Safety Instrumented Systems operated in the Intermediate Demand Mode

Kristine Tveit

Thesis for the degree of

MASTER OF SCIENCE

Modelling and Data Analysis (MOD5960)

Department of Mathematics
The Faculty of Mathematics and Natural Sciences
University of Oslo

November 2015

Acknowledgements

I would like to thank my supervisor Siegfried Eisinger at DNV GL for an interesting and enjoyable topic. You have been a source of inspiration and have always been willing to sit down and explain me things when I needed help. Thank you for also giving me the opportunity to take part in some of the projects for your customers.

During the process of publishing the article for the ESREL 2015 conference in Zürich I learned a lot, and the experience was motivating. Thank you to Luiz F. Oliveira (DNV GL, Brazil) for suggesting it, and for the time you have taken to answer my questions related to your articles, both in person and by e-mail. We are now working on a journal paper from the work of this thesis, with some additional cases.

I have had a desk at Veritas for the last 14 months. I appreciate how the section has included me in social gatherings, pleasant lunches, but also meetings and functions. It has really been an enjoyable and interesting time!

Thank you to DNV GL for giving me this opportunity to write my thesis at Høvik. It has been a phenomenal experience.

I would like to thank my supervisor Bent Natvig at UiO for giving me great assistance when needed and prompted interesting discussions. Thank you for being patient in meetings when I brought my daughter, who was not always so tolerant. Bent corrected the mislabelling in [2] (scenario-based formula).

When I started to write this thesis my daughter was 5 months, finishing now, half a year early would not have been achievable if she had not let me sleep (or work) at night, so thank you Sophie! And without question, thank you to my mother Solveig, my friends Ingvild and Thea for babysitting, and NAV for paternity leave.

I am grateful for the support I have received from my parents Solveig and Trond, as well as my friends during my time at Blindern.

And last, but not least, a big thank you to my husband Michael, who has always supported me and made it possible to study when I needed to, travel to Zürich and have a social life.

Abstract

The frequency of demands are crucial when analysing a safety instrumented system (SIS). IEC 61508 distinguishes between low and high demand mode when calculating risk for such a system. In reality there are systems that can not clearly be placed in one of the two modes. These types of systems are called intermediate demand mode systems, which we will analyse in this thesis. Not many published SIS reliability studies focus on the problems related to this borderline. Oliveira [4] predicts somewhat strange behaviour for the hazard rate in the intermediate demand mode, as well as [2] with a focus on the demand duration.

The results from the analyses of a redundant system show that the standard Probability of Failure on Demand (PFD) formulae are usable for very low demand rates, but become increasingly more conservative as one moves into the intermediate mode, while the Probability of Failure per Hour (PFH) is non-conservative. This can cause major consequences for the operator of a safety system in the sense of not obtaining the optimal testing strategy, or even worse encounter a hazard.

For more complex systems with several components the Markov approach has its limits, choice of distributions and maintenance details are also restricted. Discrete Event simulation can deal with such complex systems, and also the rare event problem that often is a challenge for safety system analysis can be handled satisfactorily.

By use of Harel Statechart and discrete event Monte Carlo simulations for different safety systems, it is shown that the intermediate demand mode is dependent on the relationship between the proof-tests, demands and repair duration. When a demand rate increases to a significant level, demands can be used as tests. With Harel Statecharts we can calculate realistic models that go beyond what a Markov model is capable of.

Contents

1. Safety Instrumented Systems

All types of systems have a risk of failing. The question is what level of risk can be tolerated by the operator. Safety is provided by layers of protection to achieve a tolerable risk level. Elements of safety are added to the system by the operator. These elements can vary in complexity from the less complicated, such as typical mechanical devices, procedures etc., to more complex instrumented systems. For example, if a firewall does not meet the given risk requirements for a system, a fire detection and a sprinkler system can be added. Though, this extra level of protection also has a risk of failing. It is increasingly common to use software-based or instrumented protection systems, or to replace mechanical devices with automated systems compared to only a few decades ago. Like today's cars, more functions are being automated. With that, knowledge and analysis about Safety Instrumented System (SIS) is becoming increasingly relevant and important for different fields.

A SIS is a safety system, added to a critical process to reduce risk by preventing hazardous events. Hazardous events are incidents or consequences that occur when there is a significant deviation from the normal situation. A critical process can be defined as a system that will cause damages to human health, the environment or financial loss for an industry, in the event a hazard failure occurs. A SIS is also known as a protection or emergency shut-down system, for example an anti-breaking system (ABS) or an automatic train protective system which makes a train reduce speed if it runs too fast or come to close to the train ahead of it (relevant especially for metro systems), or it ensures that a train will stop at a red signal even if overlooked by the operator. It typically consists of three elements [12] (illustrated in figure 1.1); a detector (or sensor), a logic solver and actuating items (final control elements such as valves, brakes). The sensors are used to detect a possible emergency situation. The logic solver performs state control, and then the actuating items implement the action determined by the logic controller.

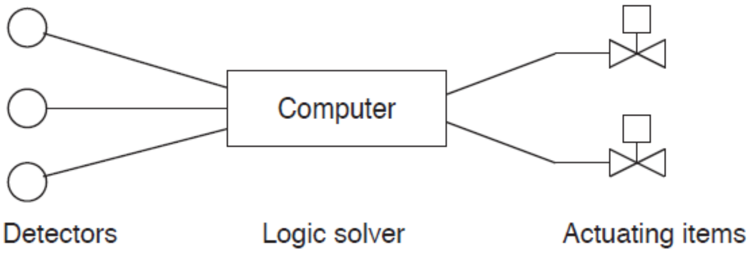


Figure 1.1: SIS

There are challenges in designing a protective system (SIS) to prevent or control dangerous failures. There are two types of dangerous failures [2]:

Dangerous undetected failure: A DU-failure is a failure on the safety system that is not yet visible for the operator. When the safety system is in this mode, it will not react correctly if a demand for it occurs, which might lead to a hazardous event. This type of failure is the main contributor to the SIS unreliability [2].

Dangerous detected failure: DD-failures are detected immediately by the safety system, normally by the controller. The repair can therefore be initiated immediately, and complete preferably before a demand occurs.

When a safety system experiences a demand, which is pre-programmed or a direct request from the operator, the system goes from its ”normal condition” to a different given mode. This can for instance be a railway signal system, a fire detector reacting to a fire, or an air-bag that inflates in a collision.

To assure the readability of the SIS many industries use the IEC 61508 [1], a generally-based standard for safety of Electronic Safety Systems. It includes a set-up method (SIS life-cycle) to implement the SIS to an existing system, and states the requirements for how to optimize the system and increase safety. The SIS life-cycle includes all aspects of a system, from the concept phase to decommissioning or disposal. There are also other such standards for other types of systems, like IEC 61511 for the process industry, IEC 62278/EN 50126 for the railway industry and ISO/ DIS 26262 for the auto-mobile industry.

When a system is designed, risk and hazard analysis are performed on the system. If the risk is intolerable, it must be reduced. To reduce the risk, the design of the system might be changed or non-SIS protection layers can be added. From the example at the beginning of this section this would be a fire wall. If this is not satisfactory, a SIS can be implemented to reduce the risk further. The SIS can perform one or more control functions to protect the system, called safety instrumented functions (SIFs) (e.g. a fire detector and a fire sprinkler system). A SIF is an electronic system that protects against a specific hazard and performs a safety function to reduce the risk to a tolerable level. The question is how much reliance on the additional SIS is needed to make the total risk acceptable. This ”reliance” is also called Safety Integrity Level (SIL). However, a SIF also adds a risk to the system which needs to be analysed. When a SIF is to be implemented to the system the SIL is determined, which sets the requirements of the necessary risk reduction for each SIF. SIL is the probability of a dangerous failure on a SIF that is targeted [6]. Thereafter the SIS is installed and the overall safety, operation, maintenance and repair are validated and tested, before a possible modification is done or a decommissioning of the system.

The IEC 61508 has stated four SILs, illustrated in table 1.1. The more reliable a system needs to be to perform satisfactorily, the greater risk reduction is needed and the higher SIL. SIL 4 represents the highest possible risk level, where the systems are required to have a very low probability to fail. Systems on this level have a very high requirement to the reliability of the SIF. It contains systems that have severe consequences on personnel, environmental and assets as well as production/ financial loss. This can be systems like railways and nuclear power plants. SIL 1 provides the lowest risk level that is accepted, and contains systems with a small ”risk gap”, such that the reliability required from the SIF can be rather low. The system can have a high failure rate. Table 1.1 says as the required probability on demand (PFD) or probability of failure per hour (PFH) decreases, the system requires a higher SIL.

From table 1.1 we can see that IEC 61508 establishes the requirements of the SIL accordingly with the two following demand modes. Each level contains a probability interval for failure on demand and failure per hour. These are found in terms of a maximum tolerable hazard rate. The two measures are [2]:
1) The average probability of failure on demand (PFD), a function based on the failure rate and the test interval.
2) Average frequency of a dangerous failure of the safety function, or the probability of failure per hour (PFH).

IEC 61508 suggests to use PFD for a low demand mode system, and PFH for a high demand mode system. The standard defines a SIS to be in the low demand mode when demands do not occur more than once a

Table 1.1: SIL requirements

Safety integrity level (SIL)	**PFD of the safety function (low-demand mode of operation)**	**PFH ($hour^{-1}$) of the safety function (high-demand mode operation)**
4	$\geq 10^{-5}$ to $< 10^{-4}$	$\geq 10^{-9}$ to $< 10^{-8}$
3	$\geq 10^{-4}$ to $< 10^{-3}$	$\geq 10^{-8}$ to $< 10^{-7}$
2	$\geq 10^{-3}$ to $< 10^{-2}$	$\geq 10^{-7}$ to $< 10^{-6}$
1	$\geq 10^{-2}$ to $< 10^{-1}$	$\geq 10^{-6}$ to $< 10^{-5}$

year and less than twice the frequency of functional tests. For the opposite case, when there is more than one demand on the safety system per year, or greater than twice the frequency of functional tests it is in a high demand mode system.

The SIL levels have the same meaning for low demand mode as for high demand mode systems. The risk level is more moderate for lower SIL. When a system is in SIL 1, the system has a higher rate of failure during an hour. For an increased frequency of demands, the more likely it is to detect a failure on a component before the whole system fails. A system in SIL 4 will have a very high frequency of demands and a small probability of failure. This is the same for a low demand system.

The relation between the failure and demand rate distinguish if the system is in a low or a high demand mode. However, why IEC 61508 makes the distinction on one year is unclear though [2].

A SIS can be analysed by various methods. Among them are approximation formulas (for example given by the IEC standards) and Markov methods (do not differentiate between a low demand mode and a high demand mode system), which has been concluded to be the most suitable [12].

PFD is calculated when a system is in a low demand mode as [6]:

$$\mathrm{PFD}_{\mathrm{avg}} = \frac{\lambda\tau}{2} \tag{1.1}$$

This is only true for $\lambda\tau << 1$, where λ is the failure rate and τ the length between tests.

The hazard rate for a low demand mode system is

$$\eta = \delta \cdot \mathrm{PFD}_{\mathrm{avg}}, \tag{1.2}$$

where δ is the demand rate.

The risk is dependent on the frequency of a hazard. The IEC 61508 states that the risk for a system to fail is dependent on the demand rate and PFD. The PFD given by the standard is not automatically a function of failure rate and test interval. It turns out that it can often be calculated as equation 1.1, because the DU-failure becomes the dominant one. When a DU-failure has occurred during a test interval, the test interval contains an unknown (failure not yet detected) and a known part. The average downtime is therefore $\frac{\tau}{2}$ [2]. Why PFD is used as a measure for determining the hazard rate is merely a tradition the process industry wants to hold on to.

For example the railway industry does not work with PFD, only PFH, which is here called Tolerable Hazard Rate (THR) [17].

For a low demand mode system it is necessary to perform regular testing to be sure to detect a possible DU-failure before a demand, to prevent a hazardous event. A fire detector, emergency shut-down systems (ESD) and air-bag system are all examples of systems in a low demand mode.

The following relationship is valid for a low demand mode system, when assuming demands occur more frequent than failures:

$$\tau_f \geq \delta >> \lambda >> \eta,$$

where τ_f is the frequency of tests and η is the hazard rate.

For a well working low demand mode system there should be more frequent tests than demands, and more frequent demands than failures on the system. To fulfil the definition of IEC 61508 for the low demand mode system we can have $\tau_f = 1$. The scale is then in per year for the different rates. During a reasonable number of failures, only a small part of them should lead to a crucial accident for the system. This is proved by using equation 1.2, the assumption $\tau_f = 1$ and maximum one demand per time unit. Hence $\delta \leq 1$. This indicates the following:

$$\delta = \frac{2\eta}{\lambda} \leq 1 \Leftrightarrow \frac{\eta}{\lambda} \leq \frac{1}{2}$$

and we have the relationship between the failure rate and the hazard rate to be:

$$\lambda >> \eta$$

For other systems where demands are more frequent than tests, like a railway signalling system, it does not make sense to run proof-tests between each demand. The proof-tests are superfluous because demands always detect the failures before the proof-tests. The high number of demands will detect a failure before the proof-tests are performed. It is cost and time consuming for the operator to perform more tests than needed. In these cases the safety is dealt with through redundancy and testing by demands. These types of systems are often activated automatically. Another example of a high demand mode system is dynamic positioning (DP) system for ships.

For the high demand mode system case, each demand represents a proof-test [3]. Here $\delta >> \lambda$ must be true to have a realistic system. When calculating the failure rate in a high demand rate case, the intervals in time are very small since the demands have a high frequency to appear ($\Delta(t) \to 0$, where $\Delta(t)$ is the length between a demand at time t and until the next demand occurs). The failure rate approximates the hazard rate [8]. The relationship is now for a single component system:

$$\delta >> \lambda \approx \eta$$

In reality there are systems that cannot be clearly placed in either a low or high demand mode system, and might be called intermediate demand mode systems. For example a blow-up preventer (BOP). During a drilling operation, it is meant to stop uncontrolled flow from oil wells. This happens seldom, but when a demand does occur it provides "sub-demands", that can be activated for hours or weeks [2].

We are looking more closely into this demand region in this thesis.

2.

Harel Statecharts

Classic state diagrams create properties and transitions for each single state/ node in a system with a finite number of states. A state diagram where it is only possible to be in one state at a time, it is a disjunctive (”xOr”) diagram [13]. Each node contains all the parameters and properties for describing the behaviour of each state, which means when a state is known all its properties are known to the system. This can lead to a large number of nodes to define a more complex system, and reduce the understanding of the state diagram [11].

Harel Statechart is a visual state diagram with relevance for describing complex discrete-event systems. It simplifies the systems compared to state diagrams because of its structure that creates super nodes/ superstates [11] that allows the machine to have the ”AND”-diagram property. The sub-state system can be serial or parallel. The properties (parameters and variables) the super-state contains are available for all its sub-states.

For a serial sub-state system, the initial state given for this series is entered, and the system will only be in one state at a time (”xOr” diagram). There are sub-conditions that define more properties than are known by the superstate. When a super-component property is known and updated in a sub-state, it is known for the whole system. In that way the model can combine the information given by the superstate with the value of the parameter on a sub-state level to define the exact condition/state of the system. This means that the number of nodes to describe each state of a system can be reduced, and the diagram is more intuitive to understand.

Transitions between the sub-state system happen when the event of entering a specific node is true. Each node has the variables from the super node available for use. It goes from one node to another when the event (entering condition) for a specific node is true.

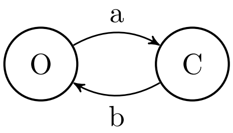


Figure 2.1: State diagram for a door

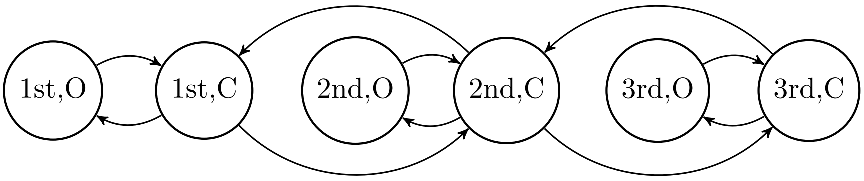


Figure 2.2: Traditional state diagram for an elevator door at three floors.

From [5] a statechart is described as:

$$\begin{aligned} \text{statecharts} = {} & \text{state diagrams} + \text{depth} \\ & + \text{orthogonality} + \text{broadcast-communication} \end{aligned} \tag{2.1}$$

A statechart is an enlargement of classic state diagrams, with some extra properties added. This is described below, based on [5] and [14]:

State diagram A state diagram can for example be a door that can be open and closed, illustrated in figure 2.1. State "O" defines that the door is open, while in state "C" the door is closed. Transition rate a means that someone is closing the door, while b defines that it is being opened. The trigger condition to enter state "O" is that b has to be true, while for entering state "C", a must be true.

This example can be extended to be an elevator door, where we want the system states that define which floor the elevator door is opened or closed on. A state diagram needs six components for describing a system where an elevator runs between three floors. This example is illustrated in figure 2.2.

Depth (Hierarchy) and Orthogonality (Concurrency) We can easily see that the state diagram can quickly be very chaotic if there are many more states (more floors) in the state diagram in figure 2.2. However, figure 2.3 illustrates how this is solved by Harel Statechart. Superstate "Elevator" consists of AND components, where "Door" is one sub-level of the super-state consisting of the nodes in a serial system

"O" and "C" ("xOr" diagram). "Floor" is the other sub-level with a serial system containing the three floors "1st", "2nd" and "3rd". This set-up reduces the transitions between each component since the states do not have to be directly linked. We can easily move back and forth between each level. "Door" and "Floor" are both synchronized and independent. When the system is in "Elevator", it also has to be in "O" or "C" and "1st", "2nd" or "3rd".

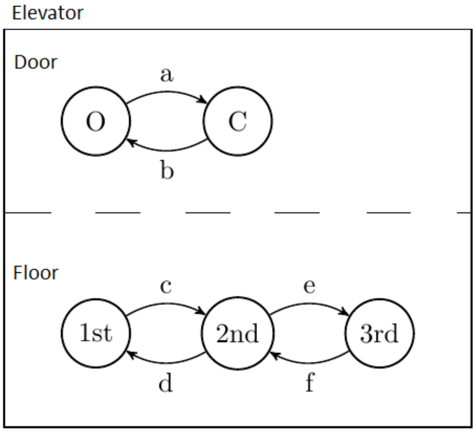


Figure 2.3: Harel State chart diagram for the behaviour of an elevator door at three floors.

Broadcast-communication The model from the superstate point of view sees all events that occur "below" hierarchically. This is illustrated for the elevator example in figure 2.4.

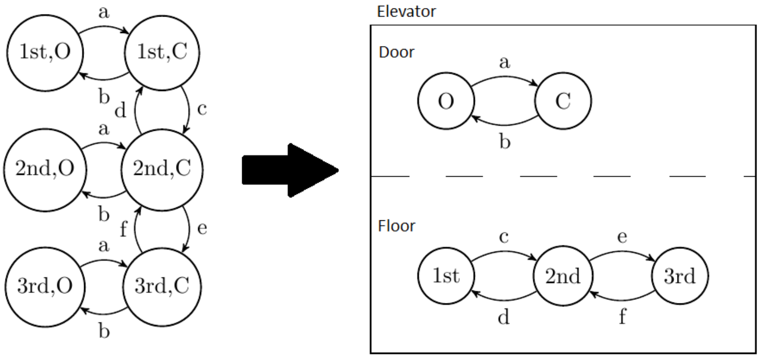


Figure 2.4: State chart illustrating orthogonality and broadcasting

For more detailed information about Harel Statechart, check out [5].

2.1 Harel Statechart in ExtendSim

The properties of Harel Statechart are implemented in ExtendSim 9. The hierarchical property is considered here with use of a parentOfParents block as the superstate. The variables are inherited hierarchically.

The system models simulated in this thesis are constructed in a similar way. There is a "Global" state, which is the superstate of the system (parentOfParents block). This state contains all parameters and variables that are on a system level, which are available to the sub-states of the "Global" state. The sub-states represent parallel modules consisting of serial sub-states that constitute the different states for a component of the system. Each sub-state block also contains local variables that are only revealed to the sub-system and its sub-systems (if they exist). During the simulation time the system is at all times in the "Global" state as well as in one of the serial states of all parallel modules.

Each Harel State block contains details about which conditions are valid for entering and exiting the block, as well as its duration time. When a block is entered it has the possibility to calculate values that are of relevance to it. The system variables that are calculated and updated throughout the simulation in a sub-state block, are updated in all other blocks on the system containing this variable. With the updated information the system makes the necessary change of state.

Each Harel State block gives results of how many times it has been visited through the simulation time, with mean and standard deviation.

In this thesis we simulate a 1oo1, 1oo2 and a 2oo3 system model. For the 1oo1 system model, technically there is no point to have a "Global" block, because there is only one component consisting of a series of states (state diagram). However, when we have a component as a sub-state system to the "Global" block, it is very easy to extend the model to a 1oo2, 2oo3, or a even more complex system model that goes beyond what a Markov model can calculate. We will look closer into this.

For more detailed information about the relevant blocks used in ExtendSim for the different simulation models in this thesis, see appendix B.

3.

Single channel system with repair

A SIS is said to have a k-out-of-n configuration. We will first start with a 1oo1 system. The following system is based on a nuclear power plant with repair ([3]).

3.1 Description of the system

- λ (Failure rate): The failure rate is constant. In this thesis it is strictly equal to 1. A failure is detected by either a demand or a proof-test.
- δ (Demand rate): Measured as a rate of the failure rate. Exponentially distributed.
- μ (Repair rate): Measured as a rate of the failure rate. Exponentially distributed. Mean time of repair is $\frac{1}{\mu}$.
- τ (Proof-test interval): It is a fixed length between each proof-test. It is measured as a length of time of the time between each failure.

The model illustrated in figure 3.1, describes a simple safety system for a nuclear power plant. The three states for the model are:
State 1: the system is up
State 2: the system is down, but failure is undetected
State 3: the system is down, failure has been detected and is under repair

The assumptions made for the single-channel model are:

- The model consists of a failure rate λ, a repair rate μ and demand rate δ that are constant over time.
- It is periodically tested, with a constant length τ.
- For time 0, the system is up and running (state 1).

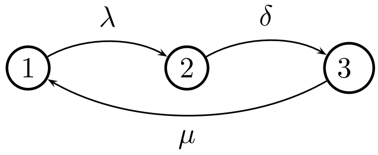


Figure 3.1: State diagram illustrating orthogonality and broadcasting

- There can be a maximum of one hazardous event on a proof-test interval.

The model treats two types of cases:

Offline model: The plant is shut down when the safety system is known to be down. No demands on the system occur. In this model, the system is turned off when it is in state 3.

Online model: The model assumes that the operator lets the plant run when the safety system is in repair, state 3. Demands on the system during repair of the safety system can occur, which leads to an increased frequency of hazard events.

For a single channel system the online case is not realistic. Oliveira explains the motives for including an online case in the article [3] in an e-mail to Bent Natvig (quoting): "the plant can get back to operation after an accident before the safety system is restored to an operating condition. This is quite an unusual situation, but not impossible, especially if the "plant" accident is not really catastrophic, but one that causes a temporary disruption or a loss of production". For a two-channel system this makes more sense, and can be more common.

The plant hazard rate is described [3] as a "plant" transition, meaning the system does not necessarily break down even if the safety system does. The hazard rate for the plant will be obtained from the simulation of the safety systems. A hazard failure can only happen when a demand occurs when the system is in a down state. For the offline model, this is in state 2. A demand happens before a proof-test when there is an undetected failure. The hazard rate for the plant is in this case:

$$\eta(t) = \delta P_2(t) \tag{3.1}$$

For the online case, the plant may still have demands when the safety system is under repair. The hazard rate is therefore:

$$\eta(t) = \delta \left[P_2(t) + P_3(t) \right] \tag{3.2}$$

3.2 Asymptotes for the single channel protective system

The asymptotes for a hazard rate of the single channel protective system without repair, is as stated in section 1. For a low demand rate it is: $\delta\frac{\lambda\tau}{2}$ (equation 1.2). And for a high demand rate it approximates the failure rate λ, as shown in section 1.

The Probability of Failure on Demand (PFD) for a system with repair is [9]:

$$\text{PFD} = \frac{\lambda}{2}\bigg(\tau + \text{MTTR}\bigg),$$

where MTTR is Mean Time to Repair, $1/\mu$. The asymptote of a hazard rate for a system with low demand rate and repair is then:

$$\eta_{1oo1,l} = \delta \cdot \frac{\lambda}{2}\bigg(\tau + \frac{1}{\mu}\bigg) \tag{3.3}$$

We can assume this is also valid for the online case, since there is close to zero demands during the repair when the demand rate is low.

For systems with high demand rates, we can find the asymptote of the hazard rate by looking at the steady-state condition for the Markov model (figure 3.1). This model is designed to solve a high demand problem since none of the states are dependent on proof-tests. With an expression of P_2 and P_3 the asymptotes of a hazard rate for the offline and online models are obtained. The system equation is when $\frac{\delta P_i(t)}{\delta t} = 0$ for $i = 1, 2, 3$:

1) $\delta P_2 = \lambda P_1$
2) $\mu P_3 = \delta P_2$
3) $P_1 + P_2 + P_3 = 1$

From equation 1 and 2 we obtain:

$$P_1 = \frac{\delta}{\lambda}P_2, \text{ and}$$

$$P_3 = \frac{\delta}{\mu}P_2$$

By substituting this into equation 3 we get an expression for P_2:

$$P_2 = \frac{\lambda\mu}{\mu\delta + \delta\lambda + \lambda\mu}$$

Using this expression in equation 3.1, and letting $\delta \to \infty$, the asymptote for a hazard rate of an offline model with repair is:

$$\eta_{1001,h,off} = \frac{\lambda\mu}{\mu + \lambda} \tag{3.4}$$

This states that including a repair time to a system decreases the hazard rate, $\frac{\lambda\mu}{\mu+\lambda} < \lambda$, since there cannot be hazardous events during repair for the offline case.

For the asymptote of the hazard rate for the online model, we obtain P_3 from the equations above by substituting for P_2:

$$P_3 = \frac{\lambda\delta}{\mu\delta + \mu\lambda + \lambda\delta},$$

substituting P_2 and P_3 into equation 3.2:

$$\eta = \delta \left[\frac{\lambda\mu}{\mu\delta + \delta\lambda + \lambda\mu} + \frac{\lambda\delta}{\mu\delta + \mu\lambda + \lambda\delta} \right],$$

and finally have $\delta \to \infty$ on this expression:

$$\eta_{1001,h,on} = \frac{\lambda\delta}{\mu + \lambda} \tag{3.5}$$

3.3 Simulation of the single channel system

To analyse this single component safety system in figure 3.1 we use Harel Statecharts in ExtendSim, introduced in section 2. We are interested in obtaining the hazard rate for the offline and online model (equation 3.1 and 3.2). If we make a model which simulates all demands and proof-tests on a system this costs a lot of processing time. The computer processes a lot more events than are necessary to find the hazardous events, which makes the time to run the simulation longer. Especially for a high demand mode system the majority of the simulated demands and proof-tests does not detect a failure. To reduce the processing time, the simulation time can be decreased. There will be less demands that can detect the failures, which leads to a less accurate result of the hazard rate, and we have a Rare Event Problem. To deal with this we can simulate only those proof-tests and demands that actually detect a failure. This is performed using Harel State Models simulated using Discrete Event Monte Carlo Simulation. In section 4.2.1 we will compare the results of a simulation model for a safety system model that simulates all demands and proof-tests on the system, with a model only simulating the crucial events.

Discrete event Monte Carlo Simulation is explained in [24]. It is based on the simulation model calculating the next event in the system with randomly drawn times for the specified distribution. The computer keeps track of the near future events that will happen to the system. The simulation model can in this way go from event to event, and is much more efficient, compared to doing traditional simulations with constant time steps [24]. The simulation model can therefore calculate the next wanted event at the specific time it is relevant, and does not ”waste” processing time on calculating information that is not crucial for the simulation model at all times. This solves a Rare Event Problem in an adequate way.

For this model it is assumed that demands are exponentially distributed, and by its memoryless property, the next demand to occur is not dependent on the previous one. Proof-tests happen with a constant length of time, and the next proof-test can easily be found. Hence, the rare event problem is resolved by calculating the time for next proof-test and demand when the system has an undetected failure.

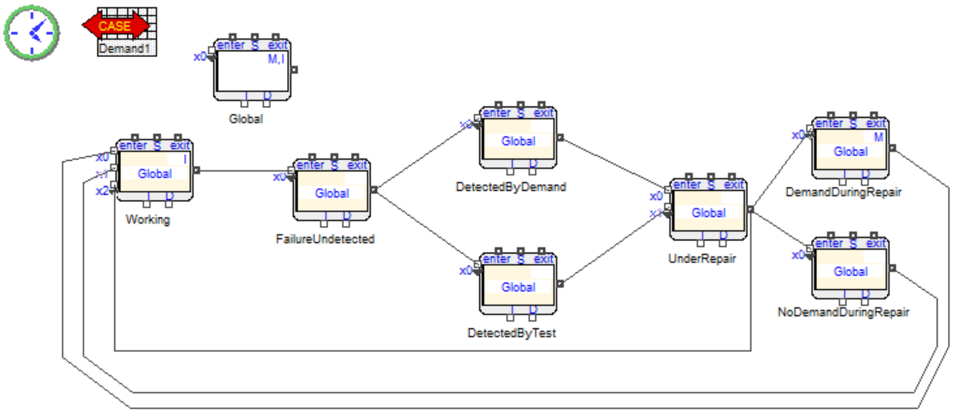


Figure 3.2: Simulation model of a single-channel model using ExtendSim

Figure 3.2 illustrates the one component model in figure 3.1 simulated in ExtendSim.

”Global” contains the system parameters λ, δ, μ and τ. These parameters have a given value. ”Global” also contains the variables; $t_{nextDemand}$ (time for next demand) and $t_{nextTest}$ (time for next test) which are thus for all sub-systems.

The system starts in ”Working” (state 1), and stays there until a failure occurs on the system. The model goes to ”FailureUndetected” (state 2). Here the failure is detected either by test or demand, and the system will go to either ”DetectedByDemand” or ”DetectedByTest”, and on to be repaired in "UnderRepair" (state 3). During the repair time for the online case a demand might occur, and in that case the system can go to "DemandDuringRepair" otherwise it goes to "NoDemandDuringRepair".

A more detailed explanation of each sub-state block follows:

Working: The initial state of the system, state 1 in figure 3.1, is represented by the block "Working". This block has λ as a parameter from "Global". This block is configured such that the duration the system will stay here is exponentially distributed with rate λ. In this way the model knows when there is a failure in the system, and the time in which this block will finish. Thereafter it moves on to the connected block "FailureUndetected".

Since the next demands and next tests are not crucial for the system in this state they are not calculated here.

FailureUndetected: This block represents state 2 in figure 3.1. This block has the global parameter δ, and variables $t_{nextDemand}$ and $t_{nextTest}$.

In this block it is crucial to calculate the time for the next demand and the next test, since we know that this will lead to detecting the failure on the system. Therefore when this block is entered these two times are calculated with the following formulae:

$$t_{nextDemand} = t + \text{DExp}(\delta), \text{ and} \tag{3.6}$$

$$t_{nextTest} = t + \tau - (t \bmod \tau), \tag{3.7}$$

where t is the current time and DExp is the interval between events, that is exponentially distributed with δ as the expected number of events per time. The next demand can be calculated since it contains the memoryless property, meaning it is independent of the time since the previous demand. We can calculate the time to the next test since they are dependent, and happen with a constant length. These calculations are now updated in "Global", and other blocks that use these two variables. If the time to the next demand is smaller than the time to next test, the model goes to "DetectedByDemand". Otherwise it goes to "DetectedByTest". The algorithm in this block is shown in listing 3.1.

DetectedByDemand: This block is used to count how many times the system gets a hazardous event. The duration time is approximately zero, and the system goes directly to "UnderRepair".

Listing 3.1: Calculations in ”FailureUndetected”

```
// when entered
nextTest = currentTime + tau - RealMod(currentTime, tau);
nextDemand = currentTime + DExponential(demand);

// triggerOut condition
if(nextDemand < nextTest) -> "DetectedByDemand"
else -> "DetectedByTest"
```

Listing 3.2: Calculations in ”UnderRepair”

```
// when entered
if(nextDemand < currentTime) nextDemand = currentTime +
   DExponential(demand);
repairTimeOver = currentTime + DExponential(mu);

// triggerOut condition
if(mu==0) -> "Working";
else if(nextDemand < repairTimeOver) -> "DemandDuringRepair";
else -> "NoDemandDuringRepair";
```

DetectedByTest: This block is used to count how many times the safety system has failed, but a test detected the failure and a hazardous event is avoided. The duration time is approximately zero, and the system goes directly to ”UnderRepair”.

UnderRepair: This block represents state 3. It contains the global parameter μ and the variable $t_{nextDemand}$. The local variable is $t_{repairTimeOver}$ (time when the repair is finished). The calculations done in this block is shown in listing 3.2. Note that the block has the information of the time to the next demand that was calculated in ”FailureUndetected”. This is updated here if necessary.

The duration in this block depends on the condition set for the model. If there is no repair, the duration time is 0. If the system is turned off during repair (offline case), the duration time is $t_{repairTimeOver} - t$. The system then goes to ”Working”.

For the case where the system is not turned off (online case), the duration of this block is $t_{nextDemand} - t$. If there is a demand, the model goes to ”DemandDuringRepair”, where the rest of repair is being done.

DemandDuringRepair: It contains the local variables $t_{nextDemand}$ and

Listing 3.3: Calculations in ”DemandDuringRepair”

```
//when entered
if(timeIn < currentTime) timeIn = currentTime;

//triggerOut condition
if(timeOut < currentTime) timeOut = currentTime;
-> "Working"

//Expression evaluated on Exit from state.
//Demands during repair
f = (timeOut-timeIn)*demand
f = DPoisson(f);
addDemand += f;
```

$t_{repairTimeOver}$, calculated in ”UnderRepair”, as well as t_{timeIn} (time for entering the block), $t_{timeOut}$ (time for exiting the block) and ”addDemand”.

The duration of this block is the remaining repair time $t_{repairTimeOver}$ - $t_{nextDemand}$.

Demands during the repair time can be modelled as a homogeneous Poisson process [12] with rate $\delta \cdot t_{systemDownTime}$ where $t_{systemDownTime}$ is the time from one demand occurring and until the repair time is over. ”addDemand” sums up each demand that is calculated within the time in this block. The calculations are in listing 3.3.

NoDemandDuringRepair: The duration is approximately zero, and the system goes straight to ”Working” because the repair time for the component is finished.

Calculating the hazard rate for the simulated model

As mentioned in section 2 and above in the explanation of each block, the Harel State blocks contain information on how many times the system visits each of them. There is a hazardous event when a failure is detected by demand, when the system is in ”DetectedByDemand”. The hazard rate for an offline model, equation 3.1 is calculated by:

$$\eta_{sim,off} = \frac{\#\text{ events in state 2}}{\#\text{ simulation time}} = \frac{\#\text{ events in ”DetectedByDemand”}}{\#\text{ simulation time}}$$

When simulating equation 3.2, the result of the local variable "addDemand" from "DemandsDuringRepair" has to be included.

$$\eta_{sim,on} = \frac{\#\text{ events in state 2 and 3}}{\#\text{ simulation time}} = \eta_{sim,off} + \frac{\text{"addDemands"}}{\#\text{ simulation time}}$$

3.4 Simulated results

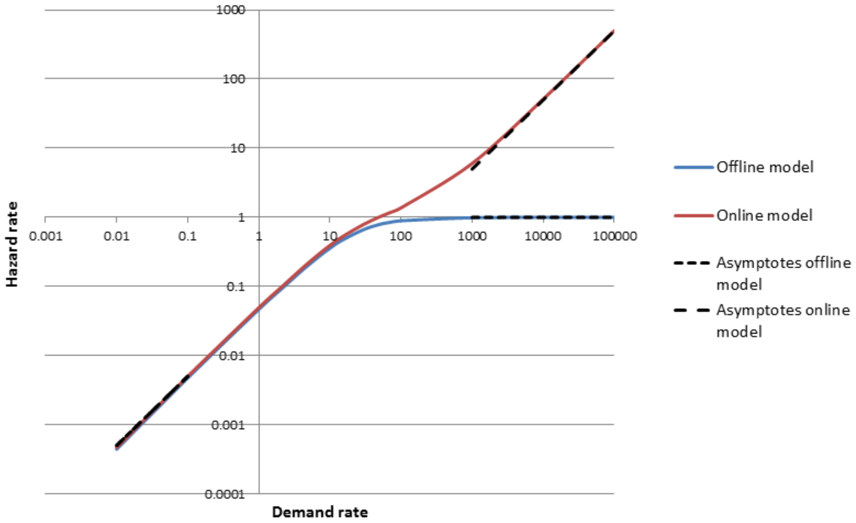


Figure 3.3: Reliability of a single-channel protective system. $\lambda = 1$, $\tau = 0.1$ and $\mu = 200$. Rates per year.

The hazard rates are plotted in figure 3.3, as well as their calculated asymptotes (equation 3.3, 3.4 and 3.5). The results from the simulated model are well approximated to the asymptotes for the hazard rate of the offline and online models. The curve of the hazard rate for the online model flattens out slightly around $\delta = 50$, before becoming steep after $\delta = 100$. The hazard rate of the offline model approximates a constant as demands occur with a high frequency, while the online model approaches infinity.

When the $\delta > 10\lambda$ ($\lambda = 1$) there is a significant difference between the offline and online model. For a higher frequency of demands there is a negative effect on the hazard rate if the operator runs the system while the safety system is down for repair, because of increased chance of demands to cause hazardous events.

Table 3.1 illustrates the numerical values obtained for the asymptote, numerical calculation from the Markov model [3] (where P_2 and P_3 are the

Table 3.1: Comparing plant hazard rates of the single-channel system. Failure rate $\lambda = 1$, repair rate $\mu = 200$ and proof test interval $\tau = 0.1$

	Off-line			On-line		
δ	Asymptote	Markovian approach [3]	Simulated	Asymptote	Markovian approach [3]	Simulated
0.1	0.005	0.0048	0.0047	0.005	0.0048	0.00498
1	0.05	0.0468	0.0465	0.05	0.047	0.0492
10		0.3573	0.3563		0.3737	0.3911
30		0.6678	0.6656		0.7611	0.7913
50		0.7866	0.7886		0.9712	1.0137
70		0.8439	0.844		1.1222	1.168
100		0.8884	0.8899		1.3079	1.3673
1000	0.995	0.9844	0.985	4.98	5.6579	6.0034
10000	0.995	0.9942	0.9946	49.8	48.216	51.2626

calculated probabilities in section 3.1 substituted in equations 3.1 and 3.2) and the simulated values.

From these results there is not much difference from using a Markovian approach and a simulated model.

4.

Two channel Protective System

The model in this section is a two channel safety system model based on [4]. The components in the single channel system from section 3 are a super-component for each of the components in this 1-out-of-2 system (parallel system).

A parallel system will not fail until all of the components have failed, which means the last one to fail and cause a hazardous event is the most important one. As the number of components increases so will the reliability of the system. The component with the lowest hazard rate is the upper limit for a parallel system.

The assumptions made for the two channel model are:

- The model consists of a failure rate λ, a repair rate μ and demand rate δ that are constant in time. The rates are equal for both components (similar as for the single channel system in section 3).
- It is periodically tested, with a constant length τ. All components are tested at the same time.
- For time 0, the system is up and running (state 1).
- The system is in state "working" as long as minimum one component is up.
- The repairs are independent. When a failure is detected due to demand or proof-test on a component the operator will start repair even when the system is up.
- When the safety system is down for repair, and one component has finished repair before the other, the system starts to run immediately. There is perfect repair on the components. These are systems that can recover after an accident.

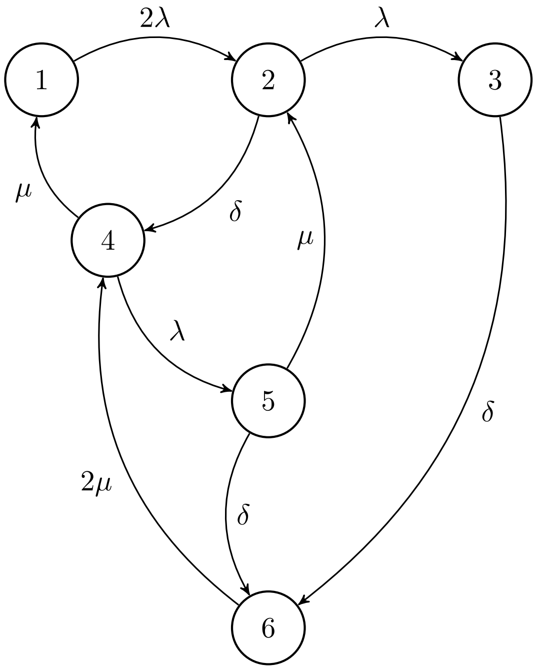


Figure 4.1: State diagram for a two channel safety system with repair

- The solutions are only valid within each proof test interval.

The states of this model are as follows:
State 1: both channels are up
State 2: one channel is up, and the other is down, but failure is undetected
State 3: both channels are down, but failures are undetected
State 4: one channel is up, and the other is under repair (its failure has been detected due to demand)
State 5: one channel is down, but undetected, and the other is under repair
State 6: both channels are down, and their failures have been detected due to demand

Its state diagram is shown in figure 4.1.

With the same reasoning as for the single channel system, the hazard rate for an offline model is [4]:

$$\eta = \delta[P_3 + P_5], \tag{4.1}$$

since a demand is crucial when the system is down for state 3 (both channels are down, but failures are undetected) and state 5 (one channel is down, but undetected, and the other is under repair).

For the online model, the operator lets the system run, and state 6 (both channels are down, and their failures have been detected due to demand) also has to be accounted for. The hazard rate is:

$$\eta = \delta[P_3 + P_5 + P_6] \tag{4.2}$$

4.1 Asymptotes

4.1.1 Asymptote of the hazard rate for the no-repair case

The Probability of Failure on Demand (PFD) in a low demand case with no repair is for a k-out-of-n system [7]:

$$\mathrm{PFD}_p = 1 - \sum_{x=k}^{n} \left[\sum_{i=k}^{x} \left[\binom{n}{x} \binom{x}{i} (-1)^{x-i} \right] \cdot \frac{1 - e^{-x \cdot \lambda\tau}}{x \cdot \lambda\tau} \right], \tag{4.3}$$

where τ is the length of the proof-test interval. For a 1oo2 system, where n=2 and k=1, the PFD is:

$$\mathrm{PFD}_p = 1 - 2 \cdot \frac{1 - e^{-\lambda\tau}}{\lambda\tau} + \frac{1 - e^{-2\lambda\tau}}{2\lambda\tau} \tag{4.4}$$

By expanding the exponentials by a Taylor series, for a small $\lambda\tau$ ($\lambda\tau <<$ 1) we get:

$$\mathrm{PFD}_p \approx \frac{(\lambda\tau)^2}{3} \tag{4.5}$$

And the hazard rate for a low demand mode system without repair is:

$$\hat{\eta}_{1oo2} = \delta \left[\frac{(\lambda\tau)^2}{3} \right] \tag{4.6}$$

For the high demand case, as mentioned in section 1, demands behave as proof-tests. Equation 4.3 cannot be used to estimate the hazard rate because it is given for constant test intervals, which happen with too low frequency compared to the demands. For a parallel system, when the demand rate is very high it is assumed that each failure will be detected by demand, and the system is protected by redundancy. Compared to the single system, it will not necessarily lead to a hazardous event.

The asymptote for the high demand case, hazard rates are found in the same way as for the single channel system, by steady-state conditions using the Markov approach.

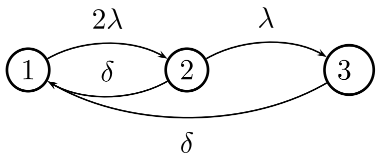


Figure 4.2: State diagram for the two channel system without repair

The two channel system without repair reduces to only contain the three first states. Its state diagram is shown in figure 4.2. By including the Markov property and $\sum_{i=1}^{n} P_i = 1$ the system equation is when $\frac{\delta P_i(t)}{\delta t} = 0$ for $i = 1, 2, 3$:

1) $(\delta + \lambda)P_2 = 2\lambda P_1$
2) $\delta P_3 = \lambda P_2$
3) $P_1 + P_2 + P_3 = 1$

By solving for P_1 in 1), and substitute P_2 from 2) into 1):

$$P_1 = \frac{(\delta + \lambda)P_2}{2\lambda} = \frac{(\delta^2 + \delta\lambda)P_3}{2\lambda^2}$$

Using this in 3):

$$\frac{(\delta^2 + \delta\lambda)P_3}{2\lambda^2} + \frac{\delta P_3}{\lambda} + P_3 = 1$$

Solving for P_3 :

$$P_3 = \frac{2\lambda^2}{2\lambda^2 + 3\lambda\delta + \delta^2}$$

The hazard rate is in this case $\delta \cdot P_3$. When $\delta \to \infty$, the hazard rate is:

$$\eta_{high,no\ rep} = \frac{2\lambda^2}{\delta} \tag{4.7}$$

This approach cannot be used to obtain the asymptote of a low demand mode system, because as stated above, the model does not contain proof tests. For example, by letting $\delta \to 0$ for P_3, then $P_3 \simeq 1$, which does not make sense since it is not an absorbing state. The hazard rate becomes much too conservative.

4.1.2 Asymptote of the hazard rate for a two channel system with independent repairs

The probability of failure on demand for a 1oo2 system with repair is [1]:

$$\mathrm{PFD}_{1oo2} = 2\lambda^2\left(\frac{\tau}{2}+\frac{1}{\mu}\right)\cdot\left(\frac{\tau}{3}+\frac{1}{\mu}\right)$$

The asymptote of a hazard rate for a low demand mode system with independent repairs (offline and online model) is:

$$\hat{\eta}_{1oo2,low} = \delta\cdot 2\lambda^2\left(\frac{\tau}{2}+\frac{1}{\mu}\right)\cdot\left(\frac{\tau}{3}+\frac{1}{\mu}\right) \tag{4.8}$$

The same approach as in section 4.1.1 with the Markovian model (figure 4.1) is used to calculate the asymptote of the hazard rate for a high demand mode system. The equation system when $\frac{\delta P_i(t)}{\delta t} = 0$ for $i = 1,..,6$ is:

1) $2\lambda P_1 = \mu P_4$
2) $(\delta+\lambda)P_2 = 2\lambda P_1 + \mu P_5$
3) $\delta P_3 = \lambda P_2$
4) $(\delta+\mu)P_5 = \lambda P_4$
5) $\delta P_3 + \delta P_5 = 2\mu P_6$
6) $P_1 + P_2 + P_3 + P_4 + P_5 + P_6 = 1$

From 1) - 6) the following probabilities for each state are obtained:

4) $P_4 = \frac{\delta+\mu}{\lambda}P_5$
1) $P_1 = \frac{\mu(\delta+\mu)}{2\lambda^2}P_5$
2) $P_2 = \frac{2\lambda P_1+\mu P_5}{(\delta+\lambda)} = P_5\left[\frac{\mu}{\delta+\lambda}\left(\frac{\delta+\mu}{\lambda}+1\right)\right]$
3) $P_3 = \frac{\lambda}{\delta}P_2 = P_5\left[\frac{\lambda\mu}{\delta(\delta+\lambda)}\left(\frac{\delta+\mu}{\lambda}+1\right)\right]$
5) $P_6 = \frac{\delta(P_3+P_5)}{2\mu} = P_5\left[\frac{\lambda}{2(\delta+\lambda)}\left(\frac{\delta+\mu}{\lambda}+1\right)+\frac{\delta}{2\mu}\right]$

The probabilities above are substituted into equation 6):

$$\begin{aligned}
P_5 =& 1-\frac{\mu(\delta+\mu)}{2\lambda^2}P_5 - P_5\left[\frac{\mu}{\delta+\lambda}\left(\frac{\delta+\mu}{\lambda}+1\right)\right] - P_5\left[\frac{\lambda\mu}{\delta(\delta+\lambda)}\left(\frac{\delta+\mu}{\lambda}+1\right)\right] \\
&-\frac{\delta+\mu}{\lambda}P_5 - P_5\left[\frac{\lambda}{2(\delta+\lambda)}\left(\frac{\delta+\mu}{\lambda}+1\right)+\frac{\delta}{2\mu}\right] \\
1 =& P_5\left[1+\left(\frac{\delta+\mu}{\lambda}\right)\left(\frac{\mu}{2\lambda}+\frac{\mu}{\delta+\lambda}+\frac{\lambda\mu}{\delta(\delta+\lambda)}+1+\frac{\lambda}{2(\delta+\lambda)}\right)\right. \\
&\left.+\frac{\mu}{\delta+\lambda}+\frac{\lambda\mu}{\delta(\delta+\lambda)}+\frac{\lambda}{2(\delta+\lambda)}+\frac{\delta}{2\mu}\right] \\
P_5 =& \frac{1}{1+\left(\frac{\delta+\mu}{\lambda}\right)\left(\frac{\mu}{2\lambda}+\frac{\mu}{\delta+\lambda}+\frac{\lambda\mu}{\delta(\delta+\lambda)}+1+\frac{\lambda}{2(\delta+\lambda)}\right)+\frac{\mu}{\delta+\lambda}+\frac{\lambda\mu}{\delta(\delta+\lambda)}+\frac{\lambda}{2(\delta+\lambda)}+\frac{\delta}{2\mu}}
\end{aligned} \tag{4.9}$$

The hazard rate is

$$\eta_{high,offline} = \delta(P_3+P_5) = \delta(P_5[\frac{\lambda\mu}{\delta(\delta+\lambda)}(\frac{\delta+\mu}{\lambda}+1)]+P_5)$$

When $\delta \to \infty$, $\eta_{high,offline} \to \delta P_5$. And substituting for P_5 the asymptote of the hazard rate for the offline model is:

$$\hat{\eta}_{1oo2,high,off} \simeq \frac{2\lambda^2\mu}{\mu^2+2\lambda\mu+\lambda^2} \tag{4.10}$$

For the online case, we substitute P_3 and P_6 into equation 4.2;

$$\eta_{high,online} = \delta\left[P_5[\frac{\lambda\mu}{\delta(\delta+\lambda)}(\frac{\delta+\mu}{\lambda}+1)]+P_5+P_5[\frac{\lambda}{2(\delta+\lambda)}(\frac{\delta+\mu}{\lambda}+1)+\frac{\delta}{2\mu}]\right],$$

and $\eta_{1oo2,high,on} \to P_5\frac{\delta^2}{\mu}$ when $\delta \to \infty$.

With similar approach and the result of $\delta \cdot P_5 \to \hat{\eta}_{high,offline}$ when $\delta \to \infty$, the asymptote of the hazard rate for the high demand online case is

$$\hat{\eta}_{online,high} \simeq \frac{\delta\lambda^2}{\mu^2+2\lambda\mu+\lambda^2} \tag{4.11}$$

It increases with the demand rate, as previously implied.

4.2 Simulation models of the two channel system

The two channel model from figure 4.1 is simulated in three ways using ExtendSim to illustrate the accuracy of the method we use to overcome the Rare Event Problem introduced in section 3.3. The first model described in section 4.2.1 is seen from the operators' perspective where tests and demands run continuously through the whole simulation time, even when the system is working. The computer processes a lot more events than are necessary to find the hazardous events, which makes the time to run the simulation longer. We call this model a direct model.

The second model in section 4.2.2, tries to deal with the low demand rare event problem by simulating proof-tests only when necessary, namely when at least one component is down. This model also experiences a long processing time when the demand rate increases.

To deal with the Rare Event Problem for demands, the solution proposed for the single channel model, is used for both the components. Similar to the calculation of proof-tests, the time for next demand is calculated when it is crucial for a demand or a proof-test. In this case, when at least one component is down. This approach is explained in the third model, section 4.2.3, the optimized model.

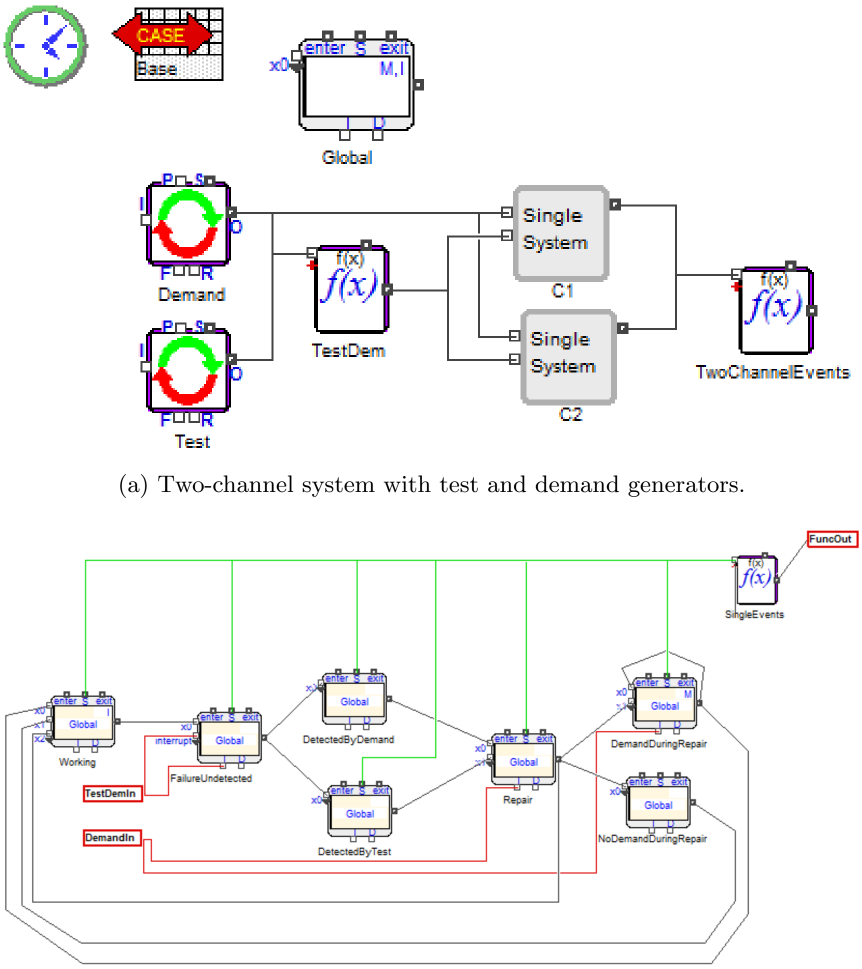
For models with more than one component there is a system and a component level of detail. Failure, proof-tests and repairs occur on each component, while demand and hazards happen on system level.

The methods and results are explained in more details in the following sections.

4.2.1 Simulation model with a test and demand generator

The two channel safety system is modelled in figure 4.3. (a) illustrating it from a system point of view, while (b) shows how each of the system components "C1" and "C2" are modelled.

For a two channel system we start to see the advantages of simulating the model on different hierarchy blocks, and levels. The simulation model is at the same time in "Global", one of its sub-states in "C1" and one of its sub-states in "C2". The system states are obtained by the relevant combination of being in each of these blocks. This clearly demonstrates the advantage of Harel Statecharts compared to traditional state systems.



(a) Two-channel system with test and demand generators.

(b) Single system ”Ci” for i=1,2

Figure 4.3: Direct reliability model of two-channel system with its components.

Linking the Markov state diagram (figure 4.1) and the simulated model is not as trivial as for the single component system. Each of the Markov states are not directly seen through different blocks in this case. Instead, each of the Markov states can be programmed in "TwoChannelEvents" to get the frequency for each system state. This is a unique set of state combinations from "C1" and "C2". A closer description of this, and each block follow:

Global The ParentOfParents block. It contains the parameters λ and μ that are accessible for all the blocks in each of the two components, "C1" and "C2". Note that these variables are on component level. Since both components are assumed to have the same value, it is more convenient to place them here.

Demand It occurs on system level, and is generated by an exponential distribution throughout the whole simulation time. Even when the system is in state "Working". This is programmed so that the component is considered "down" with a mean time $1/\delta$. When a demand occurs the block is "up" for a very short time, and sends this information to the relevant components, "FailureUndetected" (through "TestDem"), "Repair" and "DemandDuringRepair".

There is not a point to connect it to for example "Working", since it will not give us any valuable information.

Test Each test is generated continuously throughout the whole simulation time with a constant interval on the system level, but is performed on component level. Even when the system is in a working or repair state. It is "down" for the length of test time, and when the operator performs a test the component is "up" for a very short time. This message is given to "TestDem".

TestDem This block takes all demands and proof-tests as inputs. The reason is to distinguish them from the interruption on the "Failure-Undetected" blocks, to know which one occurs first.

It has "1" as an output when a demand happens, and "2" when a test occurs.

C1 and C2 These are single components based on the same structure as the single system from section 3.1.

Working is identical to the one in the model of figure 3.2.

Listing 4.1: Calculations in ”FailureUndetected”

```
// triggerOut condition
if(interrupt==1 && TestDem==1) -> "DetectedByDemand"
else -> "DetectedByTest"
```

Listing 4.2: Calculations in ”Repair”

```
// when entered
repairTimeOver = currentTime + DExponential(mu);

// triggerOut condition
if(mu==0) -> "Working"
else if(interrupt==1) -> "DemandDuringRepair"
else -> "NoDemandDuringRepair"
```

FailureUndetected has two additional inputs, connected with ”TestDem”. The connector under the block has an ”interrupted” function. This means when the condition given for interruption is true, the simulation model leaves the block. ”TestDem” helps this block to know whether there is a demand, or a test interruption.

The duration of the block is dependent on whether a demand or a test occurs first, and that specific time. The algorithm is shown in listing 4.1.

DetectedByDemand and **DetectedByTest** are identical to the one in the model in figure 3.2.

The system is then sent to **Repair**. The block contains the local variable $t_{repairTimeOver}$, and the global parameter μ. ”TestDem” is not attached here, because only the information of a demand is relevant for this block in an online model. If it is interrupted by a demand, the rest of the repair is done in ”DemandDuringRepair”. The code for ”Repair” is in listing 4.2.

DemandDuringRepair is connected with ”Demand” through the ”interrupt” input. Now that every demand is generated, the demand is not calculated by the Poisson distribution as in section 3. The simulation will make a loop for every demand that occurs during the repair time, shown in listing 4.3.

The number of entries of the block is counted, which gives us the information of how many demands there have been during the whole

Listing 4.3: Calculations in "DemandDuringRepair"

```
// triggerOut condition
if(interrupt==1 && currentTime < repairTimeOver) -> "
   DemandDuringRepair"
else -> "Working"
```

Listing 4.4: Calculations in "SingleEvents"

```
if("Working"==TRUE) f = 5;
else if ("DemandDuringRepair==TRUE) f = 4;
else if ("Repair==TRUE) f = 3;
else if ("FailureUndetected==TRUE) f = 2;
else if ("DetectedByTest==TRUE) f = 1;
else if ("DetectedByDemand==TRUE) f = 0;
else f = f0;
```

simulation time.

Each relevant block in the single system is connected to a function block **SingleEvents**. This function block keeps track of when and where the single system is. Each block has its unique value (where f0 equals the previous output). The algorithm is shown in listing 4.4.

TwoChannelEvents The "SingleEvent" from "C1" and "C2" is connected to "TwoChannelEvents". Because of the Harel Statechart property, the simulation model is at the same time in "Global", "C1" and "C2". In that way, the "TwoChannelEvents" function block knows which output each of the "SingleEvent" blocks have at all times. In this way we get the relevant information about the two-channel model, as a multi-state system. This block gives the system states from figure 4.1.

The "TwoChannelEvents" block has the following states:
0: The system is detected by demand, and there is a hazardous failure on the system. One component is in "DetectedByDemand", while the other is in either "FailureUndetected" or "Repair". This is represented as state 6 in the Markov state diagram.
1: The system is detected by test. One component is in "DetectedByTest", while the other is in either "FailureUndetected" or "Repair". The system has been in state 3 or 5 in the Markov model, but a test detecting the failure provides it to go to state 6.
2: The system has demands during downtime (only relevant for online

Listing 4.5: Calculations in "TwoChannelEvents"

```
if((C1==0 && C2<=4)|| (C1<=4 && C2==0)) f = 0; //detected by demand
else if ((C1==1 && C2<=4) || (C1<=4 && C2==1)) f = 1; //detected by test
else if ((C1==4 && C2<=4) || (C1<=4 && C2==4)) f = 2; //demand during the system is down
else if ((C1==5 && C2<=5) || (C1<=5 && C2==5)) f = 3; //working
else f = f0;
```

model). One component is in "DemandDuringRepair" while the other is in "FailureUndetected" or "Repair". The system is in state 6 from the Markov model.
3: The system is working. At least one component is in "Working". This is represented by state 1, 2 and 4 in the Markov model.

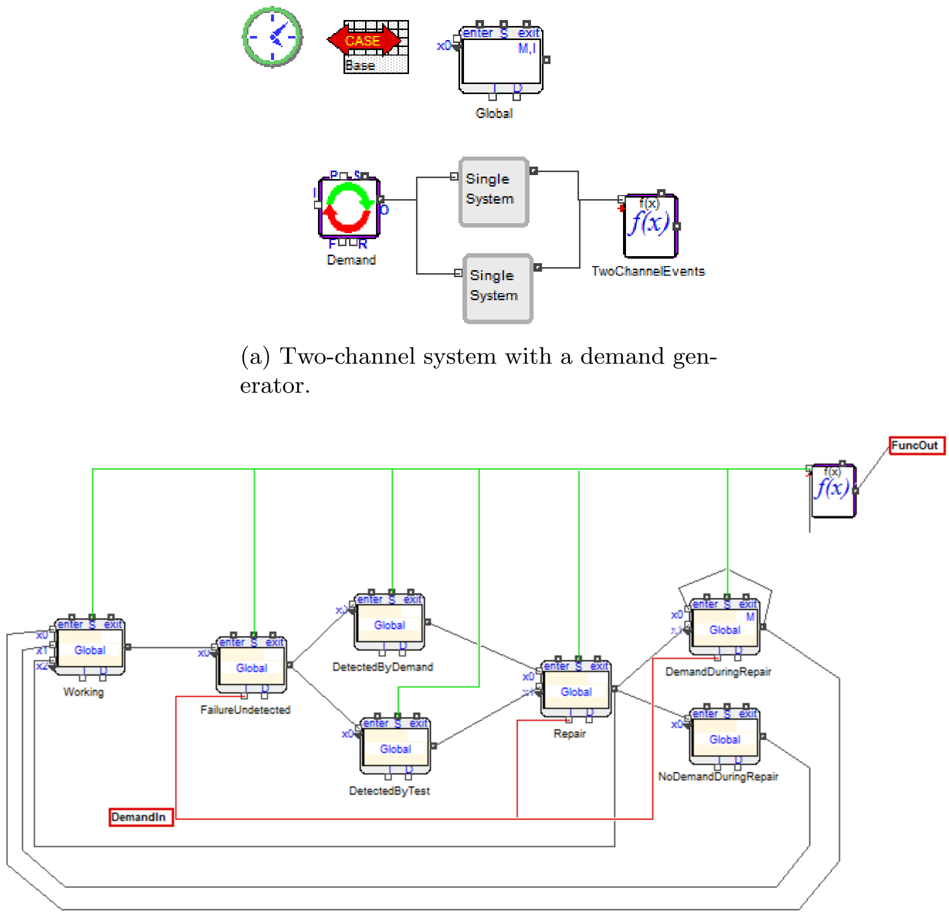
"TwoChannelEvents" gives information on how many times the system is in each of the states above, and the mean time for each of them. This is programmed in listing 4.5.

4.2.2 Simulation model with a demand generator

By first trying to handle the rare event problem for the low demand mode system, this model only consists of a demand generator. The simulation model is illustrated in figure 4.4.

The global parameters are now λ, μ and τ, and $t_{nextTestTime}$ (time for next test) is a global variable. τ and $t_{nextTestTime}$ are on a system level and only relevant for "FailureUndetected".

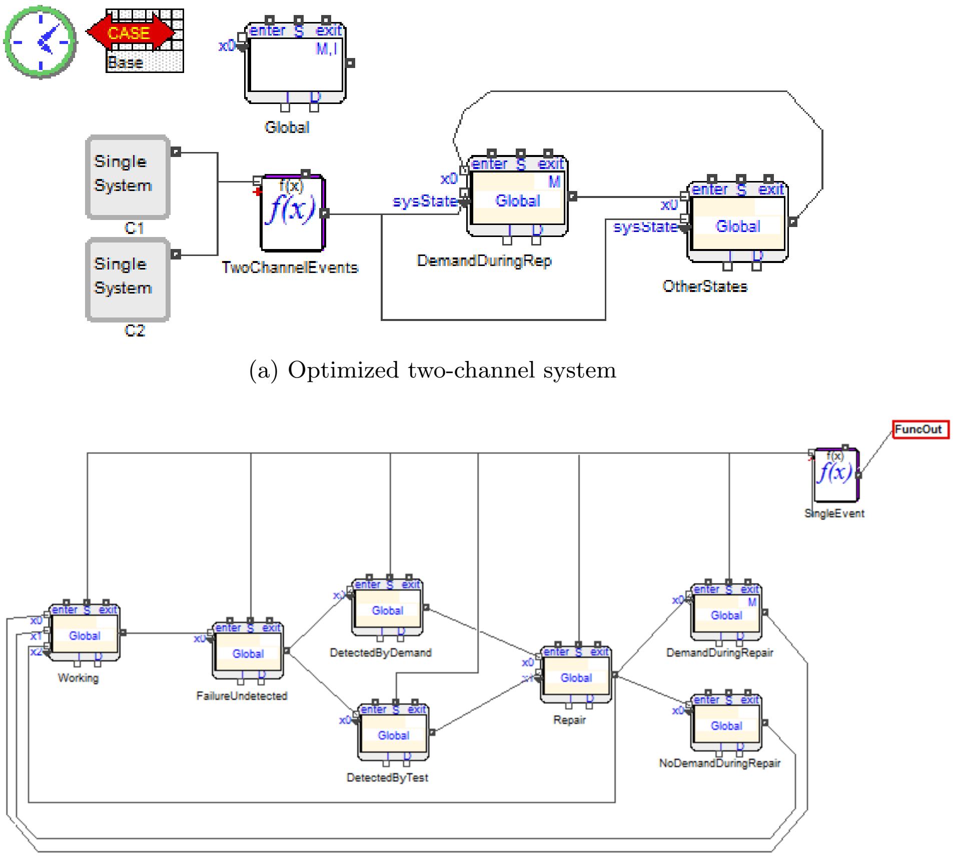
The main difference from the simulated model in the previous section is that tests are now only performed when necessary, following the formula given in equation 3.7. When a component fails the sub-state "FailureUndetected" contains the relevant information for calculating the time for the next test to appear. "FailureUndetected" is now only connected with "Demand" as a possible interruption variable. If "interrupted"==TRUE $< t_{nextTestTime}$, the simulation leaves the state for "DetectedByDemand".
The time to next test is calculated in "FailureUndetected" for each component, meaning the system only does proof-tests when at least one component is down.



(a) Two-channel system with a demand generator.

(b) Single system ”Ci” for i=1,2

Figure 4.4: Reliability model of two-channel system with its components. Demands are generated throughout the simulation time and tests are performed when needed.



(a) Optimized two-channel system

(b) Single system "Ci" for i=1,2

Figure 4.5: Reliability model of two-channel system optimized for a rare event problem.

The rest of the serial state system for each component is exactly the same as in the previous section, where the demand generator affects the same blocks according to the same assumptions.

The "TwoChannelEvents" is identical to the one in the previous model.

4.2.3 Optimized simulation model for a two channel system

Figure 4.5 illustrates the model that takes care of the rare event problem for both unnecessary tests and non-hazardous demands, in a similar way as for the single system in section 3.3.

The global parameters are in this case λ, μ, τ, δ and the global variables $t_{nextDemand}$ and $t_{nextTest}$.

Listing 4.6: Calculations in ”DemandDuringRep”

```
// condition for triggerInn
"TwoChannelEvents"==2

// when entering
previousTime = currentTime;

// triggerOut
"TwoChannelEvents"!=2

// expression on Exit from state
f = (currentTime - previousTime)*demand
f = DPoisson(f);
addDemand += f;
```

The components have access to the information needed to calculate when the next test or demand is. In that way tests and demands are now only calculated and will happen when at least one component is down. ”C1” and ”C2” are identical to the single system in the model in figure 3.1, except for the block ”DemandDuringRepair”. Demands during repair is on the system level, such that number of demands during repair cannot be calculated on the component level. Here ”DemandDuringRepair” only registers how many repairs where at least one demand happens. The ”SingleEvent” introduced in section 4.2.1 is included.

”TwoChannelEvents” is the same as in section 4.2.1.

The online case is dealt with differently than in the other models by adding the blocks ”DemandDuringRep” and ”OtherStates” connected to ”TwoChannelEvents”. As mentioned in section 3.3 demands during the repair time can be modelled as a homogeneous Poisson process [12] with rate $\delta \cdot t_{systemDownTime}$. $t_{systemDownTime}$ is the time from when a demand has occurred when the two channel system is in repair, until the first component is done with the repair. As long as the ”TwoChannelEvents” registers that the two channel system is in state 0, 1 or 3, the simulation will at the same time be in ”OtherStates”. When the output value from ”TwoChannelEvents” is 2, the simulation model will go to ”DemandDuringRep”. In this block ”addDemand” sums up all demands that are calculated for each simulation. This is programmed in the block ”DemandDuringRep”, shown in listing 4.6.

The results are plotted in figure 4.7.

4.3 Calculating the hazard rates

The offline model

The system will have a hazardous failure when one component is in "DetectedByDemand" and the other component is in either "UndetectedFailure" or "UnderRepair". Every time this combination occurs, the "TwoChannelEvent" registers output "0". The hazard rate is the number of times event "0" is registered divided by the simulation time. Or explained by the Markov state diagram when a demand appears:

$$\eta_{offline} = \frac{\text{\# events in state 3 and 5}}{\text{simulation time}}$$

The online model

From the simulation model the hazard rate is obtained by the number of events in "0", number of events in "2" and the Poisson calculated values in "DemandDuringRep", this is divided by the simulation time. From the Markov state diagram, we have the following when a demand occurs:

$$\eta_{online} = \frac{\text{\# events in state 3, 5 and 6}}{\text{simulation time}}$$

4.4 Results from the simulated models

4.4.1 The model without repair

Figure 4.6 illustrates the results for a 1oo2 system model without repair ($\mu = 0$) presented in section 4.1.1. The simulation model used for this model is the optimized model from section 4.2.3. The simulated hazard rate results approximate the asymptotes calculated in equations 4.6 and 4.7 very well. The hazard rate increases as more failures are detected by demands in the low demand region.

As introduced in section 1, the IEC standard suggests to use the PFD formula for a system that is defined to be in a low demand mode, and the PFH formula when it is in a high demand mode. For a system with $\delta \leq 20$ the PFD is a good estimate of the hazard rate. While for $\delta > 20$ we can see the effect of demands taking over as tests, which makes the system stronger and the hazard rate to decrease. For this region the PFH is an accurate estimate to use.

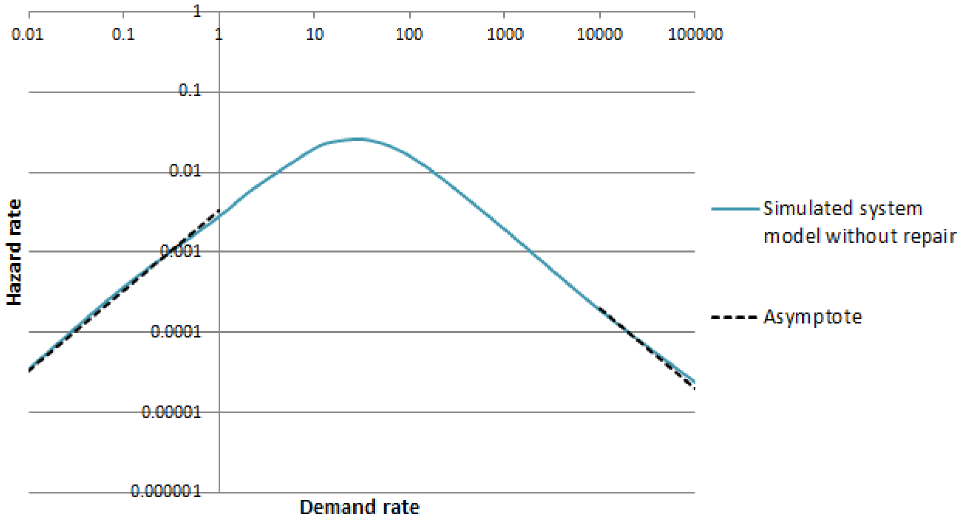


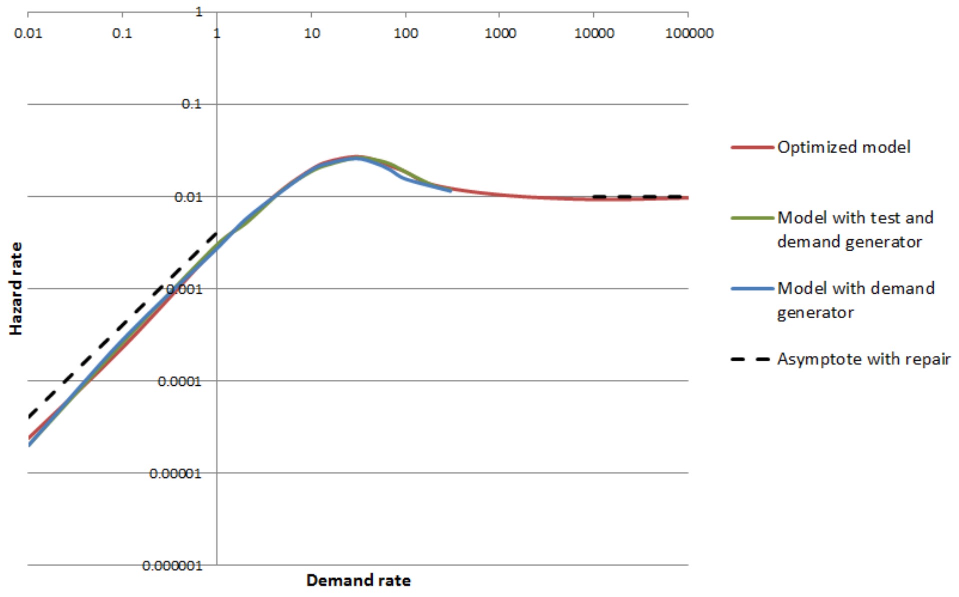
Figure 4.6: 1oo2 system model without repair

4.4.2 The distinction between the three methods of simulating the models

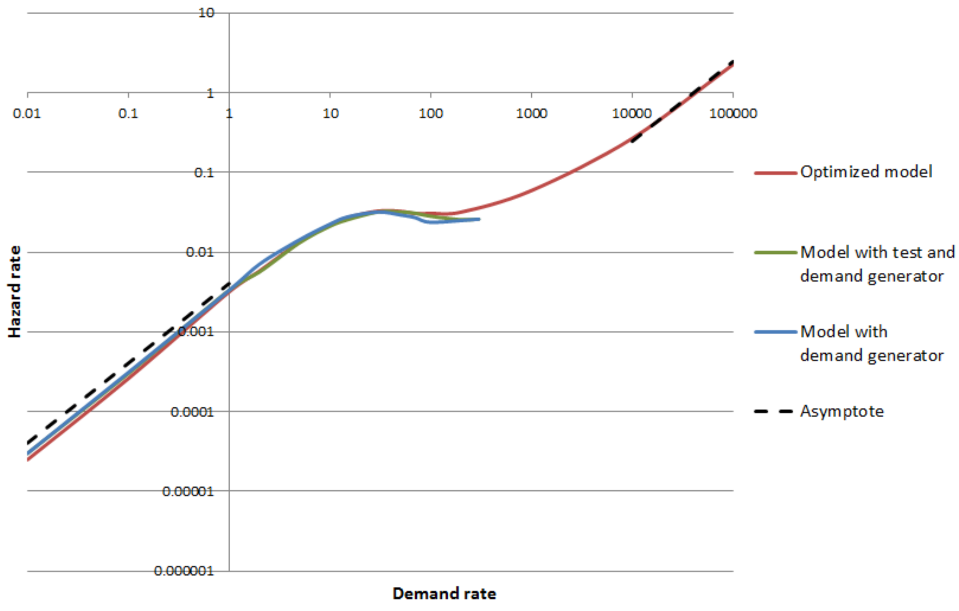
The calculations of the hazard rate for the three models give the following result shown in figure 4.7.

For $\delta < 100\lambda$ $(\lambda = 1)$ the three models are identical. When the demand rate increases further for the model generating every single demand and test, and the model only generating demands (section 4.2.1 and 4.2.2), the result starts to deviate for the online case. At $\delta = 300$ the time to run both the offline and online cases for the models generating tests and demands, and only demands takes respectively 10 minutes and 6 minutes. We can see very clearly that demands occurring before a failure is not significant for calculating the hazard rate since the results plotted for the optimized model is the same as for the models calculating all demands. By eliminating the tests and demands that are superfluous, and only calculating the demands and tests when needed, we get a much more powerful and time efficient model. The Harel State model simulated with discrete event Monte Carlo approach gives accurate results for very high demand rates, which means the rare event problem is solved in a satisfactory way. The simulation model for $\delta = 100000$ takes 23 seconds.

The simulated model is well approximated when considering the asymptote for both the low demand mode (equation 4.8) and the high demand mode system (equations 4.10 and 4.11) for the offline and online models.



(a) Offline model with repair



(b) Online model

Figure 4.7: Results for the model without repair and with repair for the three simulation models when $\lambda = 1$, $\mu = 200$ and $\tau = 0.1$.

For the intermediate demand region there is an unexpected behaviour for the two cases, offline and online. The offline model has a maximum point, while the online model also has a minimum point. The PFD which is suitable as a hazard rate estimate when the system is in a low demand mode, is only a good approximation when $\delta \leq 20$. For a $\delta \geq 200$ the PFH is a good estimate for the high demand mode system. However, when $\delta > 20$ and $\delta < 200$ there is a large deviance for both offline and online models. The PFD gives estimates of the hazard rate that are too conservative (plotted as "asymptote, low demand", equation 4.8), while the PFH is non-conservative (plotted as "asymptote, high demand", equations 4.10 and 4.11).

The unexpected behaviour of the hazard rate in the intermediate demand region can be explained. When there is a very low demand rate most of the failures are detected by proof-tests. At this point the undetected failure time can be very long, which means the downtime (undetected failure time + repair time) for one component is even longer. If there is a failure on the other component as well, it is more likely to be detected by a demand as the demand increases in frequency. The hazard rate increases.

At a certain point the demands occur with such a high frequency that the demands detects the majority of failures. The proof-tests are no longer necessary. The undetected failure time, and hence the downtime of a component, is drastically reduced. If there is a failure on one component now, it is detected within a shorter period of time and up running before a possible failure is detected for the other component. The hazard rate decreases, and a maximum point appears.

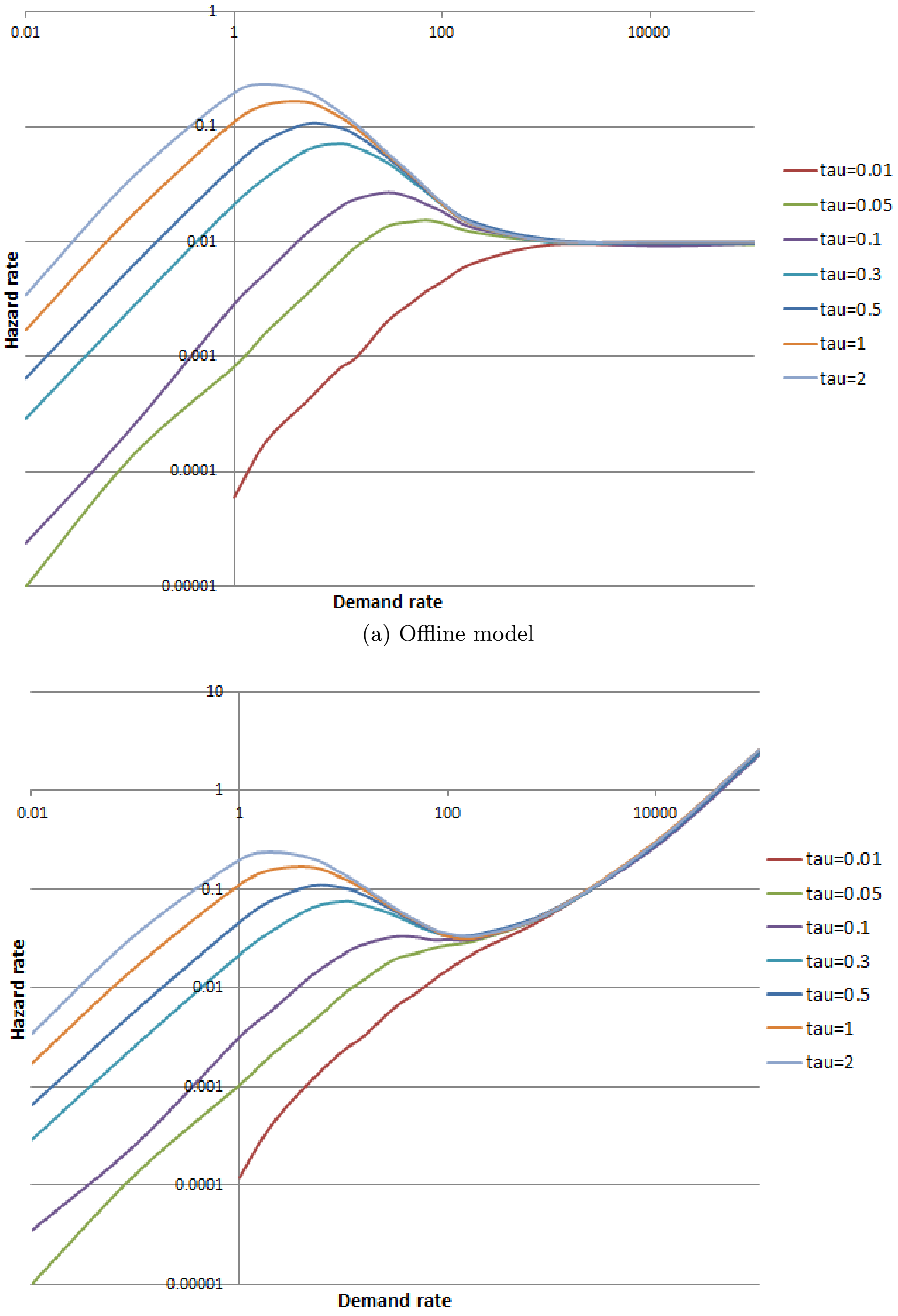
As the demand rate increases even further, the undetected failure time and the components downtime become very short. A failed component starts repair almost immediately. If a failure occurs in that period, a hazardous event happens with a large probability. The hazard rate keeps steady for the offline model, while for the online model when demands are still produced the hazard rate increases.

This will be better illustrated in the following sections, which gives the results of an analysis of various test intervals and repair rates on the two channel model.

4.4.3 Results with various test intervals

The results for various proof-test intervals are seen in figure 4.8 for the offline and online models.

The plots, both the offline and online case illustrate the change in be-



(a) Offline model

(b) Online model

Figure 4.8: Results for the optimized model with various test intervals, $\mu = 200$ and $\lambda = 1$.

haviour in the intermediate region as the length between tests varies. For $\tau > 0.05$ both the models have a maximum point, and the online model also has a minimum point.

The online plot illustrates that there is only the maximum point that is affected by the various lengths of proof-tests. We can see as the demand rate increases the maximum points move slightly to the right. We can conclude the maximum point is dependent on the relationship between proof-tests and the demand rate. For systems with a low demand rate, the proof test interval is significant for the risk level of the system. The hazard rates for high demand rates are clearly not dependent on proof-tests.

When $\tau = 2$ there are tests performed for about every second failure ($\lambda = 1$) on one component. As expected these results give a high hazard rate compared to models with a higher test frequency. This is also the case that has the most impact on the intermediate region, where the estimates between a PFD/PFH approach deviate the most compared to simulated results. The undetected failure time is now so long that the model has a lot to gain when demands happen with such a frequency to work as tests. The hazard rate decreases significantly.

As the frequency of proof-tests increases the intermediate demand region becomes narrower. The undetected failure time and the downtime for the system decreases, which is indicated by a decreased hazard rate. The right side of the maximum point becomes less steep because of the effect of demands taking over as tests becomes less significant. The maximum point moves because the demand rate has to be significantly larger than proof-tests for it to occur.

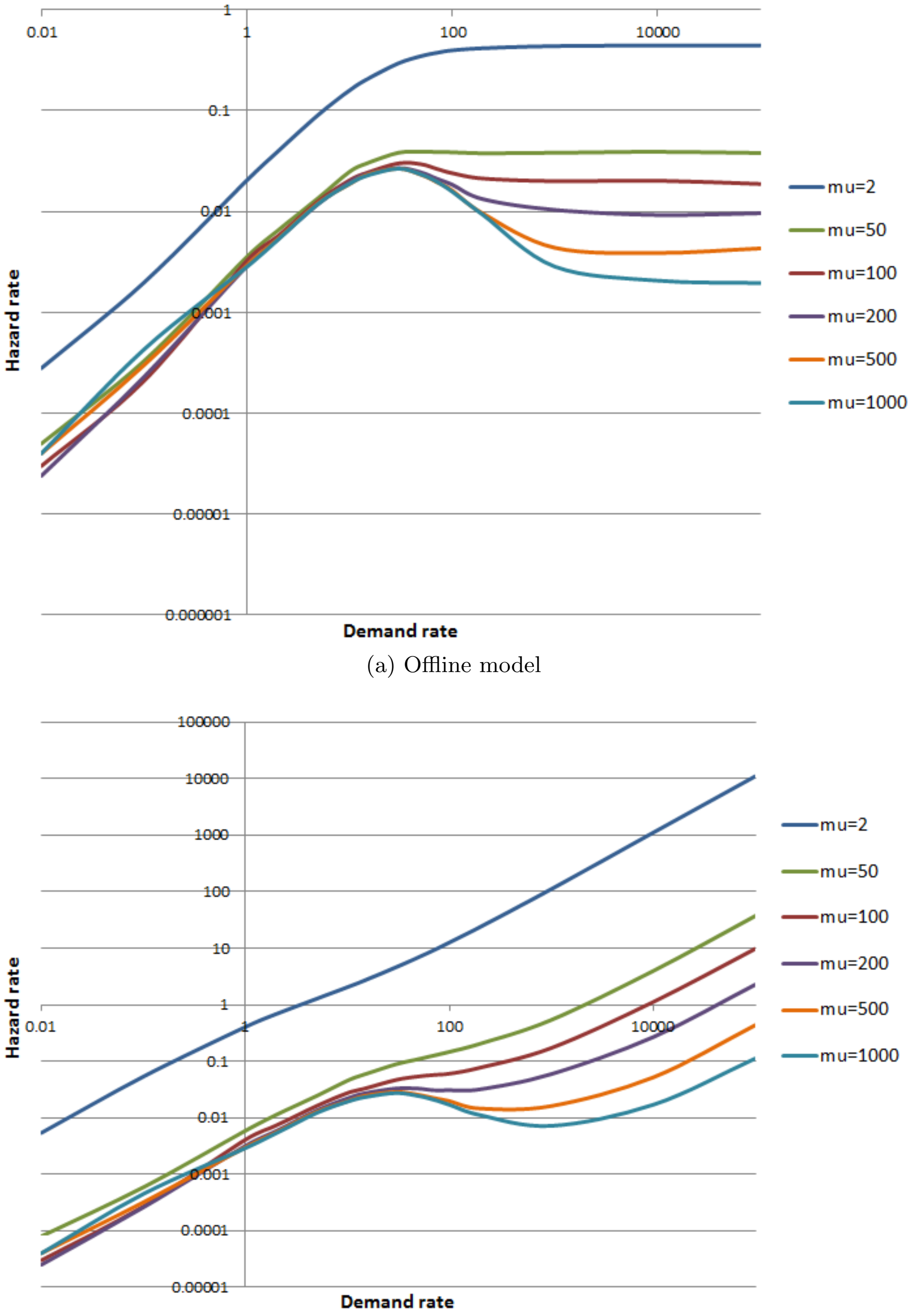
For all cases of τ the hazard rate approximates the same value when the demand rate is high.

When $\tau = 0.01$ there are 100 tests within one failure of a component. The undetected failure time is very short. The demand never has the chance to take over as tests before the effect of the repair takes place, which is shown through a steadily increasing hazard rate.

4.4.4 Results with various repair rates

The hazard rates for various demand rates with different values of the repair rate, μ (mean repair duration equals $1/\mu$) are illustrated in figure 4.9.

The plots illustrate that the repair rate gives an effect on the maximum point, and especially for the minimum point in the online model. When $\mu \geq 100$ the maximum and minimum points start to stand out, and there is an effect on the intermediate region. The repair rate has a small effect



(a) Offline model

(b) Online model

Figure 4.9: Results for the optimized model with various repair rates, $\lambda = 1$ and $\tau = 0.1$.

for low demand systems, compared to high demand systems where there are wide differences in the results for the hazard rates.

For higher repair rates, the repair time is shorter, and the system goes back to "perfect" condition quicker. The intermediate demand region becomes wider. Since the downtime for a failed component is short, the possibility of having a failure and a demand during this time is low. This supports a lower hazard rate. There is a stronger redundant system.

The minimum point shifts slightly to the right as the repair rate increases, while the maximum point does not. The minimum point occurs when there are demands during the systems downtime, which means that the demand rate must be significantly larger than the repair rate. This appears in the online plot when $\delta > \mu$.

When $\mu < 100$, the repair time is so long that it will for a lot of the time only have one component up running. The system has a weak redundancy, and there are no maximum or minimum points. The point where the demand takes over as tests seems to never have an impact, because the system is for a long time up with only one component. And there is a big chance of having a failure and a demand to occur when the other component is repaired. This makes a hazard rate increase when a demand detects the failure and not a test. The 1oo2 system approximates a 1oo1 system.

If we look at the plot for $\lambda = 1,\ \mu = 2$ in figure 4.9 (both models), a component is on average down for half of the time between failures. It is likely that a failure occurs on the other component while the first one is in the repair mode, and if it continues like that, the two channel system has the majority of time with only one working component. The hazard rate does not have any minimum or maximum points, as it is for a single channel model, and the hazard rate approximates λ.

The hazard rate for all spectres of the demand rate is therefore strongly dependent on the repair rate. The plot illustrates it is most significant for the intermediate and the high demand mode system.

4.5 Two channel model with an ergodic state

This model illustrated in figure 4.10 is based on the model from section 4. The restoration time a system experiences when a hazardous event happens until it is "as good as new" has been discussed with Oliveira. However, the assumption of a "perfect repair" when the system is down makes it possible to calculate the steady-state probabilities. It is now assumed that there has been a catastrophic accident which is impossible to recover from, which might be a more realistic assumption for many systems. This means state

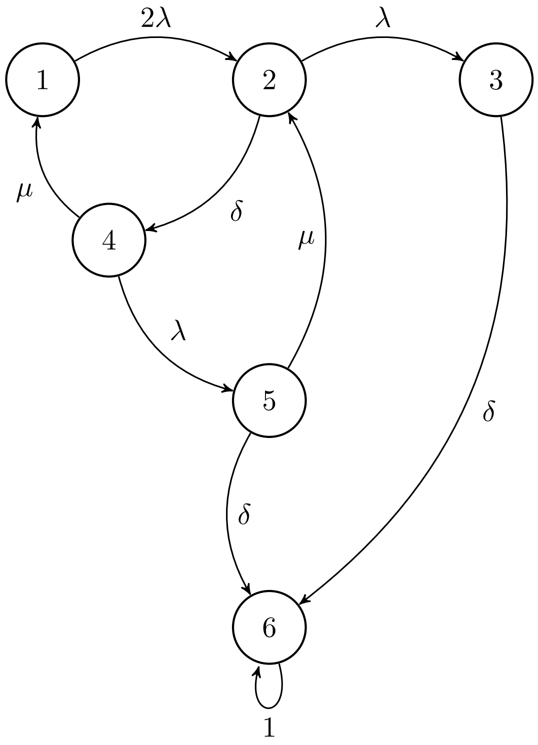


Figure 4.10: State diagram for the two channel repair system with a recurrent state

6 is an absorbing state, there is no transition out of the state.

The asymptote for a hazard rate in the low demand mode is like the previous model, equation 4.8. For a high demand mode system the hazard rate approximates 1, but an asymptote can be calculated for a given assumption of lifetime on the system. We assume the lifetime of the system equals the time the optimized model in section 4.2.3 is in state "Working". The asymptote of a hazard rate for a high demand mode can therefore be set to be the same as for this model, equation 4.10.

The simulation model is illustrated in figure 4.11. "Global", "C1", "C2" and "TwoChannelEvents" are identical as in section 4.2.3. When there is a hazardous failure (output from "TwoChannelEvent" equal 0), it goes to "HazardFailure" which sends this message as TRUE to the "interrupt" input in "Global". When the system has broken down, "Global" "resets" its variables ($t_{nextDemandTime}$, $t_{nextTestTime}$) so the two channel system is starting over again with both of the components in "Working".

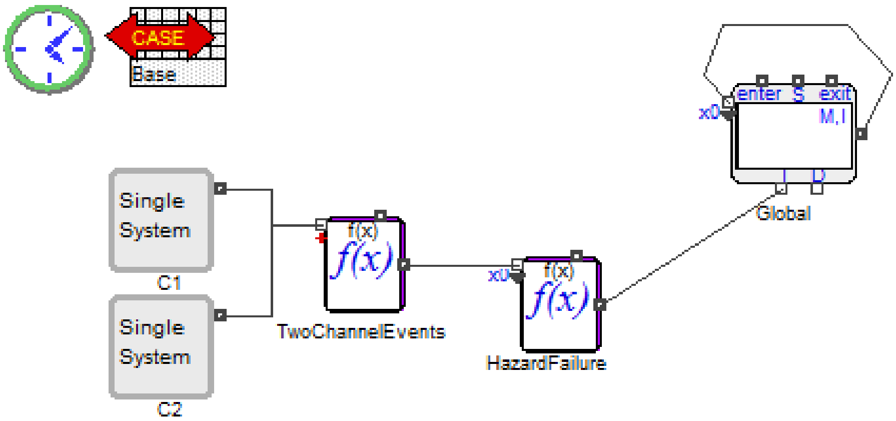


Figure 4.11: ExtendSim model of the 1oo2 model with recurrent state

The result is illustrated in figure 4.12 with the hazard rate for the offline model and its asymptotes. This model suits the asymptote perfectly for both low and high demand rates. For a model with low demand rates it fits the asymptote better than the model with a "perfect repair" assumption (section 4.2.3). From $\delta > 1$ the two models are more or less identical. We can conclude that the renewal rate is only significant for systems with a very low demand rate. However as the demand rate increases, a system that breaks down completely (replaced by a new system) and a system with perfect repair are not significant when calculating the hazard rate.

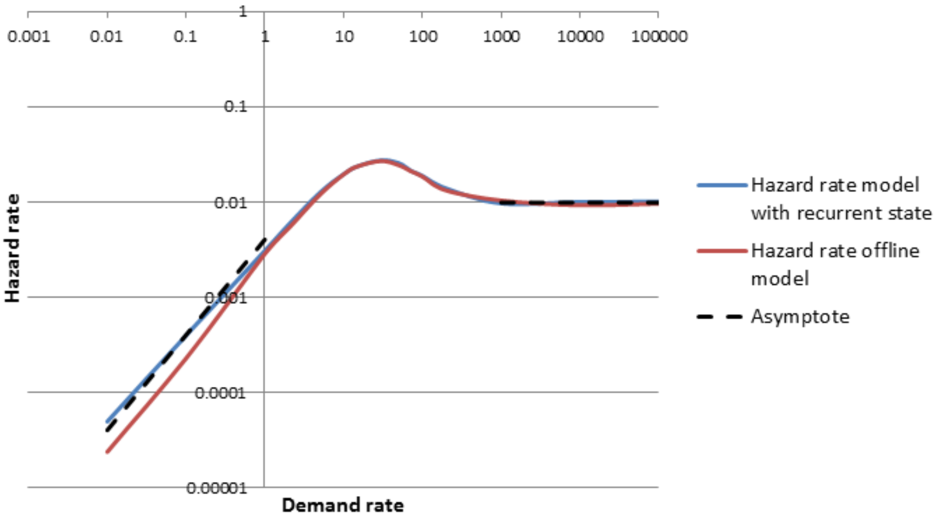


Figure 4.12: Results for the 1oo2 model with a recurrent state for various demand rates. $\lambda = 1$, $\mu = 200$ and $\tau = 0.1$.

5.

A 2oo3 safety system model

The 1oo2 safety system model from section 4 can easily be extended to a 2oo3 model with the simulation program ExtendSim. For a 2oo3 model at least two components have to work for the system to be functional.

The PFD for a 2oo3 system is [10]:

$$\mathrm{PFD}_{2oo3} = (\lambda\tau)^2 \tag{5.1}$$

The asymptote of the hazard rate for the 2oo3 system in low demand mode is:

$$\eta_{2oo3,low} = \delta \cdot (\lambda\tau)^2 \tag{5.2}$$

We do not account for the repair rate since it will not have a big impact when the demand rate is low.

To calculate the asymptote for the high demand rate case is a more complex Markov equation to solve when systems contain many components. However, to simulate the model with a Harel Statechart it is just to add an extra component and change the system logic somewhat.

We have simulated a direct model with test and demand generators, similar to the approach for the 1oo2 model in section 4.2.1 and the optimized model, where only the necessary tests and demands are calculated in section 4.2.3.

Figure 5.1 (a) illustrates the direct way of simulating it. The blocks contain the same calculations and work in the same way as explained in section 4.2.1. $C3$ is identical to $C1$ and $C2$. We have changed the "SingleEvents" function block in "Ci" for $i = 1, 2, 3$ to a more suitable function in "SystemEvents" to a 2oo3 model. The algorithm is now shown in listing 5.1.

The "SystemEvents" block contains information of which states the three components are in at all times, and gives the system states in a similar way

Listing 5.1: Calculations in "SingleEvents"

```
if("Working == TRUE) f = 10;
else if ("DemandDuringRepair" == TRUE) f = 6;
else if ("Repair" == TRUE) f = 3;
else if ("FailureUndetected" == TRUE) f = 2;
else if ("DetectedByTest" == TRUE) f = 1;
else if ("DetectedByDemand" == TRUE) f = 0;
else f = f0;
```

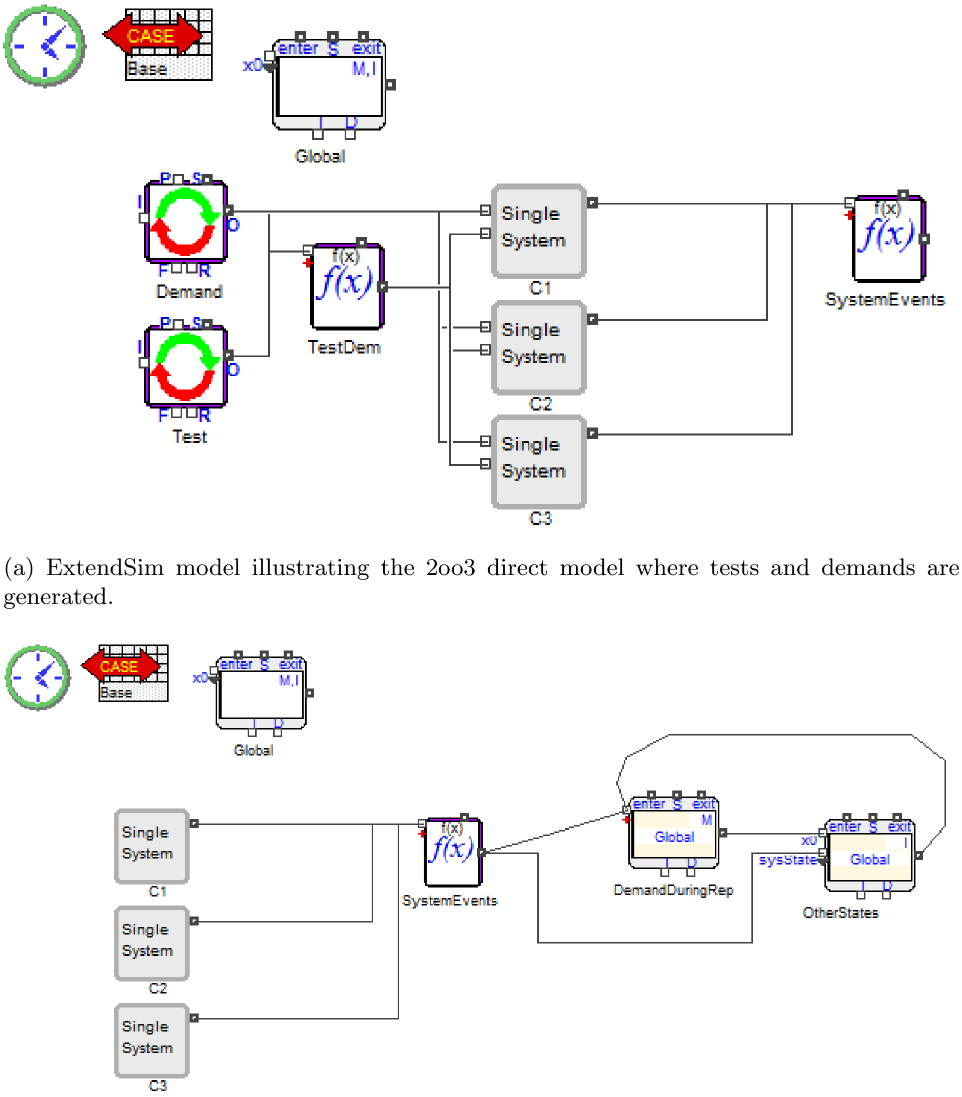
Listing 5.2: Calculations in "SystemEvents"

```
f = C1+C2+C3
if((f <= 16 && C1==0) || (f <= 16 && C2==0) || (f <= 16 &&
   C3==0)) f=0; //hazard event
else if((f <= 17 && C1==0) || (f <= 17 && C2==0) || (f <= 17
   && C3==0)) f=1; //detected by test
else if((f <= 18 && C1==0) || (f <= 18 && C2==0) || (f <= 18
   && C3==0)) f=2; //repair
else if((f <= 19 && C1==0) || (f <= 19 && C2==0) || (f <= 19
   && C3==0)) f=3; //demand during downtime
else if(f >= 20) f = 4; //working
else f = f0;
```

as "TwoChannelEvents" in section 4.2.1. The simulation model is now in "Global", a serial-block in "C1", "C2" and "C3". The formula is described in listing 5.2.

Figure 5.1 (b) illustrates the ExtendSim model for the 2oo3 system for the optimized case. "C1", "C2" and "C3" are the same as the components for the 1oo2 simulated model in section 4.2.3. "DemandDuringRep" and "OtherStates" are also identical to the ones in the 1oo2 model. "SystemEvents" is the same for the direct model, listing 5.2.

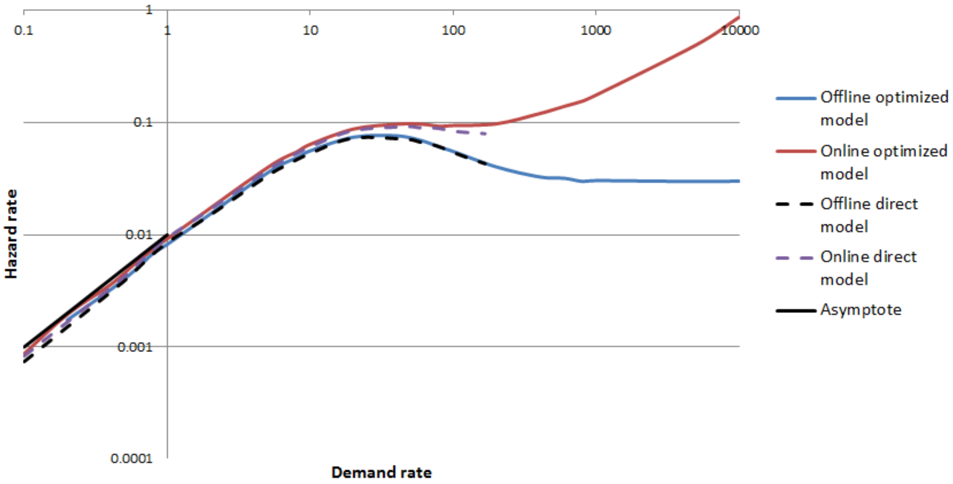
The low demand rate can easily be verified by the asymptote in equation 5.2. Since it is more complex to calculate the asymptote for a high demand mode system, we compare the results of the hazard rates for the optimized model to the direct model. This model is the correct one since it takes into account all tests and demands that appear. Figure 5.2 (a) illustrates that the optimized model is more or less identical to the results for the direct model until the processing time for the direct model becomes too long, which is around $\delta = 150$. The hazard rate follows the same pattern



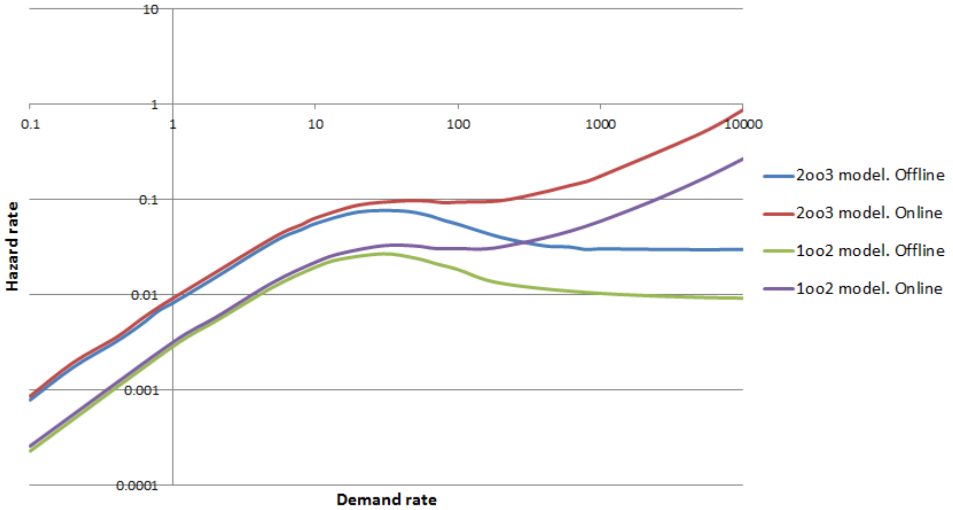
(a) ExtendSim model illustrating the 2oo3 direct model where tests and demands are generated.

(b) ExtendSim model illustrating the 2oo3 optimized model.

Figure 5.1: ExtendSim models of a 2oo3 system.



(a) The results of the hazard rate for a 2oo3 model with the direct and the optimized model for offline and online case.



(b) Comparison of the hazard rate between a 1oo2 and a 2oo3 system hazard rate.

Figure 5.2: 2oo3 system hazard rate results. $\lambda = 1$, $\tau = 0.1$ and $\mu = 200$.

as the optimized model when the demand rate gets larger than the direct model. We interpret these results as accurate for high demand rates. This applies for both the offline and online model.

Figure 5.2 (b) compares the result with the 1oo2 system hazard rate for the offline and online model. As we can see the hazard rate for the 2oo3 model is higher, but parallel. This is logical since the 2oo3 system has lower availability.

The behaviour of the hazard rate in the intermediate demand region is the same as for the 1oo2 system, it has a maximum point for both the offline and online model, as well as a minimum point for the online model. The effects of the relation between the demand rate, proof-tests and the repair rate are the same as for the 1oo2 system, which contributes to the maximum and minimum points.

The maximum point occurs because of a change in the redundant system, demands takes over for proof-tests, which makes the redundancy stronger and the hazard rate decreases. While the minimum point arises for the online model when the demand rate exceeds the repair rate, and more demands happens during the system's downtime, which increases the hazard rate. This is explained more thorough in section 4.4.2.

Therefore the conclusion of PFD being too conservative and the PFH non-conservative is valid for the intermediate demand mode in a 2oo3 system as well.

5.1 2oo3 model with repair crew

We can extend the 2oo3 model to a more realistic model. The industry has limited resources to repair their systems, or it is common that a broken system has to wait for the right parts to start the repair.

Figure 5.3 illustrates the simulation model of this case. "ResourcePool" and 20 of the "Component" blocks are added to the 2oo3 model from figure 5.1 (b).

"ResourcePool" contains the number of specified repair crews. In this case, the model has one repair crew. This means that only one failed component can be repaired at a time. If other components fail during the time the repair crew is occupied, the component has to wait to be repaired and its downtime gets longer.

The 20 components run at the same time as in the 2oo3 model. They do not interact with the 2oo3 system, but are components the repair crew is in charge of when failing. The components have the same failure and repair rates as the 2oo3 system, and are either "up" or "down".

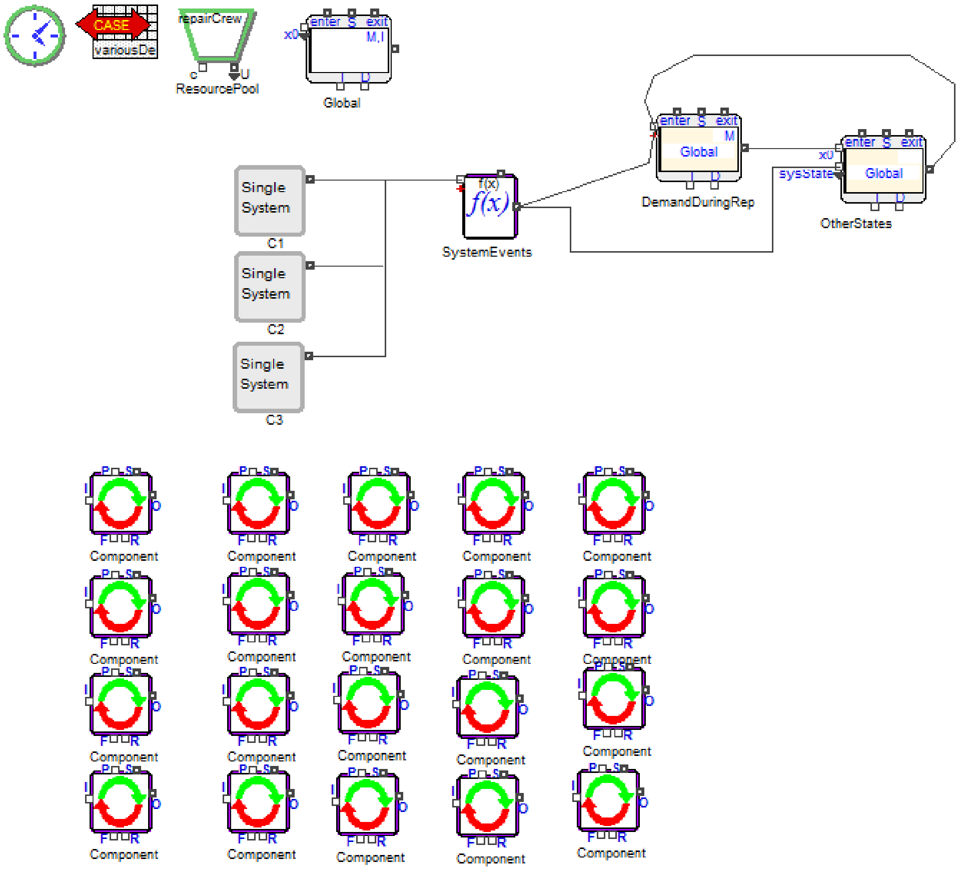


Figure 5.3: ExtendSim model of a 2oo3 system with components and repair crew

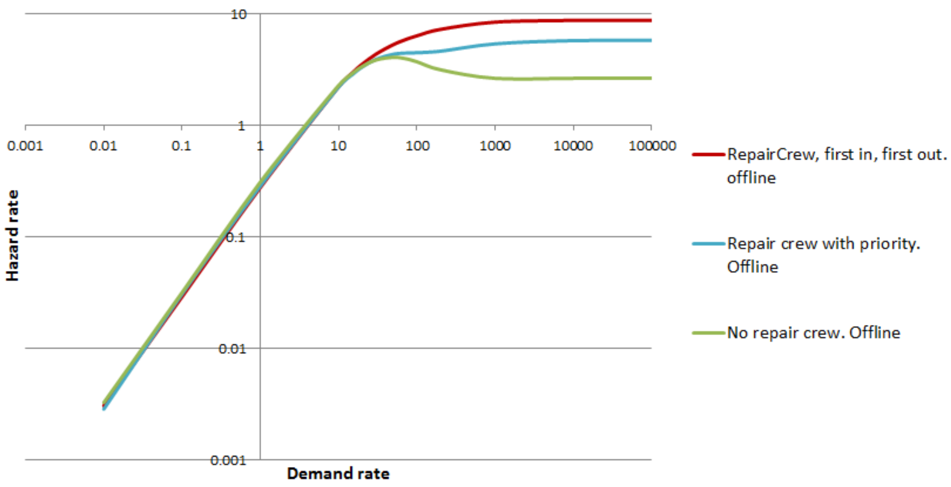


Figure 5.4: 2oo3 model with repair crew versus not repair crew. $\lambda = 10$, $\mu = 200$ and $\tau = 0.1$.

We have simulated three cases for an offline model where $\lambda = 10$, $\mu = 200$ and $\tau = 0.1$:

Case 1: "First in-first out". The components are repaired in the same order that they fail.

Case 2: Prioritized repair. The 2oo3 components ("C1", "C2" and "C3") are prioritized to be repaired before a "Component" block after a failure.

Case 3: No repair crew. All failed components are repaired immediately when the failure is detected, no matter how many other components are repaired at the same time.

The results plotted in figure 5.4 illustrate the difference between an offline 2oo3 model with the three cases explained above.

We can see that the model with no repair crew, case 3, has a maximum point. Case 2, "First in - first out" shows a large deviance from the model without repair crew. It does not have a maximum point. This is explained by the fact that the failed components in the model with repair crew have a much longer downtime. It can be compared to the effect of the components having a very long repair rate, explained in section 4.4.4. Case 3, the prioritized repair model, gives a bit lower hazard rate, but still no maximum point. The downtime for each failed component is not as long for the three prioritized system components, which means it has a slightly stronger redundant system.

6.

SIS model with demand duration

We are now looking at how the hazard rate for a single system is behaving when it has a demand duration, compared to the models above where a demand is just a spike. There are also two types of failures, a dangerous undetected and detected failure. The following model is from [2]. We will also compare the results of the hazard rate that we think are calculated wrongly by [2] with the corrected calculations and simulations.

The Markov state diagram of the model is illustrated in figure 6.1. The transition rates are described in table 6.1 with its parameters used in this thesis.

Description of each state:
State 5: The system is working. (Initial state)
State 4: Safe state.
State 3: Functioning. The system has a demand duration.
State 2: DD-failure.
State 1: DU-failure.
State 0: D-failure (DU or DD). Hazardous events,

where DD-failure (dangerous detected failure) and DU-failure (dangerous undetected failure) are explained in section 1. D-failure is dangerous failure.

Assumptions of the model [2]:

- The system consists of a "safe state", which cannot lead the system to a hazardous event. The system is in this state in case of spurious activation. The transition rate from state 4 to state 5, μ_s, is calculated as $1/\text{MTTR}_s$, where MTTR_s is the mean restoration time.
- All transition rates are constant in time.
- A repair starts immediately when a DD-failure occurs. The DD repair

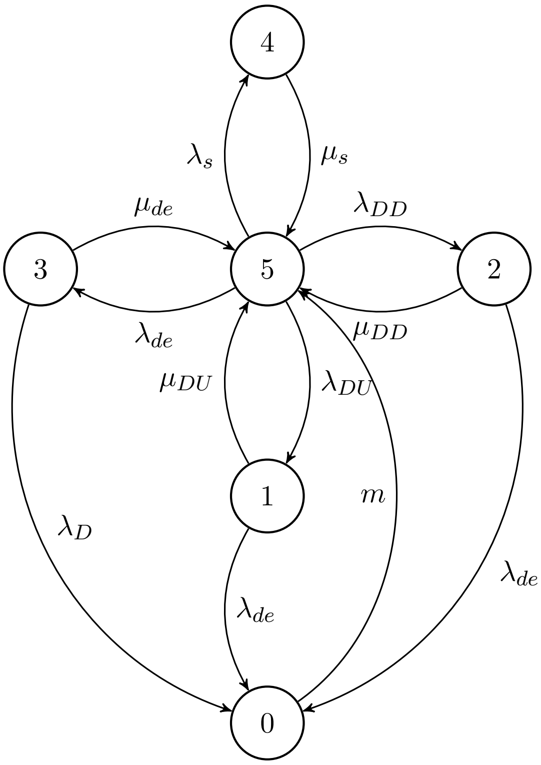


Figure 6.1: Markov transition diagram

Table 6.1: Description of the transition rates, system specifications and its values (in hours)

Transition rate	Description	Value
λ_{de}	Demand rate	varies
λ_s	Transition rate to safe state	$5 \cdot 10^{-7}$
μ_s	Restoration rate	$2 \cdot 10^{6}$
τ_{de}	Mean demand duration	0.2
μ_{de}	Demand duration rate	5
λ_{DD}	DD-failure rate	$3 \cdot 10^{-7}$
λ_{DU}	DU-failure rate	$5 \cdot 10^{-7}$
$\lambda_D (= \lambda_{DU} + \lambda_{DD})$	Dangerous failure rate	$8 \cdot 10^{-7}$
τ_{DU}/τ_{DD}	Mean repair time (DU and DD)	8
μ_{DD}	DD repair rate	0.125
μ_{DU}	DU repair time	$4.5496 \cdot 10^{-4}$
τ	Proof-test	4380
m	Renewal rate	0.00595
τ_m	Mean renewal time	168
τ_s	Mean restoration time	24

rate is

$$\mu_{DD} = \frac{1}{\mathrm{MTTR}_{DD}}.$$

- A repair starts after a DU-failure. The DU repair rate is

  $$\mu_{DU} = \frac{1}{\frac{\tau}{2} + MTTR_{DU}},$$

  where $\tau/2$ is the average time before the failure is detected.

  Note that this is a rough estimation that is wrong. Adding two expected downtimes $\tau/2$ and MTTR_{DU}, where we assume each random variable is exponentially distributed, is correctly assumed to be the expected downtime after a DU-failure. But it is not correct that taking the inverse of this expression gives the μ_{DU} of an exponential distribution. From probability theory we have that the sum of two variables that are exponentially distributed with a common scale parameter is gamma distributed with the common scale parameter and shape parameter equal to 2.

- The time between demands are exponentially distributed with parameter λ_{de}. The mean duration for a demand is hence $1/\mu_{de}$.

- The renewal time is exponentially distributed with rate m.

- No more than one hazardous event can occur during a test interval.

6.1 Asymptote

The PFD with a dangerous detected and undetected failure is [2]:

$$PFD = 1 - e^{-\lambda_D t_{CE}}, \tag{6.1}$$

where t_{CE} is the average downtime after a dangerous failure.

$$t_{CE} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{\tau}{2} + MTTR_{DU}\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR_{DD} \tag{6.2}$$

The asymptote of the hazard rate for a low demand mode system becomes:

$$\eta_{PFD} = \lambda_{de} \cdot (1 - e^{-\lambda_D t_{CE}}) \tag{6.3}$$

The asymptote of the hazard rate for the high demand mode is

$$\eta_{high} = \lambda_D$$

This is according to what discussed in section 1, when the system is a single channel system the hazard rate approximates the dangerous failure rate.

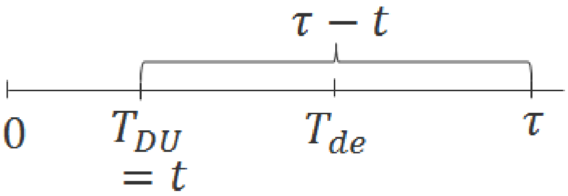


Figure 6.2: $Pr(T_{DU} < T_{de} \leq \tau)$

6.2 Calculation of the hazard rate by scenario-based formulae

Jin [2] introduces a different way of calculating the hazard rate for a safety system that suits both low- and high demand modes. We believe that the calculations of the probabilities for each of these scenarios are wrong. We will look at the solutions for both cases, as well as results obtained from a simulation model. The calculations are done in appendix C.

Each scenario is quoted from [2] below:

Scenario 1

A DU-failure occurs at time t. The demand happens after the DU-failure, but before the next scheduled functional test at time τ. In this case, an incorrect probability of having a hazardous event with scenario 1 is

$$P_1 = Pr(T_{DU} < T_{de} \leq \tau) = \int_0^\tau \lambda_{DU} e^{-\lambda_{DU} t} \left(1 - e^{-\lambda_{de}(\tau - t)}\right) dt \qquad (6.4)$$

Scenario 1 ($Pr(T_{DU} < T_{de} \leq \tau)$) is correctly illustrated in figure 6.2. This is according to the description above from [2] which gives the impression that there is only one demand happening during a test interval (quoting: "The demand occurs after the DU-failure..."). A test interval with only one demand is only valid for a very low demand rate case, but not for a high demand case. The correct occurrence of a demand is between time t and τ. By accounting for the probability of a demand to happen after time $t = T_{DU}$, we include:

$$Pr(t < T_{de}) = e^{-\lambda_{de} t}.$$

Since for T_{DU} happening at time t we have:

$$Pr(t < T_{de} \leq \tau) = Pr(t < T_{de}) \cdot Pr(T_{de} \leq \tau - t)$$

The correct probability of scenario 1 is then:

$$P_1 = Pr(T_{DU} < T_{de} \leq \tau) = \int_0^\tau \lambda_{DU} e^{-\lambda_{DU} t} e^{-\lambda_{de} t} (1 - e^{-\lambda_{de}(\tau - t)}) \; dt \quad (6.5)$$

Equation 6.4 does not state the same. It says that a DU-failure happens at time t and demand occurs between time 0 and $\tau - t$. By having $Pr(t < T_{de}) = 1$, we can interpret scenario 1 in the same way as we have done with the other safety systems introduced in this thesis. This scenario also has a rare event problem, discussed in section 3.3. When a DU-failure happens we know the length $\tau - T_{DU}$, and a demand needs to occur within this length of time for a hazardous event to apply. At this time the memoryless property for a demand comes in, and the next demand is calculated. If it occurs within the length of $\tau - T_{DU}$ a hazardous event apply. However, making the assumption that a demand survives time t overestimates the hazard frequency for the scenario.

The adjusted expression of probability that represents equation 6.4 is

$$Pr(T_{DU} = t \, \cap \, 0 < T_{de} \leq \tau - t) = \int_0^\tau \lambda_{DU} e^{-\lambda_{DU} t} \left(1 - e^{-\lambda_{de}(\tau - t)}\right) \; dt \quad (6.6)$$

Scenario 2

A DD-failure occurs at time t. A demand occurs after the DD-failure and before the failure is repaired, which happens before τ. The repair starts immediately after a DD-failure. In this case, the probability of having a hazardous event is:

$$\begin{aligned} P_2 =& Pr(T_{DD} < T_{de} < T_{DD} + \tilde{T}_{DD} \leq \tau) \\ =& \int_0^\tau \lambda_{DD} e^{-\lambda_{DD} t} Pr(t < T_{de} < t + \tilde{T}_{DD} \leq \tau) \; dt, \end{aligned} \quad (6.7)$$

where $\tilde{T}_{DD}$ is the repair time of a DD-failure, and an incorrect probability is given by

$$Pr(t < T_{de} < t + \tilde{T}_{DD} \leq \tau) = \int_0^{\tau - t} (1 - e^{-\lambda_{de} u}) e^{-\mu_{DD} u} \; du \quad (6.8)$$

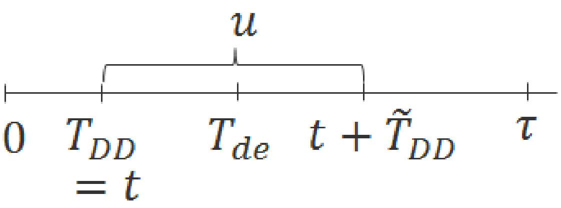


Figure 6.3: $Pr(t < T_{de} < t + \tilde{T}_{DD} \leq \tau)$

$Pr(t < T_{de} < t + \tilde{T}_{DD} \leq \tau)$ in equation 6.8 is correctly illustrated in figure 6.3. The equation on the other-hand implies that a demand occurs between time 0 and u, and the repair is finished after time u. When we solve the equation, and calculate it for various scales, we obtain different results. This should not be possible for a probability, and indicates that the stated equation is wrong. Our calculations give expressions that do not cancel out the difference. This is because [2] calculates the probability of having a repair to survive time u, and not to be finished at time u. The μ_{DD} in front of $\mu_{DD}e^{-\mu_{DD}u}$ is missing.

Equation 6.8 and figure 6.3 are supposed to state the probability of a demand to occur after the DD-failure at time t: $e^{-\lambda_{de}t}$, but before an additional time u, when the repair is finished.

$$Pr(t < T_{de} < t+\tilde{T}_{DD} \leq \tau) = Pr(T_{de} > t) \cdot Pr(T_{de} \leq u) \cdot Pr(\tilde{T}_{DD} = u \leq \tau - t),$$

where u equals the time of repair, and the correct probability must be:

$$Pr(t < T_{de} < t+\tilde{T}_{DD} \leq \tau) = \int_0^{\tau - t} e^{-\lambda_{de}t}(1 - e^{-\lambda_{de}u})\mu_{DD}e^{-\mu_{DD}u}\, du \quad (6.9)$$

However, we have a similar type of a rare event problem here as for scenario 1, where a demand leads to a hazardous event when happening within time u, which is $\leq \tau - T_{DD}$. This scenario is also too conservative, because of the assumption of having a demand to survive the time where a DD-failure occurs.

The adjusted probability for scenario 2 when the demand is exponentially distributed is therefore:

$$\begin{aligned} &Pr(T_{DD} = t \,\cap\, 0 < T_{de} \leq u \,\cap\, \tilde{T}_{DD} = u \leq \tau - t) \\ &= \int_0^{\tau} \lambda_{DD}e^{-\lambda_{DD}t}\, dt \times \int_0^{\tau - t} (1 - e^{-\lambda_{de}u})\mu_{DD}e^{-\mu_{DD}u}\, du \end{aligned} \quad (6.10)$$

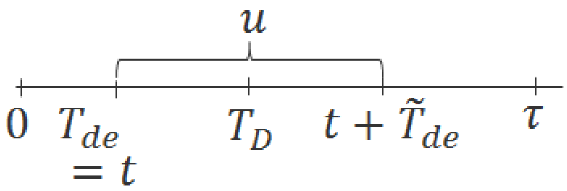


Figure 6.4: $Pr(t < T_{de} < t + \tilde{T}_{DD} \leq \tau)$

Scenario 3

A demand occurs at time t. A dangerous failure occurs during the demand, but before the end of demand which happens before τ. Hence the probability of having a hazardous failure is

$$\begin{aligned} P_3 =& Pr(T_{de} < T_D < T_{de} + \tilde{T}_{de} \leq \tau) \\ =& \int_0^{\tau} \lambda_{de} e^{-\lambda_{de} t} Pr(t < T_D < t + \tilde{T}_{de} \leq \tau)\ dt, \end{aligned} \tag{6.11}$$

where $\tilde{T}_{de}$ is the demand duration, and an incorrect probability is given by

$$Pr(t < T_D < t + \tilde{T}_{de} \leq \tau) = \int_0^{\tau - t} (1 - e^{-\lambda_D u}) e^{-\mu_{de} u}\ du \tag{6.12}$$

$Pr(T_{de} < T_D < T_{de} + \tilde{T}_{de} \leq \tau)$ from equation 6.11 is correctly illustrated in figure 6.4.

There is a similar probability error here as for scenario 2, where μ_{de} is missing in front of $\mu_{de} e^{-\mu_{de}}$.

Since

$$Pr(t < T_D < t + \tilde{T}_{de} \leq \tau) = Pr(t < t_D) \cdot Pr(T_D \leq u) \cdot Pr(\tilde{T}_{de} = u \leq \tau - t),$$

where u is the time of demand duration, the correct probability is:

$$Pr(t < T_D < t + \tilde{T}_{de} \leq \tau) = \int_0^{\tau - t} e^{-\lambda_D t} (1 - e^{-\lambda_D u}) \mu_{de} e^{-\mu_{de} u}\ du \tag{6.13}$$

Here a dangerous failure to occur during demand is a rare event problem. Since the D-failure is exponentially distributed, the same approach is valid to use as for scenario 1 and 2. The adjusted probability is:

$$\begin{aligned}&Pr(T_{de} = t\ \cap\ 0 < T_D \leq u\ \cap\ \tilde{T}_{de} = u \leq \tau - t)\\&= \int_0^{\tau} \lambda_{de} e^{-\lambda_{de} t}\ dt \times \int_0^{\tau - t} (1 - e^{-\lambda_D u}) \mu_{de} e^{-\mu_{de} u}\ du\end{aligned} \tag{6.14}$$

Calculation of the hazard rate

Since a hazard only takes place when a demand occurs during the system's downtime, or when a failure occurs during demand the hazard rate is,

$$\mathrm{HEF}(t) = \lambda_{de} \cdot P_1(t) + \lambda_{de} \cdot P_2(t) + \lambda_D \cdot P_3(t),$$

where P_i for $i = 1, 2, 3$ are the long-run probabilities that the system ends in state i from figure 6.1.

Jin [2] uses the scenarios above to define the hazard rate to be:

$$HEF = \frac{P_1 + P_2 + P_3}{\tau}, \tag{6.15}$$

From now on the scenario-based formula containing the equations 6.6, 6.10 and 6.14 is called scenario-based adjusted method. While the equations 6.5, for scenario 2 including equation 6.9 and for scenario 3 with equation 6.13 is called scenario-based probability method.

6.3 Simulation models of the correct probability scenarios

All three scenarios are based on the same ExtendSim model as illustrated in figure 6.5, but with distinctive parameters and formulas. Each scenario model is explained in more detail below.

Scenario 1

We want to simulate $Pr(T_{DU} < T_{de} \leq \tau)$, by the proportion of events where a demand occurs after a DU-failure within a test interval. This is according to figure 6.2. For this case we have to calculate T_{DU} and T_{de} in time 0, to know where they are in relation to each other within the test interval.

The parameters in "Scenario" are: λ_{de}, λ_{DU} and τ, while "nextDemand", "nextDUFailure" and "nextTest" are variables. The simulation code is in listing 6.1, where we do a rejection sampling:

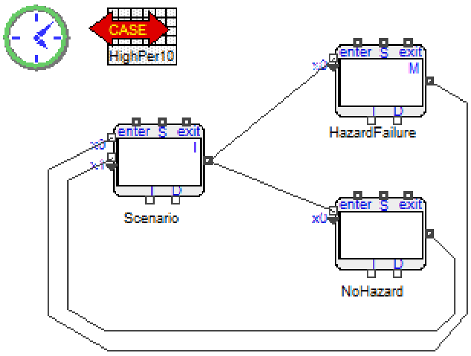


Figure 6.5: ExtendSim model of a scenario.

Listing 6.1: Calculations in "Scenario" for the probability in scenario 1

```
// when entered
nextTest = tau;
nextDemand = DExponential(lambda_de);
nextDUFailure = DExponential(lambda_DU);

// triggerOut condition
if(nextDUFailure < nextDemand < nextTest) -> HazardFailure
else -> NoHazard
```

Listing 6.2: Calculations in ”Scenario” for the probability in scenario 2

```
// when entered
nextTest = tau;
nextDemand = DExponential(lambda_de);
nextDDFailure = DExponential(lambda_DD);
repairDone = DExponential(mu_DD);

// triggerOut condition
if(nextDDFailure < nextDemand < repairDone <= nextTest) ->
   HazardFailure
else -> NoHazard
```

Scenario 2

We want to simulate from figure 6.3, $Pr(T_{DD} < T_{de} < T_{DD} + \tilde{T}_{DD} \leq \tau)$, by the proportion of events where a demand occurs after a DD-failure, and during its repair within the test interval. T_{DD}, T_{de} and $\tilde{T}_{DD}$ are calculated in time 0 for each test interval. The simulation model is illustrated in figure 6.5.

”Scenario” now contains the following parameters: λ_{DD}, λ_{de}, μ_{DD} and τ, and the variables: ”nextTest”, ”nextDemand”, ”nextDDFailure” and ”repairDone”. The algorithm is in listing 6.2.

Scenario 3

We want to simulate $Pr(T_{de} < T_D < T_{de} + \tilde{T}_{de} \leq \tau)$ as in figure 6.4, by the proportion of events where a D-failure occurs after demand, and during the demand duration within a test interval.

The simulation model is similar to the one for Scenario 2, ”nextDDFailure” is replaced by ”nextDemand”, ”nextDemand” is replaced by ”nextD-Failure”, and ”repairDone” by ”demandDone”.

The hazard rate according to equation 6.15 is obtained by adding the simulation result for each of the three scenarios.

Listing 6.3: Calculations in "Scenario" for equation in scenario 1

```
// when entered
nextTest = tau;
nextDemand = DExponential(lambda_de);
nextDUFailure = DExponential(lambda_DU);
if(nextTest > nextDUFailure) tToTau = nextTest -
    nextDUFailure;
else tToTau = 0;

// triggerOut condition
if((nextDUFailure < nextTest) && (nextDemand <= tToTau)) ->
    HazardFailure
else -> NoHazard
```

6.4 Simulation models of the adjusted scenarios

Scenario 1

The integral in equation 6.4 is equal to:

$$Pr(T_{DU} = t \ \cap \ 0 < T_{de} \leq \tau - t), \quad t \in [0, \tau] \tag{6.16}$$

The simulated model is built up in the same way as in figure 6.2, but is now based on the probability in equation 6.16. We want to simulate the probability of a failure to be within a test interval, and a demand to happen before the test given a failure has happened.

The parameters are the same as for scenario 1 in section 6.3, except for "tToTau" which is added. This variable is the time between a DU-failure and the next test. There is a hazardous event if there is a DU-failure within the test interval, and a demand occurring within the time to the next test. The simulation is described in listing 6.3.

Scenario 2

We want to simulate from equation 6.10 $Pr(T_{DD} = t \cap 0 < T_{de} \leq u \cap \tilde{T}_{DD} = u \leq \tau - t)$, where u is the DD-repair time. Note that we have simulated the repair to be finished in time u, and not to survive it. As stated above in section 6.2, this makes more sense.

"Scenario" contains the same parameters and variables as in section 6.3, as well as "repairDuration" and "tToTau".

Listing 6.4: Calculations in ”Scenario” for equation in scenario 2

```
// when entered
nextTest = tau;
nextDemand = DExponential(lambda_de);
nextDDFailure = DExponential(lambda_DD);
repairDone = DExponential(mu_DD);
if(nextDDFailure < nextTest) tToTau = nextTest -
   nextDDFailure;
else tToTau = 0;

//triggerOut condition
if((nextDDFailure < nextTest) && (nextDemand < repairDone <=
   tToTau))-> HazardFailure
else -> NoHazard
```

Scenario 3

We want to simulate equation 6.14, $Pr(T_{de} = t \cap 0 < T_D \leq u \cap \tilde{T}_{de} = u \leq \tau - t)$, where u is the demand duration time given a demand has happened.

Scenario 3 has an algorithm that is basically the same as for scenario 2 (listing 6.4), where ”nextDDFailure” is replaced by ”nextDemand”, ”nextDemand” is replaced by ”nextDFailure”, and ”repairDone” by ”demandDone”.

There is a hazardous event if there is a demand within the test interval, and a D-failure between time 0 and the duration of demand.

6.5 Results of the calculations of the scenarios

Figure 6.6 illustrates that both the scenario-based models depending on the probabilities and the adjusted one have identical simulated and analytical results. Both methods approximate the asymptote well for very low demand rates. The deviation between the adjusted one and the probability based hazard rates deviate more as the demand rate increases. We notice that the hazard rate decreases for the probability based calculation when the demand rate increases. The reason is the probabilities for scenario 1 and 2, where it assumes that there is only one demand per test interval that is a threat. Scenario 3 does not make much of a difference since it depends on a D-failure to occur during a demand, where the D-failure rate is very small and does not appear more than once during a test interval. Each of the two probabilities for scenario 1 and 2 is dependent on the frequency of demands and respectively the frequency of a DU-failure and a DD-failure,

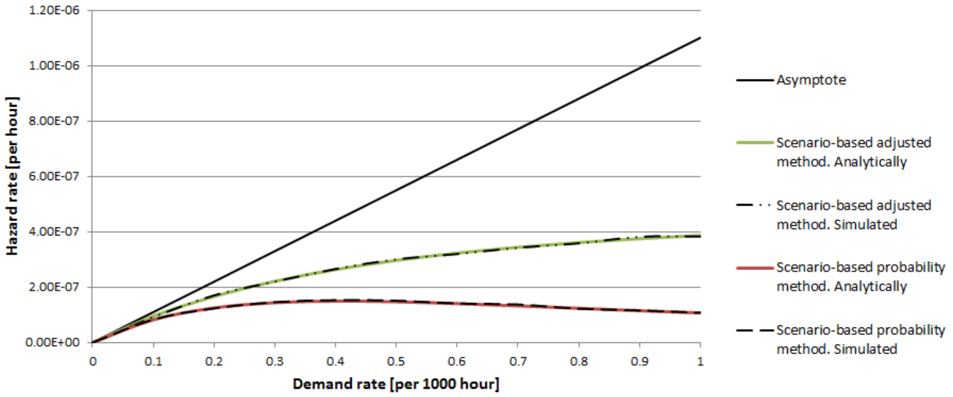
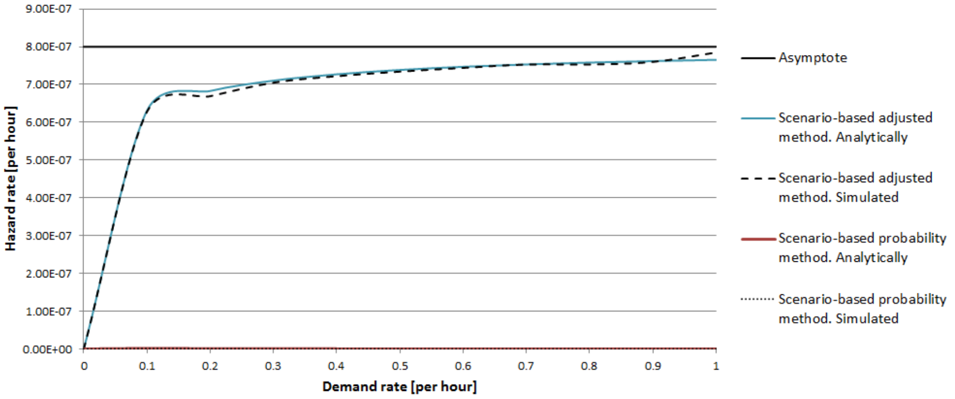
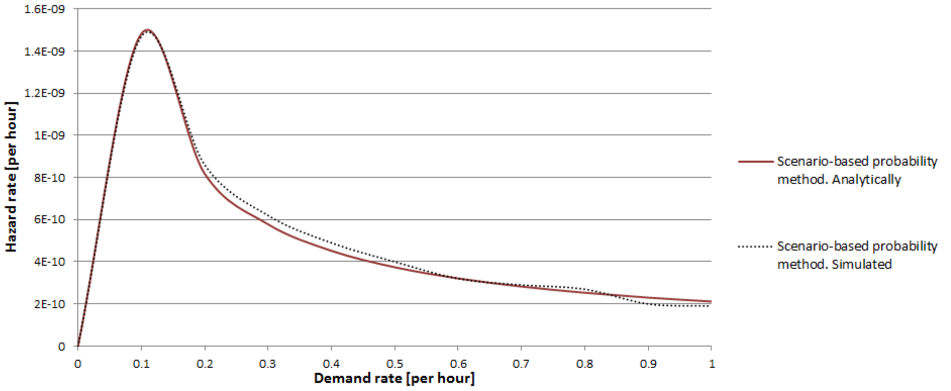


Figure 6.6: The hazard rate as a function of demand rate in low-demand mode per 1000 hours



(a) Comparing the adjusted and correct calculated and simulated scenario-based hazard rate with the asymptote. One demand per hour. Note the scales.



(b) The simulated and the calculated hazard rate for the correct scenario-based formula. One demand per hour. Note the scales.

Figure 6.7: ExtendSim model of a scenario.

which also are stochastic variables. These are dependent for a hazardous failure to occur. For example for the first scenario, we must know where there is a DU-failure and where there is a demand during the interval. This is determined immediately when entering a new test interval. For a very low demand rate it is correct that there is only one demand per test interval, which shows through a well approximated calculation to the asymptote for this case. But as the demand rate increases, there will be more than one demand on each test interval. Since the probability and the corresponding simulation model only account for the first demand happening after time 0 in the test interval, there are demands after the DU-failure that are missed. The hazard rate decreases.

This becomes even more clear when we look at the results for the high demand rate case in figure 6.7 (a), where the correct calculation of the scenario-based formula, shown more explicitly in (b), gives a very small estimate of the hazard rate. It is underestimated compared to the asymptote. The simulated model and the calculations are more or less identical.

The scenario-based adjusted model approximates the asymptote very well for the high demand rate case.

6.6 Simulation model of the system

We have simulated the system model in two ways. The first model simulates the Markovian transition diagram (figure 6.1) with assumptions from [2].

As mentioned, we think the repair rate of a DU-failure μ_{DU} is not correct. In the second model we want to look closer into this by simulating demands continuously throughout the simulation time (in a similar way as we did in the model for the two channel protective system in section 4.2.2, where demands are generated).

Note that the big difference is in state 1, the undetected failure state. The assumption of [2] says that when the system comes to this state, it stays here with a mean duration $\frac{\tau}{2} + \text{MTTR}_{DU}$. There is a hazardous event if a demand occurs during this time.

In real life the system is in an undetected failure state until it is detected by either a demand or a test, for so to be repaired. This is what simulation model 2 implies, and should be the more accurate system model.

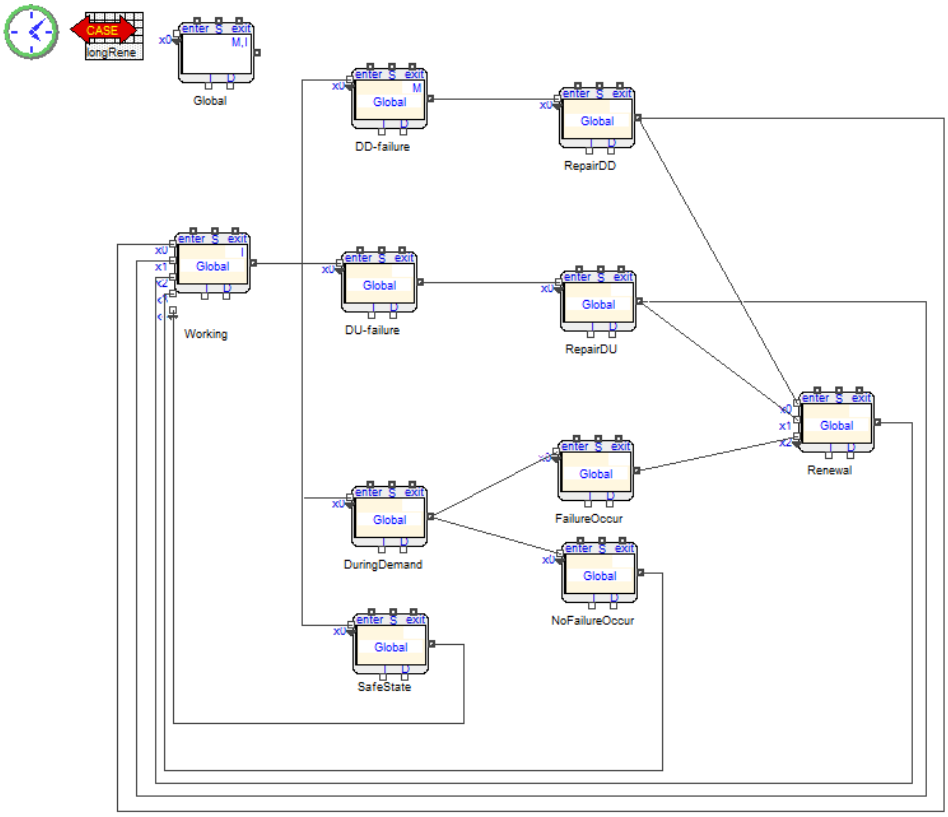


Figure 6.8: The simulation model of the single component safety system.

6.6.1 Simulation model 1: exponentially distributed DU-repair rate

The system model, in figure 6.1, is simulated in much the same way as the single component simulation model explained in section 3.3. The model is illustrated in figure 6.8.

The simulated model starts in "Working". The time for next DD-failure, DU-failure, safe state and demand is calculated. For the event that occurs first, the model goes to the represented block. In the DD-block and DU-block, the event of having a demand before its repairs are finished is calculated. If the time of repair is less than the time to next demand, the model goes to "Working". Otherwise, it goes to "Renewal", which is the hazardous event state of the simulated model, represented as state 0 in the Markov model.

For the demand block, the event of having a failure before the duration of the demand ends is calculated. If the time of failure is less than the time of the demand duration, the model goes to "Renewal", and we register another hazardous event on the system. If not, the model goes to "Working".

In the safe state block it has a given duration before it goes back to "Working".

A more detailed explanation of the simulated model and each block follows:

Global The parent state. The global parameters for the model are all the transition rates from figure 6.1: λ_{DD}, λ_{DU}, λ_D, λ_{de}, λ_s, μ_{de}, μ_{DU}, μ_{DD}, μ_s and m. τ is also a variable here, with the other variables that are calculated in other blocks continuously throughout the simulation; nextDDFailure (time for next DD-failure), nextDUFailure (time for next DU-failure), nextDangerousFailure (time for next dangerous failure), nextDemand (time for next demand) and nextSafeState (time for next safe state). They are stored here, accessible for all other blocks in the model.

Working This is the initial state and represents state 5 in the Markov model (figure 6.1). From state 5 there are four possibilities: there can be a DD-failure, DU-failure, a demand or the system goes to a safe state. The calculations done in the block is shown in listing 6.5.

DD-failure This block represents state 2 in figure 6.1. When a DD-failure occurs, repair starts immediately, and the model goes to "RepairDD" instantly.

RepairDD This block also represents state 2. The transition rate to state 5 is used giving the repair time for a DD-failure. There is a hazardous event if a demand appears during the repair. Listing 6.6 shows the algorithm.

DU-failure It represents state 1 in figure 6.1. The model leaves immediately to "RepairDU", since the undetected failure time is included in the repair time.

RepairDU Here the component is being repaired after a DU-failure. This is the transition from state 1 to state 5. The block calculates the duration of the repair, which is the time spent in this block DExp(μ_{DU}). When the repair is finished the system is back to "Working". If a demand has occurred during the repair it goes to "Renewal".

DuringDemand It represents state 3 in figure 6.1. If a demand occurs, the mean duration time for a demand is $1/\mu_{de}$. This is the time spent in this block, if a D-failure does not occur during the time of the demand duration, which leads to a hazardous event. The block is illustrated in listing 6.7.

Listing 6.5: Calculations in ”Working”

```
// when entered
if (nextDemand < currentTime) nextDemand = currentTime +
   DExponential(lambda_de);
if (nextDDFailure < currentTime) nextDDFailure = currentTime
   + DExponential(lambda_DD);
if (nextDUFailure < currentTime) nextDUFailure = currentTime
   + DExponential(lambda_DU);
if (nextSafeState < currentTime) nextSafeState = currentTime
   + DExponential(lambda_s);

// triggerOut condition
if((nextDDFailure < nextDUFailure) && (nextDDFailure <
   nextDemand) && (nextDDFailure < nextSafeState)) ->
   DD-failure
else if ((nextDUFailure < nextDDFailure) && (nextDUFailure <
   nextDemand) && (nextDUFailure < nextSafeState)) ->
   DU-failure
else if ((nextDemand < nextDDFailure) && (nextDemand <
   nextDUFailure) && (nextDemand < nextSafeState)) ->
   DuringDemand
else -> SafeState
```

Listing 6.6: Calculations in ”RepairDD”

```
// when entered
if(nextDemand < currentTime) nextDemand = currentTime +
   DExponential(lambda_de);
repairDDOver = currentTime + DExponential(mu_DD);

// triggerOut condition
if(nextDemand < repairDDOver) -> Renewal
else -> Working
```

Listing 6.7: Calculations in ”DuringDemand”

```
// when entered
if (demandOver < currentTime) demandOver = currentTime +
   DExponential(lambda_de);
if (nextDFailure < currentTime) nextDFailure = currentTime +
   DExponential(lambda_D);

// triggerOut condition
if(nextDFailure < demandOver) -> Renewal
else -> Working
```

SafeState It represents state 4 in figure 6.1. The mean duration of this state is $1/\mu_s$. After that it goes back to ”Working”.

Renewal This block collects the hazardous events that occur from the three scenarios; demand during a DD-repair, a demand during a DU-repair and a failure happening in the duration of a demand. From the results in this block we get the number of times this state has been visited, which means how many times there has been a hazardous event during the simulation time. The mean duration in this state is $1/m$. After the renewal time is over, the system is working perfectly again.

6.6.2 Simulation model 2: generated demands

Figure 6.9 illustrates the second version of the model from the Markovian state diagram. To get the most realistic model, demands are generated continuously throughout the whole simulation time, in the ”Demand” block. The blocks that calculated ”nextDemand” in the previous model, now has an ”interrupt” connection to the ”Demand”. When a demand happens that is relevant for the system, the simulation model changes state according to its conditions (the same as in the listings describing the calculations for each block for the previous model).

This has the most impact on the ”DU-failure” block. The time for the next test is calculated when entering, and if a demand occurs before that time, interrupting the block, the system goes to ”DetectedByDemand”. This block only keeps track of how many of the undetected failures are detected by demand, and the simulation model goes immediately to ”Renewal”. If a demand does not occur before the next test is done, the system goes to ”DetectedByTest”, and immediately to ”RepairDU”. Now the model spends the correct amount of time in the undetected failure mode.

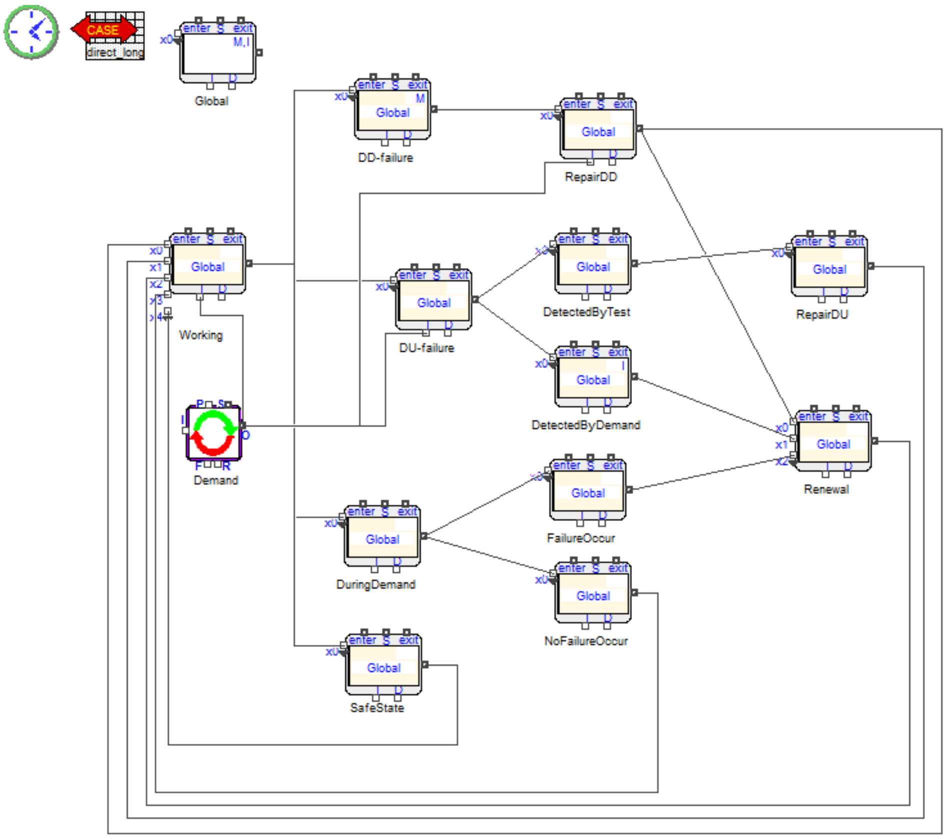


Figure 6.9: The simulation model of the single component safety system with a demand generator.

Calculation of the hazard rate

$$\eta_{sim} = \frac{\text{\# events in "Renewal"}}{\text{simulation time}}, \tag{6.17}$$

because all scenarios that lead to a hazardous event end in "Renewal".

6.7 Results for the simulated system model

The plots in figure 6.10 show that the simulated system with a demand generator (simulation model 2, section 6.6.2) approximates the asymptote slightly better than the system model based on assumptions from [2] (simulation model 1, section 6.6.1) for very low demand rates. The latter model actually gives non-conservative hazard rates, when we use the correct simulated system model as benchmark for the low demand mode (figure 6.10 (a)).

While for systems in the high demand mode (figure 6.10 (b)) the simulation model 1 approximates the failure rate (asymptote) before simulation model 2, and is slightly too conservative before they both approximates the asymptote.

We can see by assuming $Pr(T_{de} > t)$ that the scenario-based adjusted formula gives a good approximation to the correct simulated system model, even though the assumptions made here are conservative. It is slightly too non-conservative compared to the system model with demand. This can be explained by a very low failure rate compared to the demand rate which makes it very likely for a demand to occur after a DU-failure ($5 \cdot 10^{-7}$ and demand rates from $1 \cdot 10^{-4}$ per hour [2]).

The probability based formula however is much too non-conservative.

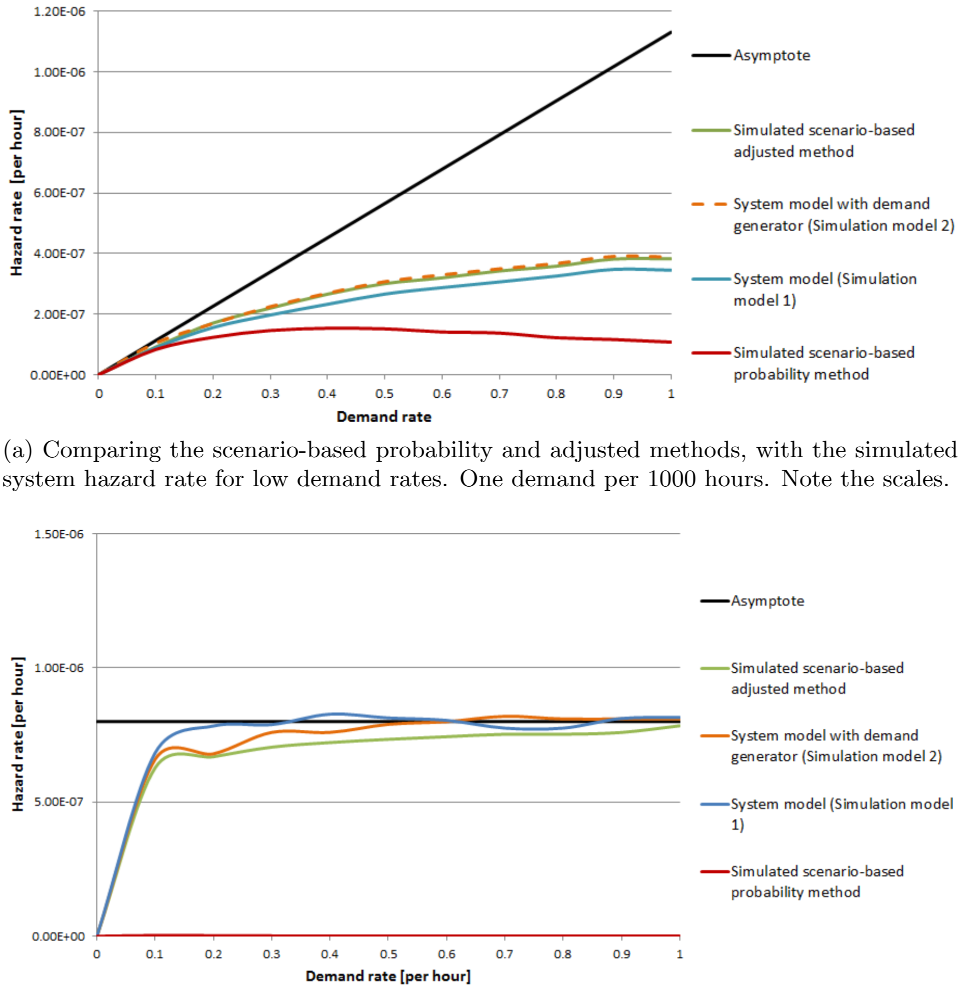
Even though the scenario-based adjusted formula is a good approximation to the hazard rate of the simulated system, it does not take into account a safe state (λ_s and μ_s), the renewal time (m) or MTTR$_{DU}$.

Increased duration time in repair, safe state or renewal time do not lead to a hazardous event directly, but they do affect the availability of the system. However, this has an impact on the hazard frequency of the system.

We have simulated four additional simulated system cases compared to the standard one given in [2] for low demand rates. They are described below, where the change in different parameters for each case are mentioned, the rest of the parameters for the model are as in table 6.1.

Case 1: There is no safe state, $m = 0$ and MTTR$_{DU} = 0$.

Case 2: The repair duration after a DU-failure is about 3 months compared to 8 hours. This means that $\mu_{DU} = 0.0005$ per hour.



(a) Comparing the scenario-based probability and adjusted methods, with the simulated system hazard rate for low demand rates. One demand per 1000 hours. Note the scales.

(b) Comparing the scenario-based probability and adjusted methods, with the simulated system hazard rate. One demand per hour. Note the scales.

Figure 6.10: Results comparing the simulated hazard rate for the systems and the two versions of the scenario-based formula.

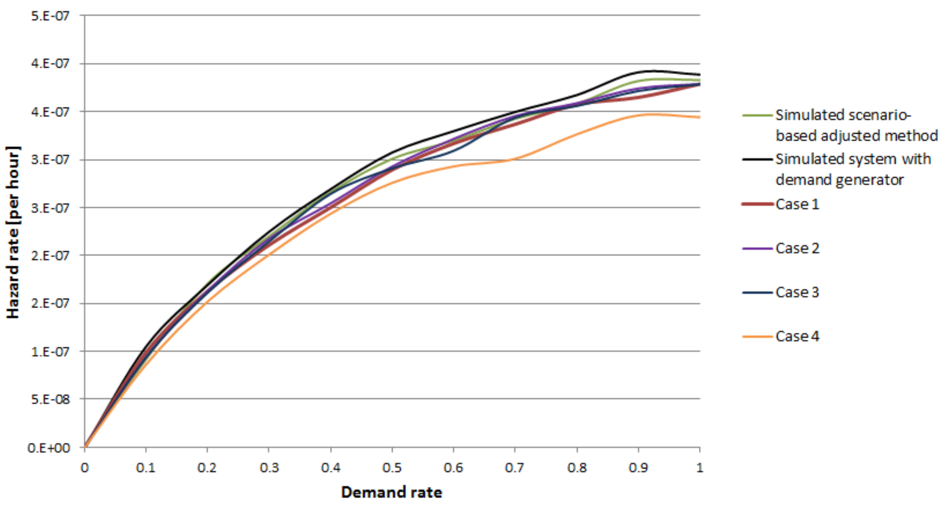


Figure 6.11: Simulated results with additional cases for low demand rates. One demand per 1000 hours.

Case 3: The mean renewal time is longer, 6 months (compared to 7 days). $m = 0.000228$ per hour.

Case 4: There is a higher frequency for entering safe state, ($\lambda_s = 5 \cdot 10^{-5}$ per hour). And the duration in safe state is 3 months, compared to 1 day. $\mu_s = 4.1667 \cdot 10^{-4}$ per hour.

From figure 6.11 we see that case 4 stands out. A long time in a safe state decreases the availability of the system, which results in a decreased hazard rate. The adjusted scenario-based formula is too conservative compared to it.

It is also slightly too conservative compared to the other cases, but not significantly. The parameters chosen might not give the necessary unavailability for the system to influence the hazard rate.

This means that the scenario-based adjusted approach is a good approximation method to the estimated hazard rate as long as the failure rate is much smaller than the demand rate, and the parameters not included in the formula are not of significant length.

7. Conclusion

In this thesis we have looked at the effects on the hazard rate for various demand rates in a 1oo1, 1oo2 and a 2oo3 safey system model. We have modelled the systems with various assumptions of demand duration, different failure rates, time between proof-tests, time of repair and priority of repair. This to get a better understanding of the hazard rate in the different models, and the borderline between low demand and high demand mode systems, where we have the intermediate demand region.

We have illustrated that there is an unexpected behaviour for a parallel system [4] compared to a single system [3] and [2]. The maximum and minimum points are dependent on the downtime for the system (section 4.4.2). The effect of demands utilized as tests, which implies a stronger redundant system and an increased availability for a system in the intermediate demand region is shown well for a 1oo2 and 2oo3 system. The maximum point occurs where the demand rate is significantly larger than the proof-tests. The repair rate for a failed component is also a significant factor for the downtime of system, impacting how strong the redundant system is. For the online system there is a minimum point where there is a significant possibility of having a demand during the system repair time, which means the system has a weak redundant system. A long downtime approximates a serial system for a 1oo2 system.

We can conclude that the intermediate demand mode is very much dependent on the relationship between proof-tests, demands and repair.

The hazard rate for a system in the intermediate demand mode is not suitable with the IEC standard's calculation of PFD or PFH. It can lead to major consequences for the operator and the people/ environment around it. If the system has a SIL that is too high, there are too many tests performed, which allocate resources in a cost-efficient way. Or even worse, the system has a SIL that is too low. In that case the system can cause hazard costs, loss of production and reduces profit.

A new formula for calculating the probability of failure between the maximum and minimum point, or between the maximum point and the steady point should be constructed.

We have shown an adequate and efficient way to deal with the rare event problem when modelling safety systems for high demand rates. By calculating the crucial information when needed saves a lot of processing time, and gives a much more powerful model. The Harel Statechart modelling gives satisfying results for various demand rates from very low to very high for various models with distinctive properties. It shows that we can model realistic problems in an intuitive and uncomplicated way, which far exceeds the possibilities we have with a Markov model.

Through Harel Statechart we have shown that using $\mu_{DU} = 1/(\frac{\tau}{2} + \text{MTTR}_{DU})$ as the transition rate from a state where the system has an undetected failure until it works again (model from [2] in section 6) gives a non-conservative hazard rate compared to simulating each demand and let the system go to repair when it is actually detected.

The adjusted scenario-based formula for calculating the hazard rate by [2] is a good approximation to the system model, even though it has a conservative assumption. However, the failure rate has to be much smaller than the demand rate, and not spend a significant time in a state that decreases the availability on the system.

A.

List of abbreviations

D-failure	Dangerous failure (DU+DD)
DD-failure	Dangerous detected failure
DU-failure	Dangerous undetected failure
PFD	Probability of failure on demand
PFH	Probability of failure per hour
SIF	Safety instrumented function
SIL	Safety integrity level
SIS	Safety instrumented system

B.

ExtendSim

A brief description of each block used in the ExtendSim simulation models, illustrated in figure B.1. This is taken from the "Help" function for each block in the program [13].

(a) Clock: For discrete event modelling this must be included to the left of the model. It provides the settings for simulation control, attributes, item contents and the discrete rate for doing event scheduling.

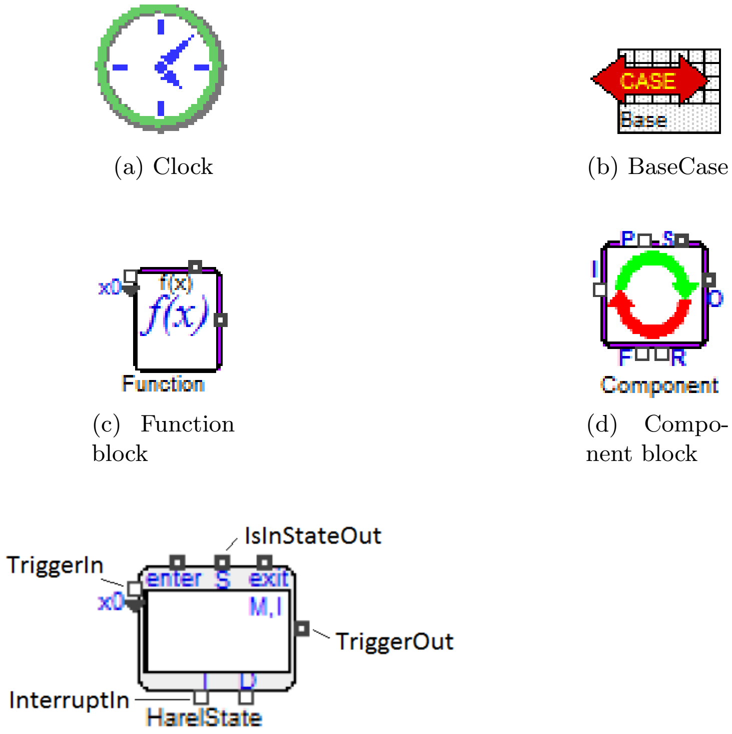
(b) Case Study: The link between Extend and Excel. Each desired case is given in Excel. These cases are simulated by starting the simulation from this block. The results is given in a "result" tab in Excel.

(c) Function Block: It can take maximum 7 input variables. There are fixed functions sat up like sum, max and min. But there is also a "General" option where a desired function can be specified. The Function blocks has the result of its calculation performed, or a given variable for a specified equation of its function as an output variable.

(d) Component block: Modelling maintainable items. In the simulations done in this thesis only the output on the right hand side "O" (state-Out) is used. Blocks that are connected to these blocks are informed by TRUE/FALSE when there is a change in the component block (binary output).

The user can give the information for the component regarding its initial state, up-value, down-value, MTTF, MTTR, distribution details and its parameters, among other details about test and maintenance.

(e) Harel State block The following are/ can be given by the user for each block: condition for triggering this state, expressions evaluated on entry, calculation of duration time and expressions evaluated on



(a) Clock

(b) BaseCase

(c) Function block

(d) Component block

(e) Harel State block

Figure B.1: Illustration of relevant blocks in ExtendSim

trigger out of the block. The Harel State block has several input and output connectors. In this thesis we have used:

TriggerInn is for the input connector(s) (can be extended to have maximum 20 input channels). Is most often connected to another HarelState block, or a Component block.

TriggerOut contains the value that is given for triggerOut condition. Is often connected to another HarelState block or a Function block.

InterruptIn If another block/component connected here has a value > 0, this block is interrupted, and the model leaves this block immediately.

IsInStateOut has output equal to TRUE when the block is entered, and FALSE when it has left. It is often connected to a Function block. In that way the Function block has the possibility to know which block that is active at all times.

The results given in each block states how many times each block has been activated, mean duration time in this block, the standard deviation, percentage immediate leave and enter.

C.

Calculations of the scenario based formulae

C.1 Scenario 1

Calculation of the adjusted probability in equation 6.4:

$$\begin{aligned}
P_{R1} =& \int_0^\tau \lambda_{DU} e^{-\lambda_{DU} t}(1 - e^{-\lambda_{de}(\tau - t)})\; dt \\
=& \lambda_{DU} \int_0^\tau \left(e^{-\lambda_{DU} t} - e^{-(\lambda_{DU} - \lambda_{de})t - \lambda_{de}\tau}\right)\; dt \\
=& \lambda_{DU} \left[-\frac{e^{-\lambda_{DU} t}}{\lambda_{DU}} + \frac{e^{-(\lambda_{DU} - \lambda_{de})t - \lambda_{de}\tau}}{\lambda_{DU} - \lambda_{de}} \right]_0^\tau \\
=& \frac{\lambda_{DU}}{\lambda_{DU} - \lambda_{de}} \left(e^{-\lambda_{DU}\tau} - e^{-\lambda_{de}\tau} \right) - e^{-\lambda_{DU}\tau} + 1
\end{aligned} \tag{C.1}$$

Calculation of the correct integral for the probability in equation 6.5:

$$\begin{aligned}
P_{corr1} =& \int_0^\tau \lambda_{DU} e^{-\lambda_{DU} t} e^{-\lambda_{de} t}(1 - e^{-\lambda_{de}(\tau - t)})\; dt \\
=& \lambda_{DU} \int_0^\tau \left(e^{-(\lambda_{DU} + \lambda_{de})t} - e^{-(\lambda_{DU})t - \lambda_{de}\tau}\right)\; dt \\
=& \lambda_{DU} \left[-\frac{e^{-(\lambda_{DU} + \lambda_{de})t}}{\lambda_{DU} + \lambda_{de}} + \frac{e^{-\lambda_{DU} t - \lambda_{de}\tau}}{\lambda_{DU}} \right]_0^\tau \\
=& \frac{\lambda_{DU}}{\lambda_{DU} + \lambda_{de}} \left(-e^{-(\lambda_{DU} + \lambda_{de})\tau} + 1 \right) + e^{-(\lambda_{DU} + \lambda_{de})\tau} - e^{-\lambda_{de}\tau}
\end{aligned} \tag{C.2}$$

C.2 Scenario 2

Calculation of the adjusted $Pr(t < T_{de} < t + \tilde{T}_{DD} \leq \tau)$ gives:

$$
\begin{aligned}
\int_0^{\tau-t} (1 - e^{-\lambda_{de}u})\mu_{DD}e^{-\mu_{DD}u}\ du =& \mu_{DD}\int_0^{\tau-t} \left(e^{-\mu_{DD}u} - e^{-(\lambda_{de}+\mu_{DD})u}\right)\ du \\
=& \mu_{DD}\left[-\frac{e^{-\mu_{DD}u}}{\mu_{DD}} + \frac{e^{-(\lambda_{de}+\mu_{DD})u}}{\lambda_{de}+\mu_{DD}}\right]_0^{\tau-t} \\
=& \mu_{DD}\left(\frac{e^{(\lambda_{de}+\mu_{DD})(t-\tau)} - 1}{\lambda_{de}+\mu_{DD}}\right) - e^{\mu_{DD}(t-\tau)} + 1
\end{aligned}
\tag{C.3}
$$

The adjusted probability of having scenario 2, equation 6.10:

$$
\begin{aligned}
P_{R2} =& \int_0^{\tau} \mu_{DD}\lambda_{DD}e^{-\lambda_{DD}t}\left[\frac{e^{(\lambda_{de}+\mu_{DD})(t-\tau)} - 1}{\lambda_{de}+\mu_{DD}} - \frac{e^{\mu_{DD}(t-\tau)} - 1}{\mu_{DD}}\right]\ dt \\
=& \frac{\mu_{DD}\lambda_{DD}}{\lambda_{de}+\mu_{DD}}\int_0^{\tau}\left(e^{(\lambda_{de}+\mu_{DD}-\lambda_{DD})t-(\lambda_{de}+\mu_{DD})\tau} - e^{-\lambda_{DD}t}\right)\ dt \\
& - \lambda_{DD}\int_0^{\tau}\left(e^{(\mu_{DD}-\lambda_{DD})t-\mu_{DD}\tau} - e^{\lambda_{DD}t}\right)\ dt \\
=& \frac{\mu_{DD}\lambda_{DD}}{\lambda_{de}+\mu_{DD}}\left[\frac{e^{(\lambda_{de}+\mu_{DD}-\lambda DD)t-(\lambda_{de}+\mu_{DD})\tau}}{\lambda_{de}+\mu_{DD}-\lambda_{DD}} + \frac{e^{-\lambda_{DD}t}}{\lambda_{DD}}\right]_0^{\tau} \\
& - \lambda_{DD}\left[\frac{e^{(\mu_{DD}-\lambda_{DD})t-\mu_{DD}\tau}}{\mu_{DD}-\lambda_{DD}} + \frac{e^{-\lambda_{DD}t}}{\lambda_{DD}}\right]_0^{\tau} \\
=& \frac{\mu_{DD}\lambda_{DD}}{\lambda_{de}+\mu_{DD}}\left(\frac{e^{-\lambda_{DD}\tau} - e^{-(\lambda_{de}+\mu_{DD})\tau}}{\lambda_{de}+\mu_{DD}-\lambda_{DD}}\right) + \mu_{DD}\left(\frac{e^{-\lambda_{DD}\tau} - 1}{\lambda_{de}+\mu_{DD}}\right) \\
& - \lambda_{DD}\left(\frac{e^{-\lambda_{DD}\tau} - e^{-\mu_{DD}\tau}}{\mu_{DD}-\lambda_{DD}}\right) - e^{-\lambda_{DD}\tau} + 1
\end{aligned}
\tag{C.4}
$$

The correct integrals and probabilities, for equation 6.9:

$$\begin{aligned}
P(t < T_{de} < t + \tilde{T}_{DD} \leq \tau) = & \int_0^{\tau - t} e^{-\lambda_{de} t}(1 - e^{-\lambda_{de} u})\mu_{DD} e^{-\mu_{DD} u}\, du \\
= & \mu_{DD} e^{-\lambda_{de} t} \int_0^{\tau - t} \left(e^{-\mu_{DD} u} - e^{-(\lambda_{de} + \mu_{DD})u} \right)\, du \\
= & \mu_{DD} e^{-\lambda_{de} t} \left[-\frac{e^{-\mu_{DD} u}}{\mu_{DD}} + \frac{e^{-(\lambda_{de} + \mu_{DD})u}}{\lambda_{de} + \mu_{DD}} \right]_0^{\tau - t} \\
= & \mu_{DD} e^{-\lambda_{de} t} \left(\frac{e^{(\lambda_{de} + \mu_{DD})(t - \tau)} - 1}{\lambda_{de} + \mu_{DD}} \right) \\
& - e^{-\lambda_{de} t}(e^{\mu_{DD}(t - \tau)} - 1)
\end{aligned} \tag{C.5}$$

The correct probability of having scenario 2 is:

$$\begin{aligned}
P_{corr2} = & \int_0^{\tau} \lambda_{DD}\mu_{DD} e^{-\lambda_{DD} t} e^{-\lambda_{de} t} \left[\frac{e^{(\lambda_{de} + \mu_{DD})(t - \tau)} - 1}{\lambda_{de} + \mu_{DD}} - \frac{e^{\mu_{DD}(t - \tau)} - 1}{\mu_{DD}} \right]\, dt \\
= & \frac{\lambda_{DD}\mu_{DD}}{\lambda_{de} + \mu_{DD}} \int_0^{\tau} \left(e^{(\lambda_{DD} - \mu_{DD})t - (\lambda_{de} + \mu_{DD})\tau} - e^{-(\lambda_{DD} + \lambda_{de})t} \right)\, dt \\
& - \lambda_{DD} \int_0^{\tau} \left(e^{-(\lambda_{DD} + \lambda_{de} - \mu_{DD})t - \mu_{DD}\tau} - e^{-(\lambda_{DD} + \lambda_{de})t} \right)\, dt \\
= & \frac{\lambda_{DD}\mu_{DD}}{\lambda_{de} + \mu_{DD}} \left[-\frac{e^{-(\lambda_{DD} - \mu_{DD})t - (\lambda_{de} + \mu_{DD})\tau}}{\lambda_{DD} - \mu_{DD}} + \frac{e^{-(\lambda_{DD} + \lambda_{de})t}}{\lambda_{DD} + \lambda_{de}} \right]_0^{\tau} \\
& - \lambda_{DD} \left[-\frac{e^{-(\lambda_{DD} + \lambda_{de} - \mu_{DD})t - \mu_{DD}\tau}}{\lambda_{DD} + \lambda_{de} - \mu_{DD}} + \frac{e^{-(\lambda_{DD} + \lambda_{de})t}}{\lambda_{DD} + \lambda_{de}} \right]_0^{\tau} \\
= & \frac{\lambda_{DD}\mu_{DD}}{\lambda_{de} + \mu_{DD}} \left(\frac{e^{-(\lambda_{DD} + \lambda_{de})\tau} - 1}{\lambda_{DD} + \lambda_{de}} - \frac{e^{-(\lambda_{DD} + \lambda_{de})\tau} - e^{-(\lambda_{de} + \mu_{DD})\tau}}{\lambda_{DD} - \mu_{DD}} \right) \\
& - \lambda_{DD} \left(\frac{e^{-(\lambda_{DD} + \lambda_{de})\tau} - 1}{\lambda_{DD} + \lambda_{de}} - \frac{e^{-(\lambda_{DD} + \lambda_{de})\tau} - e^{-\mu_{DD}\tau}}{\lambda_{DD} + \lambda_{de} - \mu_{DD}} \right)
\end{aligned} \tag{C.6}$$

C.3 Scenario 3

By letting T_{de} be replaced by T_D, T_{DD} by T_{de} and $\tilde{T}_{DD}$ by $\tilde{T}_{de}$, and hence λ_{de} is replaced by λ_D, λ_{DD} by λ_{de} and μ_{DD} by μ_{de} we obtain scenario 3 from scenario 2.

Calculation of the adjusted $Pr(t < T_D < t + \tilde{T}_{de} \leq \tau)$ gives:

$$\int_0^{\tau-t}(1-e^{-\lambda_D u})\mu_{de}e^{-\mu_{de}u}\ du = \mu_{de}\left(\frac{e^{(\lambda_D+\mu_{de})(t-\tau)}-1}{\lambda_D+\mu_{de}}\right) - e^{\mu_{de}(t-\tau)}+1 \tag{C.7}$$

The adjusted probability of having scenario 3, equation 6.14 gives:

$$\begin{aligned} P_{R3} =& \frac{\mu_{de}\lambda_{de}}{\lambda_D+\mu_{de}}\left(\frac{e^{-\lambda_{de}\tau}-e^{-(\lambda_D+\mu_{de})\tau}}{\lambda_D+\mu_{de}-\lambda_{de}}\right)+\mu_{de}\left(\frac{e^{-\lambda_{de}\tau}-1}{\lambda_D+\mu_{de}}\right) \\ & -\lambda_{de}\left(\frac{e^{-\lambda_{de}\tau}-e^{-\mu_{de}\tau}}{\mu_{de}-\lambda_{de}}\right)-e^{-\lambda_{de}\tau}+1 \end{aligned} \tag{C.8}$$

The result of the correct probability in equation 6.13:

$$\begin{aligned} Pr(t<T_D<t+\tilde{T}_{de}\leq\tau) =& \mu_{de}e^{-\lambda_D t}\left[\frac{e^{(\lambda_D+\mu_{de})(t-\tau)}-1}{\lambda_D+\mu_{de}}\right] \\ & -e^{-\lambda_D t}(e^{\mu_{de}(t-\tau)}-1) \end{aligned} \tag{C.9}$$

The correct probability of scenario 3 is now:

$$\begin{aligned} P_{corr3} =& \frac{\lambda_{de}\mu_{de}}{\lambda_D+\mu_{de}}\left(\frac{e^{-(\lambda_{de}+\lambda_D)\tau}-1}{\lambda_{de}+\lambda_D}-\frac{e^{-(\lambda_{de}+\lambda_D)\tau}-e^{-(\lambda_D+\mu_{de})\tau}}{\lambda_{de}-\mu_{de}}\right) \\ & -\lambda_{de}\left(\frac{e^{-(\lambda_{de}+\lambda_D)\tau}-1}{\lambda_{de}+\lambda_D}-\frac{e^{-(\lambda_{de}+\lambda_D)\tau}-e^{-\mu_{de}\tau}}{\lambda_{de}+\lambda_D-\mu_{de}}\right) \end{aligned} \tag{C.10}$$

D.

Safety Instrumented Systems operated in the Intermediate Demand Mode

The following article by Siegfried Eisinger (DNV GL), Bent Natvig(UiO), Luiz F. Oliveira (DNV GL) and Kristine Tveit (UiO) was published and presented at the European Safety and Reliability Conference (ESREL) in Zurich, september 2015.

Abstract

When analysing critical systems the demand frequency is crucial. Often the low and the high demand mode are distinguished. In this paper the intermediate demand mode is analysed.

The results from the analyses of the example (two channel) model show that the hazard rate exhibits unexpected behaviour in the intermediate demand region. As far as can be seen from the analysis, the standard Probability of Failure on Demand (PFD) formulas are usable, but they become exceedingly conservative as one moves into the intermediate demand region. On the other hand, usage of the standard formulas for the hazard rate (PFH) (high demand mode) in the intermediate region may lead to non-conservative results. Therefore, whenever a system seems to be operated in this intermediate demand mode, or even only close to it is advisable to perform more accurate analysis compared to standard PFD and PFH formulas. It has been demonstrated that such analysis is readily feasible using modern simulation tools. Operational or maintenance details should be easy to accommodate on top of the issues handled in this article. The knowledge of rare event handling techniques may be necessary. For the operator it is necessary to perform the required tests and documentation after demands in a proper way.

1 INTRODUCTION

For Safety Instrumented Systems, demands on the Safety Function are obviously crucial and may lead to hazards if the Safety System does not react in the specified way. Safety-critical component failures are often not detectable during normal operation. For such systems, if demands happen relatively seldom proof tests may be specified which detect the failures. Obviously proof tests should be performed more frequent than the occurrence of demands. Systems where this is clearly possible are said to be operated in low demand mode. Fire detection represents an example for such a system.

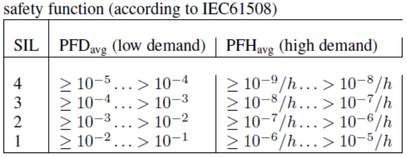
On the other hand systems exist where demands occur relatively frequent and proof tests with an even higher frequency do not make sense. The safety protection must be established in different ways, e.g. through redundancy. Such systems are said to be operated in high demand mode. An example of such a system is given by railway interlocking systems.

Safety Standards like (IEC61508 2010) treat the two demand modes as completely distinguishable with requirements that seem to be separate from each other. Table 1 shows the target failure measures for both low and high demand mode. For low demand mode the average Probability of Failure on Demand (PFD) is used and for high demand mode the average frequency for dangerous failures (PFH). Note that the latter is called Tolerable Hazard Rate (THR) in the railway industry (see (EN50126 1999)). Note also that the PFD cannot directly be used as acceptance criteria - the expected demand rate needs always to be specified. (IEC61508) uses a criterion $\delta < 1y$ (with δ: demand frequency) for the low demand range.

In reality systems exist, which cannot be clearly placed and might be called "intermediate demand mode systems". The present paper discusses this intermediate mode.

The issue of utilising demands as test has not been discussed extensively, but some authors have addressed it with varying focus. In (L.F.Oliveira, R.Youngblood, & P.F.F.Melo 1990) similar systems as the one discussed here have been analysed. More recently (Y.Liu & M.Rausand 2011) have taken up the issue again using similar systems but focusing on the demand duration. All publications that we are aware of are restricted to the Markov assumption which can be overcome using the analysis techniques discussed here.

Table 1: Safety Integrity Levels - target failure measures for a safety function (according to IEC61508)



SIL	PFD_{avg} (low demand)	PFH_{avg} (high demand)
4	$\geq 10^{-5} \ldots > 10^{-4}$	$\geq 10^{-9}/h \ldots > 10^{-8}/h$
3	$\geq 10^{-4} \ldots > 10^{-3}$	$\geq 10^{-8}/h \ldots > 10^{-7}/h$
2	$\geq 10^{-3} \ldots > 10^{-2}$	$\geq 10^{-7}/h \ldots > 10^{-6}/h$
1	$\geq 10^{-2} \ldots > 10^{-1}$	$\geq 10^{-6}/h \ldots > 10^{-5}/h$

2 THE MODELS

The analysis of intermediate demand mode systems is not straight forward due to the fact that there is a combination of periodic tests, repair times and demands. The latter are at least not periodic and are often assumed random, with a constant demand rate δ. In the extreme regions of (very) low demand rate or (very) high demand rate the system reliability can be readily approximated to a good level of accuracy (see Section 2.1). Another complication is given by the component and system level of detail. While failures, repair and proof testing happens on component level, demand and hazards happen on system level. Component level analysis can be performed by (partial) Markov Analysis, but the extension to the system level renders the analysis at least rather complex and limited to the Markov assumptions.

One method which overcomes all these difficulties is given by Discrete Event Simulation. It shall be demonstrated that even the Rare Events Problem (see (rareEvent 2015)), which is often a challenge in safety system analysis based on simulation can be solved in a satisfactory way.

As the system to be analysed here clearly involves states, generalised state modelling represents a good choice for model representation both on the component and on the system level. The following generalisations with respect to standard Markov State Models are utilised

- The standard Markov assumption that a state transition is only dependent on the current state is not needed. This means also that the involved statistical distributions do not need to be exponential.
- States can have a structure including sub-state systems as serial or parallel systems. This feature is implemented to counter the general tendency that 'flat' state systems can get rather involved even with a moderate amount of states. For the present purpose components are implemented as parallel sub-state systems. The system level issues are modelled in another parallel sub-state system. In this way the model is kept modular, easy to understand and straightforward to extend to e.g. other system configurations like 2oo4.
- States can have variables related to the whole state system or to sub-systems. This feature turns states into pseudo states in the sense that a state may contain many states as always only the state occupation together with all related variable values fully define the state.

State models thus generalised where proposed by Harel(see (Harel 1987)), which represents also the implementation chosen in this project. The modelling techniques resembles the Petri Net Models (see (Y.Dutuit, F.Innal, A.Rauzy, & J.Signoret 2008)) which have recently been suggested for safety sytem calculations. Both modelling techniques fall into the same class of state-based discrete event simulation, but we believe that the Harel State Charts used here are more intuitive to understand and communicate.

On the component level a rather simple repairable component is modelled. Failures happen with a constant rate λ and are assumed hidden until they are detected by either a demand or a test. Repair takes a time MTTR $= 1/\mu$. In the present article we assume that also this time is exponentially distributed. The simplicity of the model is mostly triggered by the wish to be able to compare our results with previously published results. Most assumptions can be made less stringent and more realistic within the framework of the present analysis.

The Harel State Chart model simulated by ExtendSim for one component is shown in figure 1. The model shows the main states working, undetected failure and repair. The model knows if a failure is detected by demand or proof-test, and is aware of possible demands during the repair time of the system.

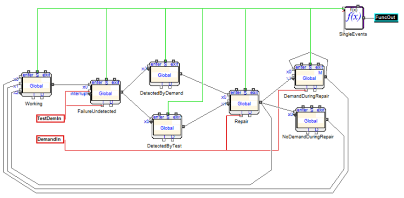


Figure 1: State model of component sub-system

Note that this component model has two inputs for the triggers when proof tests are performed or when a demand happens. These are system properties which must therefore come from the super model.

The model of figure 1 runs into a rare-event problem (see (rareEvents 2015)) for high demand rates. This rare event problem is caused by the fact that most demands find the system with all components working and only relatively few demands find one component in the failed state - thereby detecting this failure and initiating repair actions. Even fewer demands cause a system hazard, namely the demands which find both components in a non-working state. Obviously, the system hazard represents the rare event and the many demands which find everything working represent the events which are not really interesting for the analysis, but which use up most of the processing time during a simulation. This problem description contains already the solution to the problem: demands do not really need to be made explicit when not needed - only when at least one component has failed the demand has a function to the system. Moreover, as it is assumed that demands arrive independently from each other, demand generation is not dependent on previous demands and it is thus sufficient to calculate the next demand when a situation arises where this needs to be known, namely when at least one component has failed. This strategy is followed in a variant model to figure 1, where the demand input is omitted and the time for the next demand is kept as a system variable. The time for the next demand is calculated by any component which fails and is available for all components in the system.

A similar rare event problem exists in the low demand mode region. When the demand frequency gets low the hazard frequency gets likewise low, but system proof tests are still performed using valuable processing time. Similar to the discussion above, observing that most tests are not actually important for the analysis (namely the tests when all components are working), tests can simply be generated when needed, i.e. when at least one component has failed. In the case of the proof test there is complete dependency between tests, such that it is again possible to calculate the next proof test time at any time of the simulation using the formula 1.

$$t_{\text{nextTest}} = t + \tau - (t \mod \tau) \quad (1)$$

Also in this case the test input is omitted and the time for the next test is kept as a system variable which is only updated 'just in time', when at least one component fails.

In high demand mode a single component system does not really make sense in critical applications: either the failure mode in question can be excluded as incredible or redundancy is needed as it is impossible to detect failures and bring the system into a safe state if there is only one component and a high demand frequency. This article is restricted to two component systems as the simplest extension to a single system. The two component system is shown in figure 2 and follows the same rules as given in (L.F.Oliveira, R.Youngblood, & P.F.F.Melo 1990). "C1" and "C2" represents the single channel system illustrated in figure 2.

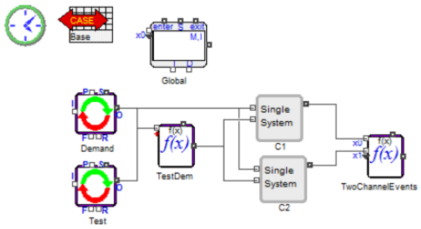


Figure 2: Direct model - reliability model of two component system

We distinguish between two models:

online model During repair the system is fully in use. This includes also the possibility that demands are received during repair, even if both components are not working.

offline model The system is still in use if one component has failed. If both components have failed and the failures are detected, the system is taken offline for repair.

The related state diagram is shown in figure 3, implementing the states

State 1 both channels are up

State 2 one channel is up, and the other is down, but failure is undetected

State 3 both channels are down, but failures are undetected

State 4 one channel is up, and the other is under repair (its failure has been detected due to demand)

State 5 one channel is down, but undetected, and the other is under repair

State 6 both channels are down, and their failures have been detected due to demand. Note that the transitions from state 6 are somewhat different from (L.F.Oliveira, R.Youngblood, & P.F.F.Melo 1990) due to the fact we assume that both repairs can be done simultaneously.

Figure 2 represents the two component system model while figure 3 represents the component sub-model, which resides in the blocks ”C1” and ”C2” of the system model. The two figures 2 and 3 illustrate very well the different approach in Harel State Charts modelling compared to traditional state charts. In many ways the system model of figure 2 resembles a Reliability Block Diagram (ref. (A.Høyland & M.Rausand 1994)), but it is in fact more than that because the ”TwoChannelEvents” block keeps track of which state each of the components are in at all times. In that way this block contains the relevant states that are illustrated in the state diagram. The model in figure 2 is very well modularised and can be extended to more components in a straightforward way. The model of figure 3 does not offer that. Moreover, Markov modelling is also limited when it comes to the choice of distributions, maintenance details and system safety strategy. This simplified model has mainly been chosen for comparison with previous work.

The blocks in addition to ”C1” and ”C2” in the system model of figure 2 have the following purpose

Global Global variable settings which are available for all substate models. In our case these are λ, δ, μ, τ, $t_{\text{nextDemand}}$ and t_{nextTest}. Note that the first three of these could be component level variables (and be chosen different from each other). Here they are only added for convenience, since they are chosen equal for all components.

Demand The demand generator. This block triggers demands and communicates them to the components.

Test The test generator. This block triggers proof tests and communicates them to the components.

TestDem Since the component blocks need only to know the demands and the combined demands and tests, “TestDem” generates the combined signal from these triggers.

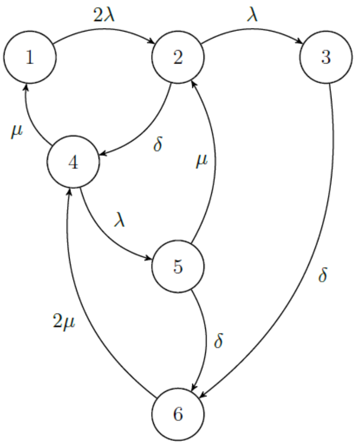


Figure 3: State diagram of the two-channel model where each transfer of state is exponentially distributed with the corresponding parameter.

The model of figure 4 represents the model without treatment of rare events. For optimised treatment of rare events the model must be modified into the model shown in figure 4.

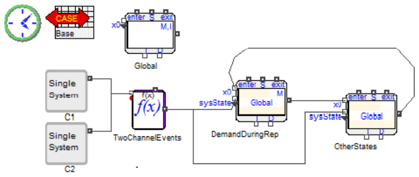


Figure 4: Reliability model of two component system optimised for dealing with the rare event problem

Clearly the explicit generation of demands and tests is not present any more in figure 4. In the case of online repair another rare event problem is revealed, namely the demands during repair of both components, which become many events in the case of high demand frequency. Instead of explicitly generating these demands, only the state 'DemandDuringRep' is modelled. When this state finishes a representative number of additional demands is sampled through a Poisson distribution according to the demand rate and the time interval. This issue represents a solution to a system level rare event problem.

With respect to figure 3 the hazard rate for an offline case is found by:

$$\hat{\eta}_{\text{offline}} = \frac{\#\text{ events in states 3 and 5}}{\text{simulation time}} \tag{2}$$

because these two states represent hazardous events when a demand occurs. For the online case the hazard rate is:

$$\hat{\eta}_{\text{online}} = \frac{\#\text{ events in states 3, 5 and 6}}{\text{simulation time}} \tag{3}$$

where the additional state 6 represents the additional demands during repair discussed above.

2.1 Asymptotes

The asymptotes for the hazard rate for small and high demand rates can be calculated analytically.

In the low demand range the demands de-couple from the failures such that the traditional PFD can be calculated for a two channel system. The hazard rate becomes (see (IEC 61508), part 6, B.3.2.2)

$$\eta_{\text{low demand}} \simeq \delta{\cdot}2\lambda^2\left(\frac{\tau}{2}+\frac{1}{\mu}\right)\cdot\left(\frac{\tau}{3}+\frac{1}{\mu}\right) \tag{4}$$

This formula can be derived through Markov analysis or through reasoning about failure rates and equivalent down times

In the high demand range the repair time dominates the hazards. In the case of offline repair the state 5 of figure 3 dominates. I.e. one channel is under repair and the other fails and the failure is detected by the demand. This leads to the formula

$$\eta_{\text{high demand offline}} \simeq \frac{2\lambda^2\mu}{\lambda^2+2\lambda\mu+\mu^2} \tag{5}$$

In the case of online repair, the additional failures during the time when both components are repaired come in addition and are dominant for very high demand rates. The respective formula becomes

$$\eta_{\text{high demand online}} \simeq \frac{\delta\lambda^2}{\lambda^2+2\lambda\mu+\mu^2} \tag{6}$$

The last two equations are either obtained through calculating the equilibrium Markov solutions or through approximations with respect to repairable systems (see also (L.F.Oliveira, R.Youngblood, & P.F.F.Melo 1990)).

3 RESULTS

The problem at hand and the models introduced in section 2 contain the following parameters

Failure rate λ The rate at which the components of the system fail. It is assumed that λ is constant and that the failure rates of all components of the system are equal.

Demand rate δ The rate of demands on the safety system. This is a system parameter.

Proof test interval τ The interval for proof tests of the system components. It is assumed that proof tests are performed periodically and that all components are tested at the same time.

Repair rate μ The repair rate $\mu = 1/\text{MTTR}$ for a component after a failure is detected. Within generalised state modelling it is not necessary to assume a constant repair rate. In any case, when a failure mode is known it is often more realistic to assume a constant repair time. Still, in this paper a constant rate is assumed for easy comparison with previous work.

Without loss of generality $\lambda = 1$ is set throughout this paper, i.e. the time unit is set equal to the mean time between failures of a single component. As repair rate $\mu = 200$ is used as a 'typical' repair rate.

Results for $\tau = 0.1$ are shown in figure 5.

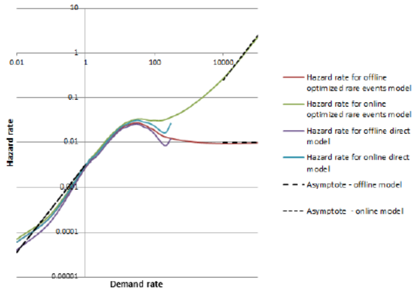


Figure 5: Offline repair results for the direct model and the rare events optimised model for $\tau = 0.1$ over a wide demand range

It is clearly seen that the direct model is limited in the demand range at least in the high demand mode area both for online and offline repair. For demand rates above about 100λ the simulation times for the direct model become too long to be practicably feasible. In the area where both models can produce results, the results coincide well within statistical accuracy. The rare events problem in the low demand range does not become visible for demand rates down to 0.01λ.

With the choice of time scale as λ and $\mu = 200$ as typical values, this leaves two parameters to be varied in a suitable range and the resulting system hazard rate. The results for offline repair are shown in figure 6. Similar results for online repair are shown in figure 7. As the frequency of proof-tests decreases, the intermediate mode has a greater effect on the system. There is a larger deviation from the simulated results and the asymptotic formulas normally used for PFD and PFH calculation.

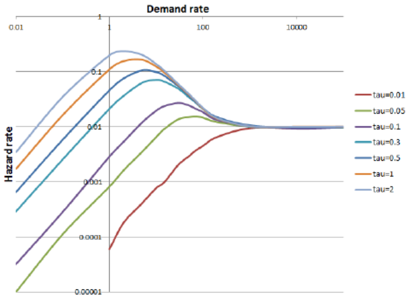
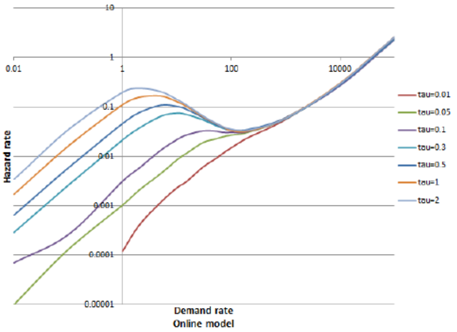


Figure 6: Offline repair results using the rare events optimised model for various τ over a wide demand range. $\mu = 200$



The asymptotes as discussed in section 2.1 are confirmed well in all plots, as illustrated for one case ($\tau = 0.1$) in figure 5. When the demand rate increases the hazard rate for the offline model approaches towards the hazard rate given by the asymptotic equation 5. For the online model, equation 6 shows that the hazard rate increases with the demand rate.

The plots exhibit an unexpected pair of extreme points which are most marked for large proof test times. The top point is due to the fact that demands become effective as tests when the demand rate increases. In this way failures of single components are detected earlier, reducing the chance for double failures and hazards. On the other hand there is the repair time which contradicts this effect since failures and demands during repair can increase the hazard rate. The asymptotic formula which explains the low demand region does not take these effects into account. The effect diminishes when the proof test interval is reduced and seems to vanish altogether for very small proof test intervals. There is a similar dependency on μ which is not elaborated here. Together these results confirm the above explanation of the pair of extreme points.

4 DISCUSSION AND CONCLUSIONS

The results from section 3 show clearly that the hazard rate exhibits unexpected behaviour in the intermediate demand region. As far as can be seen from the analysis, the standard PFD formulas are usable, but they become exceedingly conservative as one moves into the intermediate demand region. According to [1] the PFH formula should be used for $\delta > 1y$. In this case the asymptotic hazard rate renders non-conservative results in the intermediate region . Therefore, whenever a system seems to be operated in this demand mode, or even only close to it, it is advisable to perform more accurate analysis compared to standard PFD and PFH formulas. It has been demonstrated that such analysis is readily feasible using modern simulation tools. Operational or maintenance details should be easy to accommodate on top of the issues handled in this article. The knowledge of rare event handling techniques may be necessary.

As the usage of demands as effective tests is crucial for gaining the advantage of an improved hazard rate it is important that

- demands are properly recorded in relevant systems
- the necessary tests on the components are performed and the results recorded, such that the demand can actually be used as an effective test

The analysis performed here can be extended towards a number of additional points in order to better understand the details. Without claiming completeness the following issues would be interesting

- systematic analysis of the dependencies on the repair rate μ
- more realistic distributions (e.g. constant repair time)
- other system architectures (e.g. more general koon architectures including also common cause failures)
- other possible maintenance strategies (e.g. take the system offline when only one working component is left)

Bibliography

[1] IEC 61508. *Functional safety of electrical/ electronic programmable electronic safety-related systems. Part 1 General requirements.* IEC. Edition 2.0. (2010-04)

[2] H. Jin, M.A. Lundteigen, M. Rausand. *Reliability performance of safety instrumented systems: A common approach for both low- and high demand mode of operation.* Reliability Engineering and System Safety. 96, 365-373, 2011.

[3] L.F. Oliveira, J.D. Amaral Netto. *Influence of the demand rate and repair rate on the reliability of a single-channel protective system.* Reliability Engineering 17, 267-276, 1987.

[4] L.F. Oliveira, R. Youngblood, P.F.F Melo. *Hazard rate of a plant equipped with a two-channel protective system subject to a high demand rate.* Reliability Engineering and System Safety 28, 35-58, 1990.

[5] D. Harel *Statecharts: A visual formalism for complex systems* Science of Computer Programming 8, 231-274, 1987.

[6] The MTL Instruments Group. *An introduction to Functional Safety and IEC 61508.* AN9025. Retrieved 27th of April 2015 from http://www.mtl-inst.com/images/uploads/datasheets/App_Notes/AN9025.pdf

[7] F. Brissaud, A. Barros, C. Brenguer. *Probability of failure of safety-critical systems subject to partial tests* Retrieved 13th of Feb 2015 from http://arxiv.org/ftp/arxiv/papers/1007/1007.5448.pdf.

[8] Failure rate. *Wikipedia.* Retrieved 2nd of March 2015 from http://en.wikipedia.org/wiki/Failure_rate

[9] M. Spellemaeker. Retrieved 26th of March 2015 from www.jlab.org/eng/ssg/safety/sil.pdf

[10] J. Börcksök. Retrivied 1st of April 2015 from http://www.iceweb.com.au/sis/Hima/HIMA%20-%20Comparison%20of%20PFD%20Calculation.pdf

[11] State diagram. Harel statecharts. Retrieved 1st of April 2015 from http://en.wikipedia.org/wiki/State_diagram#Harel_statechart

[12] M. Rausand. Retrieved 1st of June 2015 from http://frigg.ivt.ntnu.no/ross/srt/slides/chapt10-sis.pdf

[13] Description of Harel State in ExtendSim 9

[14] B. Franke. Retrieved 13th of September 2015 from http://www.inf.ed.ac.uk/teaching/courses/es/PDFs/lecture_4.pdf

[15] Montecarlo. Retrieved 14th of September from https://en.wikipedia.org/wiki/Monte_Carlo_method

[16] IEC 61511 *Functional safety - Safety instrumented systems for the process industry sector.* IEC. Edition 1 (2003)

[17] EN50126 *Railway applications - The specification and demonstration of Reliability, Availability, Maintainability and Safety (RAMS).* CENELEC. Edition 1 (1999)

[18] EN50129 *Railway applications - Communication, signalling and processing systems. Safety related electronic systems for signalling.* CENELEC. Edition 1 (2003)

[19] A. Høyland, M. Rausand *System reliability theory; models, statistical methods and applications.* Wiley. Edition 2 (1994)

[20] Y. Liu, M. Rausand *Reliability assessment of safety instrumented systems subject to different demand modes.* Journal of Loss Prevention in the Process Industries. 24, 49-56, 2011.

[21] *https://en.wikipedia.org/wiki/Rare_Event_Sampling* retrieved 20th of May 2015

[22] J. Bukowsky *Incorporating process demand into models for assessment of safety system performance.* Proceedings of RAM'06 Symposium. 2006

[23] Y. Dutuit, F. Innal, A. Rauzy, J. Signoret *Probabilistic assessments in relationship with safety integrity levels by using fault trees.* Reliability Engineering and System Safety. 93, 1867-1876, 2008

[24] S. Eisinger *Discrete Event Simulation.* Det Norske Veritas, Høvik, 1997. Obtained from ExtendSim.