_{DOP}}$ is detected and the faultless component fail SD due to a common cause.
	8	$(1-\beta)\lambda_{\mathrm{DU,LCP}}$	A $\mathrm{DU_{LCP}}$ failure occurs in the faultless component.
	7	$(1-\beta)\lambda_{\mathrm{DU,FTC}}$	A $\mathrm{DU_{FTC}}$ failure occurs in the faultless component.
	6	$(1-\beta)\lambda_{\mathrm{SU,DOP}}$	A $\mathrm{SU_{DOP}}$ failure occurs in the faultless component.
13	17	μ_{DD}	DD failure is restored.
	17	$(1-\beta)\lambda_{\mathrm{SD}}$	A SD failure occurs in the faultless component. Because $\mu_{\mathrm{SD}} \to \infty$ the transition is directly to initial state and not to SD state.
	12	$(1-\beta)\lambda_{\mathrm{DD}}$	A DD failure occurs in the faultless component.
	5	$(1-\beta)\lambda_{\mathrm{DU,LCP}}$	A $\mathrm{DU_{LCP}}$ failure occurs in the faultless component.
	4	$(1-\beta)\lambda_{\mathrm{DU,FTC}}$	A $\mathrm{DU_{FTC}}$ failure occurs in the faultless component.
	3	$(1-\beta)\lambda_{\mathrm{SU,DOP}}$	A $\mathrm{SU_{DOP}}$ failure occurs in the faultless component.
12	17	μ_{DD}	DD failures are restored simultaneously.
11	17	$\mu_{\mathrm{DU,1oo2}}$	$\mathrm{DU_{LCP}}$ failures are restored simultaneously.
10	17	$\mu_{\mathrm{DU,1oo2}}$	$\mathrm{DU_{FTC}}$ and $\mathrm{DU_{LCP}}$ failure are restored simultaneously.
	5	$(1-\beta)\lambda_{\mathrm{SD}}$	$\mathrm{DU_{FTC}}$ failure are revealed by a SD failure.
9	17	$\mu_{\mathrm{DU,1oo2}}$	$\mathrm{DU_{FTC}}$ failures are restored simultaneously.
	12	$\beta\lambda_{\mathrm{SD}}$	$\mathrm{DU_{FTC}}$ failures are revealed by a common cause SD failure.
	4	$2(1-\beta)\lambda_{\mathrm{SD}}$	One of the $\mathrm{DU_{FTC}}$ failures are revealed by a SD failure.
8	17	$\mu_{\mathrm{DU,1oo2}}$	$\mathrm{SU_{DOP}}$ and $\mathrm{DU_{LCP}}$ failures are restored simultaneously.
	16	$(1-\beta)\lambda_{\mathrm{SD}}$	$\mathrm{SU_{DOP}}$ is revealed by a SD failure.
7	17	$\mu_{\mathrm{DU,1oo2}}$	$\mathrm{SU_{DOP}}$ and $\mathrm{DU_{FTC}}$ failures are restored simultaneously.
	15	$(1-\beta)\lambda_{\mathrm{SD}}$	$\mathrm{SU_{DOP}}$ failure is revealed by a SD failure.
	13	$\beta\lambda_{\mathrm{SD}}$	$\mathrm{SU_{DOP}}$ and $\mathrm{DU_{FTC}}$ failures are revealed by a common cause SD failure.
	3	$(1-\beta)\lambda_{\mathrm{SD}}$	$\mathrm{DU_{FTC}}$ failures is revealed by a SD failure.
6	17	$\mu_{\mathrm{DU,1oo2}}$	$\mathrm{SU_{DOP}}$ failures are restored simultaneously.
	17	$\beta\lambda_{\mathrm{SD}}$	$\mathrm{SU_{DOP}}$ failures are revealed by a common cause SD failure. Because $\mu_{\mathrm{SD}} \to \infty$ the transition is directly to initial state and not to SD state.
	14	$2(1-\beta)\lambda_{\mathrm{SD}}$	One of the $\mathrm{SU_{DOP}}$ failures are revealed by a SD failure. .
5	16	μ_{DD}	DD failure is restored.
	13	$\mu_{\mathrm{DU,LCP}}$	$\mathrm{DU_{LCP}}$ is restored.
4	15	μ_{DD}	DD failure is restored.
	13	$\mu_{\mathrm{DU,FTC}}$	$\mathrm{DU_{FTC}}$ is restored.
3	14	μ_{DD}	DD failure is restored.
	13	$\mu_{\mathrm{SU,DOP}}$	$\mathrm{SU_{DOP}}$ is restored.
2	17	μ_{DU}	$\mathrm{CCDU_{LCP}}$ failures are restored simultaneously.
1	17	$\mu_{\mathrm{DU,FTC}}$	$\mathrm{CCDU_{FTC}}$ failures are restored simultaneously.
0	17	$\mu_{\mathrm{SU,DOP}}$	$\mathrm{CCSU_{DOP}}$ failures are restored simultaneously.

CHAPTER 4

Results

The Markov models obtained in the previous chapter are written in the programming language R. R is commonly used for statistical computing and graphics at Industrial Mathematics and is well suited for the tasks related to this thesis. The results are presented in the following sections, where all models have the same basic plots. The PFD is plotted against SFF and for illustrative purposes the positive effect on PFD by increasing SFF one percent is plotted. The formula for calculating this percentage effect is:

$$\text{Effect} = \frac{\text{PFD}_{\text{SFF[i-1]}} - \text{PFD}_{\text{SFF[i]}}}{\text{PFD}_{\text{SFF[i-1]}}}. \tag{4.1}$$

The chapter starts with a sensitivity analysis for the different parameters used in reliability calculations as shown in Equations 2.7 and 2.8. The last section compares the results obtained for the two different system configurations, 1oo1 and 1oo2.

4.1 Variation of restoration time of safe failures for a 1oo1 system

The result of running Algorithm 1 for different values of MTTR_S is displayed in Figure 4.1. The figure displays a common tendency for the PFD to decrease as a function of SFF for all values of MTTR_S. The result can be explained by looking at the state transition diagram in Figure 3.1. An increased fraction of safe failures means that there is an increased fraction of transitions from initial state to the safe states 3 and 4. The exposure time for transitions to dangerous state decreases and the result is a lower value of PFD. The percentage effect on PFD by increasing SFF one percent, expressed by Equation 4.1, is plotted in Figure 4.2. The graph shows a marginally increasing effect from increasing MTTR_S but the effect is so small that it can be considered neglectable.

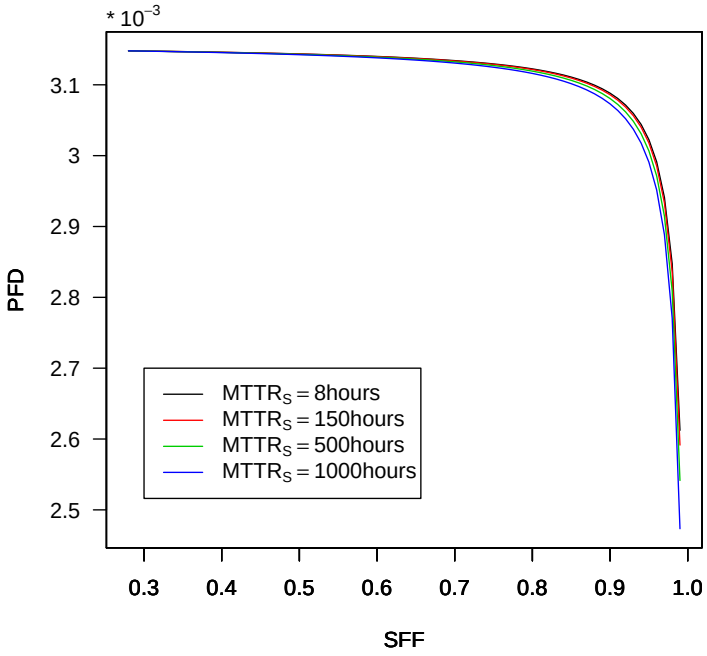

Figure 4.1: PFD for a 1oo1 system when considering the effect of restoration time.

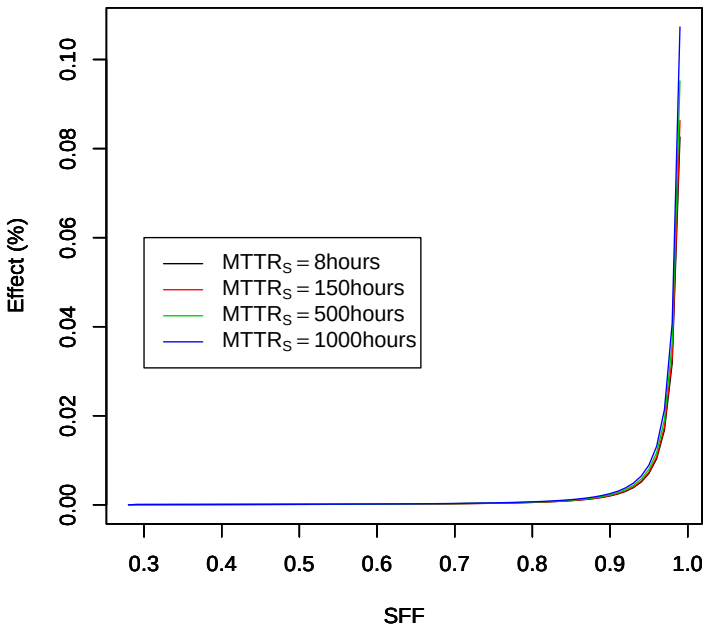

Figure 4.2: Percentage reduction of PFD by increasing SFF by one percent.

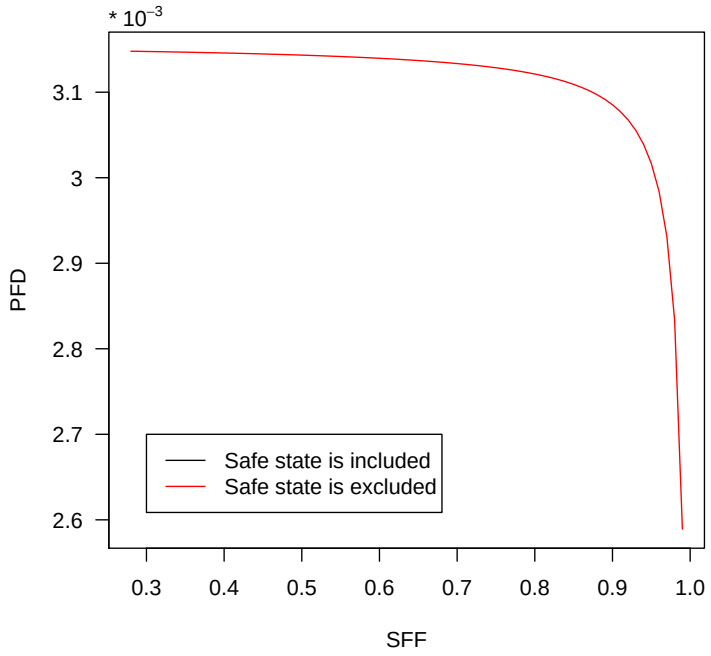

Figure 4.3: Comparison of the PFD for a 1oo1 system after introducing instantaneous restoration from safe state.

4.2 The effect of assuming instantaneous restoration from safe state for a 1oo1 system

Based on the results obtained in the previous section it is interesting to see what happens when it is assumed that safe state is instantaneous, i.e., when $\mu_{SD} \rightarrow \infty$. The restoration time from safe failure is set equal to the restoration time from dangerous failures, i.e. $\text{MTTR}_S = \text{MTTR}_D = 8\text{hours}$. The rationale for this choice is that most of the restoration time is not active repair of the failed component. Normal procedures includes isolation of the item, flushing with inert gas, demolition, re installation and so forth. These activities are not affected by the failure mode and it is reasonable to assume equal restoration time.

The results of running Algorithm 1 are shown in Figure 4.3 and 4.4. The graphs are almost identical, and it can be concluded that the time spent in safe state does not affect the PFD noticeable. It follows that the restoration rate from SU state is assumed instantaneous in the rest of the implementation.

4.3 Variation of dangerous failure rate for a 1oo1 system

The plot in Figure 4.5 shows the PFD for a 1oo1 system for different values of λ_D. The PFD reduces considerable when λ_D reduces. From Figure 4.6 it can be seen that the percentage effect of increasing SFF is small when λ_D is small. This is probably due to the decreasing frequency of a DU failure and a safe failure occurring in the same test interval. The results demonstrates that safe failures have significant effect only when

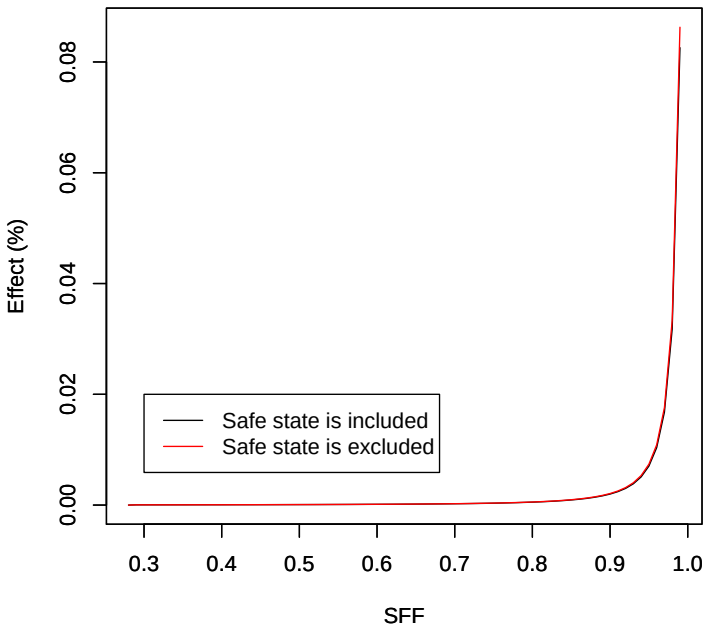

Figure 4.4: Percentage reduction of PFD by increasing SFF by one percent.

the dangerous failure rate is high.

According to Hauge et al. [2006b], OREDA [2002] the most reasonable value of the dangerous failure rate is 10^{-6}, so this is the value used in the rest of the implementation.

4.4 Variation of β-factor for a 1oo2

When an identical, redundant component is introduced, the safety system is run as a 1oo2 system. There is a possibility that both components fail due to a shared event, and β determines the rate at which this scenario occur. Thus, as β increases, there is an increased probability that an event resulting in a component failure will affect both components and not just one of them. If this failure is dangerous, the safety function is unavailable and will contribute to the PFD. Figure 4.7 shows that the PFD increases as β increases. From Figure 4.8 it is seen that the percentage effect of increasing SFF actually is reduced when β increases, the opposite of what happens to be the case for λ_D. The reason for this is that the combination of high β-factor and high SFF will probably result in an increased fraction of safe CCF. Thus the effect of revealing DU failures is of minor importance when the frequency of a DU failure and a safe failure occurring in the same test interval decreases.

The problem of finding a correct value of β for the safety system seems to be more important than to incorporate the effect of safe failures. According to Hauge et al. [2006b] the reasonable choice of β for this system is 0.02.

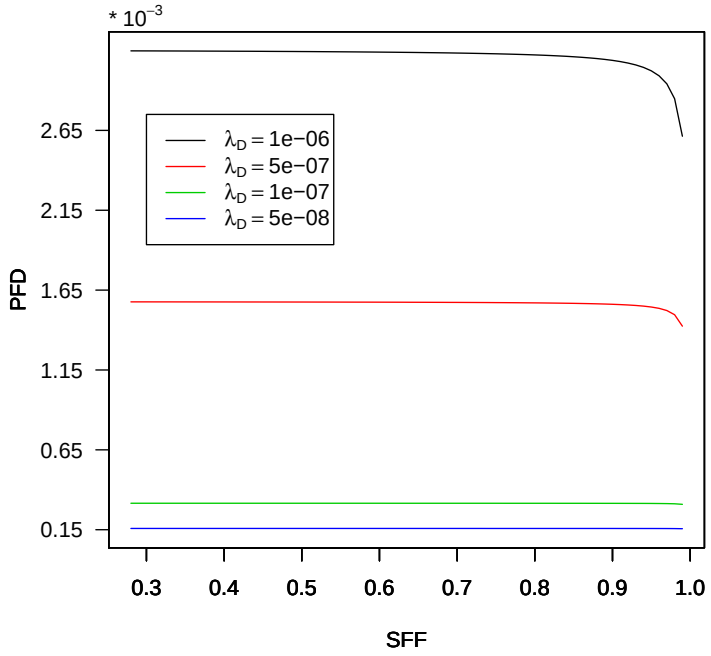

Figure 4.5: Comparison of the PFD for a 1oo1 system when assuming

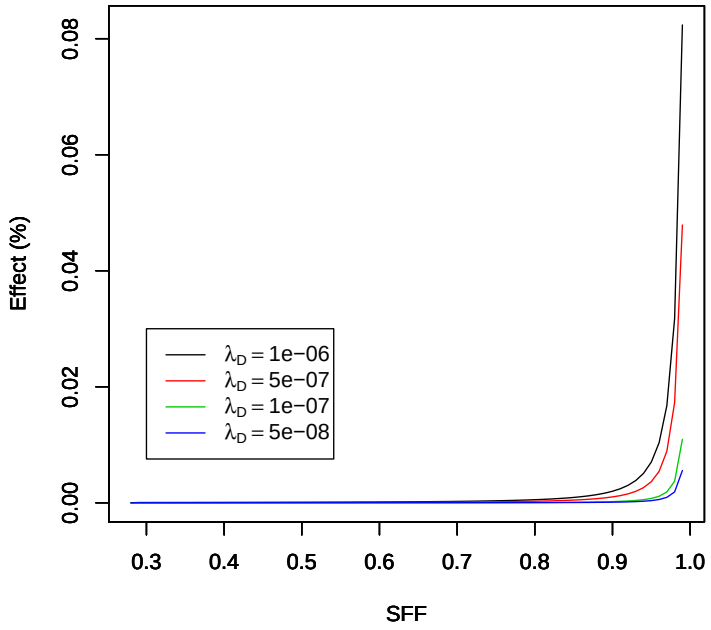

Figure 4.6: Percentage reduction of PFD by increasing SFF by one percent.

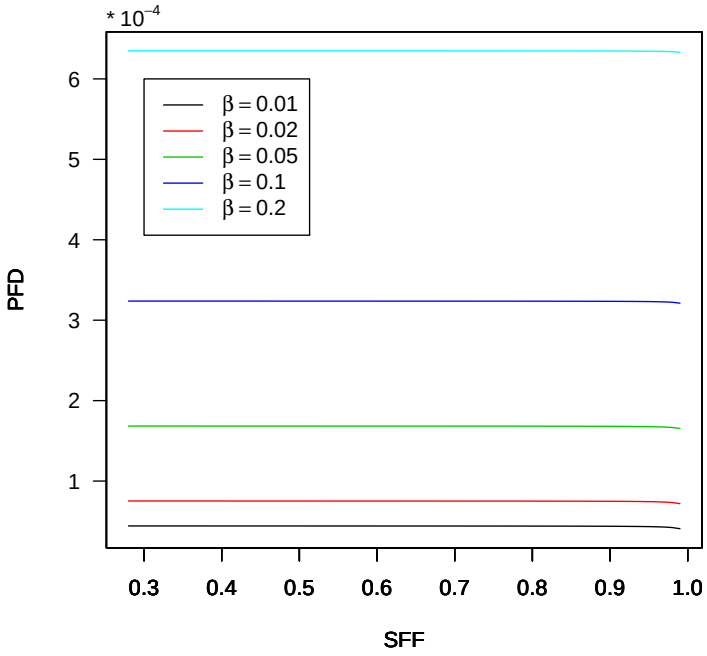

Figure 4.7: PFD for a 1oo2 system with various β-factors.

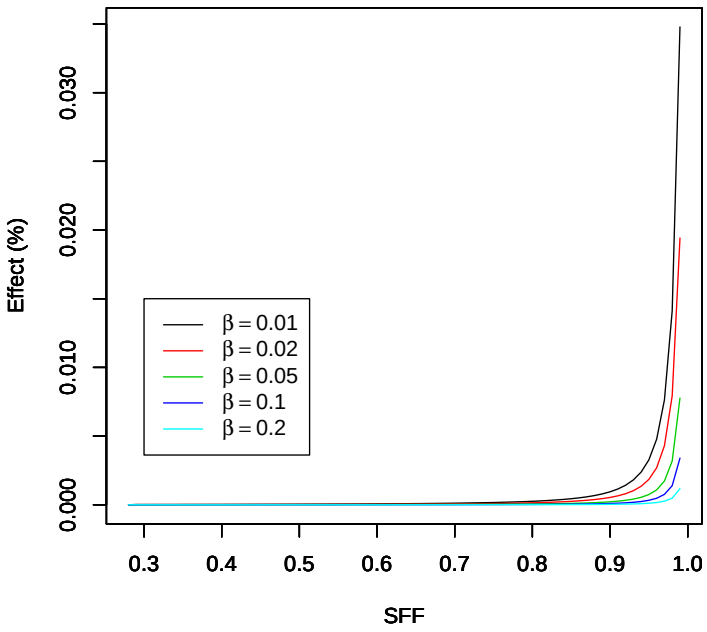

Figure 4.8: Comparison of the percentage reduction of PFD for various β-factors.

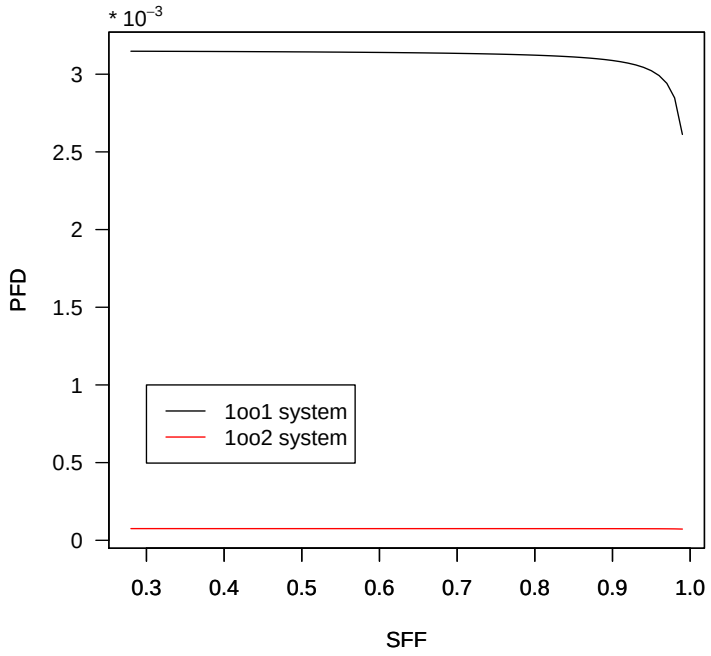

Figure 4.9: Comparison of the PFD for the two different system configurations 1oo1 and 1oo2.

4.5 Comparison of PFD for 1oo1 and 1oo2 system

The simulations in this section are run for the following parameter values: $\mu_S \rightarrow \infty$, $\text{MTTR}_S = 8$ hours, $\lambda_D = 10^{-6}$ and $\beta = 0.02$. The PFD for the two different system configurations, 1oo1 and 1oo2, is plotted in Figure 4.9. It shows that there is a large reduction in PFD when introducing a redundant component. The difference can be explained by course of HFT. The 1oo2 configuration has HFT equals 1 while the 1oo1 system has HFT equals 0. In the redundant system both components must fail dangerous before the system is unable to respond adequately on demand. On the other hand the rate of safe failures is twice as big for a 1oo2 system compared to a 1oo1 system. The percentage effect on PFD by increasing SFF is plotted in Figure 4.10 and shows that the effect is greater for the 1oo1 configuration, especially when SFF lies between 75 and 95% which is common values in practical analysis. Since the effect is smaller than 1% when SFF equals 95%, safe failures does not have a significant effect on PFD calculations.

4.6 Comparison of PFD calculated by the Markov model and normal probability calculations

Figure 4.11 shows the results obtained when applying the parameters in the previous section for a 1oo1 system together with the numerical results obtained by applying these parameters in Equation 2.7. Figure 4.12 is derived in a similar manner for a 1oo2 system where the numerical result is obtained by assuming identical components in Equation 2.8. The numerical values, $\text{PFD}_{1oo1} = 3.210^{-3}$ and $\text{PFD}_{1oo2} = 7.610^{-5}$, correspond to

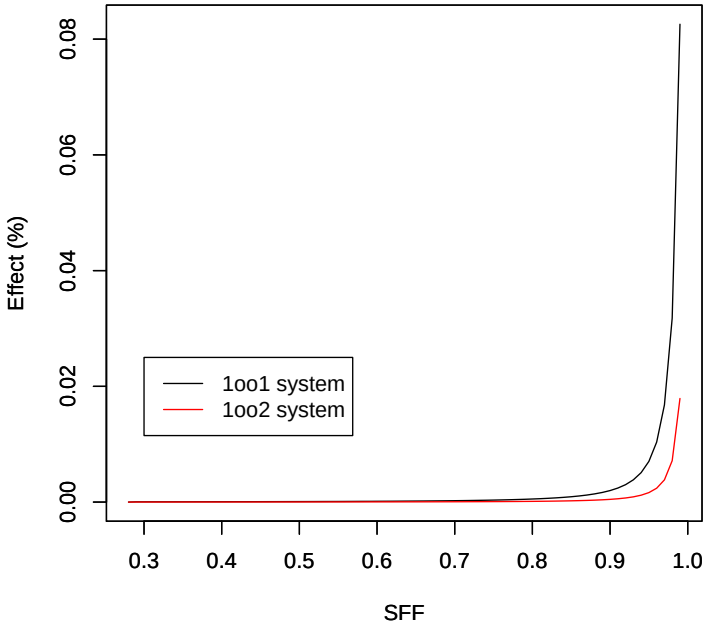

Figure 4.10: Comparison of the percentage reduction of PFD for two different system configurations, 1oo1 and 1oo2.

the points on their respective graphs where SFF = 0. This is because standard PFD calculations are done under the assumption of no influence from safe failures, or, in other words, when assuming SFF = 0.

The figures shows that the SFF must be close to 100% to have a significant effect. Since the fraction of dangerous failures is always kept at a minimum, the number of safe failures must be increased to obtain a higher SFF. This is probably not the intent of SFF and is negative with respect to production downtime.

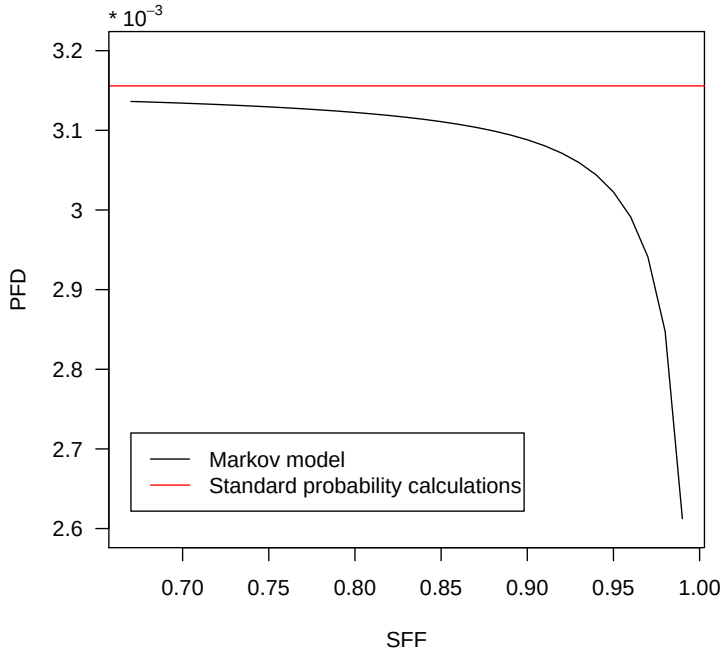

Figure 4.11: Comparison of the calculated PFD for a 1oo1 system by normal probability calculations and Markov model.

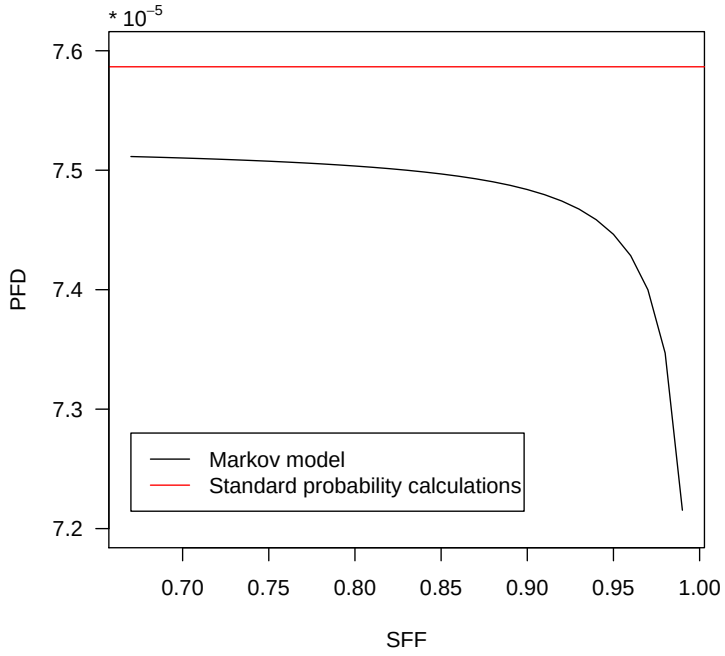

Figure 4.12: Comparison of the calculated PFD for a 1oo2 system by normal probability calculations and Markov model.

CHAPTER 5

Conclusions

Safety engineers have questioned the suitability of SFF as an indicator of safe design, where the negative effects of safe failures have achieved great importance. A literature survey has been carried out to clarify the relationship between safe failures and SIS reliability and to identify the positive effects of safe failures.

The application includes a detailed analysis of a safety shutdown valve and the possible effects have been incorporated into a Markov model for two different system configurations, 1oo1 and 1oo2.

From the results it can be concluded that the time needed to restore the system back to initial state after a safe failure does not have a significant effect on PFD.

The main contributor to PFD is the long run probability of being in DU state. DU failures are normally detected by function tests or sometimes upon demand, but they can also be revealed by a spurious closure. This effect is based on the assumption of perfect repair of safe failures, which means that all possible failure modes are detected and the failed items are repaired or replaced after restoration of safe failures. The ability to reveal DU failures is clearly dependent on the frequency of a DU failure and safe failure occurring in the same test interval. This thesis demonstrates that safe failures only have significant effect when the dangerous failure rate is high. Other parameters affect the PFD to a greater extent, and the importance of exact parameter estimation is crucial and more important than the positive effects of safe failures.

When redundant components are introduced, there is a possibility that both components fail due to a shared event. These CCFs will reduce the reliability of the safety system, and the PFD is highly dependent on the value of the β factor.

The positive effects of safe failures is not so considerable that it can justify a lower degree of redundancy. The SFF must be close to 100% to indicate a lower SIL level, and since the minimum number of dangerous failures always are arrived at, the alternative is to introduce more safe failures. This is probably not the intent of SFF and is negative with respect to production downtime. On the other hand, the positive effects of safe failures show a satisfactory reason for adopting a longer test interval. This is an optimisation of PFD and can reduce costs or even the frequency of dangerous situations during start-up and shutdown.

This thesis demonstrates that the PFD is not affected by safe failures, and indicates

no reason to be in doubt about this parameter as a measure of reliability. The SFF gives hardly any information and the choice of SIS architecture should not be based on SFF alone. An alternative parameter that considers different means of revealing DU failures seems to be a better choice.

Recommendations for further work

The models in this thesis could be developed further to also include systematic failures or to take negative effects into consideration as well. A particular safety valve is analysed and some effects may be valid only for this specific system. To get the entire picture of the effects of safe failures, another systems should be analysed. Alternative methods could be used to verify the results obtained in this thesis.

A simplification in the analysis is that the possibility of three or more failure modes of a component during a test interval is assumed negligible. This limitation in the model should be checked to find out whether or not several failures in the same test interval will affect the PFD.

This thesis concluded that accurate estimation of the β-factor was crucial to obtain reliable results. A development of the models is to apply specific β-factors for the different failure modes. More research should be developed to find correct β-values and to incorporate them into the model.

Bibliography

CCPS. *Guidelines for Safe and Reliable Instrumented Protective Systems*. John Wiley and Sons, 2007. Center for Chemical Process Safety.

K.N Fleming. A reliability model for common cause failures in redundant safety systems. *General Atomic Report, GA-13284*, 1974.

William M. Goble and Harry Cheddie. *Safety Instrumented Systems Verification: Practical Probabilistic Calculations*. ISA-Instrumentation, Systems, and Automation Society, 2005.

Stein Hauge, Per Hokstad, Helge Langseth, and Knut Øien. *Reliability Prediction Method for Safety Instrumented Systems, PDS Method handbook*. SINTEF, 2006a.

Stein Hauge, Helge Langseth, and Tor Onshus. *Reliability Data for Safety Instrumented System*. SINTEF, 2006b.

IEC 61508. *Functional safety of electrical/electronic/programmable electronic safety related systems*. International Electrotechnical Commision, 1998. Part 1: General requirements.

IEC 61511. *Functional safety - Safety instrumented systems for the process industry sector*. International Electrotechnical Commision, 2003. Part 1: Framework, definitions, system, hardware and software requirements.

Yves Langeron, Anne Barros, Antoine Grall, and Christophe Bérenguer. Combination of safety integrity levels (SILs): A study of IEC61508 merging rules. *Journal of Loss Prevention in the Process Industries*, 21(4):437–449, July 2008.

B. Littlewood and J. L. Verrall. A bayesian reliability growth model for computer software. *Applied Statistics*, 22(3):332–346, 1973. ISSN 00359254.

Mary Ann Lundteigen and Marvin Rausand. Architectural constraints in IEC 61508: Do they have the intended effect? *Reliability Engineering & System Safety*, 94(2): 520–525, February 2008a.

Mary Ann Lundteigen and Marvin Rausand. Spurious activation of safety instrumented systems in the oil and gas industry: Basic concepts and formulas. *Reliability Engineering & System Safety*, 93(8):1208–1217, August 2008b.

Einar Munkeby. Effect of safe failures on the reliability of safety instrumented systems. Master's thesis, 2008. Norwegian University of Science and Technology (NTNU).

OLF-070. *Application of IEC 61508 and IEC 61511 in the Norwegian Petroleum Industry*. The Norwegian Oil Industry Association, Stavanger, Norway, 2004.

OREDA. *Offshore Reliability Data Handbook 2002*. SINTEF Industrial Management, 4th edition, 2002.

Marvin Rausand and Arnljot Høyland. *System Reliability Theory: Models, Statistical Methods, and Applications*. Wiley-IEEE, 2004.

Sheldon M. Ross. *Introduction to Probability Models*. Academic Press, 8th edition edition, January 2003.

Iwan van Beurden and Rachel Amkreutz. What does Proven In Use imply? *Hydrocarbon Processing*, November 2004.

Itaru Yoshimura and Yoshinobu Sato. Safety Achieved by the Safe Failure Fraction (SFF) in IEC 61508. 2008. Accepted for inclusion in a future issue of IEEE Transactions on Reliability.

Appendices

APPENDIX A

Supplementary theory

A.1 Markov model

A Markov process is a continuous-time Markov chain which means that events, such as failure modes, can occur at any point in time in contrast to Markov chains where they take place only at discrete times. The basic assumption of a Markov process is that the behaviour of a system is memoryless. This means that future states of a system is characterised only by its present state and not the past. Suppose the random variable $Z(t)$ denotes the state of the system at time t and that the collection of all possible states, χ, is finite, i.e $\chi = \{0, 1, 2, ..., r\}$. Suppose that the state of the process at time t is i, the Markov property says that:

$$P\left[Z(t+s) = j \mid Z(t) = i, Z(u) = z(u); u < t\right] = P\left[Z(t+s) = j | Z(t) = i\right] \tag{A.1}$$

for all possible values $z(u); u < t$.
If, in addition,

$$P\left[Z(t+s) = j | Z(t) = i\right] = P\left[Z(s) = j | Z(0) = i\right], \tag{A.2}$$

the Markov process is said to have *stationary* or *homogeneous* transition probabilities. In words, this means that the probability of a transition from one state to another does not depend on the global time but only on the time interval available for the transition to take place.

Let T_i denote the amount of time that the Markov process stays in state i before making a transition into a different state. Consider a Markov process that enters state i at time 0 and suppose that we observe that the process is still in state i at time t, i.e $T_i > t$. What is the probability of finding the process in state i after another s time units, i.e what is $P\left[T_i > t + s | T_i > t\right]$? Since the process have the Markov property, we know that the probability that the process remains in state i for s more time units is determined only by the current state i. Thus

$$P\left[T_i > t + s | T_i > t\right] = P\left[T_i > s\right] \tag{A.3}$$

for all $s, t > 0$. Hence T_i is memoryless, and we conclude that the random variable T_i is exponentially distributed.[1]

The number of possible states in an industrial process is countable, thus finite, and the transition probability distribution can therefore be represented by a matrix. The (i,j) element of the transition probability matrix $\mathbb{P}$, equals

$$P_{ij} = P\left[Z(t) = j | Z(0) = i\right] \quad \forall i, j \in \chi. \tag{A.4}$$

When we use Markov models to represent the construction of a safety system, it is more common to find values for the transition *rates* instead of the transition *probabilities*. There is a clear connection between these two parameters, given by the following formula:

$$a_{ij} = \alpha_i P_{ij} \qquad \forall i \neq j. \tag{A.5}$$

In formula A.5, α_i denotes the rate at which the process leaves state i and P_{ij} is the probability that it makes a transition from state i to state j. It follows that a_{ij} is the transition rate from state i to state j and thus the (i,j) element of the transition rate matrix $\mathbb{A}$. Earlier in this section we defined T_i as the amount of time spent in state i before making a transition, and proved that it is exponentially distributed. The rate parameter is actually α_j and it follows that the mean time spent in state i, $E(T_i)$, is equal to $1/\alpha_i$.

Equation A.5 explains that specifying the transition rate matrix of a Markov process determines the transition probability matrix. The transition rates can be found by constructing a state transition diagram where circles are used to represent states and directed arcs are used to represent transitions between the states. The diagonal elements a_{ii} can be found by the following equation

$$a_{ii} = -\alpha_i = -\alpha_i \sum_{j \neq i} P_{ij} = -\sum_{j \neq i} a_{ij}, \tag{A.6}$$

where the first transition follows from the fact that the transition rate back to its own state is minus the transition rate out of that state. The second transition is true because $\sum_{j \neq i} P_{ij} = 1$ [2].

By using the Markov property and the law of total probability, we derive at:

$$\begin{aligned} P_{ij}(t+s) &= P[Z(t+s) = j | Z(0) = i] \\ &= \sum_{n=0}^{k} P[Z(t+s) = j | Z(t) = k, Z(0) = i] P[Z(t) = k | Z(0) = i] \\ &= \sum_{n=0}^{k} P[Z(t+s) = j | Z(t) = k] P[Z(t) = k | Z(0) = i] \\ &= \sum_{n=0}^{k} P_{kj}(s) P_{ik}(t) \end{aligned} \tag{A.7}$$

We have shown that

$$P_{ij}(t+s) = \sum_{n=0}^{k} P_{kj}(s) P_{ik}(t) \qquad \forall s, t \geq 0, \tag{A.8}$$

[1] The exponential distribution is the only continuous distribution that models a memoryless process.

[2] The summation runs over all values of j except i and is a consequence of treating a continuous time Markov chain where the probability of making a jump back to its present state, P_{ii}, equals 0

which is known as the Chapman Kolmogorov equation.

From equation A.8 we get that

$$\begin{aligned} P_{ij}(t+\Delta t) - P_{ij}(t) &= \sum_{k=0}^{r} P_{kj}(\Delta t)P_{ik}(t) - P_{ij}(t) \\ &= \sum_{k=0}^{r} P_{ik}(\Delta t)P_{kj}(t) - P_{ij}(t) \\ &= \sum_{k\neq i} P_{ik}(\Delta t)P_{kj}(t) + P_{ii}(\Delta t)P_{ij}(t) - P_{ij}(t) \\ &= \sum_{k\neq i} P_{ik}(\Delta t)P_{kj}(t) - [1 - P_{ii}(\Delta t)]P_{ij}(t) \end{aligned} \tag{A.9}$$

If we divide by Δt and takes the limit as $\Delta t \to 0$ we get:

$$\lim_{\Delta t\to 0} \frac{P_{ij}(t+\Delta t) - P_{ij}(t)}{\Delta t} = \lim_{\Delta t\to 0} \left\{ \sum_{k\neq i} \frac{P_{ik}(\Delta t)}{\Delta t} P_{kj}(t) - \frac{1 - P_{ii}(\Delta t)}{\Delta t} P_{ij}(t) \right\}. \tag{A.10}$$

Since the summation index is finite we are able to interchange the summation and limit, and equation A.10 can be simplified using the following equalities:

$$\lim_{\Delta t\to 0} \frac{P_{ij}(t+\Delta t) - P_{ij}(t)}{\Delta t} = P'_{ij}(t) \tag{A.11}$$

$$P_{ij}(\Delta t) = \Delta t \alpha_i P_{ij} \tag{A.12}$$

$$1 - P_{ii}(\Delta t) = \alpha_i \Delta t \tag{A.13}$$

The first equation follows from the definition of a derivative. $P_{ij}(\Delta t)$,the probability that the process goes from state i to state j in a time h, equals the rate at which the process makes a transition when in state i multiplied by the time interval, Δt, multiplied by the probability of making a transition from state i to state j. The last equation can be derived when recognizing that $1 - P_{ii}(\Delta t)$, the probability that a process in state i at time h, equals the rate at which the process makes a transition when in state i multiplied with Δt.

The resulting equation is called **Kolmogorov's forward equations**:

$$P'_{ij}(t) = \sum_{k\neq j} a_{ik}P_{kj}(t) - \alpha_i P_{ij}(t) = \sum_{k\neq i} a_{kj}P_{ik}(t) \tag{A.14}$$

In matrix term this equation may be written as

$$\mathbb{P}'(t) = \mathbb{P}(t)\mathbb{A}. \tag{A.15}$$

In many modeling situations it is interesting to know what happens with a process that has been running for a long time, or in the probabilistic sense, to know something about the limiting probabilities. Let Π_j be the limiting probability that the Markov process will be in state j, i.e

$$\Pi_j = lim_{t\to\infty} P_{ij}(t). \tag{A.16}$$

Since $P_{ij}(t)$ tends to a constant value when $t \to \infty$ it implies that the derivative must be equal to 0. As a consequence, the limiting probabilities can be drawn from the simplified forward equation given in equation A.14:

$$\begin{aligned} \lim_{\Delta t \to \infty} P'_{ij}(t) &= \lim_{\Delta t \to \infty} \sum_{k \neq j} a_{ik} P_{kj}(t) - \alpha_i P_{ij}(t) \\ 0 &= \sum_{k \neq j} a_{ik} \Pi_k - \alpha_j \Pi_j \\ \alpha_j \Pi_j &= \sum_{k \neq j} a_{ik} \Pi_k \end{aligned} \tag{A.17}$$

It is important to remark that the transition rate matrix, $\mathbb{A}$ is singular because the sum of each row equals 0. This means that we need one extra independent equation in the Π_j's in order to solve for the limiting probabilities. For this purpose we use the normalization equation,

$$\sum_{j=1}^{r} \pi_j = 1. \tag{A.18}$$

Equation A.17 has a nice interpretation as the left-hand side equals the rate at which the process leaves state j and the right-hand side equals the rate at which the process enters state j. As a result equation A.17 are sometimes referred to as the balance equation while, in the long run, the rate at which transitions *into* state j occur equals the rate at which transitions *out of* state j occur.

APPENDIX B

R files

B.1 Variation of restoration time of safe failures

```
#################################################################################
#
#---------------------------------- 1oo1 model  ----------------------------------
#---------- Variation og MTTR_S, safe state is included ----------
#
#################################################################################

#-------Parameters-------#
tau=24*365
Cd=0.28
lambda_D=1e-6
LCP=0.35
FTC=0.65
DOP=0.2

SFF=seq(C_d,0.99,0.01)
MTTR_D=8
MTTR_S=c(8,50,150,300,500,750,1000,2000,3000)

mu_DD=1/MTTR_D
mu_LCP=1/(tau/2+MTTR_D)
lambda_DD=lambda_D*Cd
lambda_DU=lambda_D*(1-Cd)
lamLCP=lambda_DU*LCP
lamFTC=lambda_DU*FTC

#----Probability matrices ----------#
P0=matrix(0,length(SFF),length(MTTR_S))
P1=matrix(0,length(SFF),length(MTTR_S))
PFD1oo1=matrix(0,length(SFF),length(MTTR_S))

for (i in 1:length(SFF)){
        lambda_Si=(SFF[i]*lambda_D-lambda_DD)/(1-SFF[i])
        lamDOP=lambda_Si*DOP
        lamS=lambda_Si*(1-DOP)

        P_FTC=lamS*lamFTC*tau^2/2
        P_DOP=lamS*lamDOP*tau^2/2
        mu_FTC=1/(tau/3*P_FTC+tau/2*(1-P_FTC))
        mu_DOP=1/(tau/3*P_DOP+tau/2*(1-P_DOP))

        for (j in 1:length(MTTR_S)) {
                mu_Sj=1/MTTR_S[j]

                P_5=1/(lamFTC/(mu_FTC+lamS)+lamLCP/mu_LCP+lambda_DD/mu_DD
                        +lamS*lamFTC/(mu_DD*(mu_FTC+lamS))+lamDOP/(mu_DOP+lamS)
                          +lamS/mu_Sj+lamS*lamDOP/(mu_Sj*(mu_DOP+lamS))+1)
                P_4=(lamS/mu_Sj+lamS*lamDOP/(mu_Sj*(mu_DOP+lamS)))*P_5
                P_3=(lamDOP/(mu_DOP+lamS))*P_5
                P_2=(lambda_DD/mu_DD+lamS*lamFTC/(mu_DD*(mu_FTC+lamS)))*P_5
                P_1=(lamLCP/mu_LCP)*P_5
                P_0=(lamFTC/(mu_FTC+lamS))*P_5

                tot=P_0+P_1+P_2+P_3+P_4+P_5
                if(round(tot,10)!=1)print(c("Sum of probabilities not equal to 1!",tot))
```

B.1 Variation of restoration time of safe failures

```
57                  P0[i,j]=P_0
                    P1[i,j]=P_1
59
                    PFD1oo1[i,j]=P_0+P_1+P_2
61          }
    }
63
    print(c("Portion of the PFD that comes from DU equals:",(sum(P0[,1])+sum(P1[,1]))/sum(PFD1oo1[,1])))
65
    #-------------------------------------------------------------------------------
67  #------ Plots the PFD for a 1oo1 system for different values of MTTR_S --------
    #-------------------------------------------------------------------------------
69  ymin=min(PFD1oo1[,1],PFD1oo1[,3],PFD1oo1[,5],PFD1oo1[,7])
    ymax=max(PFD1oo1[,1],PFD1oo1[,3],PFD1oo1[,5],PFD1oo1[,7])
71  x=seq(2.5e-3,3.1e-3,by=1e-4)
    plot(SFF, PFD1oo1[,1],type='l',col=1,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
73  par(new=T)
    plot(SFF, PFD1oo1[,3],type='l',col=2,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
75  par(new=T)
    plot(SFF, PFD1oo1[,5],type='l',col=3,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
77  par(new=T)
    plot(SFF, PFD1oo1[,7],type='l',col=4,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
79  axis(2, at=x,labels=x*10^3, las=2)
    mtext("*"~10^-3,side=3,line=0.2,adj=0)
81  legend(0.3,0.0027,c(expression(MTTR["S"]==8*hours),expression(MTTR["S"]==150*hours),
          expression(MTTR["S"]==500*hours),expression(MTTR["S"]==1000*hours)),lty=1,col=c(1,2,3,4))
83  dev.copy2eps(file="1oo1.eps")

85  #-------------------------------------------------------------------------------
    #------ Plots the percentage reduction of PFD by increasing SFF by one percent
87  #-------------------------------------------------------------------------------
    effect1=rep(0,length(SFF)-1)
89  effect2=rep(0,length(SFF)-1)
    effect3=rep(0,length(SFF)-1)
91  effect4=rep(0,length(SFF)-1)

93  for (i in 2:length(SFF)){
          effect1[i]=(PFD1oo1[i-1,1]-PFD1oo1[i,1])/PFD1oo1[i-1,1]
95        effect2[i]=(PFD1oo1[i-1,3]-PFD1oo1[i,3])/PFD1oo1[i-1,3]
          effect3[i]=(PFD1oo1[i-1,5]-PFD1oo1[i,5])/PFD1oo1[i-1,5]
97        effect4[i]=(PFD1oo1[i-1,7]-PFD1oo1[i,7])/PFD1oo1[i-1,7]
    }
99  ymin=min(effect1,effect2,effect3,effect4)
    ymax=max(effect1,effect2,effect3,effect4)
101
    plot(SFF, effect1,type='l',col=1,ylim=c(ymin,ymax),ylab="Effect (%)")
103 par(new=T)
    plot(SFF, effect2,type='l',col=2,ylim=c(ymin,ymax),ylab="Effect (%)")
105 par(new=T)
    plot(SFF, effect3,type='l',col=3,ylim=c(ymin,ymax),ylab="Effect (%)")
107 par(new=T)
    plot(SFF, effect4,type='l',col=4,ylim=c(ymin,ymax),ylab="Effect (%)")
109 legend(0.3,0.06,c(expression(MTTR["S"]==8*hours),expression(MTTR["S"]==150*hours),
           expression(MTTR["S"]==500*hours),expression(MTTR["S"]==1000*hours)),lty=1,col=c(1,2,3,4))
111 dev.copy2eps(file="1oo1effect.eps")
```

B.2 The effect of assuming instantaneous restoration from safe state

```
#############################################################################
#
#------------------------- 1oo1 HIPPS model ---------------------------
#---------------------- instantaneous restoration ----------------------
#
#############################################################################

"-------Parameters------"
tau=24*365
Cd=0.28
lambda_D=1e-6
LCP=0.35
FTC=0.65
DOP=0.2

SFF=seq(C_d,0.99,0.01)
MTTR_D=8

mu_DD=1/MTTR_D
mu_LCP=1/(tau/2+MTTR_D)
mu_S=1/MTTR_D
lambda_DD=lambda_D*Cd
lambda_DU=lambda_D*(1-Cd)
lamLCP=lambda_DU*LCP
lamFTC=lambda_DU*FTC

#----Probability matrices ----------#
P0=rep(0,length(SFF))
P1=rep(0,length(SFF))
PFD1oo1_b=rep(0,length(SFF))
OK=rep(0,length(SFF))

for (i in 1:length(SFF)){
        lambda_Si=(SFF[i]*lambda_D-lambda_DD)/(1-SFF[i])
        lamDOP=lambda_Si*DOP
        lamS=lambda_Si*(1-DOP)

        P_FTC=lamS*lamFTC*tau^2/2
        P_DOP=lamS*lamDOP*tau^2/2
        mu_FTC=1/(tau/3*P_FTC+tau/2*(1-P_FTC))
        mu_DOP=1/(tau/3*P_DOP+tau/2*(1-P_DOP))

        P_4=1/(lamFTC/(mu_FTC+lamS)+lamLCP/mu_LCP+lambda_DD/mu_DD
                +lamS*lamFTC/(mu_DD*(mu_FTC+lamS))+lamDOP/(mu_DOP+lamS)+1)
        P_3=(lamDOP/(mu_DOP+lamS))*P_4
        P_2=(lambda_DD/mu_DD+lamS*lamFTC/(mu_DD*(mu_FTC+lamS)))*P_4
        P_1=(lamLCP/mu_LCP)*P_4
        P_0=(lamFTC/(mu_FTC+lamS))*P_4

        tot=P_0+P_1+P_2+P_3+P_4
        if(round(tot,5)!=1)print("Sum of probabilities not equal to 1!")

        P0[i]=P_0
        P1[i]=P_1

        PFD1oo1_b[i]=P_0+P_1+P_2
        OK[i]=P_4
}
print(c("Portion of the PFD that comes from DU equals:",(sum(P0)+sum(P1))/sum(PFD1oo1_b)))

#-------------------------------------------------------------------------
#------ Compares the PFD when introducing instantaneous restoration ---------
#-------------------------------------------------------------------------
ymin=min(PFD1oo1[,1],PFD1oo1_b)
ymax=max(PFD1oo1[,1],PFD1oo1_b)
x=seq(2.5e-3,3.1e-3,by=1e-4)

plot(SFF, PFD1oo1[,1],type='l',col=1,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
par(new=T)
plot(SFF,PFD1oo1_b,type='l',col=2,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
axis(2, at=x,labels=x*10^3, las=2)
mtext("*"~10^-3,side=3,line=0.2,adj=0)
legend(0.3,0.0027,c("Effect of restoration time is considered",
        "Effect of restoration time is ignored"),lty=1,col=c(2,3))
dev.copy2eps(file="1oo1_sml.eps")

#-------------------------------------------------------------------------
#------ Plots the Percentage reduction of PFD by increasing SFF by one percent
#-------------------------------------------------------------------------
effect=rep(0,length(SFF)-1)
effect_inf=rep(0,length(SFF)-1)
for (i in 2:length(SFF)){
        effect[i]=(PFD1oo1[i-1,1]-PFD1oo1[i,1])/PFD1oo1[i-1,1]
        effect_inf[i]=(PFD1oo1_b[i-1]-PFD1oo1_b[i])/PFD1oo1_b[i-1]
}
ymin=min(effect,effect_inf)
ymax=max(effect,effect_inf)
```

```
88
plot(SFF, effect,type='l',col=1,ylim=c(ymin,ymax),ylab="Effect (%)")
90 par(new=T)
plot(SFF, effect_inf,type='l',col=2,ylim=c(ymin,ymax),ylab="Effect (%)")
92 legend(0.3,0.03,c("Effect of restoration time is considered",
        "Effect of restoration time is ignored"),lty=1,col=c(2,3))
94 dev.copy2eps(file="1oo1effect_sml.eps")
```

B.3 Variation of dangerous failure rate

```
##############################################################################
#
#---------------------------------- 1001 model  ---------------------------------------
#---------- Variation lambda_D, instantaneous restoration from safe state ------------------------
#
##############################################################################

#-------Parameters-------#
tau=24*365
Cd=0.28
LCP=0.35
FTC=0.65
DOP=0.2

SFF=seq(C_d,0.99,0.01)
MTTR_D=8
lambda_D=c(1e-6,5e-7,1e-7,5e-8)

mu_DD=1/MTTR_D
mu_LCP=1/(tau/2+MTTR_D)

#------Probability matrices----------#
P0=matrix(0,length(lambda_D),length(SFF))
P1=matrix(0,length(lambda_D),length(SFF))
PFD1oo1_lambda=matrix(0,length(lambda_D),length(SFF))

for (i in 1:length(lambda_D)){
        lambda_DD=lambda_D[i]*Cd
        lambda_DU=lambda_D[i]*(1-Cd)
        lamLCP=lambda_DU*LCP
        lamFTC=lambda_DU*FTC

        for (j in 1:length(SFF)){
                lambda_Sj=(SFF[j]*lambda_D[i]-lambda_DD)/(1-SFF[j])
                lamDOP=lambda_Sj*DOP
                lamDOP
                lamS=lambda_Sj*(1-DOP)

                P_FTC=lamS*lamFTC*tau^2/2
                P_DOP=lamS*lamDOP*tau^2/2
                mu_FTC=1/(tau/3*P_FTC+tau/2*(1-P_FTC))
                mu_DOP=1/(tau/3*P_DOP+tau/2*(1-P_DOP))

                P_4=1/(lamFTC/(mu_FTC+lamS)+lamLCP/mu_LCP+lambda_DD/mu_DD
                  +lamS*lamFTC/(mu_DD*(mu_FTC+lamS))+lamDOP/(mu_DOP+lamS)+1)
                P_3=(lamDOP/(mu_DOP+lamS))*P_4
                P_2=(lambda_DD/mu_DD+lamS*lamFTC/(mu_DD*(mu_FTC+lamS)))*P_4
                P_1=(lamLCP/mu_LCP)*P_4
                P_0=(lamFTC/(mu_FTC+lamS))*P_4

                tot=P_0+P_1+P_2+P_3+P_4
                if(round(tot,5)!=1)print("Sum of probabilities not equal to 1!")

                P0[i,j]=P_0
                P1[i,j]=P_1

                PFD1oo1_lambda[i,j]=P_0+P_1+P_2
        }
}
print(c("Portion of the PFD that comes from DU equals:",(sum(P0[1,])+sum(P1[1,]))/sum(PFD1oo1_lambda[1,])))

#-------------------------------------------------------------------------------
#------ Plots the PFD for a 1oo1 system for different values of lambda_D --------
#-------------------------------------------------------------------------------
ymin=min(PFD1oo1_lambda[1,],PFD1oo1_lambda[2,],PFD1oo1_lambda[3,],PFD1oo1_lambda[4,])
ymax=max(PFD1oo1_lambda[1,],PFD1oo1_lambda[2,],PFD1oo1_lambda[3,],PFD1oo1_lambda[4,])
x=seq(1.5e-4,3.1e-3,by=5e-4)
plot(SFF, PFD1oo1_lambda[1,],type='l',col=1,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
par(new=T)
plot(SFF, PFD1oo1_lambda[2,],type='l',col=2,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
par(new=T)
plot(SFF, PFD1oo1_lambda[3,],type='l',col=3,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
par(new=T)
plot(SFF, PFD1oo1_lambda[4,],type='l',col=4,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
axis(2, at=x,labels=x*10^3, las=2)
mtext("*"~10^-3,side=3,line=0.2,adj=0)
legend(0.3,0.00275,c(expression(lambda["D"]==1e-6),expression(lambda["D"]==5e-7),
        expression(lambda["D"]==1e-7),expression(lambda["D"]==5e-8)),lty=1,col=c(1,2,3,4))
dev.copy2eps(file="1oo1lambda.eps")

#-------------------------------------------------------------------------------
#------ Plots the percentage reduction of PFD by increasing SFF by one percent
#-------------------------------------------------------------------------------
effect1=rep(0,length(SFF)-1)
effect2=rep(0,length(SFF)-1)
effect3=rep(0,length(SFF)-1)
effect4=rep(0,length(SFF)-1)
effect3[68]

for (i in 2:length(SFF)){
```

B.3 Variation of dangerous failure rate

```
        effect1[i]=(PFD1oo1_lambda[1,i-1]-PFD1oo1_lambda[1,i])/PFD1oo1_lambda[1,i-1]
92      effect2[i]=(PFD1oo1_lambda[2,i-1]-PFD1oo1_lambda[2,i])/PFD1oo1_lambda[2,i-1]
        effect3[i]=(PFD1oo1_lambda[3,i-1]-PFD1oo1_lambda[3,i])/PFD1oo1_lambda[3,i-1]
94      effect4[i]=(PFD1oo1_lambda[4,i-1]-PFD1oo1_lambda[4,i])/PFD1oo1_lambda[4,i-1]
}
96 ymin=min(effect1,effect2,effect3,effect4)
ymax=max(effect1,effect2,effect3,effect4)
98
plot(SFF, effect1,type='l',col=1,ylim=c(ymin,ymax),ylab="Effect␣(%)")
100 par(new=T)
plot(SFF, effect2,type='l',col=2,ylim=c(ymin,ymax),ylab="Effect␣(%)")
102 par(new=T)
plot(SFF, effect3,type='l',col=3,ylim=c(ymin,ymax),ylab="Effect␣(%)")
104 par(new=T)
plot(SFF, effect4,type='l',col=4,ylim=c(ymin,ymax),ylab="Effect␣(%)")
106 legend(0.3,0.03,c(expression(lambda["D"]==1e-6),expression(lambda["D"]==5e-7),
        expression(lambda["D"]==1e-7),expression(lambda["D"]==5e-8)),lty=1,col=c(1,2,3,4))
108 dev.copy2eps(file="1oo1effect_lambda.eps")
```

B.4 Variation of beta factor

```
#############################################################################
#
#------------------------- 1oo2 HIPPS model ---------------------------------
#---------- Variation of beta, instantaneous restoration from safe state ------
#
#############################################################################

#-------Parameters-------#
tau=24*365
Cd=0.28
lambda_D=1e-6
LCP=0.35
FTC=0.65
DOP=0.2

SFF=seq(C_d,0.99,0.01)
MTTR_D=8
MTTR_S=8
beta=c(0.01,0.02,0.05,0.1,0.2)

mu_DD=1/MTTR_D
mu_LCP=1/(tau/2)
mu_S=1/MTTR_S
mu_1oo2=1/(tau/3+MTTR_D)
lambda_DD=lambda_D*Cd
lambda_DU=lambda_D*(1-Cd)
lamLCP=lambda_DU*LCP
lamFTC=lambda_DU*FTC


#----Probability matrices--------#
PFD1oo2_beta=matrix(0,length(SFF),length(beta))
OK=matrix(0,length(SFF),length(beta))

for (i in 1:length(SFF)){
        lambda_Si=(SFF[i]*lambda_D-lambda_DD)/(1-SFF[i])
        lamDOP=lambda_Si*DOP
        lamS=lambda_Si*(1-DOP)

        P_FTC=lamS*lamFTC*tau^2/2
        P_DOP=lamS*lamDOP*tau^2/2
        mu_FTC=1/(tau/3*P_FTC+tau/2*(1-P_FTC))
        mu_DOP=1/(tau/3*P_DOP+tau/2*(1-P_DOP))

        for (j in 1:length(beta)){
                r0=c(-mu_DOP,0,0,0,0,0,0,0,0,0,0,0,0,0,0,mu_DOP)
                r1=c(0,-mu_FTC,0,0,0,0,0,0,0,0,0,0,0,0,0,mu_FTC)
                r2=c(0,0,-mu_LCP,0,0,0,0,0,0,0,0,0,0,0,0,mu_LCP)
                r3=c(0,0,0,-(mu_DOP+mu_DD),0,0,0,0,0,0,0,0,0,mu_DOP,mu_DD,0,0,0)
                r4=c(0,0,0,0,-(mu_FTC+mu_DD),0,0,0,0,0,0,0,0,mu_FTC,0,mu_DD,0,0)
                r5=c(0,0,0,0,0,-(mu_LCP+mu_DD),0,0,0,0,0,0,0,mu_LCP,0,0,mu_DD,0)
                r6=c(0,0,0,0,0,0,-(mu_1oo2+beta[j]*lamS+2*(1-beta[j])*lamS),0,0,
                  0,0,0,0,0,2*(1-beta[j])*lamS,0,0,mu_1oo2+beta[j]*lamS)
                r7=c(0,0,0,(1-beta[j])*lamS,0,0,0,-(mu_1oo2+beta[j]*lamS+2*(1-beta[j])*lamS),
                      0,0,0,0,0,beta[j]*lamS,0,(1-beta[j])*lamS,0,mu_1oo2)
                r8=c(0,0,0,0,0,0,0,0,0,-(mu_1oo2+(1-beta[j])*lamS),0,0,0,0,0,0,0,
                       (1-beta[j])*lamS,mu_1oo2)
                r9=c(0,0,0,0,2*(1-beta[j])*lamS,0,0,0,0,-(mu_1oo2+beta[j]*lamS+2*(1-beta[j])*lamS),
                    0,0,beta[j]*lamS,0,0,0,0,mu_1oo2)
                r10=c(0,0,0,0,0,(1-beta[j])*lamS,0,0,0,0,-(mu_1oo2+(1-beta[j])*lamS),0,0,0,0,0,0,mu_1oo2)
                r11=c(0,0,0,0,0,0,0,0,0,0,0,-mu_1oo2,0,0,0,0,0,mu_1oo2)
                r12=c(0,0,0,0,0,0,0,0,0,0,0,0,0,-mu_DD,0,0,0,0,mu_DD)

                r13=c(0,0,0,(1-beta[j])*lambda_DU*DOP,(1-beta[j])*lambda_DU*FTC,(1-beta[j])*lambda_DU*LCP,
                        0,0,0,0,0,0,0,(1-beta[j])*lambda_DD,0,0,0,0,mu_DD+(1-beta[j])*lamS)
                r14=c(0,0,0,0,0,0,0,(1-beta[j])*lambda_DU*DOP,(1-beta[j])*lambda_DU*FTC,
                        (1-beta[j])*lambda_DU*LCP,0,0,0,0,0,0,0,0,mu_DOP+lamS)
                r15=c(0,0,0,0,0,0,0,0,0,(1-beta[j])*lambda_DU*DOP,0,(1-beta[j])*lambda_DU*FTC,
                        (1-beta[j])*lambda_DU*LCP,0,0,lamS,0,0,0,mu_FTC)
                r16=c(0,0,0,0,0,0,0,0,0,0,(1-beta[j])*lambda_DU*DOP,0,(1-beta[j])*lambda_DU*FTC,
                        (1-beta[j])*lambda_DU*LCP,0,0,0,0,0,mu_LCP)
                r17=c(beta[j]*lambda_DU*DOP,beta[j]*lambda_DU*FTC,beta[j]*lambda_DU*LCP,
                        0,0,0,0,0,0,0,0,0,0,beta[j]*lambda_DD,2*(1-beta[j])*lambda_DU*DOP,
                        2*(1-beta[j])*lambda_DU*FTC,2*(1-beta[j])*lambda_DU*LCP,0)

                r13[14]=-sum(r13)
                r14[15]=-sum(r14)
                r15[16]=-sum(r15)
                r16[17]=-sum(r16)
                r17[18]=-sum(r17)

                A=matrix(c(r0,r1,r2,r3,r4,r5,r6,r7,r8,r9,r10,r11,r12,r13,r14,r15,r16,r17),
                          byrow=TRUE,ncol=18,nrow=18)
                A[,dim(A)[2]]=rep(1,dim(A)[2])
                dP=rep(0,dim(A)[2])
                dP[dim(A)[2]]=1

                Pi=dP%*%solve(A)
                length(Pi)
```

B.4 Variation of beta factor

```
                tot=sum(Pi)
                if(round(tot,5)!=1)print("Sum of probabilities not equal to 1!")

                PFD1oo2_beta[i,j]=sum(c(Pi[2:3],Pi[5:6],Pi[10:13]))
        }
}
#-------------------------------------------------------------------------------
#------ Plots the PFD for a 1oo2 system for different values of beta ------------
#-------------------------------------------------------------------------------

ymin=min(PFD1oo2_beta[,1],PFD1oo2_beta[,2],PFD1oo2_beta[,3],PFD1oo2_beta[,4],PFD1oo2_beta[,5])
ymax=max(PFD1oo2_beta[,1],PFD1oo2_beta[,2],PFD1oo2_beta[,3],PFD1oo2_beta[,4],PFD1oo2_beta[,5])

x=seq(0,6.0e-4,by=1.0e-4)
plot(SFF, PFD1oo2_beta[,1],type='l',col=1,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
par(new=T)
plot(SFF, PFD1oo2_beta[,2],type='l',col=2,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
par(new=T)
plot(SFF, PFD1oo2_beta[,3],type='l',col=3,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
par(new=T)
plot(SFF, PFD1oo2_beta[,4],type='l',col=4,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
par(new=T)
plot(SFF, PFD1oo2_beta[,5],type='l',col=5,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
axis(2, at=x,labels=x*10^4, las=2)
mtext("*"~10^-4,side=3,line=0.2,adj=0)

legend(0.3,0.0006,c(expression(beta==0.01),expression(beta==0.02),expression(beta==0.05),
        expression(beta==0.1),expression(beta==0.2)),lty=1,col=seq(1,length(beta)))
dev.copy2eps(file="1oo2_beta.eps")

#-------------------------------------------------------------------------------
#------ Plots the percentage reduction of PFD for different values of beta
#-------------------------------------------------------------------------------
effect1=rep(0,length(SFF)-1)
effect2=rep(0,length(SFF)-1)
effect3=rep(0,length(SFF)-1)
effect4=rep(0,length(SFF)-1)
effect5=rep(0,length(SFF)-1)

for (i in 2:length(SFF)){
        effect1[i]=(PFD1oo2_beta[i-1,1]-PFD1oo2_beta[i,1])/PFD1oo2_beta[i-1,1]
        effect2[i]=(PFD1oo2_beta[i-1,2]-PFD1oo2_beta[i,2])/PFD1oo2_beta[i-1,2]
        effect3[i]=(PFD1oo2_beta[i-1,3]-PFD1oo2_beta[i,3])/PFD1oo2_beta[i-1,3]
        effect4[i]=(PFD1oo2_beta[i-1,4]-PFD1oo2_beta[i,4])/PFD1oo2_beta[i-1,4]
        effect5[i]=(PFD1oo2_beta[i-1,5]-PFD1oo2_beta[i,5])/PFD1oo2_beta[i-1,5]
}
ymin=min(effect1,effect2,effect3,effect4,effect5)
ymax=max(effect1,effect2,effect3,effect4,effect5)

plot(SFF, effect1,type='l',col=1,ylim=c(ymin,ymax),ylab="Effect (%)")
par(new=T)
plot(SFF, effect2,type='l',col=2,ylim=c(ymin,ymax),ylab="Effect (%)")
par(new=T)
plot(SFF, effect3,type='l',col=3,ylim=c(ymin,ymax),ylab="Effect (%)")
par(new=T)
plot(SFF, effect4,type='l',col=4,ylim=c(ymin,ymax),ylab="Effect (%)")
par(new=T)
plot(SFF, effect5,type='l',col=5,ylim=c(ymin,ymax),ylab="Effect (%)")
legend(0.3,0.015,c(expression(beta==0.01),expression(beta==0.02),expression(beta==0.05),
        expression(beta==0.1),expression(beta==0.2)),lty=1,col=seq(1,length(beta)))
dev.copy2eps(file="1oo2effect_beta.eps")
```

B.5 Comparison of PFD for 1oo1 and 1oo2 system

```
##############################################################################
#
#------------------------ 1002 model ------------------------------------------
#------------------------ beta = 0.02 -----------------------------------------
#
##############################################################################

#-------Parameters-------#
tau=24*365
Cd=0.28
lambda_D=1e-6
LCP=0.35
FTC=0.65
DOP=0.3

SFF=seq(C_d,0.99,0.01)
MTTR_D=8
MTTR_S=8
beta=0.02

mu_DD=1/MTTR_D
mu_LCP=1/(tau/2)
mu_S=1/MTTR_S
mu_1oo2=1/(tau/3+MTTR_D)
lambda_DD=lambda_D*Cd
lambda_DU=lambda_D*(1-Cd)
lamLCP=lambda_DU*LCP
lamFTC=lambda_DU*FTC

#----Probability matrices----------#
PFD1oo2=rep(0,length(SFF))
OK=rep(0,length(SFF))

for (i in 1:length(SFF)){
        lambda_Si=(SFF[i]*lambda_D-lambda_DD)/(1-SFF[i])
        lamDOP=lambda_Si*DOP
        lamS=lambda_Si*(1-DOP)

        P_FTC=lamS*lamFTC*tau^2/2
        P_DOP=lamS*lamDOP*tau^2/2
        mu_FTC=1/(tau/4*P_FTC+tau/2*(1-P_FTC))
        mu_DOP=1/(tau/4*P_DOP+tau/2*(1-P_DOP))

        r0=c(-mu_DOP,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,mu_DOP)
        r1=c(0,-mu_FTC,0,0,0,0,0,0,0,0,0,0,0,0,0,0,mu_FTC)
        r2=c(0,0,-mu_LCP,0,0,0,0,0,0,0,0,0,0,0,0,0,mu_LCP)
        r3=c(0,0,0,-(mu_DOP+mu_DD),0,0,0,0,0,0,0,0,0,mu_DOP,mu_DD,0,0,0)
        r4=c(0,0,0,0,-(mu_FTC+mu_DD),0,0,0,0,0,0,0,0,mu_FTC,0,mu_DD,0,0)
        r5=c(0,0,0,0,0,-(mu_LCP+mu_DD),0,0,0,0,0,0,0,mu_LCP,0,0,mu_DD,0)
        r6=c(0,0,0,0,0,0,-(mu_1oo2+beta*lamS+2*(1-beta)*lamS),0,0,0,0,0,
            0,0,2*(1-beta)*lamS,0,0,mu_1oo2+beta*lamS)
        r7=c(0,0,0,(1-beta)*lamS,0,0,0,-(mu_1oo2+beta*lamS+2*(1-beta)*lamS),
              0,0,0,0,0,beta*lamS,0,(1-beta)*lamS,0,mu_1oo2)
        r8=c(0,0,0,0,0,0,0,0,-(mu_1oo2+(1-beta)*lamS),0,0,0,0,0,0,0,(1-beta)*lamS,mu_1oo2)
        r9=c(0,0,0,0,2*(1-beta)*lamS,0,0,0,0,-(mu_1oo2+beta*lamS+2*(1-beta)*lamS),0,0,
             beta*lamS,0,0,0,0,mu_1oo2)
        r10=c(0,0,0,0,0,(1-beta)*lamS,0,0,0,0,-(mu_1oo2+(1-beta)*lamS),0,0,0,0,0,0,mu_1oo2)
        r11=c(0,0,0,0,0,0,0,0,0,0,0,0,-mu_1oo2,0,0,0,0,0,mu_1oo2)
        r12=c(0,0,0,0,0,0,0,0,0,0,0,0,0,-mu_DD,0,0,0,0,mu_DD)

        r13=c(0,0,0,(1-beta)*lambda_DU*DOP,(1-beta)*lambda_DU*FTC,(1-beta)*lambda_DU*LCP,
                0,0,0,0,0,0,(1-beta)*lambda_DD,0,0,0,0,mu_DD+(1-beta)*lamS)
        r14=c(0,0,0,0,0,0,(1-beta)*lambda_DU*DOP,(1-beta)*lambda_DU*FTC,
                (1-beta)*lambda_DU*LCP,0,0,0,0,0,0,0,0,mu_DOP+lamS)
        r15=c(0,0,0,0,0,0,0,0,(1-beta)*lambda_DU*DOP,0,(1-beta)*lambda_DU*FTC,
                (1-beta)*lambda_DU*LCP,0,0,lamS,0,0,0,mu_FTC)
        r16=c(0,0,0,0,0,0,0,0,0,(1-beta)*lambda_DU*DOP,0,(1-beta)*lambda_DU*FTC,
                (1-beta)*lambda_DU*LCP,0,0,0,0,0,mu_LCP)
        r17=c(beta*lambda_DU*DOP,beta*lambda_DU*FTC,beta*lambda_DU*LCP,0,0,0,0,0,0,0,0,0,0,
                beta*lambda_DD,2*(1-beta)*lambda_DU*DOP,2*(1-beta)*lambda_DU*FTC,2*(1-beta)*lambda_DU*LCP,0)

        r13[14]=-sum(r13)
        r14[15]=-sum(r14)
        r15[16]=-sum(r15)
        r16[17]=-sum(r16)
        r17[18]=-sum(r17)

        A=matrix(c(r0,r1,r2,r3,r4,r5,r6,r7,r8,r9,r10,r11,r12,r13,r14,r15,r16,r17),byrow=TRUE,ncol=18,nrow=18)

        A[,dim(A)[2]]=rep(1,dim(A)[2])
        dP=rep(0,dim(A)[2])
        dP[dim(A)[2]]=1

        Pi=dP%*%solve(A)

        tot=sum(Pi)
        if(round(tot,5)!=1)print("Her er det en feil i beregningen av stasjonærsannsynlighetene!")

        PFD1oo2[i]=sum(c(Pi[2:3],Pi[5:6],Pi[10:13]))
}
```

B.5 Comparison of PFD for 1oo1 and 1oo2 system

```
#-------------------------------------------------------------------------------
#------ Compares the PFD for a 1oo1 and a 1oo2 system -------------------------
#-------------------------------------------------------------------------------
ymin=min(PFD1oo1[,1],PFD1oo2)
ymax=max(PFD1oo1[,1],PFD1oo2)

x=seq(0,3.1e-3,by=5e-4)

plot(SFF, PFD1oo1[,1],type='l',col=1,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
par(new=T)
plot(SFF,PFD1oo2,type='l',col=2,ylim=c(ymin,ymax),ylab="PFD",yaxt="n")
axis(2, at=x,labels=x*10^3, las=2)
mtext("*"~10^-3,side=3,line=0.2,adj=0)
legend(0.3,0.001,c("1oo1 system","1oo2 system"),lty=1,col=c(1,2))
dev.copy2eps(file="1oo2_sml.eps")

#-------------------------------------------------------------------------------
#------ Compares the percentage reduction of PFD  for a 1oo1 and 1oo2 system
#-------------------------------------------------------------------------------
effect1oo1=rep(0,length(SFF)-1)
effect1oo2=rep(0,length(SFF)-1)

for (i in 2:length(SFF)){
        effect1oo1[i]=(PFD1oo1[i-1,1]-PFD1oo1[i,1])/PFD1oo1[i-1,1]
        effect1oo2[i]=(PFD1oo2[i-1]-PFD1oo2[i])/PFD1oo2[i-1]

}
ymin=min(effect1oo1,effect1oo2)
ymax=max(effect1oo1,effect1oo2)

plot(SFF, effect1oo1,type='l',col=1,ylim=c(ymin,ymax),ylab="Effect (%)")
par(new=T)
plot(SFF,effect1oo2,type='l',col=2,ylim=c(ymin,ymax),ylab="Effect (%)")
legend(0.3,0.025,c("1oo1 system","1oo2 system"),lty=1,col=c(1,2))
dev.copy2eps(file="1oo2effect_sml.eps")

#-------------------------------------------------------------------------------
#------ Compares the PFD for standard probability calculations ----------------
#---------------------- and the Markov model -----------------------------------
#-------------------------------------------------------------------------------
PFD1oo2_RBD=1/3*((1-beta)*lambda_DU*tau)^2+lambda_DU*tau/2*(beta+2*(1-beta)*lambda_DD*MTTR_D)+beta*lambda_DD*MTTR_
PFD1oo1_RBD=lambda_DU*tau/2+lambda_DD*MTTR_D

x=seq(2.6e-3,3.2e-3,by=1e-4)
plot(SFF[40:72], PFD1oo1[40:72,1],type='l',col=1,ylim=c(2.6e-3,3.2e-3),ylab="PFD",xlab="SFF",yaxt="n")
abline(h=PFD1oo1_RBD)
axis(2, at=x,labels=x*10^3, las=2)
mtext("*"~10^-3,side=3,line=0.2,adj=0)
dev.copy2eps(file="1oo1_PFDsml.eps")

x=seq(7.2e-5,7.6e-5,by=1e-6)
plot(SFF[40:72], PFD1oo2[40:72],type='l',col=1,ylim=c(7.2e-5,7.6e-5),ylab="PFD",xlab="SFF",yaxt="n")
abline(h=PFD1oo2_RBD)
axis(2, at=x,labels=x*10^5, las=2)
mtext("*"~10^-5,side=3,line=0.2,adj=0)
dev.copy2eps(file="1oo2_PFDsml.eps")
```