UNIVERSITÀ
DEGLI STUDI
FIRENZE

**DOTTORATO DI RICERCA IN
INGEGNERIA INDUSTRIALE**

CICLO XXIX

**COORDINATORE
Prof. Maurizio DE LUCIA**

SIMULATION MODELS AND RELIABILITY ASSESSMENT FOR GAS TURBINE AUXILIARY SYSTEMS

Settore Scientifico Disciplinare ING-INF/07

Dottorando	Tutore
Dott. Venzi Matteo	Prof. Catelani Marcantonio

**Coordinatore
Prof. De Lucia Maurizio**

Anni 2014/2016

© Università degli Studi di Firenze – Faculty of Engineering
Via di Santa Marta, 3, 50139 Firenze, Italy

Tutti i diritti riservati. Nessuna parte del testo può essere riprodotta o trasmessa in qualsiasi forma o con qualsiasi mezzo, elettronico o meccanico, incluso le fotocopie, la trasmissione fac simile, la registrazione, il riadattamento o l' uso di qualsiasi sistema di immagazzinamento e recupero di informazioni, senza il permesso scritto dell' editore .

All rights reserved. No part of the publication may be reproduced in any form by print, photoprint, microfilm, electronic or any other means without written permission from the publisher.

“Never say never, because limits, like fears, are often just an illusion. “ – M.J.

Special thanks to Prof. M.Catelani, Ing. L. Ciani, my mom Laura, my grandparents Nino and Palmina and my closest friends for the support and the affection. You played a fundamental role to achieve this goal.

Table of Contents

Chapter 6

Abstract

The interest in RAMS (Reliability, Availability, Maintainability and Safety) and diagnostics parameters is growing in many different manufacturing fields. These branches of knowledge are nowadays crucial and play a fundamental role in industrial engineering becoming focal part of performance requirements.

Modern technologies and business requirements are producing a growth in variety and complexity of manufacturing product and this trend increased number and variety of failures. System downtime and unplanned outages massively affect plant productivity. In many Oil&Gas applications an emergency shutdown produces an interruption of normal running operation, a considerable productivity reduction and a loss of thousands dollars [1-2].

This is the reason why RAMS disciplines together with fault diagnosis and condition monitoring are almost mandatory in Oil&Gas applications where products are forced to endure extreme process and environmental conditions [3].

This thesis is focused on availability improvement and takes into account maintainability and, in particular, reliability roles in order to achieve this kind of target.

The goal is to develop a procedure for availability improvement that engineers may used during the early stages of product design.

Availability means that a system is “on-line” if it is involved in continuous running condition or “ready to use” in case of on-demand” usage.

As said before, in modern systems there are a great variety of factors that can take a system off-line, ranging from scheduled maintenance downtime to catastrophic failures.

The goal of improving system availability is to detect incipient failures, minimize downtime and minimize the time needed to restore the system to normal working conditions.

Obviously the margin of downtime tolerance is directly associated with the system application and this requirement impose the complexity and the corresponding cost of the solution [4-6].

Reliability prediction is the main focus of this study since it turned out to be best method in RAM (Reliability, Availability and Maintainability) analysis for industrial applications: reliability prediction is very helpful in order to evaluate design feasibility, compare design choices, identify potential failure areas, trade-off system design factors and track reliability improvement.

This is the reason why the best solution to improve system availability in the early product design stages turned out to be reliability-oriented since it provides reliability feedback to design engineers in order to reduce re-design costs and time for upgrades.

This thesis is organized as follows: Chapter 1 contains a brief description of Life Data Analysis focusing on the comparison of two failure distributions, Exponential and Weibull.

The second Chapter shows the best Availability improvement methods starting from standby redundancy and comparing cold and warm standby solutions.

Chapter 3 deepens the Reliability Allocation procedures starting from a review of the methods described in literature and showing a new solution to achieve allocation parameters in complex systems; this Chapter contains also the description of a new Reliability Importance procedure (Credible Improvement Potential) and its application on Auxiliary Systems of a gas turbine.

Chapter 4 describes the Condition-based Maintenance using Markov models with some applications in case of complex repair solutions and standby spares; Chapter 5 shows the basis of fault detection, isolation, reconfiguration, diagnostics and Condition Monitoring. This Chapter contains both on-board and logic solver diagnostics with a detailed application on a gas turbine safety loop and corresponding Probability of Failure on Demand (PFD) assessment [7-8].

The final Chapter describes the Reliability Assessment Loop with the brand new approach proposed and show the potential of the tool that was developed to achieve a reliability prediction in the early product design stages.

Chapter 1

Life Data Analysis

1.1 Reliability introduction

The analysis of RAMS – Reliability, Availability, Maintainability and Safety requires and introduction of some basic concepts that are relevant in particular for reliability purposes i.e. time to failure T, reliability function R(t) and failure rate λ(t).
The following definitions are the reference for the whole document.

The state of an item at time t may be described by the state variable X(t), generally a random variable [9]:

$$X(t) = \begin{cases} 1 & \text{if the item is functioning at time t} \\ 0 & \text{if the item is in a failed state at time t} \end{cases} \tag{1}$$

Time to failure T is a random variable which defines the time period starting from the instant in which the item is put into operation up to the instant it fails for the first time.
If T is continuously distributed with probability density f(t) and distribution function is the following:

$$F(t) = \Pr(T \le t) = \int_0^t f(u)du \quad \text{for } t > 0 \tag{2}$$

While the probability density function is defined as follows:

$$f(t) = \frac{dF(t)}{dt} = \lim_{\Delta t \to 0} \frac{F(t+\Delta t) - F(t)}{\Delta t} = \lim_{\Delta t \to 0} \frac{\Pr\,(t<T<t+\Delta t)}{\Delta t} \tag{3}$$

F(t) represents the probability that the item fails within the time interval (0, t].
The reliability function of an item is defined by:

$$R(t) = 1 - F(t) = \Pr(T > t) \quad \text{for } t > 0 \tag{4}$$

$$R(t) = \int_t^{\infty} f(u)\,du \tag{5}$$

R(t) is the probability that the item does not fail in the time interval (0, t] or the probability that the item survives the time (0, t] and is still functioning at time t [9].

The probability that an item will fail in the time interval (t, t+Δt], with the hypothesis that the item is functioning at time t, is the following:

$$Pr(t < T \leq t + \Delta t | T > t) = \frac{F(t+\Delta t)-F(t)}{R(t)} \tag{6}$$

By dividing this probability by the length of the time interval Δt, and letting Δt→0, the following failure rate function λ(t) of the item is obtained:

$$\lambda(t) = \lim_{\Delta t \to 0} \frac{F(t+\Delta t)-F(t)}{\Delta t} \frac{1}{R(t)} = \frac{f(t)}{R(t)} \tag{7}$$

Considering Eq. 3:

$$\mathrm{f(t)} = \frac{\mathrm{dF(t)}}{\mathrm{dt}} = \frac{\mathrm{d(1-R(t))}}{\mathrm{dt}} = -\mathrm{R'(t)} \tag{8}$$

Then considering Eq. 7:

$$\lambda(\mathrm{t}) = -\frac{\mathrm{R'(t)}}{\mathrm{R(t)}} = -\frac{\mathrm{d}}{\mathrm{dt}}\ln(\mathrm{t}) \tag{9}$$

Since R(0)=1, then:

$$\int_0^{\mathrm{t}} \lambda(\mathrm{u})\,\mathrm{du} = -\ln(\mathrm{R(t)}) \tag{10}$$

And

$$\mathrm{R(t)} = \exp\left(-\int_0^{\mathrm{t}} \lambda(\mathrm{u})\,\mathrm{du}\right) \tag{11}$$

The reliability function R(t) and distribution function F(t) are therefore uniquely determined by the failure rate function λ(t) [9].

To determine the form of λ(t) it is necessary to carry out this experiment: considering n identical items functioning at time t=0, n(i) the number of down elements and T_{ij} the time when the j-th element is functioning in the interval i, then it’s possible to define:

$$\lambda(\mathrm{i}) = \frac{\mathrm{n(i)}}{\sum_{\mathrm{j=1}}^{\mathrm{n}} \mathrm{T_{ij}}} \tag{12}$$

1.2 Bathtub curve

The bathtub curve shown in Figure 1 describes the trend of the failure rate of an entire population of products over time [2]. The bathtub curve is characterized by three sections:

- Infant mortality period with a decreasing failure rate;
- Normal life period (also known as "useful life") with a low, relatively constant failure rate;
- Wear-out period that exhibits an increasing failure rate.

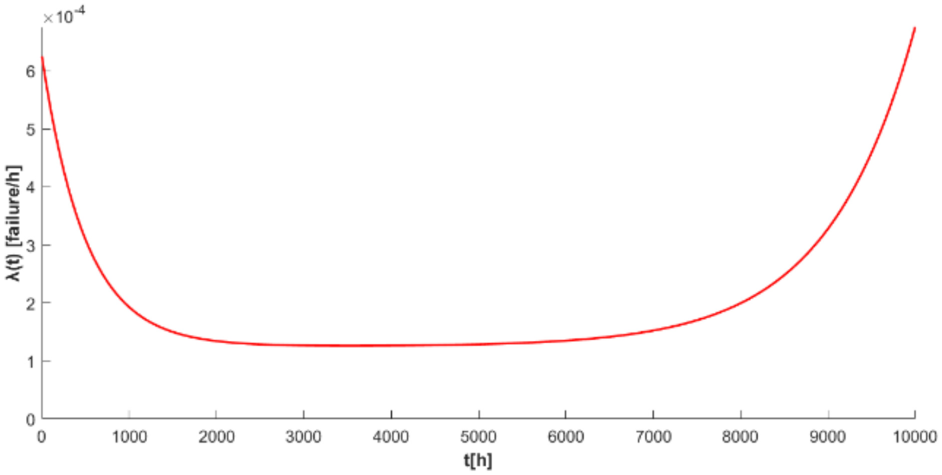


Figure 1. The bathtub curve

Failures during infant mortality are highly undesirable and are always caused by material defects, design blunders, errors in assembly, etc.

Normal life failures are normally considered to be random cases of "stress exceeding strength."

Wear-out is a fact of life due to fatigue or deterioration of materials [10].

As said before, failures during infant mortality are caused by defects designed/built into a product. Therefore, to avoid infant mortalities, the product manufacturer must determine methods to eliminate these defects: appropriate specifications, adequate design tolerance and sufficient component derating can help and should always be used but even the best design intent can fail to cover all possible interactions of components in operation.

In addition to the best design approaches, stress testing should be started at the earliest development phases in order to evaluate design weaknesses and uncover specific assembly and materials problems; these tests are called HALT (Highly Accelerated Life Test) or HAST (Highly Accelerated Stress Test) and should be applied with increasing stress levels until failures arise [11-12].

After the beginning of the manufacturing process, a stress test can still be valuable; there are two distinct uses for stress testing in production. One purpose (often called HASA, Highly Accelerated Stress Audit) is to identify defects caused by assembly or material variations that can lead to failure and to take action to remove the root causes of these defects. The other purpose (often called “burn-in”) is to use stress tests as an ongoing screen to weed out defects in a product where the root causes cannot be eliminated [13].

Some reliability specialists like to point out that real products don't exhibit constant failure rates. This is quite true for a mechanical part where wear-out is the primary failure mode. And all kinds of parts, mechanical and electronic, are subject to infant mortality failures from intrinsic defects. There are other cases, especially in electronic products, where a "constant" failure rate may be appropriate (although approximate). This is the basis for standards and other methods to estimate system failure rates from consideration of the types and quantities of components used.

For many electronic components, wear-out is not a practical failure mode. The time that the product is in use is significantly shorter than the time it takes to reach wear-out modes.

In the long run, everything wears out. For many electronic designs, wear-out will occur after a long, reasonable use-life. For many mechanical assemblies, the wear-out time will be less than the desired operational life of the whole product and replacement of failed assemblies can be used to extend the operational life of the product. The shortest-lived component will determine the location of the wear-out time in a given product so in designing a product, the engineer must assure that it lasts long enough to provide a useful service life [9].

1.3 Exponential distribution

The failure rate of an item with exponential life distribution is constant (i.e., independent of time), so it may be a realistic life distribution for an item during its useful life period [9-10].

The probability that an item is working for t time units is therefore equal to the probability that the item is still working in a different time interval of length t. So exponential distribution doesn't consider the degradation of items.

Consider that the random variable time to failure T follows an exponential probability density function with parameter λ:

$$f(t) = \begin{cases} \lambda e^{-\lambda t} & \text{for } t > 0 \text{ and } \lambda > 0 \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

As shown in Fig. 2, all curves are exponentials and increasing the parameter λ, the curves decrease quickly. The reliability function of the item is the integral of equation (13), so:

$$R(t) = \int_t^{\infty} \lambda e^{-\lambda u} du = e^{-\lambda t} \tag{14}$$

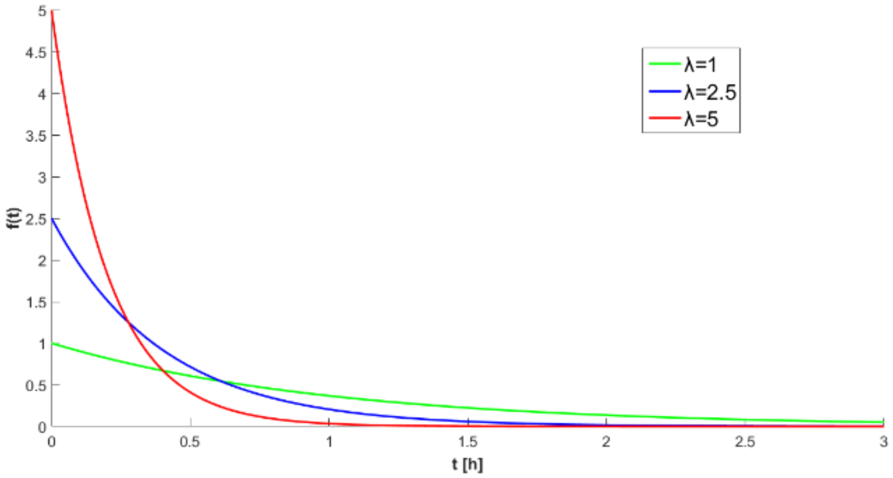


Figure 2. Exponential probability density function varying λ

Fig. 3 shows some functions starting at their maximum (1) and decreasing thereafter monotonically: the drop speed is function of λ (larger values of λ correspond to faster decreases).

The failure rate function, according with Eq. 7, is the following:

$$\lambda(t) = \frac{f(t)}{R(t)} = \lambda \tag{15}$$

The exponential distribution is the most used life distribution in applied reliability analysis since it is quite easy to manage and it represents a realistic lifetime model for a wide variety of items.

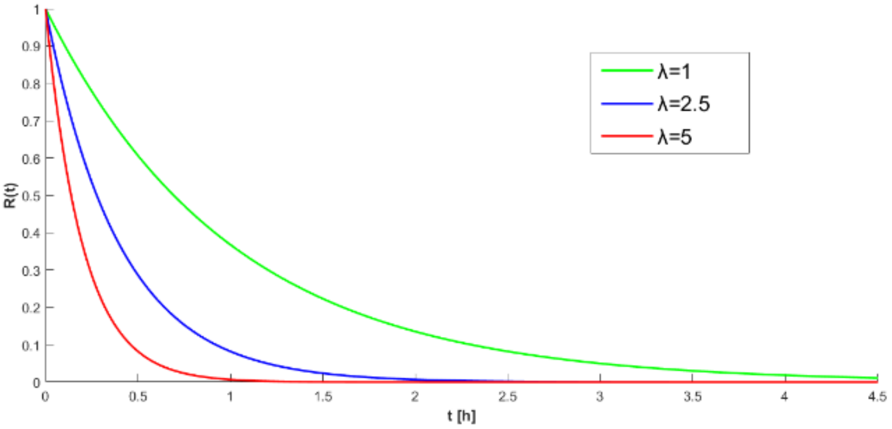


Figure 3. Exponential reliability function varying λ

1.4 Weibull distribution

The Weibull distribution is one of the most widely used life distributions in reliability analysis. It is a very flexible distribution since it is based on different parameters and it can model different behaviours of failure rate functions [9].

The time to failure of an item T is said to be Weibull distributed with parameters β and η if the distribution function is given by:

$$F(t) = \begin{cases} 1 - e^{-\left(\frac{t}{\eta}\right)^{\beta}} & \text{if } t > 0 \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

The corresponding density function is the following:

$$f(t) = \begin{cases} \frac{\beta}{\eta}\left(\frac{t}{\eta}\right)^{\beta-1} e^{-\left(\frac{t}{\eta}\right)^{\beta}} & \text{for } t > 0 \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

Where η is a scale parameter and β is the shape parameter.

The reliability function is:

$$R(t) = e^{-\left(\frac{t}{\eta}\right)^{\beta}} \quad \text{for } t > 0 \tag{18}$$

And the failure rate function is:

$$\lambda(t) = \frac{\beta}{\eta}\left(\frac{t}{\eta}\right)^{\beta-1} \quad \text{for } t > 0 \tag{19}$$

The parameter β is a pure number, i.e. it is dimensionless [12]. Different values of the shape parameter β can have marked effects on the behaviour of the distribution.

In fact, some values of the shape parameter will reduce the distribution equations to those of other distributions: for example, when β = 1, the PDF of the three-parameter Weibull equation reduces to the two-parameter exponential distribution [14-15].

Figure 4 shows the effect on PDF of different values of the shape parameter (considering a constant value of the scale parameter η=1,5h): it can be observed that the shape of the PDF can assume a variety of forms based on the β value.

Figure 5 shows the same effects of β modulation on the reliability function (in case of constant value of the scale parameter η=4h): high β values ensure a higher reliability up to the crossroad, then R(t) curves with greater β quickly decrease.

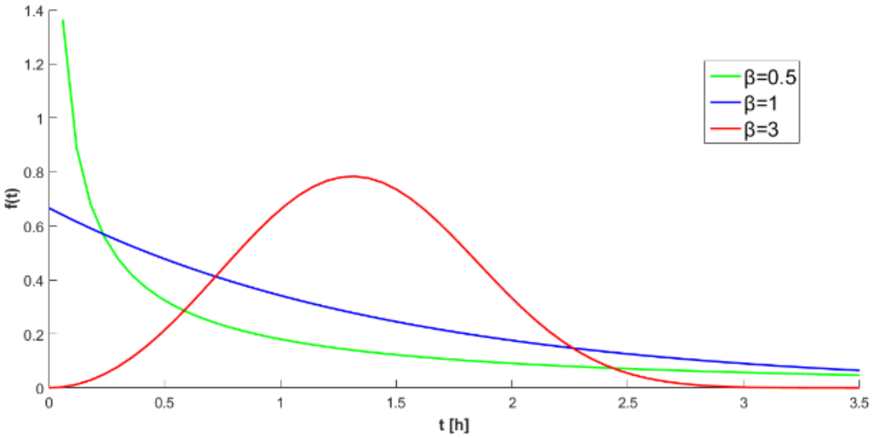


Figure 4. Weibull probability density function varying β

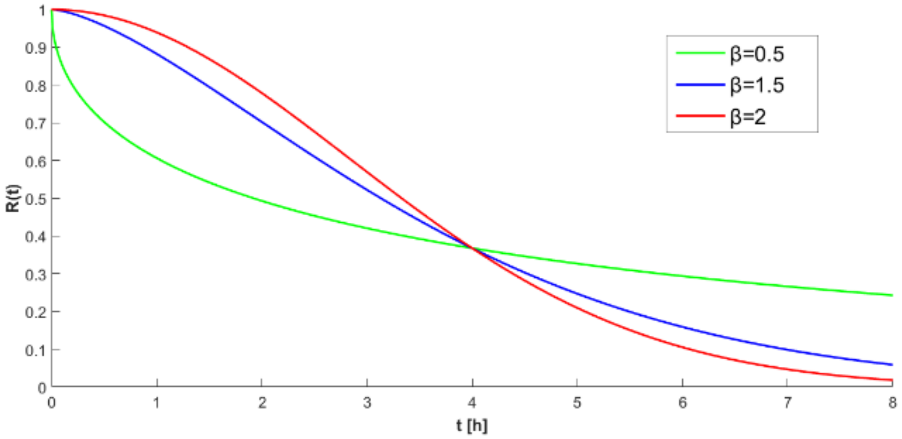


Figure 5. Weibull reliability function varying β

The modulation of β has consequences also on the failure rate function and this trend is shown in Fig. 6; this is one of the most important aspects of the Weibull distribution.

As shown in the plot, Weibull distributions with β<1 have a failure rate that decreases with time, the same trend of the first part of bathtub curve in case of infantile or early-life failures.

For β values close or equal to 1 the Weibull distributions have a fairly constant failure rate, indicative of useful life or random failures.

Finally, Weibull distributions with β >1 have a failure rate that increases with time representing the typical trend of wear-out failures.

Therefore this behaviour comprises the three sections of the classic "bathtub curve": the plot of three Weibull distributions with respectively β <1, β =1 and β >1 would reproduce the whole trend of the standard bathtub curve [10].

Also the modulation of the scale parameter η can influence the trend of curves; η has the same unit of measure as T (hours) and a change in this parameter has the same effect on the distribution of a change in the x-axis scale.

An increase of η (holding β constant) has the effect of stretching out the PDF, as shown in Fig. 7. Since the area under a PDF curve has a constant value of one, the maximum of the PDF curve will decrease together with the growth of η, as indicated. If η is increased (decreased), while β is the same (β=4), the distribution gets stretched out to the right (left) and its maximum decreases (increases), while maintaining its shape. Figure 8 and Figure 9 show reliability and failure rate functions: a growth of the parameter η produces a corresponding increase of reliability and reduction of number of failures.

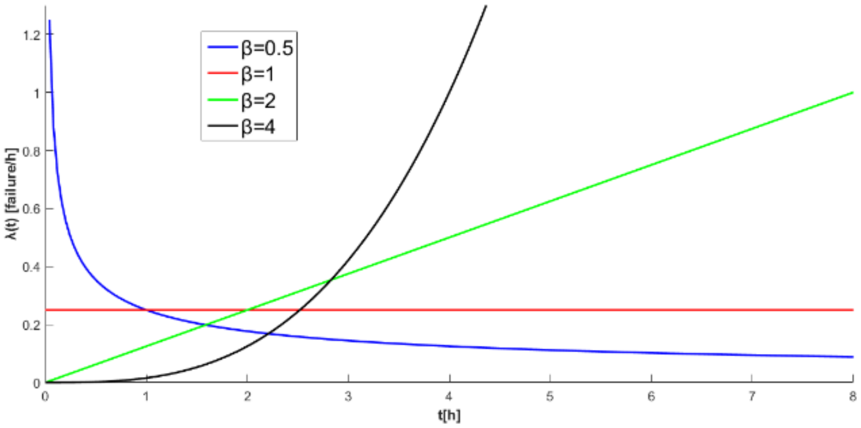


Figure 6. Weibull failure rate function varying β

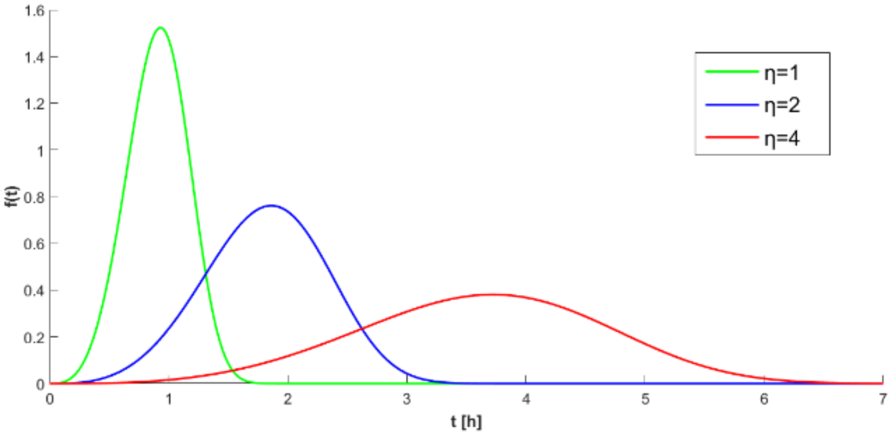


Figure 7. Weibull probability density function varying η

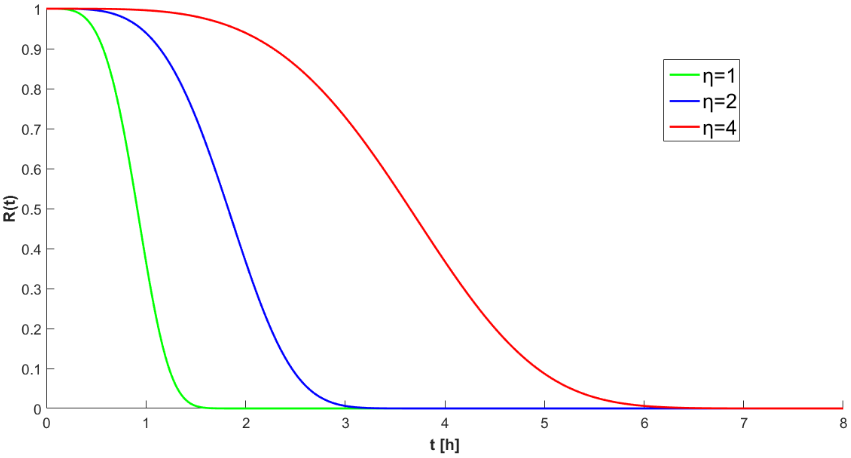


Figure 8. Weibull reliability function varying η

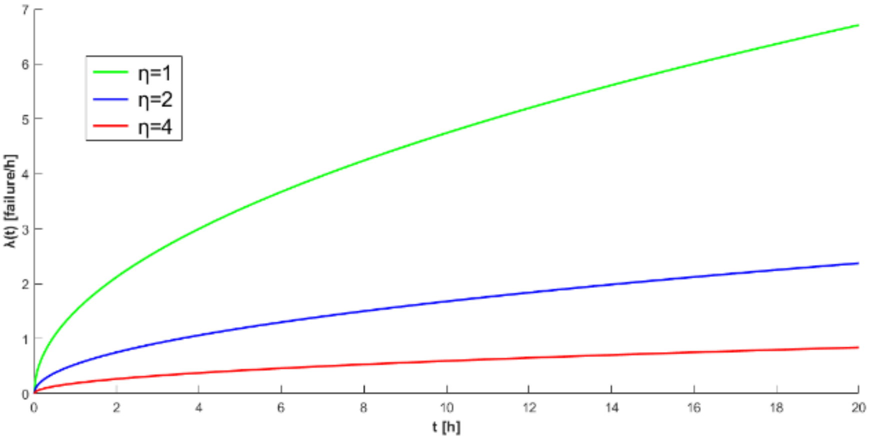


Figure 9. Weibull failure rate function varying η

A natural extension of this distribution is the three-parameter Weibull distribution (β, η, γ), where γ is called the location parameter.

In this case the reliability and the failure rate functions are the following:

$$R(t) = e^{-\left(\frac{t-\gamma}{\eta}\right)^{\beta}},\ \lambda(t) = \frac{\beta}{\eta}\left(\frac{t-\gamma}{\eta}\right)^{\beta-1} \text{ with } t \geq \gamma \tag{20}$$

The third parameter has the effect to move the distribution on the x-axis: if γ>0 the distribution is moved on the right whereas if γ<0 it is moved on the left.

1.5 Normal or Gaussian distribution

The most commonly used distribution in statistics is the normal distribution [9]. A random variable T is said to be normally distributed with mean μ and variance σ2 when the probability density of T is the following:

$$f(t) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(t-\mu)^2}{2\sigma^2}} \quad \text{for } t \in (-\infty, \infty) \tag{21}$$

If μ=0 and σ=1, then the distribution is called standard normal distribution and usually denoted by ϕ, the PDF function is:

$$\phi(t) = \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} \tag{22}$$

The reliability function is the following:

$$R(t) = \int_t^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \tag{23}$$

And the failure rate function of the normal distribution is:

$$\lambda(t) = \frac{e^{-\frac{(t-\mu)^2}{2\sigma^2}}}{\int_t^{\infty} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx} \tag{24}$$

Fig. 10 shows the curves of normal failure rate function with a constant mean value μ=0 in case of varying the standard deviation: an increase of the σ value produces a decrease of the number of failures and the curves loose the typical exponential trend.

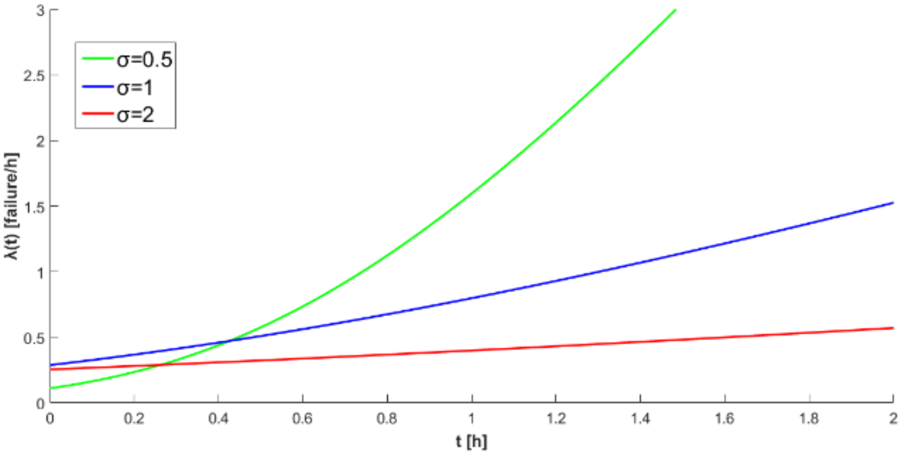


Figure 10. Normal failure rate function varying σ

1.6 Lognormal distribution

This distribution is used for the description of failures of devices characterized by a large wear out period [4]. The time to failure T of an item is said to be lognormally distributed with parameters μ and σ^2, if the random variable Y=ln(T) is normally (Gaussian) distributed with mean μ and variance σ^2 [9]. The probability density function of T is:

$$f(t) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma t} e^{-\frac{(\ln t - \mu)^2}{2\sigma^2}} & \text{for } t > 0 \\ 0 & \text{otherwise} \end{cases} \tag{25}$$

The reliability function is the following:

$$R(t) = \frac{1}{\sqrt{2\pi}\sigma} \int_t^{\infty} \frac{1}{x} e^{-\frac{[\ln (x) - \mu]^2}{2\sigma^2}} dx \tag{26}$$

The failure rate function of the lognormal distribution is:

$$\lambda(t) = \frac{\frac{1}{t} e^{-\frac{[\ln (t) - \mu]^2}{2\sigma^2}}}{\int_t^{\infty} \frac{1}{x} e^{-\frac{[\ln (x) - \mu]^2}{2\sigma^2}} dx} \tag{27}$$

Fig. 11 shows that increasing the standard deviation, the failure rate decreases; it corresponds to have a lower number of failures together with time increase. This distribution, due to its trend, describes the first part of the bathtube curve where the failures are caused by infant mortality.

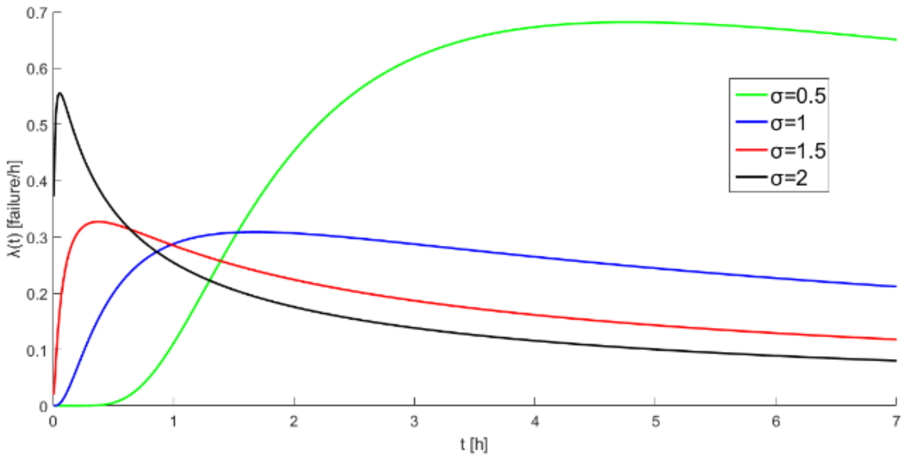


Figure 11. Lognormal failure rate function varying σ

1.7 Comparison between Exponential and Weibull distributions

Nowadays the assumption that most reliability engineering problems can be perfectly modelled by the exponential distribution is still widely held.

For the sake of simplicity, many insiders have embraced simple equations derived from the underlying assumption of an exponential distribution for different purposes such as reliability prediction, accelerated testing, reliability growth, maintainability and system reliability analyses. As said before, the exponential distribution models the behaviour of units that fail at a constant rate, regardless of the accumulated age: although this property greatly simplifies the analysis, it makes the distribution inappropriate for most reliability assessment because it does not apply to real world applications.

For example, if cars exhibited a constant failure rate, then the vehicle's mileage would not be a factor in the price of a used car because it would not affect the subsequent reliability of the vehicle: in other words, if a product can wear out over time, it should not have a constant failure rate. Similarly to cars, most items in this world are affected by wear out, even electronic components and non-physical assets such as computer software.

Anyway, despite the illustrated inadequacy of the exponential distribution to accurately model the behaviour of most products in the real world, the exponential assumption is still widely used in today's reliability practices, standards and methods [15-16].

The use of the Weibull distribution is more accurate and practical. It is shown that the Weibull best models the reliability, maintainability and availability of parts that have a variable failure rate over its useful life. The role of reliability analysis is to evaluate the behaviour of failure mechanisms on a part to understand failures and provide insights to design efficacy. It seems obvious that if a failure mechanism is incorrectly modelled then any information derived from that model is imperfect [17-19]. Graphically it can be shown that the exponential and Weibull probability density functions resulting from the same data are very different (Fig. 12).

If it is possible to assume that one or the other model is a correct characterization of the failure mechanism, then it is obvious that the wrong distribution is not close enough to be useful as an estimator. The impact of the differences between the reliability functions is shown in Fig. 13. Notice that the exponential reliability is an exponential curve for the whole time span: on the other hand, for lower values of time, Weibull guarantees a higher reliability value then decreases and after the curve crossroad the exponential distribution provides a higher reliability performance.

The comparative failure rates (Fig. 14) reveal quantifiable differences between the Weibull distribution and the exponential one: exponential considers a higher number of failure in the early time but Weibull curve increases and it considers a high number of failure increasing time.

Consider a system composed by n series elements, the system works if all the items work properly [20]. If the components of a system follow an exponential distribution, the reliability of the system is:

$$R_{SYS}(t) = e^{\lambda_{SYS} \cdot t} \tag{28}$$

Where:

$$\lambda_{SYS}(t) = \sum_{i=1}^{n} \lambda_i \tag{29}$$

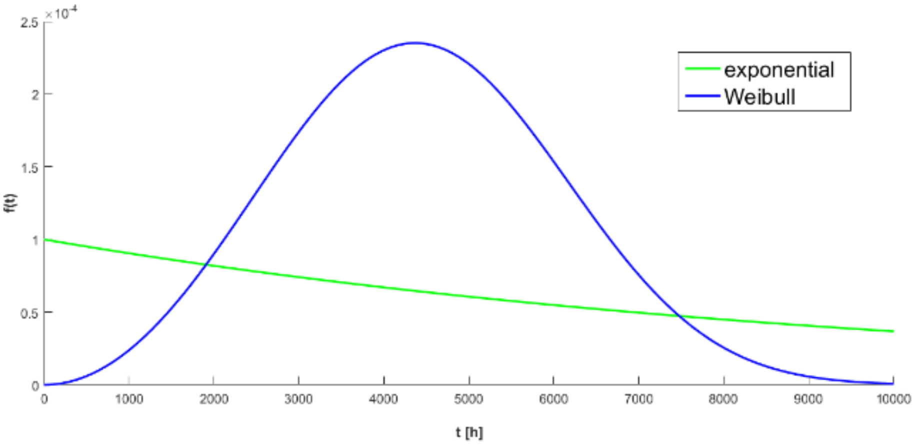


Figure 12: Comparative PDF plots

Meanwhile for processes following the Weibull distribution the equation becomes:

$$R_{SYS}(t) = e^{-\left[\sum_{i=1}^{n}\left(\frac{1}{\eta_i}\right)^{\beta}\right]t^{\beta}} \tag{30}$$

And the failure rate of the system is the following:

$$\lambda_{SYS}(t) = \beta t^{\beta-1}\left[\sum_{i=1}^{n}\left(\frac{1}{\eta_i}\right)^{\beta}\right] \tag{31}$$

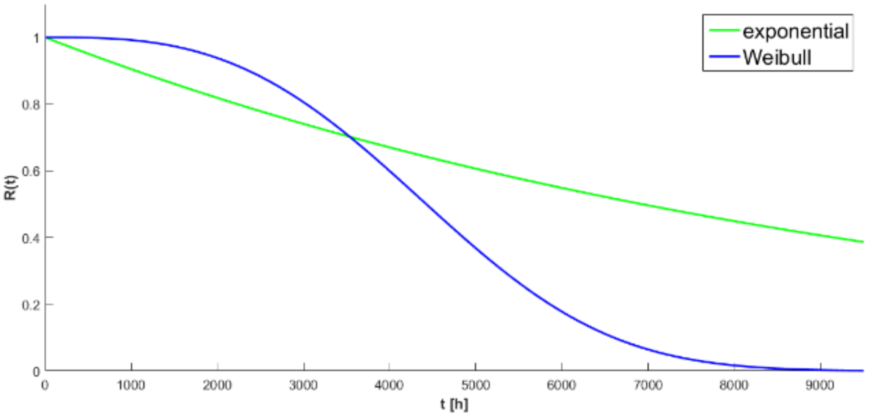


Figure 13. Comparative reliability functions

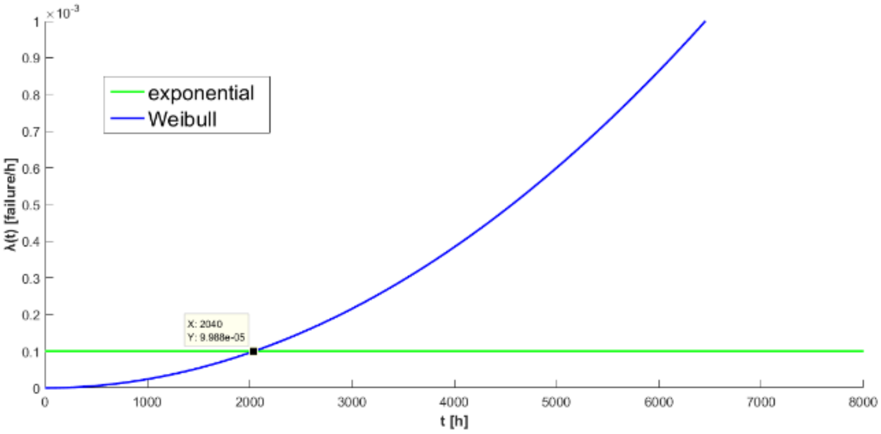


Figure 14. Comparative failure rate functions

In both cases increasing the number of the series items produces a decrease in system reliability.

In order to compare the reliability performance of Weibull and exponential distributions, a 3 elements series system (n=3) is considered: the Weibull distribution ($\eta = 5000$ h and $\beta = 3$) in Fig. 15 shows a lower number of failure before approximately 2000h (7 months), then the curves crosses each other and the exponential distribution (with constant failure rate $\lambda = 10^{-4}\ h^{-1}$) provides a better perspective than the Weibull one.

The same conclusions are achieved comparing the reliability function in Fig.16: before the intersection of the curves the Weibull distribution provides a higher reliability performance, then it decreases faster than the exponential [21-22].

Consider a system composed by two parallel items, the system works if one element at least works. If the system follows an exponential distribution and the items fail with the same failure rate λ, the reliability of the system is:

$$R_{SYS}(t) = 2e^{-\lambda t} - e^{-2\lambda t} \tag{32}$$

And the failure rate of the system is:

$$\lambda_{\mathrm{SYS}}(\mathrm{t}) = \frac{\lambda(1-\mathrm{e}^{-\lambda \mathrm{t}})}{(1-0.5\mathrm{e}^{-\lambda \mathrm{t}})} \tag{33}$$

Meanwhile the distribution follows the Weibull distribution the equation becomes:

$$R_{SYS} = 2e^{-\left(\frac{t}{\eta}\right)^{\beta}} - e^{-2\left(\frac{t}{\eta}\right)^{\beta}} \tag{34}$$

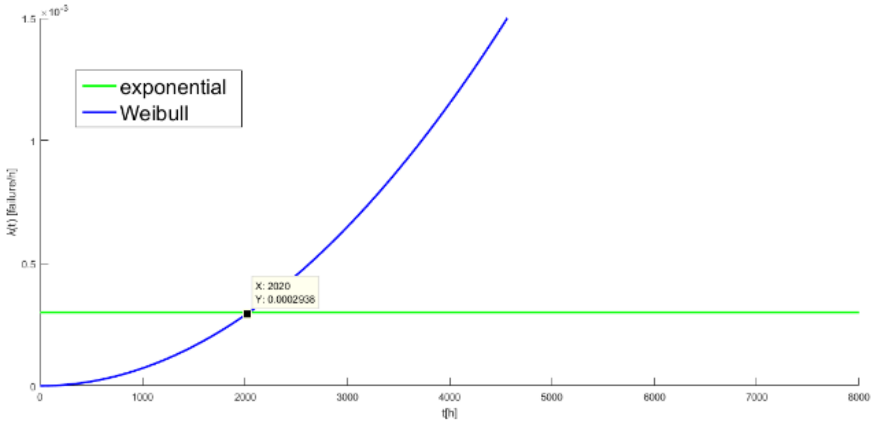


Figure 15. Comparison of failure rate functions of three series system

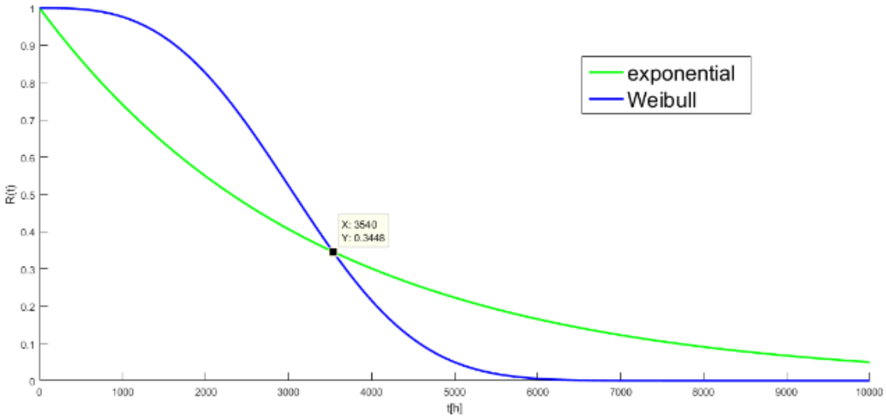


Figure 16. Comparison of reliability functions of three series system

And failure rate is the following:

$$\lambda_{\mathrm{SYS}}(\mathrm{t}) = \frac{\beta}{\eta}\left(\frac{\mathrm{t}}{\eta}\right)^{\beta-1} \frac{\left[2-2\mathrm{e}^{-\left(\frac{\mathrm{t}}{\eta}\right)^{\beta}}\right]}{2-\mathrm{e}^{-\left(\frac{\mathrm{t}}{\eta}\right)^{\beta}}} \tag{35}$$

Fig. 17 shows that in the early time the system offers a lower failure rate in case the components follows a Weibull distribution ($\eta = 5000\,\mathrm{h}$ e $\beta = 3$); anyway after the crossroad, the curve increases exponentially so the exponential distribution ($\lambda = 10^{-4}\mathrm{h}^{-1}$) gives a better prevision since it increases linearly with a lower slope.

Fig. 18 confirms the same trend, so initially Weibull distribution is convenient then it decreases and exponential gives a higher reliability values.

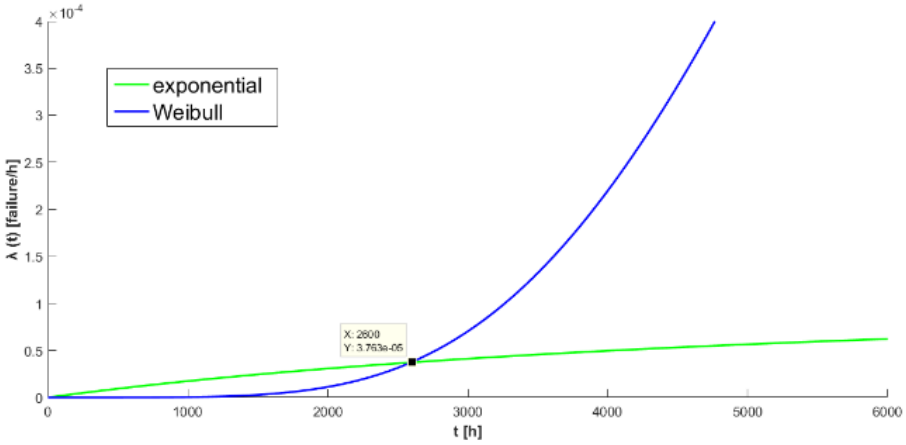


Figure 17. Comparison of failure rate functions of two parallel items

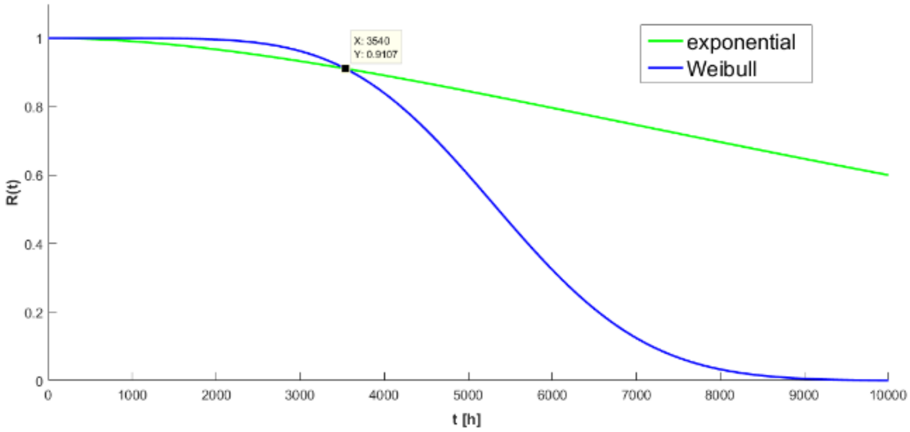


Figure 18. Comparison of reliability functions of two parallel items

1.8 Life Data Analysis and Parameter Estimation

The Weibull distribution is a quiet common model in reliability and lifetime data analysis [10]; not by chance the Life Data Analysis (LDA) is called "Weibull analysis" since the Weibull distribution is widely used to analyse the relationship between reliability and the life span of the product (or system).

Life data analysis starts with the analysis of a representative sample of units (belonging to the population of interest) in order to make a life prediction for all the products.

The resulting distribution for the data set can be used to achieve important life characteristics of the product e.g. reliability function, probability of failure at a fixed time, failure rate, etc.

Life data analysis is usually developed in four steps:

- Collect product life data;
- Select the best-fitting distribution to model the life of the products;
- Assess the parameters that ensure the distribution to the data;
- Generate plots and results to estimate product life characteristics.

One of the strong points of Weibull distribution is its ability to provide reasonably accurate analysis and failure forecasts with extremely small data samples. Because the Weibull distribution can take a variety of forms, it is effective in analysing data from increasing, constant, and decreasing failure rate applications.

Weibull analysis is typically used to determine the best-fit distribution for a set of failure data collected during testing or field operations. A said before, the distribution that best fits these data points provide info about the population from which they are drawn.

The best-fit distribution for any set of data points is quiet often the Weibull distribution but in some applications it may be another failure distribution, such as the lognormal, normal, or exponential; the behaviour of each distribution is described and influenced by characteristics and parameters that vary from distribution to distribution. In order to fit a statistical model to a life data set, the analyst should estimate the parameters of the life distribution that will make the function fitting the data in the best way. The parameters define the scale, shape and location of the PDF function [21].

Several methods have been developed to estimate the parameters that will fit a lifetime distribution to a particular data set and the most important are described in the following paragraphs. Starting from the relatively simple method of Probability Plotting and then analysing complex methods such as Rank Regression (or Least Squares) and Maximum Likelihood Estimation (MLE).

Obviously the appropriate analysis method will vary depending on the data set and, in some cases, on the life distribution selected.

1.8.1 Probability Plotting

The method of probability plotting takes the cumulative density functions of the distribution and attempts to linearize it using a dedicated paper.

This process includes linearization of the unreliability function, construction of the probability plotting paper and assessment of the X and Y positions of the plot points; then the plot is used to read any particular time or reliability/unreliability value of interest.

In the case of two-parameter Weibull, Eq.16 gives the unreliability function and this function can be linearized (i.e., following the standard format $\mathrm{y} = \mathrm{m}'\mathrm{x} + \mathrm{b}$) by setting:

$$\begin{cases} \mathrm{y} = \ln\left[\ln\left(\frac{1}{1-\mathrm{F(t)}}\right)\right] \\ \mathrm{x} = \ln \mathrm{t} \end{cases} \tag{36}$$

The equation can then be rewritten as follows:

$$y = \beta x - \beta \ln \eta \tag{37}$$

Which is now a linear equation with a slope of m=β and an intercept of $b = -\beta \ln \eta$.

The next task is to construct the Weibull probability plotting paper with the appropriate y and x axes: the x-axis transformation is simply logarithmic while the y-axis is more complex and requires a double logarithmic reciprocal transformation.

The y-axis represents unreliability and the x-axis represents time; both of these values must be known for each time-to-failure point to plot. Then, given the x and y value for each point, the points can easily be put on the plot. Once the points have been placed on the plot, the best-fitting straight line is drawn through these points [15].

Once the line has been drawn, the slope of the line can be obtained (some probability papers include a slope indicator to simplify this calculation): this is the parameter β that corresponds to the value of the slope. To determine the scale parameter, (also called the characteristic life), it is necessary to read the time from the x-axis corresponding to F(t)=63,2%.

Determining the appropriate y plotting positions, or the unreliability values, means to obtain the cumulative per cent failed for each time-to-failure. The most widely used method of determining this value is to assess the median rank for each failure. The median rank is the value that the true probability of failure F(ti) should have at the j-th failure out of a sample of N units at the 50% confidence level. The rank can be found for any percentage point, P, greater than zero and less than one, by solving the cumulative binomial equation for Z. This represents the rank, or unreliability estimate, for the j-th failure in the following equation for the cumulative binomial:

$$P = \sum_{k=j}^{N} \binom{N}{k} Z^k (1 - Z)^{N-k} \tag{38}$$

Where N is the sample size and j the order number. The median rank is obtained by solving this equation for Z at P=0,50. Besides the amount of effort required, which is the most obvious drawback to Probability Plotting, manual probability plotting is not always consistent in the results.

Two people plotting a straight line through a set of points will not always draw this line the same way, and thus will come up with slightly different results. This method was used primarily before the widespread use of computers that could easily perform the calculations for more complicated parameter estimation methods, such as the least squares and maximum likelihood methods [22].

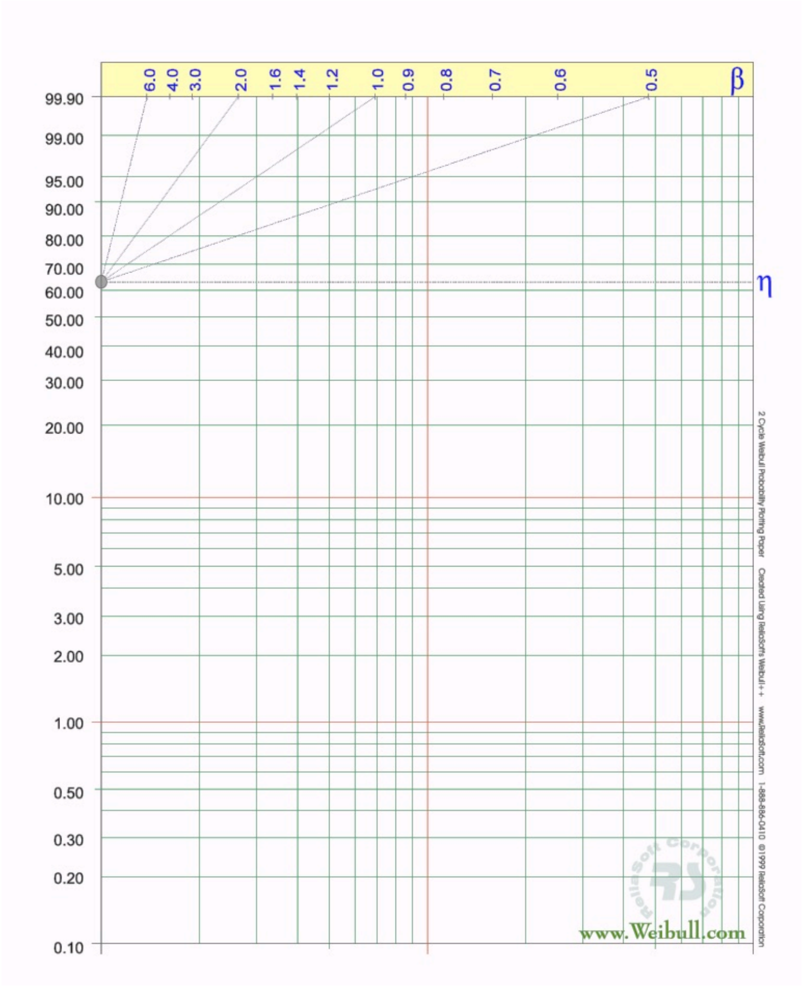


Figure 19. Plotting paper

1.8.2 Least Square Estimation (LSE)

The Least Square method requires that a straight line is fitted to a set of data points in order to minimize the sum of the squares of the distance of the points to the fitted line [15].
This minimization can be performed both vertically and horizontally: if the regression is on x, then the line is fitted so that the horizontal deviations from the points to the line are minimized. If the regression is on y, then this means that the distance of the vertical deviations from the points to the line is minimized (Fig. 20).
For the vertical regression, it is assumed that a set of data pairs (x_1, y_1), (x_2, y_2),...., (x_N, y_N) were obtained and plotted, and that the x-values are known exactly. The least squares principle minimizes the vertical distance between the data points and the straight line fitted to the data and, according to this assumption, the best fitting straight line to these data is the straight line [18]:

$$y = \hat{b}x + \hat{a} \tag{39}$$

Where:

$$\hat{a} = \frac{1}{N}\sum_{i=1}^{N} \ln[-\ln(1-F(t_i))] - \hat{b}\frac{1}{N}\sum_{i=1}^{N} \ln t_i \tag{40}$$

$$\hat{b} = \frac{N\sum_{i=1}^{N} \ln t_i \ln[-\ln(1-F(t_i))] - \sum_{i=1}^{N} \ln t_i \sum_{i=1}^{N} \ln[-\ln(1-F(t_i))]}{N\sum_{i=1}^{N} (\ln t_i)^2 - \left(\sum_{i=1}^{N} \ln t_i\right)^2} \tag{41}$$

Then the estimated parameters are:

$$\begin{cases} \eta = e^{-\frac{\hat{a}}{\hat{b}}} \\ \beta = \frac{1}{\hat{b}} \end{cases} \tag{42}$$

For the horizontal regression, the same least squares principle is applied, but this time, minimizing the horizontal distance between the data points and the straight line fitted to the data. The best fitting straight line to these data is the straight line:

$$y = -\frac{\hat{a}}{\hat{b}} + \frac{1}{\hat{b}}x \tag{43}$$

Where:

$$\hat{a} = \frac{1}{N}\sum_{i=1}^{N} \ln t_i - \hat{b}\frac{1}{N}\sum_{i=1}^{N} \ln[-\ln(1-F(t_i))] \tag{44}$$

$$\hat{b} = \frac{N\sum_{i=1}^{N} \ln t_i \ln[-\ln(1-F(t_i))] - \sum_{i=1}^{N} \ln t_i \sum_{i=1}^{N} \ln[-\ln(1-F(t_i))]}{N\sum_{i=1}^{N} (\ln[-\ln(1-F(t_i))])^2 - \left(\sum_{i=1}^{N} \ln[-\ln(1-F(t_i))]\right)^2} \tag{45}$$

Then the parameters estimated are:

$$\begin{cases} \eta = e^{-\frac{\hat{a}}{\hat{b}}\frac{1}{\beta}} \\ \beta = \frac{1}{\hat{b}} \end{cases} \tag{46}$$

The correlation coefficient is a measure of the quality of data-fitting of the linear regression model and it is usually denoted by ρ. The case of life data analysis is a measure for the strength of the linear relation (correlation) between the median ranks and the data. The correlation coefficient of the population is defined as follows:

$$\hat{\rho} = \frac{N\sum_{i=1}^{N} x_i y_i - \sum_{i=1}^{N} x_i \sum_{i=1}^{N} y_i}{\sqrt{\left(N\sum_{i=1}^{N} x_i^2 - \left(\sum_{i=1}^{N} x_i\right)^2\right)\left(N\sum_{i=1}^{N} y_i^2 - \left(\sum_{i=1}^{N} y_i\right)^2\right)}} \tag{47}$$

It assumes values in a range [-1, 1], the closer the value is to ±1, the better is the linear fitting. The least squares estimation method is a good solution for functions that can be linearized: for these distributions, the calculations are relatively easy and straightforward, since they have closed-form solutions that can yield an answer without having to resort to numerical techniques or tables. Furthermore, this technique provides a good measure of the goodness-of-fit of the chosen distribution in the correlation coefficient.

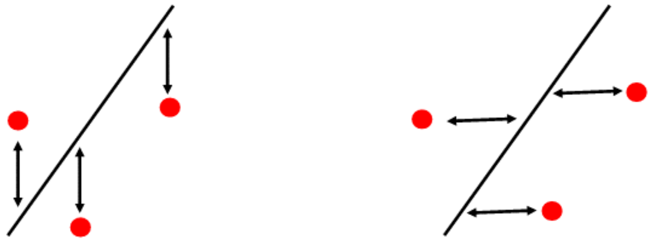


Figure 20. Rank regression on y (left) and on x (right)

1.8.3 Maximum Likelihood Estimation (MLE)

In statistics the Maximum Likelihood Estimation method is considered one of the most robust parameter estimation techniques [21]. The basic idea behind MLE is to obtain, for a given distribution, the most likely values of the parameters that will best describe the data.

Supposing T is a continuous random variable with PDF $f(t, \beta, \eta, \gamma)$ [9], where t, β, η, γ are unknown parameters which need to be estimated, with R independent observations, $x_1, x_{2,} x_3, \dots, x_R$, which correspond to failure times (in life data analysis). The likelihood function is given by:

$$L(\beta, \eta, \gamma \,|t_1, t_{2,} t_3, \dots, t_{R)} = L = \prod_{i=1}^{R} f(\, t_i; \beta, \eta, \gamma) \tag{48}$$

The logarithmic likelihood function is the following:

$$\Lambda = \ln L = \sum_{i=1}^{R} \ln f(t_i; \beta, \eta, \gamma) \tag{49}$$

Maximizing Λ or L are the two solutions to obtain the maximum likelihood estimators (or parameter values) of β, γ, η: by maximizing Λ which is much easier to work with than L, the maximum likelihood estimators (MLE) are the simultaneous solutions of equations such that:

$$\frac{\partial \Lambda}{\partial \beta} = 0 \qquad \frac{\partial \Lambda}{\partial \gamma} = 0 \qquad \frac{\partial \Lambda}{\partial \eta} = 0 \tag{50}$$

If all the three parameters are unknowns, the log-likelihood function becomes the following:

$$\Lambda = N \ln \beta - N\beta \ln \eta + (\beta - 1) \sum_{i=1}^{N} \ln(t_i - \gamma) - \sum_{i=1}^{N} \left(\frac{t_i - \gamma}{\eta}\right)^{\beta} \tag{51}$$

The parameters are achieved by maximizing the equation above. In most cases, no closed-form solution exists for this maximum or for the parameters. If η is known, by defining xi=ti-γ, Eq. 48 can be solved as follows:

$$\Lambda = N \ln \beta - N\beta \ln \eta + (\beta - 1) \sum_{i=1}^{N} \ln(t_i) - \sum_{i=1}^{N} \left(\frac{t_i}{\eta}\right)^{\beta} \tag{52}$$

Deriving and solving to η Eq. 47, the estimated parameter is:

$$\eta = \left(\frac{1}{N}\sum_{i=1}^{N} t_i^{\beta}\right)^{\frac{1}{\beta}} \tag{53}$$

Furthermore, the shape parameter has not a closed form:

$$\frac{\partial \Lambda}{\partial \beta} = N\left(\frac{1}{N}\sum_{i=1}^{N} \ln x_i + \frac{1}{\beta}\frac{\sum_{i=1}^{N}\left(x_i^{\beta}\ln x_i\right)}{\sum_{i=1}^{N} x_i^{\beta}}\right) = 0 \tag{54}$$

The equation can't be solved analytically but needs a numeric technique or a software implementation.

1.8.4 Confidence Bounds

Consider a sample Y1, … , Yn from a known distribution F(y, θ) except the parameter θ, supposed to be a real number [8]. A statistical couple $\widehat{X_1}(Y_1,\dots,Yn\ )$ and $\widehat{X_2}(Y_1,\dots,Yn\ )$ with $\mathrm{Prob}(X_1 \leq X_2\ ) = 1$ is said confidence bound for the parameter θ with confidence level $\alpha \in (0,1)$ if:

$$\mathrm{Prob}(X_1 \leq \theta \leq X_2\ ) \geq \alpha \tag{55}$$

Usually two-sided confidence bounds (or intervals) are used for closed intervals where a certain percentage of the population is likely to lie [12].

One possible methodology used to find confidence bounds is the so-called Fisher matrix bounds; for the MLE applications the Fisher matrix is the following:

$$F = \begin{bmatrix} -\frac{\partial^2\Lambda}{\partial\beta^2} & -\frac{\partial^2\Lambda}{\partial\beta\,\partial\eta} \\ -\frac{\partial^2\Lambda}{\partial\beta\,\partial\eta} & -\frac{\partial^2\Lambda}{\partial\eta^2} \end{bmatrix} \tag{56}$$

Substituting the values of the estimated parameters $\widehat{\beta,}\hat{\eta}$ and then inverting the matrix, the local estimate of the covariance matrix is achieved:

$$\begin{bmatrix} \widehat{\mathrm{Var}}(\hat{\beta}) & \widehat{\mathrm{Cov}}(\hat{\beta},\hat{\eta}) \\ \widehat{\mathrm{Cov}}(\hat{\beta},\hat{\eta}) & \widehat{\mathrm{Var}}(\hat{\eta}) \end{bmatrix} = \begin{bmatrix} -\frac{\partial^2\Lambda}{\partial\beta^2} & -\frac{\partial^2\Lambda}{\partial\beta\,\partial\eta} \\ -\frac{\partial^2\Lambda}{\partial\beta\,\partial\eta} & -\frac{\partial^2\Lambda}{\partial\eta^2} \end{bmatrix}^{-1} \tag{57}$$

Values for the variance and covariance of the parameters are obtained from Fisher Matrix equation. Once they have been obtained, the approximate confidence bounds on the function are given as:

$$\hat{\beta} - K_{\frac{1-\alpha}{2}} \cdot \sqrt{\mathrm{Var}(\hat{\beta})} < \beta < \hat{\beta} + K_{\frac{1-\alpha}{2}} \cdot \sqrt{\mathrm{Var}(\hat{\beta})} \tag{58}$$

$$\hat{\eta} - K_{\frac{1-\alpha}{2}} \cdot \sqrt{\mathrm{Var}(\hat{\eta})} < \eta < \hat{\eta} + K_{\frac{1-\alpha}{2}} \cdot \sqrt{\mathrm{Var}(\hat{\eta})} \tag{59}$$

Where Kα is defined as follows:

$$\alpha = \frac{1}{\sqrt{2\pi}} \int_{K\alpha}^{\infty} e^{-\frac{t^2}{2}} dt \tag{60}$$

1.8.5 Comparison between estimation methods

The likelihood function indicates how frequently the observed sample is as a function of possible parameter values [20]. Therefore, maximizing the likelihood function determines the parameters that are most likely to produce the observed data. From a statistical point of view, MLE is usually recommended for large samples because it produces the most precise estimates, furthermore it is versatile and applicable to most models and different types of data.

Least squares estimates are calculated by fitting a regression line to the points from a data set that has the minimal sum of the deviations squared (least square error). In reliability analysis, the line and the data are plotted on a probability plot.

For large and complete data sets, both the LSE method and the MLE method provide consistent results; anyway in reliability applications data sets are typically small or moderate in size. Extensive simulation studies show that in small sample designs where there are only a few failures, the MLE method is better than the LSE method.

The advantages of the MLE method over the LSE method are that the calculations use more of the information in the data, the distribution parameter estimates are more precise and the estimated variance is smaller.

The LSE method is also traditionally associated with the use of probability plots to assess goodness-of-fit. However, the LSE method can provide misleading results on a probability plot [20].

1.9 Case Studies

The methods described above are used for two applicative case studies found in literature [22-23] in order to describe the procedure to analyse these samples of data and find the best-fitting distribution. The data taken into account come from two different testing procedures, a test on electronic board for automatic control and accelerated testing of electronic components. Both the data have been analysed with the software Windchill Quality Solutions [18]. The data were insert in the tool Weibull Analysis and the software displays the plots generated of different functions such as reliability, unreliability, PDF and failure rate. These outcomes help to identify the best values of the parameters for the data set and provide additional insights.

Furthermore, the selected distributions are analysed to determine how well they fit the data point within the data set; once the analysis is complete, the software shows the ranking results for the selected distributions. To determine the rankings, Λ (log-likelihood function) or ρ (correlation coefficient) are used.

1.9.1 Test On Electronic Board For Automatic Control

Table I shows the failure times achieved with the testing procedure on a population of ten electronic boards while Table II lists the results of the best-fitting procedure using both methods LSE and MLE.

Table I. Failure times of components

Component	Failure Time [h]
1	1200
2	2300
3	2500
4	2800
5	3000
6	3700
7	4000
8	4100
9	4200
10	4800

The two methods give different solutions for the second and third distributions; both Weibull and normal distributions fit well the data, but the rank is different since the estimation methods are different. The other distributions, such as log-normal and exponential don't fit the data and in fact a lower value of the coefficient is obtained.

The distribution that better fits the data is the three-parameter Weibull: the correlation coefficient is almost unitary and the log-likelihood coefficient is higher than the other distributions.

Fig. 21 and Fig. 22 show the data best-fitting as a confirm of the results described above.

In particular, Fig. 21 shows that the data follows the three-parameter Weibull distribution. All the data are placed almost linearly and the distance between the real curve and the expected Weibull line is minimized. In fact the correlation coefficient ρ=0,9885 is very high and very close to one. Fig. 22 shows the fitting of the normal distribution where the data are not perfectly linear; furthermore the distance between the reference line is bigger than the Weibull distribution but however the normal distribution can be considered a good approximation of this set of data. The correlation coefficient ρ=0,9799 is very similar to the perfect correlation correspondent to one [20].

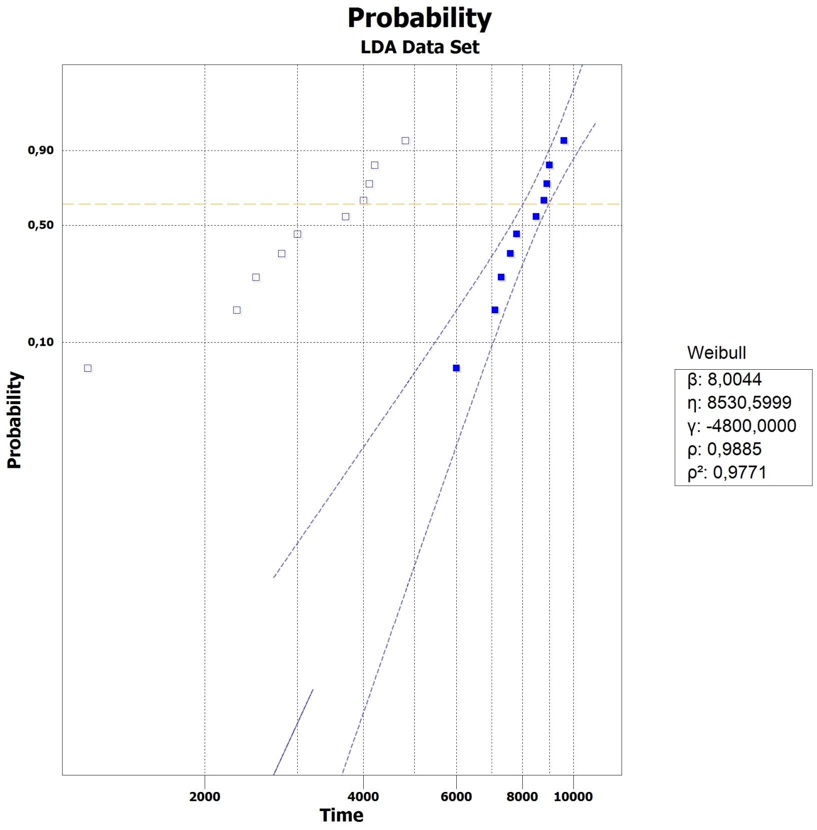


Figure 21. Three-parameter Weibull probability plot

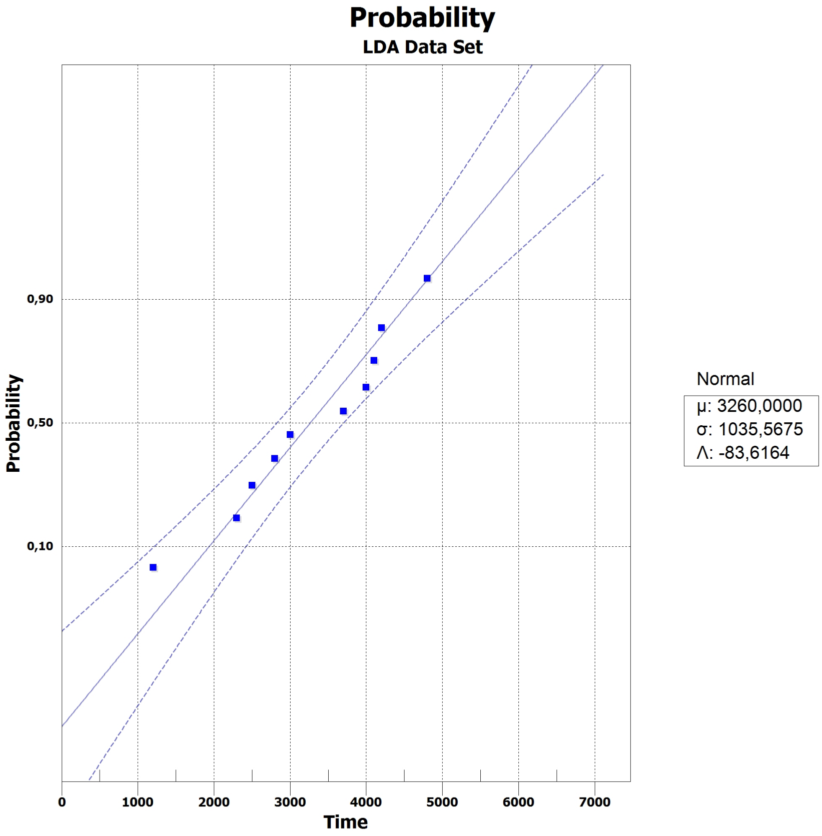


Figure 22. Normal probability plot

Table II. Ranking of the best fit distribution

Least Square Estimation		**Maximum Likelihood Estimation**	
Distribution	**ρ**	**Distribution**	**Λ**
Weibull (β, η, γ) • β= 8,0 • η=8530 h • γ= -4800 h	**0,9885**	Weibull (β, η, γ) • β= 9,45 • η=8504 h • γ= -4800 h	**-83,30**
Normal (μ,σ) • μ=3260,00 h • σ=1165 h	**0,9799**	Normal (μ, σ) • μ=3259 h • σ= 1035 h	**-83,62**
Weibull (β, η) • β=2,80 • η=3681 h	**0,9730**	Weibull (β, η) • β=3,68 • η=3621 h	**-83,47**
Log-normal (μ, σ) • μ=8,0 h • σ=0,42 h	**0,9331**	Log-normal (μ, σ) • μ=8 h • σ= 0,39 h	**-84,96**
Exponential (λ, η, γ) • λ=0,00056 h-1 • η=1785 h • γ=1188 h	**0,8864**	Exponential (λ, η, γ) • λ=0,000485 h-1 • η=2060 h • γ=1200 h	**-86,30**
Exponential (λ, η) • λ=0,00039 h-1 • η=2529 h	**0,8864**	Exponential (λ, η) • λ=0,000307 h-1 • η=3260 h	**-90,89**

1.9.2 Accelerated Test On Electronic Components

The second test case required a high stress procedure on thirty electronic components; the corresponding failure times are shown in the Table III and the best-fitting results are listed in Table IV.

Since the number of components is higher then the first test case, the method used is just the MLE technique that is particularly suited for this kind of applications [22].

The rankings show that the best-fitting distributions are the two-parameter exponential and the three-parameter Weibull.

The two-parameter exponential is defined by the standard parameter λ and a location parameter γ. The location parameter, in case it assumes positive values, shifts the beginning of the distribution by a distance of γ on the right of the origin: this means that failures may occur only after γ hours of operation, not before that time.

Table III. Failure times of components

Component	Failure time [h]
1	2100
2	3800
3	5400
4	6600
5	7600
6	7800
7	12300
8	13000
9	15200
10	15900
11	19900
12	20100
13	20400
14	21500
15	21800
16	28100
17	29500
18	31000
19	33800
20	34100
21	35400
22	35800
23	43100
24	45700
25	54500
26	56900
27	67700
28	81800
29	94600
30	148600

Table IV. Ranking of the best fit distribution

Maximum Likelihood Estimation	
Distribution	**Λ**
Exponential (λ, η, γ) • λ=0,000032 h-1 • η=31699 h • γ=2100 h	-340,92
Weibull (β, η, γ) • β=0,98 • η=31486 h • γ=2079 h	-340,93
Log-normal (μ, σ) • μ=10 h • σ= 0,96 h	-342,18
Weibull (β, η) • β=1,18 • η=35885 h	-342,20
Exponential (λ, η) • λ=0,00003 h-1 • η=33800 h	-342,85
Normal (μ, σ) • μ=33799 h • σ=30992 h	-352,81

Fig. 23 shows the probability plot of the data-set supposing a two parameters exponential distribution, where most of data are distributed on the exponential line: the software calculates a very high log likelihood coefficient Λ=-340,92 as a confirm that the distribution fits very well the data [20].

Fig. 24, instead, shows the output achieved using the Weibull distribution: it has a lower value of log-likelihood function Λ=-340,93, however it offers a good fitting and a satisfying approximation of the dataset. Fig.23 shows that data are concentrated on the Weibull line and they are located more linearly anyway both the distribution are a good approximation.

The Weibull distribution, as expected, is a satisfying approximation also in this case, even if for this application the exponential is a more accurate; in any case the Weibull distribution confirms to be very flexible and capable to describe different types of data.

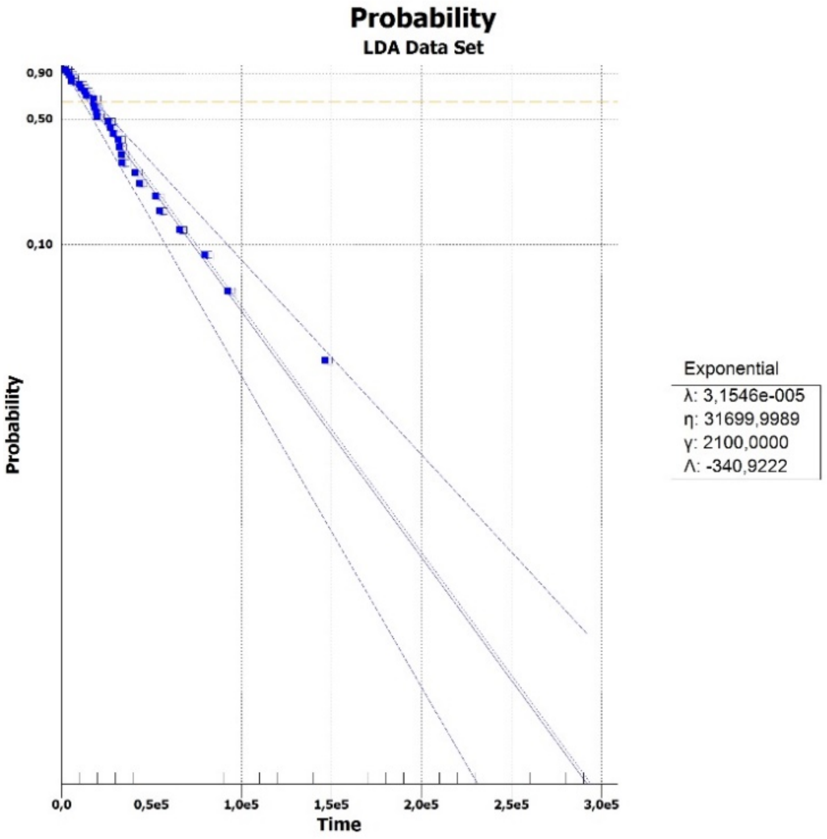


Figure 23. Two-parameter exponential probability plot

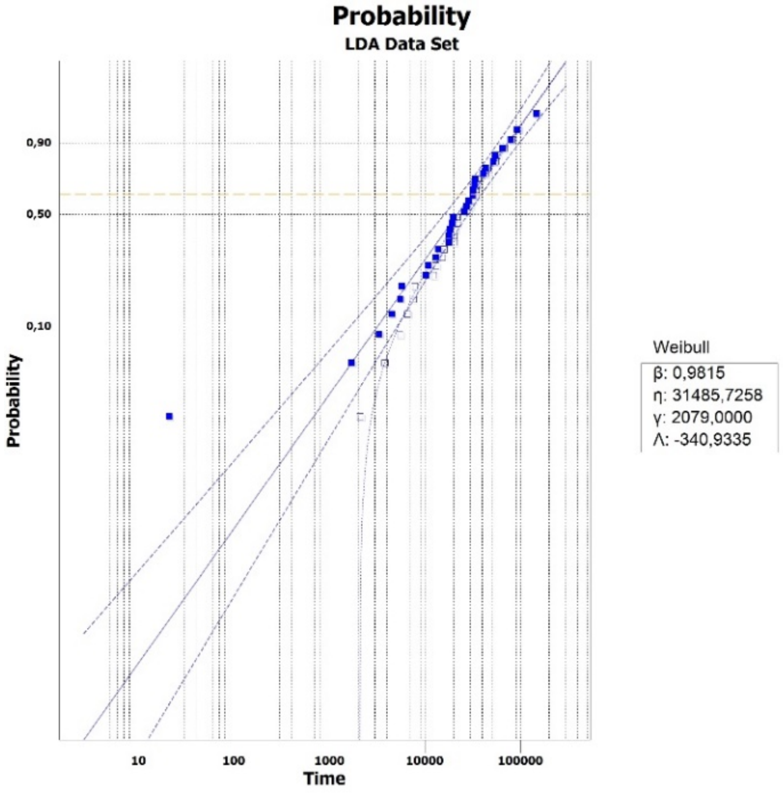


Figure 24. Three-parameter Weibull probability plot

1.10 Discussion And Remarks

The exponential distribution is widely used in reliability applications since it describes the constant failure rate section and it is used for component with a long useful life (e.g. electronic components).

In all other cases, data generally has a non-constant failure rate trend and the most used distribution to describe it is the Weibull one that is a very flexible distribution thanks to its parameters β conditioning the shape of the curves and η that extend or compress the curves [21].

The first case study described was referred to a little population and, as a result, the set of data under analysis doesn't fit a constant failure rate distribution; as expected the Weibull and normal distribution provide the best-fitting.

The second case, instead, was assessed on an elevate number of samples and the two-parameter exponential and the three-parameter Weibull resulted to be the best to fit the data. The second test provides more realistic results than the first since it was based on a larger number of samples and it confirms that the Weibull distribution is very flexible and can describe a lot of life models although for this particular test case the distribution that better fits the samples is the two-parameter exponential one [20].

As said before, the failure rates of mechanical components are not usually described by a constant failure rate distribution because of wear, fatigue and many other stress failure mechanisms. Anyway in the following chapters the exponential distribution is used as the reference for design for reliability purposes so infant mortality and wear out periods are excluded from the prediction. This choice is justifiable because:

- The infant mortality period is representative of the development of equipment or a system. Control over increasing reliability during this phase is a fundamental step in order to assess good reliability performance;
- The wear out period is usually far in the future compared with the useful life of most of the devices taken into account in this study;
- The failure mechanisms, in particular at a microscopic analysis, rarely satisfy a constant rate occurrence but the dispersion of many failure mechanisms, even if they are accumulative and increasing with time, play a fundamental role so they can be assumed to be constant over the period considered. Furthermore presence of large number and diversity of components in a complex system together with different ages (due to component replacement and overhaul) between equipment in the same system will produce a trend that is very close to a constant for an observer at system level.

This is the reason why the use of a constant failure rate is still the most relevant approach for estimating the predicted reliability of a system [9].

Chapter 2

Availability Improvement

2.1 Availability Improvement

Availability is one of the most important characteristics of repairable systems and low availability values require big efforts, and corresponding costs, to improve it.

Any improvement in availability of a system needs to be valuated in terms of costs and benefits in order to optimize the efforts and ensure that availability improvements lead to benefits for the business.

So it is important that any additional investment to improve the availability performance of the system can be cost justified.

Following the "cost-saving" trend the interest in availability is growing in many different manufacturing fields and starts in design stage.

The design of high-available systems is not associated to a specific technology nor a quantifiable attribute, it is the result of several strategies, technologies and services that are involved to achieve it.

The main solutions to assess high-availability solutions and improve the whole system availability are the following:

- Improve the availability performance of single items in the system;
- Introduce redundant architectures (fault tolerant design) for the most exposed items in the system;
- Improve maintainability operation using different techniques such as Markov models;
- Improve reliability performance using Reliability Allocation and Reliability Importance methods;
- Introduce diagnostic features on both local and system/process level.

The first and the simplest method to improve system availability is to increase the availability of each component and this is possible through increase of failure time and/or decrease of repair time. Using components with higher availability lead to an expected improvement in performance but the cost impact is often relevant.

For this reason the best solution is taking into account system availability performance from the first design stages in order to monitor the economic and technical feasibility of the process.

The availability of a system is directly influenced by uptime and downtime and, for scheduled working time, the mathematical expression is the following:

$$Availability = \frac{MTBF}{MTBF+MTTR} \tag{61}$$

MTBF represents the Mean Time Between Failures (life time), and MTTR represents the Mean Time To Repair (repair time) for machine that is defined as maintainability.

In this analysis an exponential distribution is assumed to be representative for the reliability and maintainability statistical models. The MTBF is the inverse of the failure rate.

Similarly, the MTTR is the inverse of the repair rate when it is constant and it can be influenced by technical design and availability of maintenance resources during repairing process [24-26].

The impact of technical design is obviously restricted to design stages while MTTR can be improved in the following life phases of the device working on availability of maintenance resources (human resources, necessary for machine servicing, and spare parts available for the replacement).

Furthermore the availability of manufacturing systems is directly dependent to reliability and maintainability of the system itself. An improvement in availability could be experienced by increasing MTBF, with correspondent system reliability growth, or decreasing MTTR: repairing time is influenced by two factors, technical design of the component and maintenance resources availability during repairing process so for decreasing the repairing time, a suitable allocation of maintenance resources is necessary.

2.2 Fault Tolerant Design

Fault tolerant design is a design that allow a system to continue its intended operation (with reduced efficiency level in some circumstances) rather than failing completely in case of failure of some components of the system. Fault tolerance provides a more robust approach to surviving faults and failure [26].

Fault tolerant configurations are widely used in many manufacturing fields in order to achieve continuous and successful operations despite extreme process and environmental conditions, in particular if immediate repair or maintenance is not available.

There are many different techniques to achieve fault tolerance and the most used is redundancy: this method is based on the duplication of the components that are most critical for the whole system performance in order to increase both system reliability and availability.

Redundancy techniques are usually divided in two categories in case of fault tolerant designs: static and dynamic redundancy, in compliance with MIL-HDBK 338B [27].

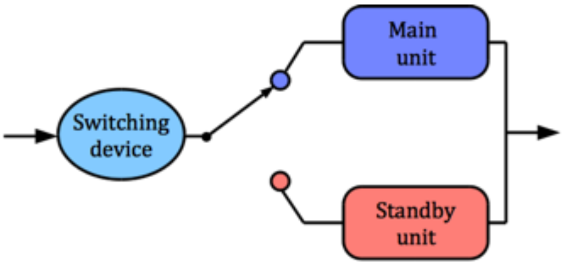


Fig. 1. Standby architecture

Static (or “active”) redundancy consists of fault masking without proper fault detection: this solution includes e.g. parallel, k-out-of-n, major voting and there isn’t a performance/status monitoring so final user is not aware of failure occurrence [28].

Dynamic (or “standby”) redundancy, instead, consists of fault detection and system reconfiguration with a standby unit; in case of main component failure, standby unit is activated to complete the mission. In particular, there are three dynamic redundancy configurations: hot standby, warm standby and cold standby [29].

The first architecture offers the same reliability performance of standard parallel configuration [9] and, for this reason, is not described in this study.

2.2.1 RBD Model Based Approach

Reliability Block Diagram (RBD) is one of the most used top-down techniques to achieve reliability assessment. A RBD is a functional diagram of all the components making up the system that shows how component reliability contributes to failure or success of the whole system. Each component in the system has a corresponding block that is described by a specific failure rate and connections with other part of the system. Despite Functional Block Diagrams (FBDs), which are focused on normal operation functionality, in RBDs the attention is shifted onto component failures and their consequences on the system.

IEC 61078 [4] shows the necessary assumptions to develop the RBD and to calculate the reliability parameters; such assumptions can be summarized as:

- System item (component or a sub-system) assumes only two states: working (“up” state) or failed (“down” state); intermediate working state is not allowed. On the basis of this assumption, the system state can be considered as a discrete random variable.
- The failures are assumed as independent events: the failure condition of a given item does not affect the probability of failure of any other block within the system modeled. On the basis of this assumption, the probability of failure of the block A, P(A) – for example – is not related with the probability of failure P(B) of the block B, and vice versa.

$$P(A \mid B) = P(A) \quad P(B \mid A) = P(B) \tag{62}$$

- Sequential events are not considered in this method; the system analysis stops when the first fault is shown. For this reason, RBDs are not suitable for modelling order-dependent or time-dependent events.
- System items are considered in "useful life" period where failures can be considered random events and failure/hazard rate is assumed as constant in the time, that is:

$$\lambda_i(t) = \lambda_i \tag{63}$$

With i=1…n, being n the number of items of the system.

- The probability density function of failure f(t) is an exponential distribution. Considering the useful-life period and assuming random failures, f(t) and reliability function R(t) can be written as follows:

$$f(t) = -\frac{dR(t)}{dt} = \lambda e^{-\lambda t} \tag{64}$$

$$R(t) = \exp\{-\int_0^\infty \lambda(t)dt\} = e^{-\lambda t} \tag{65}$$

- Not reparable system is considered with Mean Time To Failure (MTTF) as:

$$MTTF = \int_0^\infty t \cdot f(t)dt = \int_0^\infty R(t)dt \tag{66}$$

These hypotheses are mandatory to achieve a reliability prediction otherwise the proposed RBD approach would not be put in practice [9].

The following paragraphs describe a brand new approach to achieve system reliability in systems containing standby redundancies. The added value is that there is no limit to RBD complexity and this feature is essential to achieve reliability prediction of complex systems.

2.2.2 Standby Architecture

The general equation for standby redundancy is represented by:

$$R_s(t) = R_1(t) + (1-p) \cdot \int_0^t f_1(x) \cdot R_{2,sb}(x) \cdot \frac{R_{2,a}(t_e + t - x)}{R_{2,a}(t_e)} \cdot dx \tag{67}$$

Where:

- t: mission time;
- x: time of main failure and further stand-by activation;
- t_e: equivalent operating time for the stand-by branch if it had been operating at an active mode;
- R_s(t): reliability of the system;

- $R_1(t)$: reliability of the active branch (main);
- $f_1(x)$: pdf of the active branch;
- p: probability of failure of switch;
- $R_{2,sb}(t)$: reliability of the stand-by branch in quiescent mode ;
- $R_{2,a}(t)$: reliability of stand-by branch in active mode.

The following two paragraphs describe the cold and warm redundancy and show the developed standby reliability models.

2.2.3 Cold Standby

In a cold stand-by architecture the main unit is fully operative; the stand-by is inactive and completely disconnected from any kind of power source or fuel supply. This is the reason why, during inactive period, quiescent components do not age and cannot fail (this assumption can be considered supposing to store, transport, operate and maintain the equipment in totally controlled environment).

In this configuration, diagnostics plays a fundamental role in order to detect both main and stand-by unit failures. On/off-line tests, auto-diagnostics circuits and continuous monitoring on main equipment are mandatory to switch the load on demand (when failure arises) and this practice is required also on stand-by devices, in particular if are involved in industrial applications such as Oil&Gas: this equipment, although in quiescent status, are forced to endure severe environmental and process conditions and can't be considered failure free by definition.

In a cold standby architecture the switching device is included in the reliability analysis since its failure cancel all the advantages achieved through redundancy; for this reason it can't be considered failure free by definition. Switch failure modes are essentially two, not-required commutation and failure to commute on demand [28].

In these assumptions response-time required to activate and initialize stand-by unit and switch failure rate are the residual constrictions of cold stand-by employment. Therefore in a cold stand-by architecture the following assumption can be made:

$$R_{2,sb}(x) = R_{2,a}(t_e) \quad \lambda_{2,sb} = 0 \quad t_e = 0 \quad R_{2,sb}(x) = R_{2,a}(t_e) = 1 \tag{68}$$

The brand new reliability function for cold stand-by architecture is shown below:

$$R_s(t) = R_1(t) + (1-p) \cdot \int_0^t f_1(x) \cdot R_{2,a}(t-x) \cdot dx \tag{69}$$

Where, referring to the system in Figure 1, we have:

- R_s: reliability of the system;
- R_1: reliability of the active component (main);

- p: switch failure probability;
- f_1: pdf of the active component;
- $R_{2,a}$: reliability of the standby component in active mode;
- x: time of main failure and further standby activation.

2.2.4 Warm Standby

In warm stand-by architecture also the standby equipment is connected to power/fuel supply although only the main device is involved in the process. Backup unit is half operative and ready to take over if main failure occurs. One of the strengths of this configuration if compared with cold stand-by is the reduced response-time: in this architecture it's not necessary to wait for stand-by unit start-up (equipment is ready to use) so it's sufficient to switch the load from main to stand-by to keep the system working. On the other hand, as a consequence of the uninterrupted supply, stand-by units age during quiescent period and can fail before switching the load; for this reason to define the reliability behaviour of stand-by equipment two different failure rates are required [28]:

- "λ_o" when main unit is working properly so stand-by unit is half-operative (quiescent status);
- "λ" when stand-by unit is fully operative due to main equipment failure (operative status).

In a warm standby architecture the following assumption can be made:

$$R_{2,sb}(x) < 1;\ \lambda_{2,sb} \neq 0 \tag{70}$$

The brand new reliability function for warm stand-by architectures is shown below:

$$R_s(t) = R_1(t) + (1-p) \cdot \int_0^t f_1(x) \cdot R_{2,sb}(x) \cdot R_{2,a}(t-x) \cdot dx \tag{71}$$

Where:

- R_s: reliability of the system;
- R_1: reliability of the active component (main);
- p: switch failure probability;
- f_1: pdf of the active component;
- $R_{2,sb}$: reliability of the stand-by component in quiescent mode;
- $R_{2,a}$: reliability of the standby component in active mode;
- x: time of main failure and further stand-by activation.

The equations described above are necessary to put in practice the new approach proposed in this study: thanks to the cold and warm standby functions it is possible to achieve a reliability prediction of complex systems containing standby architectures. There is no limit

in structure and number of components that can be used on each branch of the redundant architecture.
These features are implemented in a dedicated tool named "RBDesigner" that was developed in order to assess reliability parameters: this tool is fully described in the Chapter 6.
In the following paragraph, instead, a case study is described to show the potential of the method.

2.2.5 Case Study

The previous paragraphs show a new broad approach for reliability assessment and without setting limits to the complexity of Reliability Block Diagrams; so it is possible to achieve reliability prediction of complex systems. Despite the classic approach in [9], the user can achieve reliability prediction of very complex structures on each branch of the architecture e.g. cold stand-by redundancy with cold stand-by blocks on each branch.

An example of the proposed methodology is shown below: for reason of space and for simplicity the system is considered made of four notional components, the redundant frameworks are 1oo2 cold stand-by and each branch contains two blocks with the same failure rate, λ_a and λ_b respectively (see Figure 2).

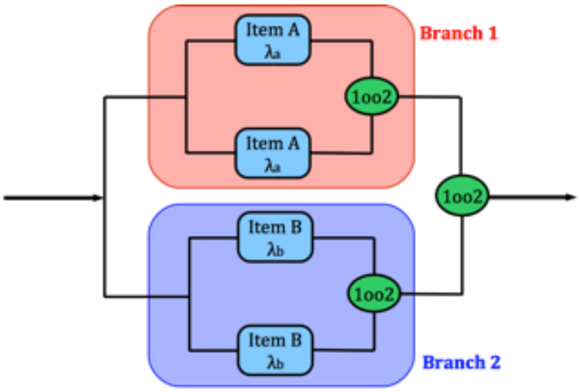


Fig. 2. Cold stand-by architecture with cold stand-by blocks

The system reliability function is described below: equation (72) is obtained from Eq. (70) considering the probability of failure of all the switches equal to zero (failure free). All terms in Eq. (72) are defined in Eqs. (73), (74) and (75). Equation (76) represents the final reliability function of the whole system.

$$R_s(t) = R_1(t) + \int_0^t f_1(x) \cdot R_{2,a}(t - x) dx \qquad (72)$$

$$\mathrm{R}_1(\mathrm{t}) = \mathrm{e}^{-\lambda \mathrm{at}} \cdot (1 + \lambda_\mathrm{a} \mathrm{t}) \qquad (73)$$

$$f_1(t) = -\frac{dR_1(t)}{dt} = -\lambda_a \cdot e^{-\lambda at} + \lambda_a \cdot e^{-\lambda at} \cdot (1 + \lambda_a t) \tag{74}$$

$$R_2(t) = e^{-\lambda bt} \cdot (1 + \lambda_b t) \tag{75}$$

$$\begin{aligned} R_s(t) &= e^{-\lambda at} \cdot \left(1 + \lambda_a t\right) + \\ &+ \int_0^t \left[-\lambda_a \cdot e^{-\lambda ax} + \lambda_a \cdot e^{-\lambda ax} \cdot \left(1 + \lambda_a x\right)\right] \cdot \left[e^{-\lambda b(t-x)} \cdot \left(1 + \lambda_b \cdot \left(t - x\right)\right)\right] \cdot dx = \\ &= \frac{\lambda_a{}^2 e^{t(-\lambda a - \lambda b)} \cdot \left(e^{\lambda at}\left(\lambda_a \lambda_b t + \lambda_a - \lambda_b \cdot \left(\lambda_b t + 3\right)\right)\right)}{\left(\lambda_a - \lambda_b\right)^3} + \\ &- \frac{\lambda_a{}^2 e^{t(-\lambda a - \lambda b)} \cdot \left(e^{\lambda bt} \cdot \left(t \cdot \left(\lambda_a - \lambda_b\right) \cdot \left(\lambda_a - 2\lambda_b\right) + \lambda_a - 3\lambda_b\right)\right)}{\left(\lambda_a - \lambda_b\right)^3} + \frac{\lambda_a t + 1}{e^{\lambda at}} \end{aligned} \tag{76}$$

In Figure 3 is shown the reliability function of the system under analysis (green) compared with two alternative solutions: a cold stand-by redundancy with serial items on each branch (red) and a cold stand-by redundancy with parallel (hot standby) blocks instead (blue).

This chart provides a close comparison between the different architectures and shows the corresponding reliability trend: as it was expected, the cold standby redundancy of cold standby blocks offers the best reliability performance. This result evidences the trustworthiness of the proposed methodology using the developed reliability functions [29-31]. The reliability functions described in this chapter are fully implemented in a dedicated software named *RBDesigner*® which plays a fundamental role in the Reliability assessment loop. The features of this software and a case study are shown in Chapter 3.

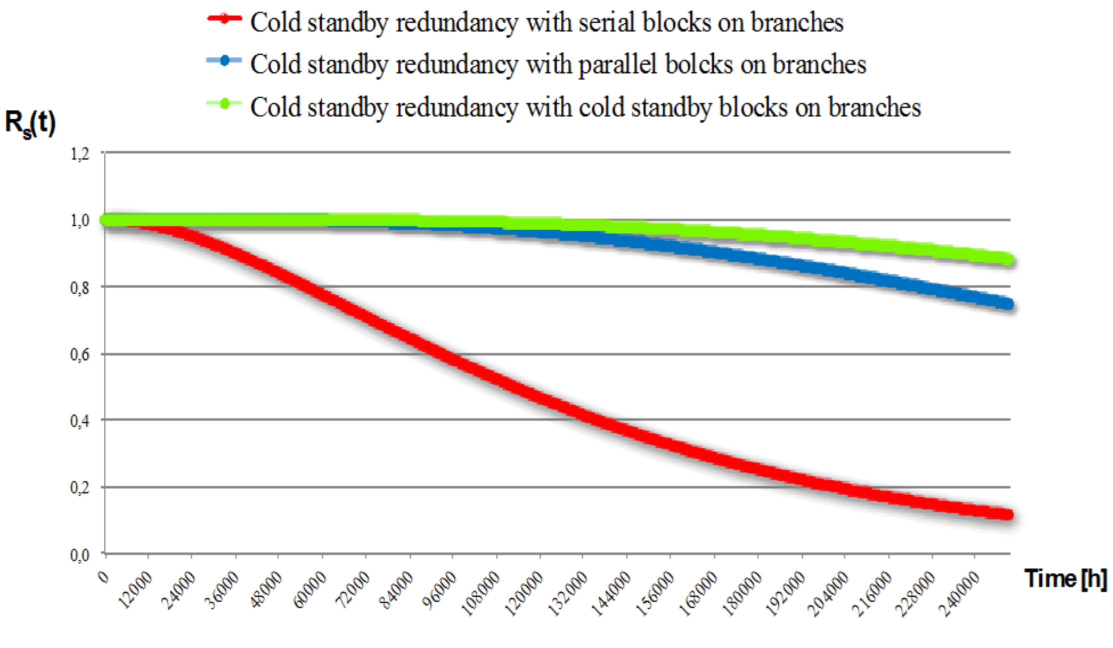


Fig. 3. Reliability vs. time plot for different system configuration

Chapter 3

Design for Reliability

3.1 Design for Reliability

Life Data Analysis and Weibull Analysis are two of the most used methods in reliability but nowadays they are not enough to achieve high reliability performance.

There are many other activities that are required to take part to an effective reliability program and develop reliable products: strategic vision, suitable planning, optimal resource allocation and the full integration of reliability practices in the development process.

Design for Reliability (DFR) is the whole process that takes into account all these activities and involves the set of tools necessary for the product and process design.

Design engineers need to achieve reliability standards to be competitive on the market, reduce warranty costs and satisfy customer expectations: these targets require that reliability be weaved into the whole development cycle [30-32].

The increasing complexity of systems involving numerous interactions and/or interfaces, diversified usages and stress profiles produced a corresponding growth in the complexity of reliability methods that nowadays are required to be well defined and incorporated into the whole design cycle.

A clarification is now necessary to distinguish reliability from quality: a quality control is done to ensure that the product will perform as expected after manufacturing process while a reliability procedure provides the probability that an item will perform its intended function for a designated period of time without failure (under specified conditions).

Furthermore, despite Quality Control methods, Design for Reliability is a process focused on achieving high long-term reliability to identify and prevent design issues early in the development phase [33-34].

The DFR process is based on the fight between stress and strength: a product fails when the stress experienced by the product exceeds its strength. The solution to reduce the failure probability and, as a consequence, increase the reliability is to cut down the interference between stress and strength; this is the goal during Design for Reliability assessment.

DFR process is organized in six steps:

- Define the reliability requirements for a product together with the expected environmental and usage conditions of the product. The definition of these requirements can be assessed following different procedures and taking into account contracts, benchmarks, competitive analysis, customer expectations and costs.

In a complex system the reliability requirement goal can be allocated to the component level with different allocation techniques that are described in the following paragraphs. After the requirement assessment the next step is translate them into design (and manufacturing) requirements.

- Identify the impact of the new product in terms of change with the past design and production: the new item can be a completely new product, an upgrade of an existing product or for example an existing product that is introduced to a new market/application. These changes produce changes in design, material, manufacturing, usage, environment, interfaces etc. and they require to identify and prioritize the key reliability risks and the corresponding risk reduction strategy. One of the best techniques for this purpose is Failure Mode and Effect Analysis.
- Analyse the product's reliability from early design phases in order to validate physics of failure, produce simulation models and deepen failure risks and mechanics.

  After this study the product weaknesses are shown so the analysts can quantify failure parameters predict product life and focus the reliability improvement efforts. At this stage the Life Data Analysis (described in Chapter 1) is a powerful method to statistically estimate the reliability of the product and calculate various reliability-related metrics. Also System Reliability Analysis with Reliability Block Diagrams (RBDs) is widely used to model the whole system reliability following the information and probabilistic data developed on the component or subsystem level.

  All of these methods are a great support for design engineers to verify whether the product meets its reliability goals, compare designs, avoid failures and achieve warranty returns.
- Validate the design using different tests to make sure that the product is ready for production; the use of statistical methods is recommended to develop a test plan and demonstrate the achievement of desired goals with the least expense of resources. The following step is to reduce or eliminate problems introduced by the manufacturing process since during this phase many variations in terms of materials, processes, manufacturing sites, human factors, etc. are involved so some changes in the design might be necessary to improve the whole system robustness.
- Monitor and Control to describe the actions required at each phase of the process to assure that all process outputs will be in control and that the requirements are achieved. Some tests like Burn-in and Screening are useful to prevent from infant mortality failures caused by manufacturing-related problems.

  Anyway continuous monitoring and field data analysis are necessary to observe the behaviour of the product in “real” applications and acquire data for improvements or future projects.

Following the steps described above a design engineer can generate a reliable product with a focus on reliability performance and requirement [30-34].

3.2 Reliability Allocation

Reliability Allocation (RA) is a top-down technique that allows apportioning the reliability goal of the system between its components: this is a very sensitive issue in industrial and commercial environments.

Furthermore in order to satisfy the product requirements, the first step in the design phase is to translate the overall system reliability goal into reliability requirements for each of the subsystems: for this reason the RA processes are fundamental to assign reliability requirements to individual units and obtain the target system reliability [34-37].

During the years a lot of methods for reliability allocation assessment were developed but they all follow the same algorithm: the failure rate to be allocated to a generic subsystem is directly proportional to the failure rate of the whole system. The proportionality constant ω is said *weight factor* and each allocation method has a dedicated procedure to assess its own weight factors [30].

Every technique shares with the others two hypothesis: all subsystem must be in series configuration and follows exponential failure distribution.

Under these assumptions, the reliability to be allocated to each subsystem is given by:

$$\boldsymbol{R_i}^*(\boldsymbol{t}) = [\boldsymbol{R_{SYS}}^*(\boldsymbol{t})]^{\boldsymbol{\omega_i}} \tag{77}$$

Where $R_{SYS}{}^*(t)$ is the system reliability target.

Nowadays many reliability allocation methods are available and the most important are the following: Equal Allocation Method (Department of Defense of USA, 1988), ARINC (Alven, 1964), Advisory Group of Reliability of Electronic Equipment (AGREE, 1957), FOO Technique (Department of Defense of USA, 1988), Bracha Technique (J.V. Bracha, 1964), Average Weighting Allocation Method (Kuo, 1999) and Maximal Entropy Ordered Weighting Average Method (Chang, 2009) [34].

3.2.1 ARINC method

The ARINC apportionment method was designed by ARINC Research Corporation, a subsidiary of Aeronautical Radio, Inc. This method is based on the assumption that the reliability of components can be assessed using previous calculations on similar components. The mathematical expression of weight factors is the following:

$$\boldsymbol{\omega_i} = \frac{\boldsymbol{\lambda_i}}{\boldsymbol{\lambda_{SYS}}} = \frac{\boldsymbol{\lambda_i}}{\sum_{j=1}^{N} \boldsymbol{\lambda_j}} \tag{78}$$

Where λ_i is the estimated failure rate of the component *i*-th obtained through a similar system and λ_{SYS} is the estimated failure rate of the whole architecture [35].

3.2.2 AGREE method

AGREE technique considers the complexity of each subsystem to calculate the weighting factors: these are assessed as the number of elements of the generic subsystem n_i compared to the total number of components N_{SYS} of overall configuration. This technique also considers the importance I_i of each subsystem i, where importance is defined as the probability that the system fails when the subsystem fails [36].
Weighting factors are given by:

$$\boldsymbol{\omega_i} = \frac{\boldsymbol{C_i}}{\boldsymbol{I_i}} \frac{\boldsymbol{t}}{\boldsymbol{t_i}} \tag{79}$$

3.2.3 FOO method

The FOO technique was first introduced in 1976 and is included in the MIL-HDBK-338B Electronic Reliability Design Handbook (Department of Defense of USA, 1988) as a method to develop and implement reliability programs for all types of military products. With the FOO method, subsystem allocation factors are computed as a function of a numerical rating of complexity (C_o), state-of-the-art (S_t), operating time (O_t), and environment condition (E_n) [34].
Each rank is based on a scale from 1 to 10 (Tab. 1) and they are estimated using design engineering and expert judgments.
The rating values are then multiplied to achieve the partial weight factor β_i.
The final product results in a value ranging from 1 to 10000 and the subsystem ratings are normalized so that their sum is equal to 1 [34].
Weighting factors are given by:

$$\boldsymbol{\omega_i} = \frac{\boldsymbol{C_{o_i} O_{t_i} E_{n_i} S_{t_i}}}{\boldsymbol{\sum_{j=1}^{N} C_{oj} O_{tj} E_{nj} S_{tj}}} = \frac{\boldsymbol{\beta_i}}{\boldsymbol{\sum_{j=1}^{N} \beta_j}} \tag{80}$$

Table I - Rules for the assessment of influence factors

Influence Factors	**Rating**
Complexity $\mathbf{C_o}$	**1** 2 3 4 5 6 7 8 9**10** Low Max
Environment Condition $\mathbf{E_n}$	**1** 2 3 4 5 6 7 8 9**10** Low Max
State of the art $\mathbf{S_t}$	**1** 2 3 4 5 6 7 8 9**10** Max Low
Operating time $\boldsymbol{O_t}$	**1** 2 3 4 5 6 7 8 9**10** Max Low

3.2.4 Bracha method

Bracha method uses the same factors of FOO technique but it privileges the S_t factor inside the expression for calculate the partial weight factors [31]:

$$\boldsymbol{D_i = S_{t_i}\left(C_{o_i} + O_{t_i} + E_{n_i}\right)} \tag{81}$$

Unlike the FOO method, in this technique each influence factor is achieved by means of special formulas and it ranges within the interval [0;1].
The subsystem rating are then normalized so the weighting factors are given by:

$$\boldsymbol{\omega_i = \frac{S_{t_i}\left(C_{o_i}+O_{t_i}+E_{n_i}\right)}{\sum_{j=1}^{N} S_{t_j}\left(C_{o_j}+O_{t_j}+E_{n_j}\right)} = \frac{D_i}{\sum_{j=1}^{N} D_{\square}}} \tag{82}$$

3.2.5 AWM method

Kuo (1999) created an average weighting allocation method as a guide for reliability allocation design. The method uses a questionnaire investigation to select the most influential system reliability factors such as complexity, state-of-the-art, system criticality, environment, safety, and maintenance in order to determine the subsystem reliability allocation ratings. Each rank is estimated on a scale from 1 to 10 using design engineering and expert judgments to obtain the subsystem reliability rate [34].
Suppose a system is composed of *N* subsystem, *m* is the number of influence factor and *p* the number of expert. Let Y_{ij} denote the *j*-th rating for subsystem $i. X_{K_{ij}}$ is the *j*-th rating for subsystem *i* set by *L*-th expert. To each of the factors is assigned the following value:

$$\boldsymbol{Y_{ij} = \frac{1}{p}\sum_{k=1}^{p} X_{K_{ij}} \,\forall i = 1, \dots, m \forall j = 1, \dots, N} \tag{83}$$

Two different models can be used to allocate weighting factors ω_i:

- Geometric model

$$\boldsymbol{\omega_i = \frac{\prod_{j=1}^{N} Y_{ij}}{\sum_{f=1}^{m}\prod_{j=1}^{N} Y_{ij}} = \frac{B_i}{\sum_{f=1}^{m} B_f}} \tag{84}$$

- Arithmetic model

$$\boldsymbol{\omega_i = \frac{\sum_{j=1}^{N} Y_{ij}}{\sum_{f=1}^{m}\sum_{j=1}^{N} Y_{ij}} = \frac{C_i}{\sum_{f=1}^{m} C_f}} \tag{85}$$

3.2.6 MEOWA method

In 1988 Yager first introduced the concept of OWA operators, which are important aggregation operators within the class of weighted aggregation methods. It has the ability to derive optimal weights of the attributes based on the rating of the weighting vectors after an aggregation process [34].

An OWA operator of dimension *n* is mapped *F* from $I^n \longrightarrow I$, where *I = [0, 1]*; the associated weighting vector $W = [w_1, w_2, \ldots, w_n]^T$ is defined as follows:

$$\boldsymbol{\sum_{i=1}^{n} w_i = 1 \quad \forall w_i \in [0, 1], \quad i = 1, 2, \ldots n} \tag{86}$$

$$\boldsymbol{f(a_1, \square_2, \ldots, a_n) = \sum_{i=1}^{n} w_i b_i} \tag{87}$$

Where b_i is the *i*-th largest element in the collection $a_1, a_2, \ldots, a_n$ and $b_1 \geq b_2 \geq \cdots \geq b_n$. [7]

Later Yager introduced two important characterizing measurements with respect to the weighting vector W of the OWA operator. One of these two measures is "*orness of the aggregation*", which is defined below (Eq. 87).

Let's assume *F* is an OWA aggregation operator with a weighting vector $W = [w_1, w_2, \ldots, w_n]^T$, the degree of orness associated with this operator is defined as:

$$Orness(W) = \alpha = \frac{1}{n-1} \sum_{i=1}^{n} (n-i)\, w_i \tag{88}$$

Where *Orness(W)* $= \alpha$ is a situation parameter [34] and can vary within the interval [0;1].

The second characterizing measurement introduced by Yager is the "*dispersion of the aggregation*" that is defined as:

$$Dispersion(W) = -\sum_{i=1}^{n} w_i \ln(w_i) \tag{89}$$

O'Hagan (1988) combined the principle of maximum entropy and OWA operators to propose a particular OWA weight that has maximum entropy with a given level of orness. This approach is based on the solution of the following mathematical problem: Maximize $Dispersion(W)$ subject to:

$$\boldsymbol{\frac{1}{n-1} \sum_{i=1}^{n} (n-i)\, w_i = \alpha} \tag{90}$$

Where $0 \leq \alpha \leq 1$; $\sum_{i=1}^{n} w_i = 1$; $0 \leq w_i \leq 1$.

Fuller and Majlender (2001) used the method of Lagrange multipliers on Yager's OWA equation to derive a polynomial equation, which can determine the optimal weighting vector under the maximal entropy. By their method, the associated weighting vector is easily obtained by the following equations:

$$\boldsymbol{w_j = \sqrt[n-1]{{w_1}^{n-j} {w_n}^{j-1}}} \tag{91}$$

$$\boldsymbol{w_n = \frac{[(n-1)\alpha - n]w_1 + 1}{(n-1)\alpha + 1 - nw_1}} \tag{92}$$

$$\boldsymbol{w_1[(n-1)\alpha + 1 - nw_1]^n = [(n-1)\alpha]^{n-1}\{[(n-1)\alpha - n]w_1 + 1\}} \tag{93}$$

With situation parameter $\alpha \in \left[\frac{1}{2}\ ; 1\right]$.

After determining the weighting vector W, the overall factor Z_k can be achieved and then allocated to each subsystem [34].

This index considers all n influence factors, each multiplied by the optimal weight.

$$\boldsymbol{Z_k = \sum_{i=1}^{n} w_i b_{i,k}} \tag{94}$$

Where$b_{1,k}, b_{2,k}, \dots, b_{n,k}$ are the values assigned to the influence factors of the k-th subsystem.

Depending on the system under analysis it is possible to choose the number and the type of influence factors. Weighting factors ω_k are given by:

$$\boldsymbol{\omega_k = \frac{Z_k}{M_\alpha} \text{ where } M_\alpha = \sum_{i=1}^{n}\left(w_i \sum_{j=1}^{N} b_{i,j}\right)} \tag{95}$$

This allocation procedure is called MEOWA method (Chang, 2009) [34].

MEOWA technique provides a situation parameter $\alpha \in [0{,}5\ ; 1]$ to assess the reliability allocation values:

- $\alpha = 1$ is used to represent the situation when the decision-maker is very confident;
- $\alpha = 0.5$ is used when the decision-maker faces a moderate uncertainty.

The conditional parameter is particularly useful when the reliability allocation procedure is achieved during design phase using imprecise, incomplete or uncertain information [34].

The situation parameters have particular effect on the value of the weighting vector component since they influence the weight of the factor with very high or very low ratings.

In conclusion, except for ARINC and AGREE methods, the weighting factor for Reliability Allocation is given by:

$$\boldsymbol{\omega_i = \frac{f(Y_{ij})}{\sum_{i=1}^{N} f(Y_{ij})}} \tag{96}$$

Where Y_{ij} denote the j-th rating for subsystem i, and f is a function of Y_{ij}.

3.2.7 Reliability Allocation in Redundant Architectures

In order to enlarge the range of applicability of allocation methods, the following hypotheses are required:

- Replacement of the reliability function R(t) with unreliability function Q(t) and successive re-conversion in terms of reliability:

$$\boldsymbol{Q_i}^*(\boldsymbol{t}) = [\boldsymbol{Q_{SYS}}^*(\boldsymbol{t})]^{\omega_i} \qquad \boldsymbol{R_i}^*(\boldsymbol{t}) = \mathbf{1} - \boldsymbol{Q_i}^*(\boldsymbol{t}) \tag{97}$$

- Inversion of the influence factor rating; this step is necessary to keep the right relationship between the factor definition and the corresponding rating.
  In traditional methods for series system the complexity factor C_0 ranges from 1 to 10, where 1 corresponds to the least complex system and 10 to the most complex one. As a consequence a growth in the complexity produces an increase in the weighting factor ω_i and a decrease in the reliability allocated.
  For parallel systems the new complexity factor $\overline{C_o}$ has to be defined as:

$$\overline{\boldsymbol{C_o}} = \mathbf{11} - \boldsymbol{C_o} \tag{98}$$

Following this definition a growth in terms of complexity produces a decrease in the weighting factors ω_i and a consequent increase of the allocated unreliability like series systems.

This approach works for parallel architectures in some methods (in particular FOO, Bracha, AWM and MEOWA) but it is not applicable for ARINC and AGREE techniques.
This is the roadblock for the applicability of these methods since it is mandatory to achieve reliability allocation parameters without restrictions for the model structure.
The explanation of this technical limit is the following: the solution to extend the applicability range of these allocation methods from simple series configurations to parallel architectures is to use the system reliability definition for series (system reliability is given by multiplication of the reliability of each items). Similarly, in parallel configurations, the system unreliability is calculated as multiplication of the unreliability of each component.
For this reason the first hypothesis is necessary in order to apply the techniques to redundant designs. Anyway it is still mandatory that the sum of all the factors ω_i is unitary:

$$\sum_{i=1}^{N} \boldsymbol{\omega_i} = \mathbf{1} \tag{99}$$

$$\boldsymbol{R_i}^*(\boldsymbol{t}) = [\boldsymbol{R_{SYS}}^*(\boldsymbol{t})]^{\omega_i} \tag{100}$$

$$\prod_{i=1}^{N} \boldsymbol{R_i}^*(\boldsymbol{t}) = \prod_{i=1}^{N} [\boldsymbol{R_{SYS}}^*(\boldsymbol{t})]^{\omega_i} = [\boldsymbol{R_{SYS}}^*(\boldsymbol{t})]^{\sum_{i=1}^{N}\omega_i} = \boldsymbol{R_{SYS}}^*(\boldsymbol{t}) \tag{101}$$

$$\boldsymbol{Q_i}^*(\boldsymbol{t}) = [\boldsymbol{Q_{SYS}}^*(\boldsymbol{t})]^{\omega_i} \tag{102}$$

$$\prod_{i=1}^{N} \boldsymbol{Q_i}^*(\boldsymbol{t}) = \prod_{i=1}^{N} [\boldsymbol{Q_{SYS}}^*(\boldsymbol{t})]^{\omega_i} = [\boldsymbol{Q_{SYS}}^*(\boldsymbol{t})]^{\sum_{i=1}^{N}\omega_i} = \boldsymbol{Q_{SYS}}^*(\boldsymbol{t}) \tag{103}$$

Following Eq. 100-101 for series and Eq. 102-103 for parallel systems the fundamental relationships of redundancy are fulfilled.

Now it can be observed why ARINC method cannot be extended to architectures with complexity greater then simple series. Generically, for all possible configuration:

$$\boldsymbol{\omega_i} = \frac{\lambda_i}{\lambda_{SYS}} \tag{104}$$

In series configuration the system failure rate is the sum of all items failure rate, so the weighting factors are normalized and equation (23) is satisfied.

In parallel configuration the system failure rate is given by a generic function of component failure rates. So:

$$\boldsymbol{\lambda_{SYS}} \neq \sum_{i=1}^{N} \boldsymbol{\lambda_i} \tag{105}$$

$$\sum_{i=1}^{N} \boldsymbol{\omega_i} = \sum_{i=1}^{N} \frac{\lambda_i}{\lambda_{SYS}} \neq \mathbf{1} \tag{106}$$

As a consequence of (1) and (30), the following relationship is achieved:

$$\prod_{i=1}^{N} \boldsymbol{Q_i}^*(\boldsymbol{t}) = \prod_{\square=1}^{N} [\boldsymbol{Q_{SYS}}^*(\boldsymbol{t})]^{\omega_i} \neq \boldsymbol{Q_{SYS}}^*(\boldsymbol{t}) \tag{107}$$

This result is produced by the conflict with the fundamental relationship of parallel configuration given.

Therefore for ARINC method the reliability allocation in case of redundant architectures is not possible with these assumptions.

Similarly AGREE method is not applicable to parallel systems because the sum of weighting factors defined by Eq. 78 is not unitary and Eq. 101 is not satisfied.

On the contrary, in FOO, Bracha, AWM and MEOWA methods the weighting factors are given by the generic Eq. 97 so:

$$\sum_{i=1}^{N} \boldsymbol{\omega_i} = \mathbf{1} \quad \prod_{i=1}^{N} \boldsymbol{Q_i}^*(\boldsymbol{t}) = \prod_{i=1}^{N} [\boldsymbol{Q_{SYS}}^*(\boldsymbol{t})]^{\omega_i} = \boldsymbol{Q_{SYS}}^*(\boldsymbol{t}) \tag{108}$$

In this case the reliability apportion works also for redundant systems [34-37]. These methods are the reference to assess Reliability Allocation in complex systems and, in general, for systems containing redundant architectures: this trend is shown in paragraphs 3.2.8, 3.2.9 and 3.2.10.

3.2.8 Reliability Allocation in Complex Systems

The Reliability Block Diagram of the system under test is shown in Fig. 1.

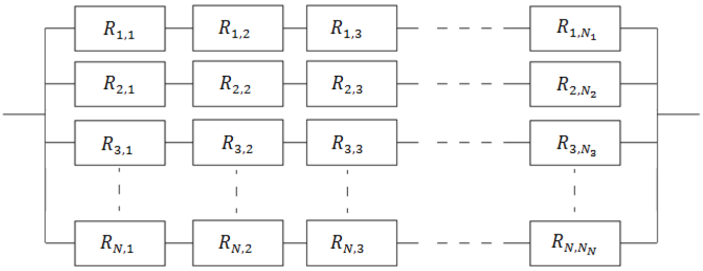


Fig. 1. RBD of generic complex system

This system is composed of N branches in parallel configuration. Each branch is in turn composed of a cascade of a generic N_i number of elements in series configuration, with $i = 1, 2, \dots N$.

In order to achieve Reliability Allocation of this system, the following steps are required:

- The reliability target is allocated using one of the techniques described in the previous paragraphs and considering an equivalent system where each branch is simplified in one block; the N branches are in parallel configuration (Fig. 2).
- The reliability target achieved in the first step is used to allocate that requirement to each branch; the RBD of the generic i-th branch is shown in Fig. 3.

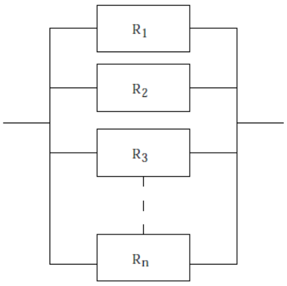


Fig. 2. – Equivalent system RBD at first step

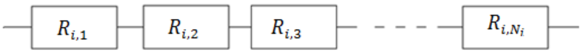


Fig. 3. RBD of generic branch to study at second step

Following the procedure described above, the first step is the assessment of equivalent influence factors of the single-branch subsystems; this procedure starts from the estimated factors of the individual elements inside the series chain once they are achieved throw expert judgments. Furthermore the following worst-case hypothesis are introduced:

- *COMPLEXITY* C_o→ the total complexity of a branch is given by the maximum complexity value between the elements belonging to the branch.

$$\boldsymbol{C_{op_i} = \max_{j=1,\dots,N_i} C_{o_j} \, \forall i = 1, 2, \dots, N} \tag{109}$$

- *STATEOF THE ART* S_t → the total state of the art of a branch is the mean between the states of the art of the elements belonging to the branch.

$$\boldsymbol{S_{tp_i} = \frac{1}{N_i}\sum_{j=1}^{N_i} S_{t_j}\; \forall i = 1, 2, \dots, N} \tag{110}$$

- *OPERATING TIME* O_t → the overall branch operating time is equal to the operating time of the most used element.

$$\boldsymbol{O_{tp_i} = \max_{j=1,\dots,N_i} O_{t_j}\; \forall i = 1, 2, \dots, N} \tag{111}$$

- *ENVIRONMENT FACTOR* E_n → the total environment factor of a branch is given by the maximum environment factor between the elements belonging to the branch.

$$\boldsymbol{E_{np_i} = \max_{j=1,\dots,N_i} E_{n_j}\; \forall i = 1, 2, \dots, N} \tag{112}$$

- *CRITICALITY* C_r→ the total criticality factor of a branch is given by the minimum criticality factor between the elements belonging to the branch.

$$\boldsymbol{C_{rp_i} = \min_{j=1,\dots,N_i} C_{r_j}\; \forall i = 1, 2, \dots, N} \tag{113}$$

- *MAINTAINABILITY* M_a→ the maintainability of a branch is the mean between the maintainability of the elements belonging to the branch.

$$\boldsymbol{M_{ap_i} = \frac{1}{N_i}\sum_{j=1}^{N_i} M_{a_j}\; \forall i = 1, 2, \dots, N} \tag{114}$$

- *SAFETY* S_a→ the total safety factor of a branch is given by the maximum safety factor between the elements belonging to the branch.

$$\boldsymbol{S_{ap_i} = \max_{j=1,\dots,N_i} S_{a_j}\; \forall i = 1, 2, \dots, N} \tag{115}$$

The Reliability Allocation procedure described above was implemented in a dedicated tool on MathWorks "Matlab r2015a" platform: the software calculates the reliability and the failure rate to be allocated to each component of the system.

The necessary inputs are system reliability goal, time to allocation, number of parallel subsystems, number of series elements of each subsystem, allocation method and influence factors [40-41].

The test of the developed tool on two dedicated case studies is shown in the following paragraphs.

3.2.9 Case Study A

Fig. 4 shows the complex system analysed in this paragraph with the developed tool.

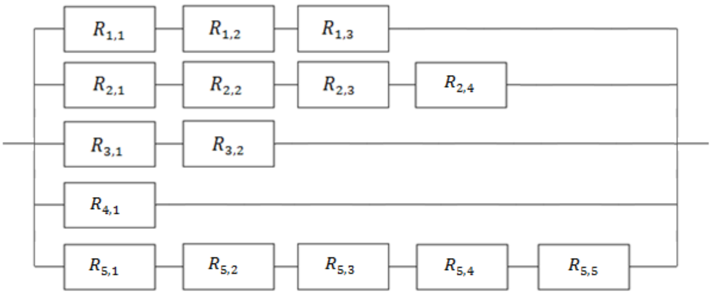


Fig. 4. RBD of case study A

The system is a special case of the generic complex system described in Fig. 1: it is composed by 5 parallel branches (N = 5), each made up of N_i components as follows: N_1= 3, N_2= 4, N_3= 2, N_4= 1, N_5= 5.

The system reliability goal to achieve through the Reliability Allocation procedure is $R_{SYS}{}^{*}(t) = 0.99$ where $t = 8760h$.

After a great number of tests and simulations, the MEOWA method shows the best results: the weighting vector w given by Eq. 92, Eq. 93 and Eq. 94 solve the problem arisen with the other methods to assign an appropriate weight to influence factors with very high/low values.

Table II shows the influence factors used in the simulation of MEOWA technique while Table III shows the reliability allocation results using the developed tool.

Table II - MEOWA influence factors for case study A

Branch	Element	C_o	E_n	S_t	C_r	M_a	S_a
1	1.1	1	4	7	7	7	6
	1.2	1	5	4	8	8	10
	1.3	1	2	10	6	9	3
2	2.1	3	5	9	9	8	8
	2.2	2	7	6	9	8	4
	2.3	4	6	7	10	6	7
	2.4	3	2	6	9	10	9
3	3.1	2	5	10	8	10	3
	3.2	2	6	6	7	6	8
4	4.1	7	3	2	6	10	6
5	5.1	2	4	9	8	6	8
	5.2	2	3	9	10	6	4
	5.3	2	5	9	5	4	3
	5.4	1	2	9	5	8	4
	5.5	1	2	9	8	6	7

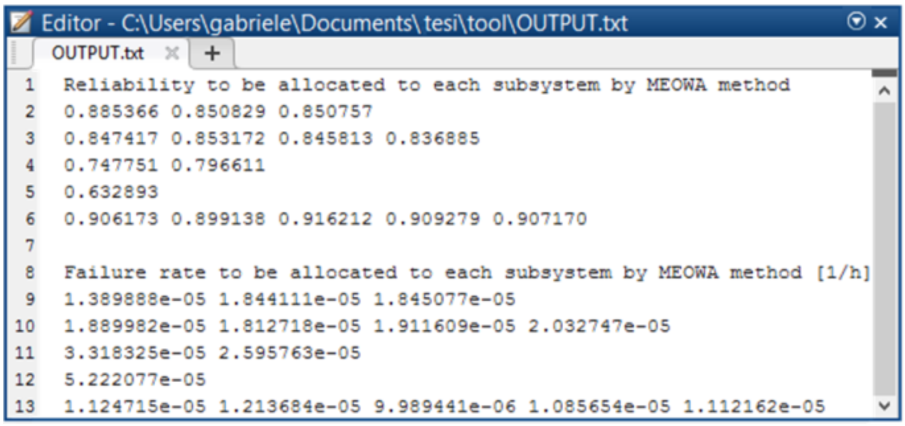
Table III - Tool output using MEOWA method

Branch	$R_{i,j}^*(t)$				
1	0,885	0,850	0,850		
2	0,847	0,853	0,845	0,836	
3	0,747	0,796			
4	0,632				
5	0,906	0,899	0,916	0,909	0,907

As a result, lower reliability is allocated to components with high influence factors (Table III); this trend validates both the tool operation and the RA method selected.

In fact other techniques (e.g. FOO, Bracha and AWM) calculate the weighting factors as a sum (or product) of influence factors: in this way the weight factors do not reflect the impact each influence factor has on the system. Therefore these methods can't take care of single factors with high/low ratings [40-43].

The tool calculates also the failure rate to be apportioned to each item, assuming that all the blocks of the system are single elements and not subsystems in turn. Figure 5 shows an example of the tool outcomes containing the simulation results for MEOWA technique.



```
Reliability to be allocated to each subsystem by MEOWA method
0.885366 0.850829 0.850757
0.847417 0.853172 0.845813 0.836885
0.747751 0.796611
0.632893
0.906173 0.899138 0.916212 0.909279 0.907170

Failure rate to be allocated to each subsystem by MEOWA method [1/h]
1.389888e-05 1.844111e-05 1.845077e-05
1.889982e-05 1.812718e-05 1.911609e-05 2.032747e-05
3.318325e-05 2.595763e-05
5.222077e-05
1.124715e-05 1.213684e-05 9.989441e-06 1.085654e-05 1.112162e-05
```

Fig. 5. RA tool output screenshot

3.2.10 Case Study B

Figure 6 shows the second case study chosen to test the proposed method and, at the same time, the potential of MEOWA technique supposing a variation of the situational parameter α.

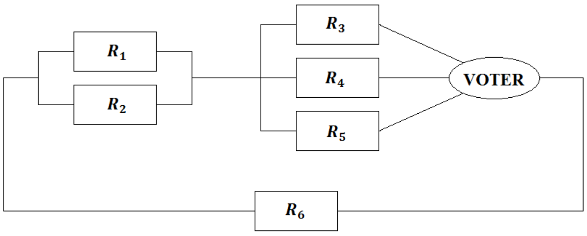


Fig. 6. RBD of the second case study

This complex system is another particular case of the generic model described above and it is composed by two parallel branches (*N* = 2), each made up of *N_i* subsystems as follows: N_1= 2, N_2= 1.

The first branch is made of two subsystems: two blocks in parallel configuration (subsystem A) and three blocks in TMR architecture (subsystem B).

The single element characterized by the R_6 reliability function will be considered as the subsystem C.

This kind of applications requires a double allocation procedure: the first step is the RA assessment at subsystem level, then the outcomes are used to allocate the final reliability target at each component.

In most applications TMR architecture is formed by three identical elements with the same reliability function. For this reason it is not possible to use the allocation methods shown in paragraph 2.2.6 since these techniques don't allocate the requirements uniformly to the items. In TMR configuration is mandatory to use the Equal Allocation Method that apportions the reliability target evenly between the redundant items; voter allocation, instead, follows the standard MEOWA procedure.

The influence factors determined by experts in reliability are shown in Table IV: the voter, as known, is the most critical component in TMR architectures and it is characterized by very low parameters with consequently high reliability performance demand [43].

Table IV – Influence factors for case study B

Subsystem	Item	C_o	E_n	S_t	C_r	M_a	S_a
A	**1**	4	10	10	4	10	3
	2	7	7	6	5	6	5
B	**3**	3	4	7	4	5	6
	4	3	4	7	4	5	6
	5	3	4	7	4	5	6
	Voter	2	2	1	1	3	3
C	**6**	3	5	4	6	7	9

Table V shows the influence factors achieved for subsystem A, B and C according to the hypothesis shown in paragraph IV.

Table V – Influence factors for subsystem A, B and C

Subsystem	Item	C_o	E_n	S_t	C_r	M_a	S_a
A	**1**	7	10	8	4	8	5
B	**2**	3	4	4	1	4	6
C	**3**	3	5	4	6	7	9

The simulation was performed with the same reliability target of the previous case study: $R_{SYS}{}^{*}(t) = 0.99$ where $t = 8760h$.

The results of the first step of simulation and the final results are shown in Table VI and Table VII (considering $R_3 = R_4 = R_5$ for the TMR architecture).

Table VI –Reliability of subsystems A, B and C calculated with the MEOWA method, varying α

	Reliability to be allocated to each subsystem		
α	Subsystem A	Subsystem B	Subsystem C
0,50	0,927	0,961	0,907
0,55	0,929	0,961	0,905
0,60	0,931	0,962	0,903
0,65	0,932	0,962	0,902
0,70	0,934	0,963	0,900
0,75	0,935	0,963	0,897
0,80	0,937	0,964	0,895
0,85	0,939	0,965	0,891
0,90	0,942	0,966	0,887

Table VII – Reliability of all items of the configuration calculated by MEOWA method, varying α

	Reliability to be allocated to each subsystem				
α	**R_1**	**R_2**	**R_3**	**R_V**	**R_6**
0,50	0,697	0,761	0,896	0,991	0,907
0,55	0,714	0,752	0,897	0,991	0,905
0,60	0,729	0,744	0,898	0,991	0,903
0,65	0,743	0,737	0,898	0,991	0,902
0,70	0,755	0,730	0,899	0,991	0,900
0,75	0,766	0,724	0,900	0,992	0,897
0,80	0,777	0,720	0,901	0,992	0,895
0,85	0,787	0,717	0,902	0,992	0,891
0,90	0,797	0,716	0,904	0,992	0,887

Figure 7 shows the RA trends in function of the situational parameter variation: the voter has the highest reliability to be allocated with a value close to 1 and its reliability allocation is not affected by the increase of α. This trend was expected since the voter is characterized by very low influence factors.

As shown in the figure, the reliability allocated to the elements involved in the parallel configuration is initially lower than the TMR ones: also this gap was expected since the elements in subsystem A are characterized by higher values than subsystem B (Table V). However the distance between the two reliabilities significantly decreases together with the growth of α [43].

Obviously redundant architecture ensures high reliability performance even for low values of reliability of its components and the reliability allocated to the parallel elements assumes very low values. For these two blocks: $R_1{}^*(t) < R_2{}^*(t)$, considering ⍰ $= 0.5$ and the presence of the maximum value (10) in three of the influence factors of the first element is the justification [40-42].

With the growth of the situational parameter the relationship between the two reliability curves is reversed, and the reliability of the component 1 becomes greater than the reliability of the other one.

This trend is produced by the influence of the situation parameter on the importance of the maximum ratings for the assessment of the weighting factors ω_i and this is possible through the weighting vector W defined by OWA operators.

Subsystem A it is the only one to take into account to select the value of situation parameter that best fits the whole system since the reliability allocated to subsystems B and C is subjected to very low variations.

Anyway this is not a standard rule: sometimes an influence factor with max rank is a sort of “out of standard” and the corresponding component may require a higher reliability allocation. For this reason the influence factors can’t be assigned by default and they need to be appointed case by case in order to achieve an optimum allocation [38-40].

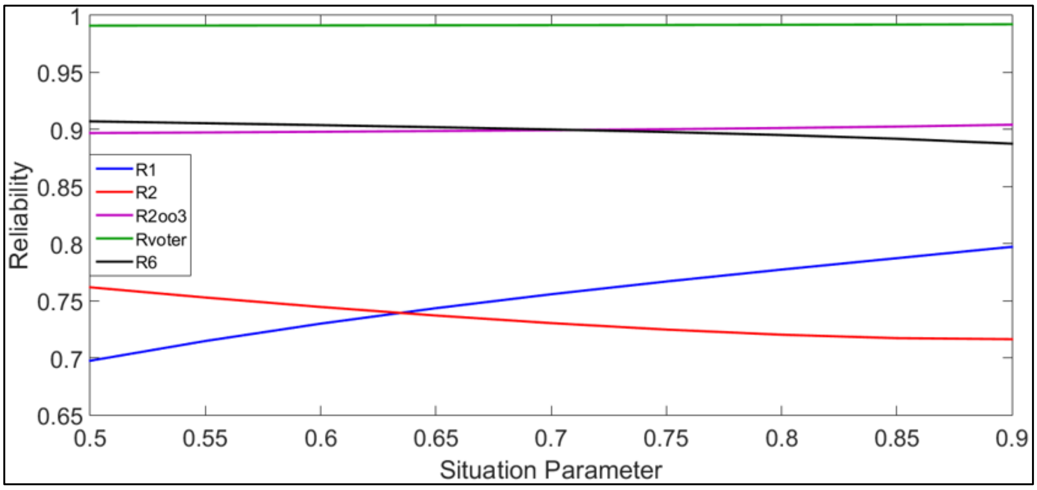


Fig. 7. Reliability allocation values using MEOWA method vs. situation parameter α

3.2.11 Discussion and Remarks

The test cases described in the previous paragraphs are necessary to summarize and compare all the Reliability Allocation techniques found in literature such as ARINC, AGREE, FOO, Bracha, AWM and MEOWA: the first test bench (standard parallel architecture) showed that the ARINC and AGREE techniques are not suitable for this kind of application due to mathematical problems, so their employment is limited to series configurations.

In case of redundant architectures FOO, Bracha, AWM and MEOWA techniques should be applied but the assumptions described in paragraph 2.2.7 are required.

The two case studies showed how the new approach proposed in this study is useful to achieve RA parameters in complex systems and that all the techniques applicable to the parallel configuration are also applicable to the complex architectures under analysis: the proposed method requires the assessment of the allocation process twice, both at subsystem and component level.

Furthermore in this study an innovative procedure is described and implemented in the dedicated RA tool developed on Matlab platform: this software is useful to achieve the subsystem influence factors starting from the component ones using MEOWA method [43].

In conclusion, the Reliability Allocation procedure in complex systems containing standby architectures can be summarized as follows:

- The first step is the assessment of the reliability requirements using the MEOWA technique (the results are in function of the situational parameter α);
- The second step is plotting the reliability values allocated to each element of the system;
- The final step is the analysis of the collected data and their trends in order to decide which situational parameter best-fits the system and achieve the optimal reliability allocation.

3.3 Reliability Importance

Reliability Importance (RI) is one of the most trustworthy and efficient procedures to measure the impact each component has on the overall system reliability.

RI methods are widely used during design stage since engineers can optimize efforts to improve the system reliability focusing on the components that have the greatest effect on the whole system.

One of the main advantages that this procedure can offer is cost and time saving due to the possibility to achieve a trustworthy prediction of reliability importance parameters also in the early stages of industrial product development; Reliability Importance offers real-time feedbacks to designers and this information are mandatory to found the system advancement on reliability importance outcomes. Furthermore it is possible to compare different solutions, prove system robustness and reduce time for improvements [44-45].

The analysis of the system and the reliability assessment are the first steps for RI assessment and they obviously require a deep knowledge of the system itself: for this purpose one of the most used technique is Reliability Block Diagram (RBD) that is a top-down technique based on a functional diagram of all the components making up the system.

The RBD outcomes are useful to take into account the contribution of each component to system failure using a one-to-one correspondence between components and blocks: each block in the diagram is described by a specific failure rate and its connections with the rest of the system.

After system reliability assessment design engineers can identify the least reliable component in the system, improve the whole system reliability, prioritize re-design actions to be taken (reliability improvement) or suggest the most effective way to operate and maintain system status [31].

3.3.1. Reliability Importance Measures

In literature there are many reliability importance indices, most of them are specific methods for dedicated practice while others have a wide range of applications and they are suitable also for generic complex systems; after many test on different test cases, Improvement Potential (IP) and Credible Improvement Potential (CIP) turned out to be the best two metrics for our purpose [31].

Improvement Potential index establish how much the system reliability would benefit from making one component completely reliable; in other words it assess the maximum potential in improving a specific component reliability [46-48]. IP measure is the difference between the system reliability with a perfect component *i* and the system reliability with the actual component, as follows:

$$I_i^{IP}(t) = R_s[t; R_i(t) = 1] - R_s(t) \tag{116}$$

Where:

- $I_I^{IP}(t)$ is Improvement Potential index of component *i* at time *t*;
- $R_S(t)$ is system reliability at time *t*;
- $R_i(t)$ is reliability of component *i* at time *t*.

The main criticality of this reliability index is that the supposed improvement is not physically achievable since it is not actually possible improving component reliability $R_i(t)$ to 100%.
In order to solve this issue a new reliability importance measure was introduced, the Credible Improvement Potential metric.
CIP solves the limit described above: following this procedure, the $R_i(t)$ value is improved to a new one $R_i^+(t)$ that represents the reliability corresponding to the state of the art for this type of components. CIP definition is shown below:

$$I_i^{CIP}(t) = R_s[t; R_i(t) = R_i^n(t)] - R_s(t) = \Delta R_s(t) \tag{117}$$

Where:

- $I_I^{CIP}(t)$ is Credible Improvement Potential index of component *i* at time *t*;
- $R_S(t)$ is system reliability at time *t*;
- $R_i(t)$ is reliability of component *i* at time *t*;
- $R_i^+(t)$ is the improved reliability of component *i* at time *t*.

The use of a component with reliability $R_i^+(t)$ in place of a correspondent component defined by $R_i(t)$ produces a system reliability improvement defined by $I_I^{CIP}(t)$: CIP measure solves the issue arisen with Improvement Potential metric and is usable in presence of standby redundancy architectures [31].
For this reason Credible Improvement Potential turned out to be the best metric for our purpose and the reliability improvement was set as follows:

$$n = \frac{\lambda_{old}}{\lambda_{new}} = \frac{\lambda_i}{\lambda_i^+} \qquad R_i(t) = e^{-\lambda_i t} \qquad R_i^+(t) = e^{-\frac{\lambda_i}{n}t} \tag{118}$$

The introduction of "*n*" as the Improvement Factor (IF) is necessary because, as said before, component reliability improvement to 100% is not physically achievable and *n* concerns the quality, the application and the effort that designers would accept to improve system reliability.
The range of suitable values for this parameter was defined using the quality factor π_Q in MIL-HDBK-217 [23] where it is used to calculate the failure rate of a specific item assuming different values in function on equipment quality. Therefore, following the definition and the range of values of π_Q defined in [49], in this study the Improvement Factor *n* is considered fixed at value of 4: this assumption is justified for *Oil&Gas* systems and for any other kind of application with high quality standards and requirements. In [31] Quality Factor assumes values greater than 4 when a low quality equipment is compared to a top quality one but, for

this study, low quality is not taken into account so the chosen value is adequate for standard/mid-quality to top-quality improvements.

CIP metric, in other words, performs the achievable percent improvement with a component that offers higher reliability standard. It assumes values in the following interval:

$$0 \leq I_i^{CIP}(t) < 1 \tag{119}$$

Where:

- $I_I^{CIP}(t)$=0 means that there is no improvement, $R_i(t)$=$R_i^+(t)$ and system reliability is the same then before;
- $I_I^{CIP}(t)$>0 low CIP value corresponds to little reliability improvement;
- $I_I^{CIP}(t)$<1 high CIP value corresponds to high reliability improvement;
- $I_I^{CIP}(t)$=1 is not included in the CIP range because it corresponds to a 100% system reliability improvement and it is possible only in case of $R_i(t)$=0 and $R_i^+(t)$=1.

The system reliability, as obvious, will improve when a component is replaced with another one that offers higher reliability performance: this procedure is done for all the items making-up the system and produces a wide insight to understand which item has the greatest impact on the whole system reliability [31]. A test case of CIP procedure is shown in the following paragraph.

3.3.2. Test Case: Fault Tolerant Complex System

The ideal test bench to validate the Credible Improvement Potential method is a generic complex system containing standby redundant blocks (Fig. 8).

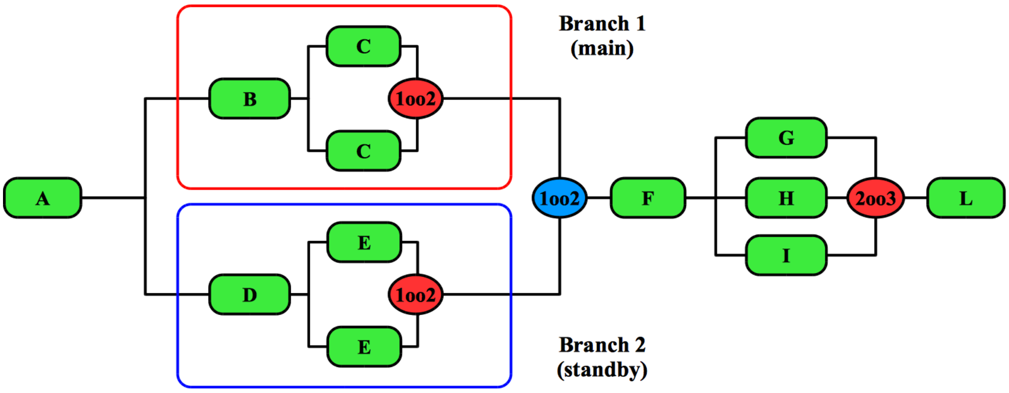


Fig. 8. Case study Reliability Block Diagram

Redundancy is the most common technique to achieve fault tolerance as it was described in Chapter 2.

The system under analysis consists of 12 blocks where all redundant configurations are hot standby, 1oo2 and 2oo3 respectively, except for the 1oo2 cold standby architecture between the two branches: Branch 1 – Main and Branch 2 - Standby.

The reliability functions necessary to assess Reliability Importance measures were achieved using the method described in Chapter 2 and the functions are listed below.

Items in series (*A,B,D,F,L*):

$$R_i(t) = e^{-\lambda_i t} \tag{120}$$

1oo2 hot standby node (*C*):

$$R_{2C}(t) = 2e^{-\lambda_C t} - e^{-2\lambda_C t} \tag{121}$$

1oo2 hot standby node (*E*):

$$R_{2E}(t) = 2e^{-\lambda_E t} - e^{-2\lambda_E t} \tag{122}$$

1oo2 cold standby node (*Branch 1-2*):

$$\begin{aligned} R_{cold}(t) = {} & e^{-\lambda_B t}(-e^{-2\lambda_C t} + 2e^{-\lambda_C t}) + e^{-(\lambda_E+2\lambda_F)t}\left[-\frac{2(\lambda_B+\lambda_C)}{\lambda_B+\lambda_C-\lambda_E-2\lambda_F} + \frac{\lambda_B+2\lambda_C}{\lambda_B+2\lambda_C-\lambda_E-2\lambda_F}\right] + \\ & + \frac{4(\lambda_B+\lambda_C)e^{-(\lambda_E+\lambda_F)t}}{\lambda_B+\lambda_C-\lambda_E-\lambda_F} + \frac{2(\lambda_B+2\lambda_C)e^{-(\lambda_E+\lambda_F)t}}{-\lambda_B-2\lambda_C+\lambda_E+\lambda_F} - e^{-(\lambda_B+2\lambda_C)t}\left[-\frac{2(\lambda_B+\lambda_C)e^{\lambda_C t}}{\lambda_B+\lambda_C-\lambda_E-2\lambda_F} + \frac{\lambda_B+2\lambda_C}{\lambda_B+2\lambda_C-\lambda_E-2\lambda_F} + \right. \\ & \left. + \frac{4(\lambda_B+\lambda_C)e^{\lambda_C t}}{\lambda_B+\lambda_C-\lambda_E-\lambda_F} + \frac{2(\lambda_B+2\lambda_C)}{-\lambda_B-2\lambda_C+\lambda_E+\lambda_F}\right] \end{aligned} \tag{123}$$

2oo3 hot standby node:

$$R_{2oo3}(t) = e^{-(\lambda_G+\lambda_H)t} + e^{-(\lambda_H+\lambda_I)t} + e^{-(\lambda_G+\lambda_I)t} - 2e^{-(\lambda_G+\lambda_H+\lambda_I)t} \tag{124}$$

System reliability:

$$\begin{aligned} R_s(t) = {} & e^{-(\lambda_A+\lambda_H+\lambda_N)t}\left[\mathrm{e}^{-(\lambda_I+\lambda_L)\mathrm{t}} + \mathrm{e}^{-(\lambda_I+\lambda_M)\mathrm{t}} + \mathrm{e}^{-(\lambda_L+\lambda_M)\mathrm{t}} - 2\mathrm{e}^{-(\lambda_I+\lambda_L+\lambda_M)\mathrm{t}}\right]\cdot\left\{e^{-\lambda_B t}(-e^{-2\lambda_C t} + 2e^{-\lambda_C t}) + \right. \\ & + e^{-(\lambda_E+2\lambda_F)t}\left[-\frac{2(\lambda_B+\lambda_C)}{\lambda_B+\lambda_C-\lambda_E-2\lambda_F} + \frac{\lambda_B+2\lambda_C}{\lambda_B+2\lambda_C-\lambda_E-2\lambda_F}\right] + \frac{4(\lambda_B+\lambda_C)e^{-(\lambda_E+\lambda_F)t}}{\lambda_B+\lambda_C-\lambda_E-\lambda_F} + \frac{2(\lambda_B+2\lambda_C)e^{-(\lambda_E+\lambda_F)t}}{-\lambda_B-2\lambda_C+\lambda_E+\lambda_F} + \\ & \left. - e^{-(\lambda_B+2\lambda_C)t}\left[-\frac{2(\lambda_B+\lambda_C)e^{\lambda_C t}}{\lambda_B+\lambda_C-\lambda_E-2\lambda_F} + \frac{\lambda_B+2\lambda_C}{\lambda_B+2\lambda_C-\lambda_E-2\lambda_F} + \frac{4(\lambda_B+\lambda_C)e^{\lambda_C t}}{\lambda_B+\lambda_C-\lambda_E-\lambda_F} + \frac{2(\lambda_B+2\lambda_C)}{-\lambda_B-2\lambda_C+\lambda_E+\lambda_F}\right]\right\} \end{aligned} \tag{125}$$

The following table shows the failure rates and MTTFs of all the components in the system while Fig. 9 and Fig. 10 shows the system reliability and system failure rate in function of time.

Table VIII – Failure rates and MTBF

Component	Failure rate [failures/10^6h]	MTTF [h]
Item A	1,51	662252
Item B	7,05	141844
Item C	8,10	123457
Item D	5,31	188324
Item E	40,42	24740
Item F	1,11	900901
Item G	5,23	191205
Item H	6,12	163399
Item I	2,26	442478
Item L	3,11	321543

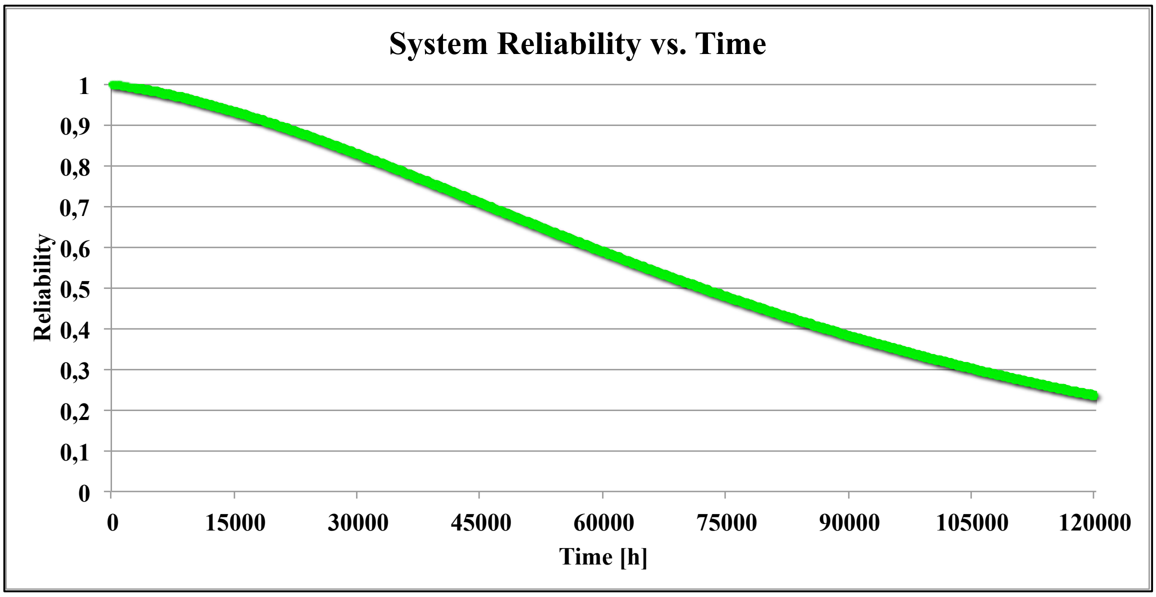


Fig. 9. System Reliability vs. Time

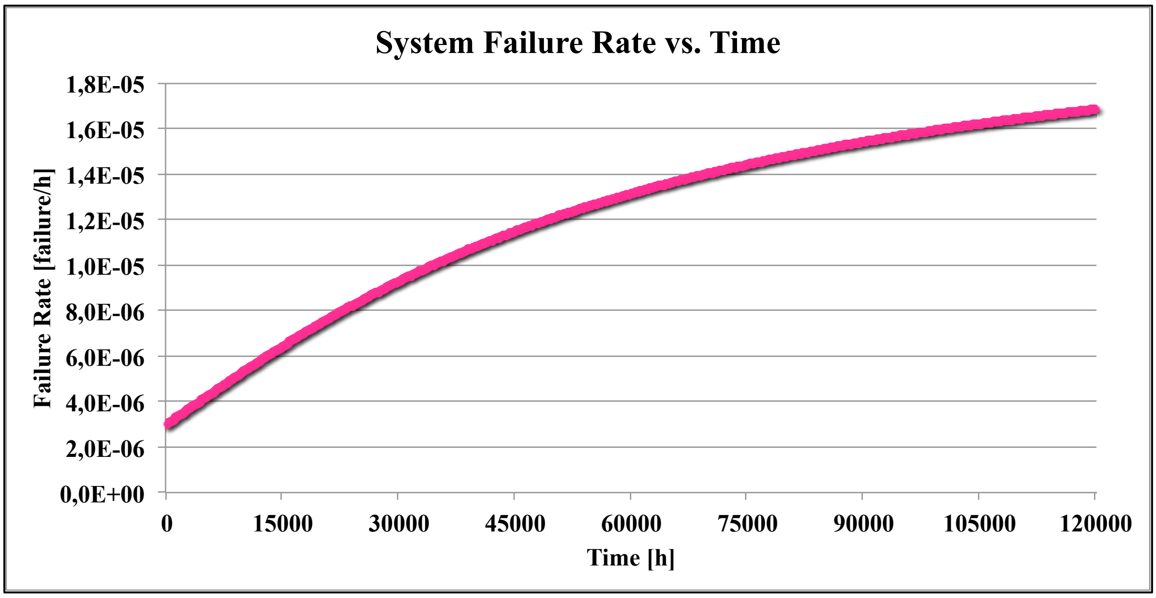


Fig. 10. System Failure Rate vs. Time

3.3.3. Reliability Importance Assessment

Figure 11 shows the system reliability considering a single component upgrade at once: each curve in the chart corresponds to the reliability of the system where the selected item was replaced with the improved one (in terms of reliability performance) and they are compared with the standard reliability function of the system (green curve). The enhancement is based on the Improvement Factor *n* described above [49-50].

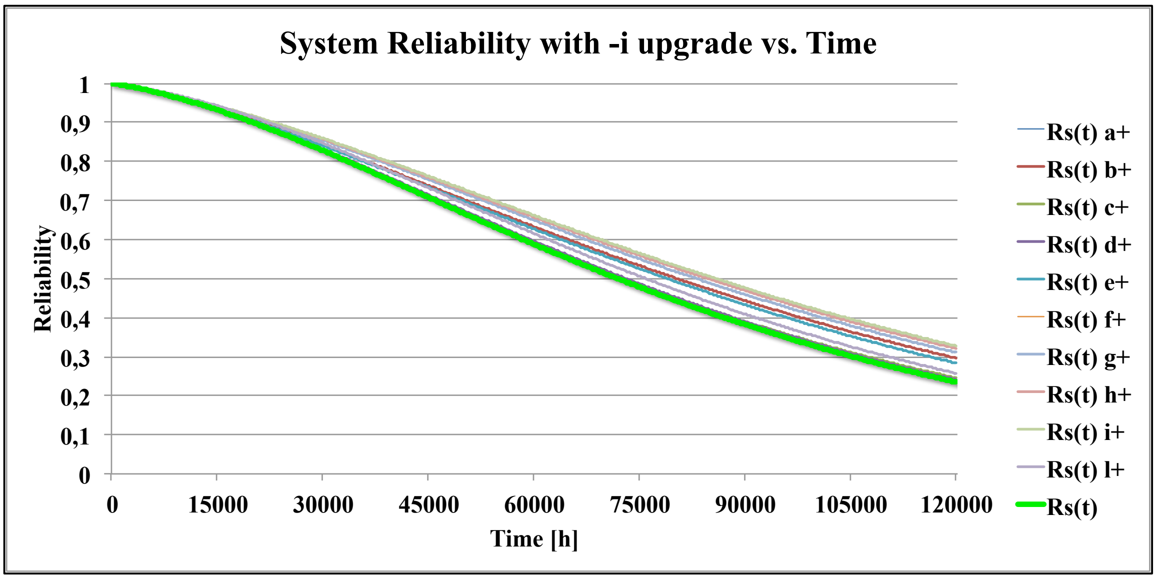


Fig. 11. System Reliability with single item upgrade vs. Time

Figure 12 shows the Credible Improvement Potential measure of each component in function of time so each line represents the percent improvement achievable at any time considering the upgrade of that particular component.

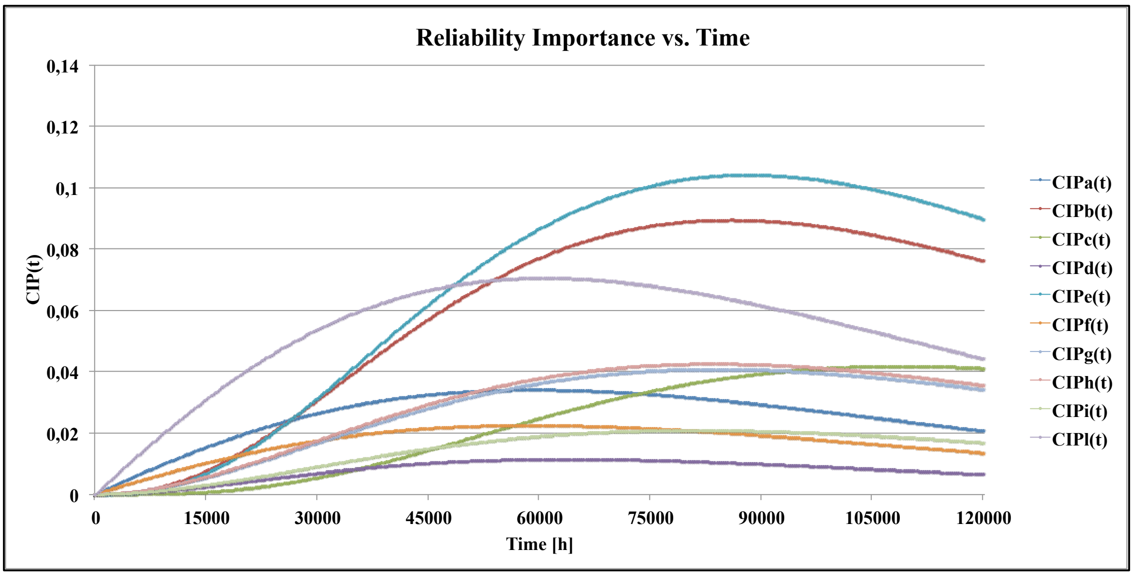


Fig. 12. Component Reliability Importance vs. Time

Static reliability importance is a different plot-type introduced to focus on the behavior of the system at a specific time. It is a sort of reliability snap-shot of the system using histograms and pie charts. Fig. 13, 14, 15 show CIP indices at 24000, 48000 and 72000 hours respectively and it is possible to notice the different contribution of each component to system reliability importance [52].

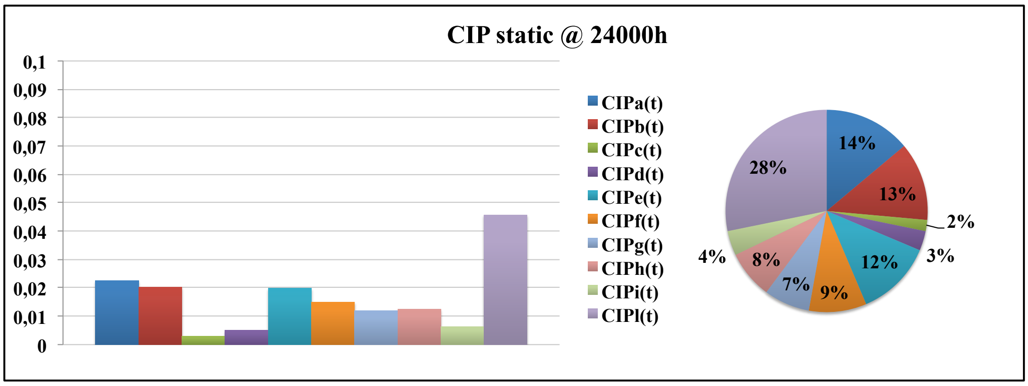


Fig. 13. Static Component Reliability Importance at 24000h

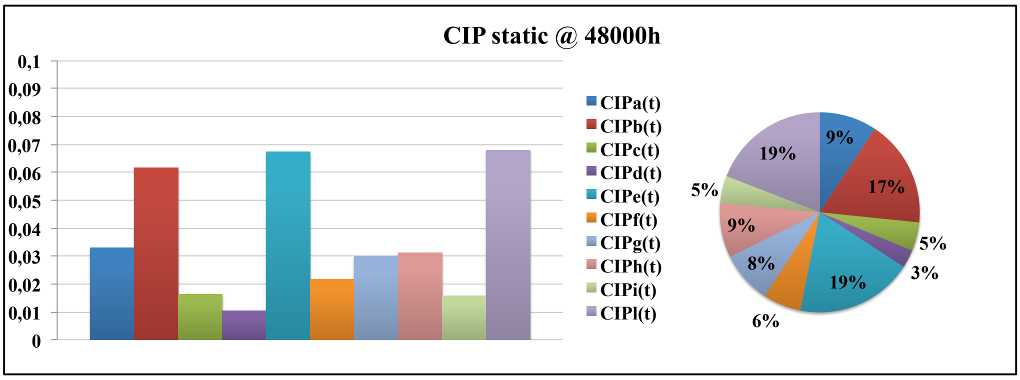


Fig. 14. Static Component Reliability Importance at 48000h

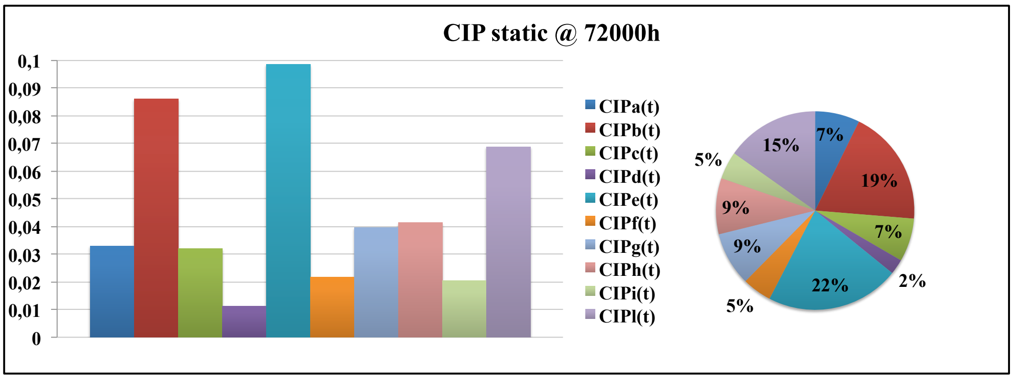


Fig. 15. Static Component Reliability Importance at 72000h

The following figure shows a brand new chart: the MTBF upgrade assessment is displayed in case of improving the reliability performance of each component in the system. The x-axis crosses at MTBF value of the starting system (68077h): this is helpful to underline the estimated improvement that is achievable with each component upgrade [52].

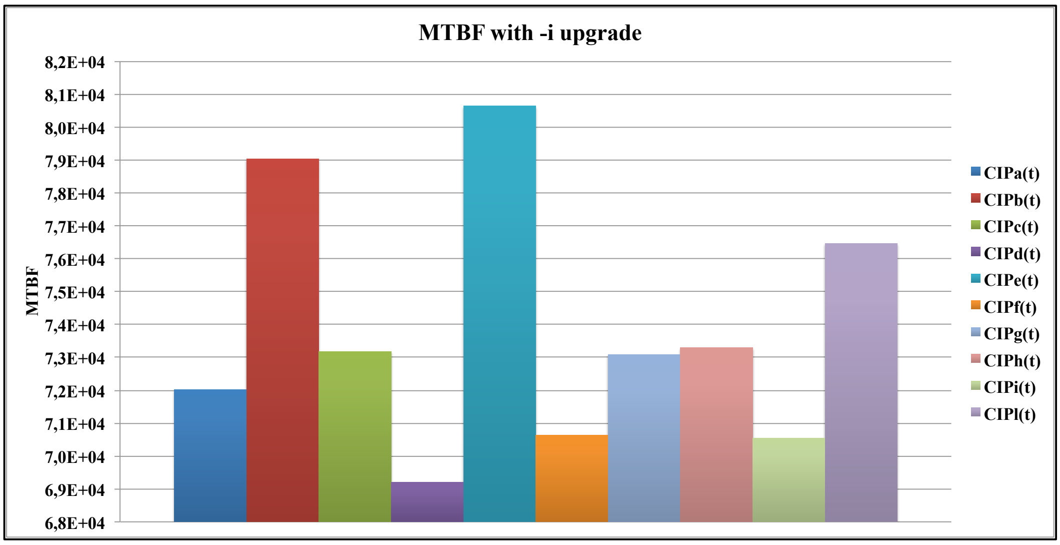


Fig. 16. MTBF improvement comparison with single item upgrade

Time-dependent and static CIP measures offer a clear overview of Reliability Importance trend of each component in the system; in order to discuss the results achieved on the test case a comparison of two components is shown. In this way the achievable benefits of RI assessment are highlighted.

The chosen components are item *E* and item *L*, which differs for the characteristics listed below:

- Components have different failure rates, $\lambda_E = 40{,}42 \bullet 10^{-6}$ failure/h and $\lambda_L = 3{,}11 \bullet 10^{-6}$ failure/h so *E* is one order of magnitude greater than *L*;
- Components are linked to the rest of the system with different connections; *L* is connected in series whereas *E* is involved in 1oo2 hot redundancy that, in turn, is inside the standby branch of a 1oo2 cold standby architecture.

At 24k hours the Reliability Importance of *L* is greater than the other item: *L* offers 28% contribution to system RI and allow a 5% improvement to the whole reliability while *E* has a reduced weight (12%) and a corresponding 2% upgrade on system reliability.

At 48k hours the impact of the two components is the same: both items have 19% significance and ensure approximately 7% gain in terms of system reliability performance.

Finally, at 72k hours the inversion is remarkable: component *L* has 15% weight on overall importance offering 7% reliability improvement while the Reliability Importance of component *E* has 22% contribution to system RI and allow 10% improvement [52].

The trend of component Reliability Importance measures underlines the time-dependency behavior and the significance of RI assessment that is central to focus on the "right" components and take the best design decisions for money and time saving.

3.3.4. Time-dependent CIP Measure Analysis

Component Reliability Importance vs. Time shows a complex and interesting trend that lends to multiple interpretations, for this reason it is extensively examined in this section:

- Maximum point: the "maximum" of a function is the largest value that the function takes at a point either within a given range (named local or relative maximum) or on the whole function domain (named global or absolute maximum).

  In this case the point of interest is the absolute maximum that correspond to the peak of the percent reliability improvement at a precise instant of time [31].

  Figure 17 shows an example that gives readers the right interpretation of absolute maximum in this particular application: for the sake of simplicity the chart represent the CIP measure in function of time for only three items and the highlighted points represent the absolute maximum value of the function.

  CIP index for the highest curve in the chart (pink) reaches the maximum at τ = 100000h and the corresponding CIP value is approximately 0,09. So the highest system percent reliability improvement is 9%.

$$I_i^{CIP}(t)\Big|_{t=\tau} = R_S^+(\tau) - R_s(\tau) \cong 0{,}09 \tag{126}$$

$$R_S^+(\tau) \cong R_s(\tau) + 9\% \tag{127}$$

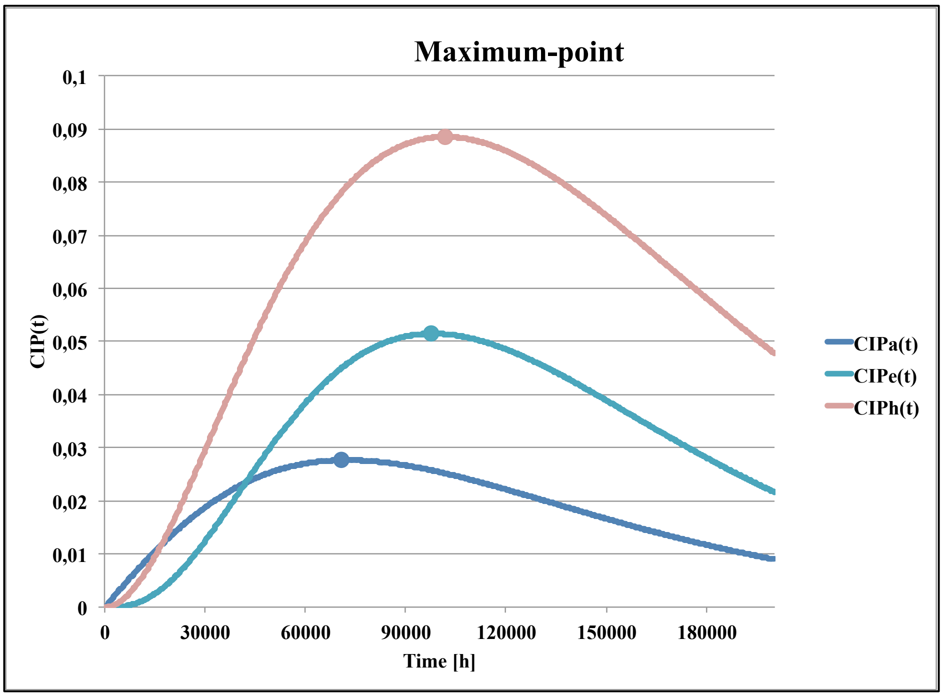


Fig. 17. CIP curves showing maximum-points

- Area subtended from the curve: the *CIP_i(t)* function, in graphic terms, corresponds to the gap between the standard reliability function and the improved one as a consequence of the upgrade of the component reliability performance (Fig. 18). The improvement in terms of reliability is the area under CIP curves, as shown in Fig. 19.

  For the purpose of clarification, the area between the two curves doesn't have to be confused with *MTTF* - Mean Time Between Failures, which is the predicted elapsed time before item showing the first failure .

$$\int_0^t I_i^{CIP}(\tau)d\tau = \int_0^t \Delta R_s(\tau)d\tau \tag{128}$$

$$MTTF = \int_0^\infty R_i(t)dt = \int_0^\infty e^{-\lambda t}dt \tag{129}$$

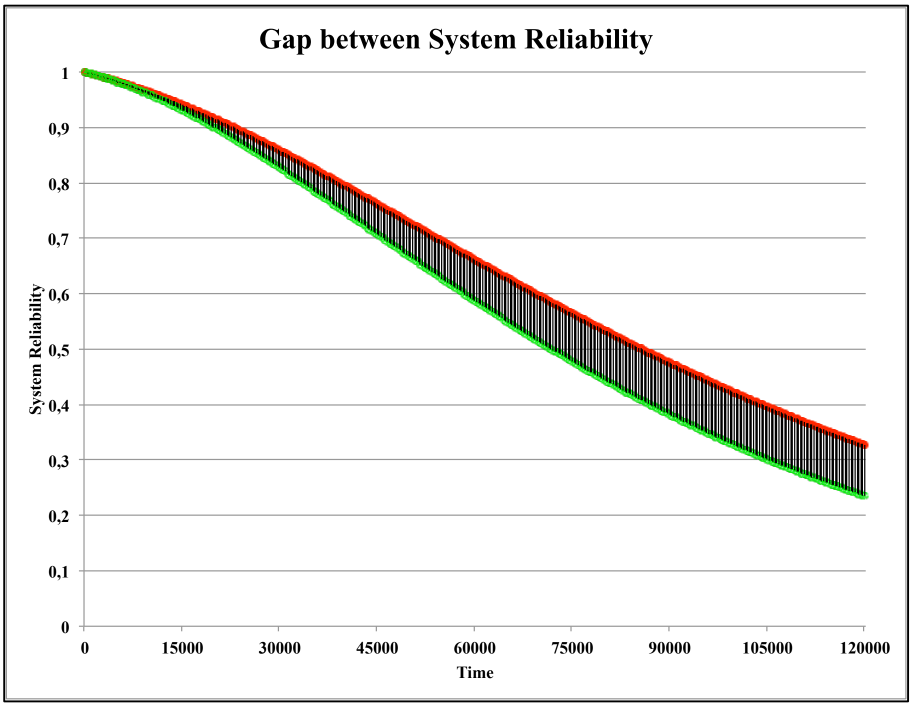


Fig. 18. System Reliability vs. Time

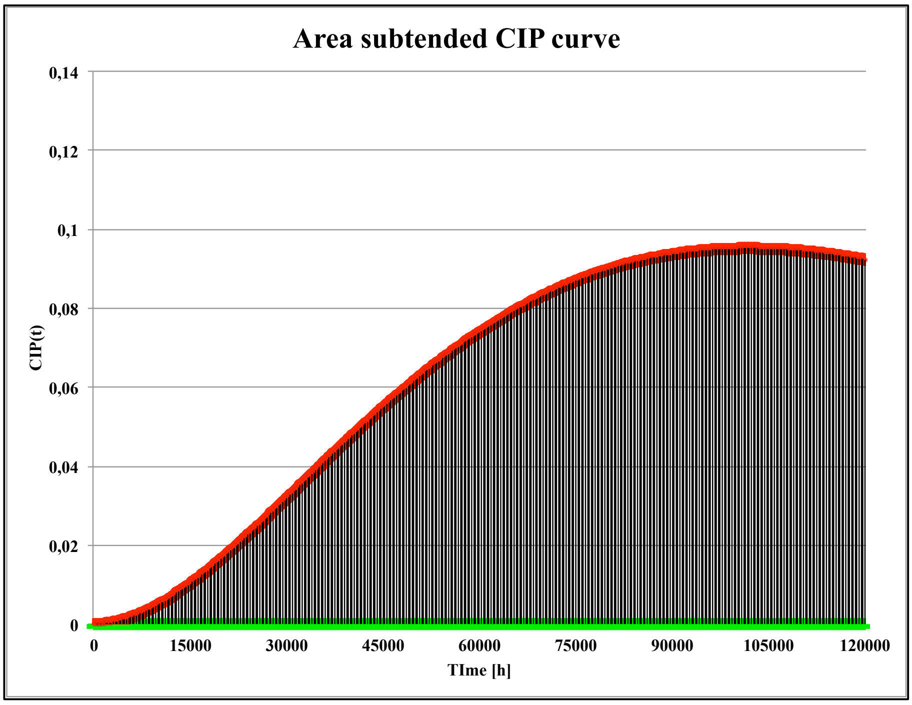


Fig. 19. CIP vs. Time with subtended area

- Crossroad point: two or more curves can create some crossroads where the components have the same reliability importance at that time (since they have the same CIP value). Usually crossroad points are produced by the intersection of the rising edge of a CIP curve with the falling edge of another one, as shown in Fig. 20: the first (blue) is represents by the growth of the CIP value and this trend means that the reliability importance of the component is rising. On the contrary the latter (violet) is associated with a reduction of the CIP trend and the corresponding decrease of RI value.

This distinction between rising and falling edges is useful to allow the designer to take the best decision for his application, taking into account the period of interest: if the focus of the achievable improvement is in the period of time before the crossroad, the designer should improve the first item; otherwise the second item can offer a higher long-term reliability performance [52].

- Inflection point: an “inflection” (or flex) is a point at which a curve changes its curvature from concave (concave downward) to convex (concave upward), or vice versa [31].

  A tangent, i.e. the straight line that touches the curve at a single point, is the best solution to underline the two different trends: the curve is concave when it is above its own tangent; in this case the curve shows a sort of parabolic trend and its growth is more than linear so the CIP value quickly increases. The curve is convex when it is below its own tangent; in this case the curve shows a sort of root square trend and its growth is less than linear so the CIP value slowly increases.

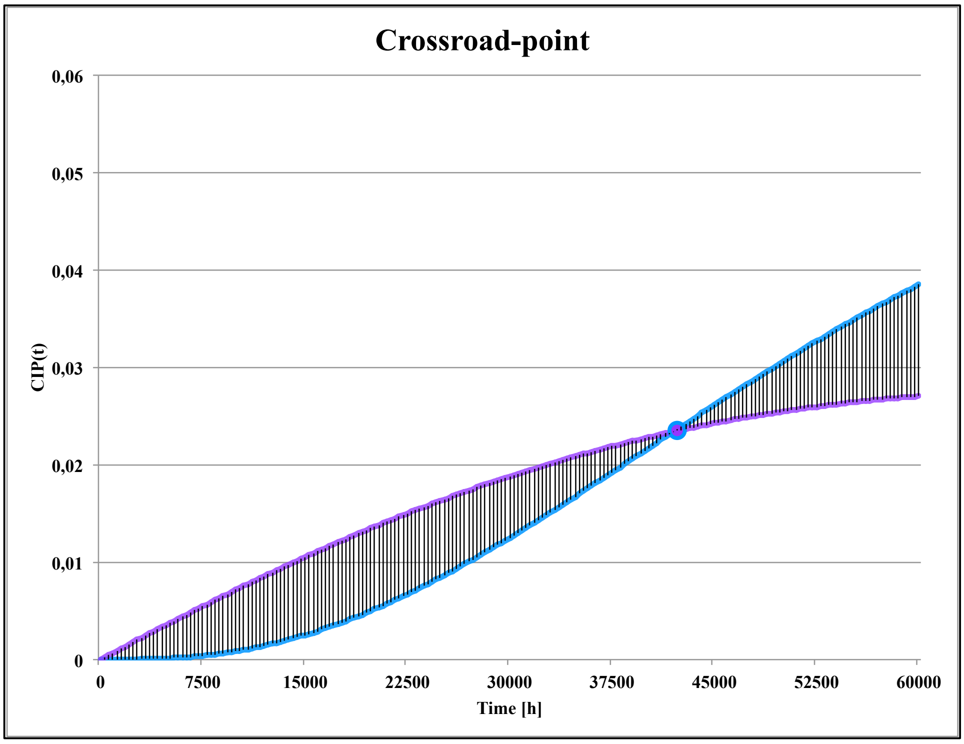


Fig. 20. CIP curve showing crossroad-point

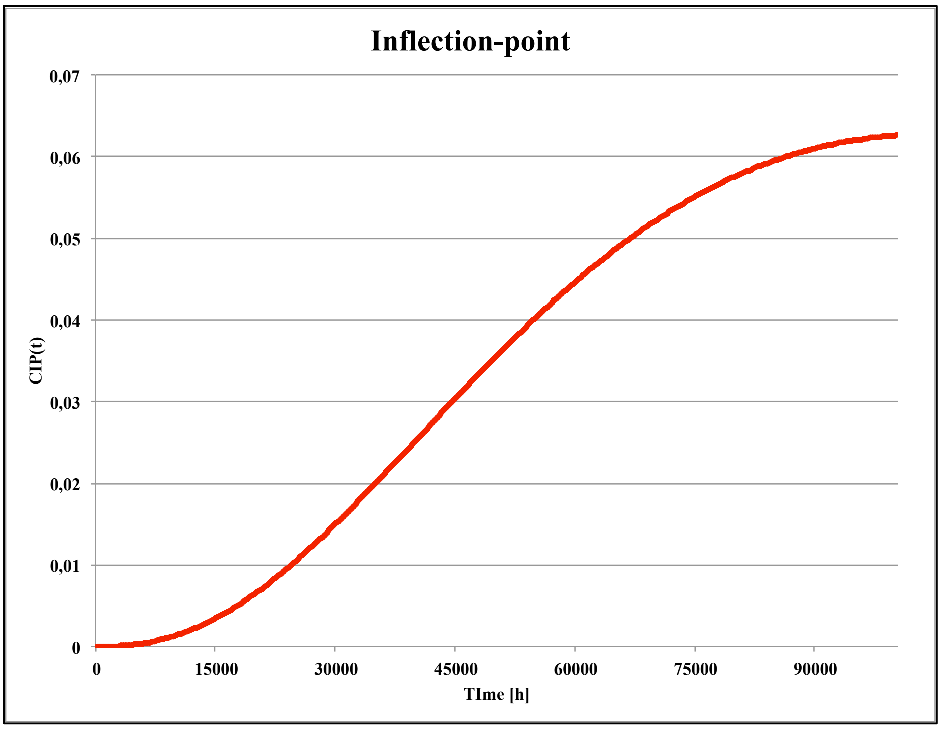


Fig. 21. CIP curve showing inflection-point

- Same-ordinate points: many CIP values are achievable in two different time instants and in these cases component reliability importance is the same. It's worth to notice that CIP represent the achievable percent improvement so the importance measure is referred to different absolute values and, as a consequence, the magnitude of the improvement is different.

  Fig. 22 shows a CIP curve assuming the value 0,06 in both τ₁=36000h and τ₂=91000h.

  The importance is the same but the percentage value is referred to different reliability values: this feature is crtical to balance the benefits of the improvement and the cost of the component upgrade.

$$I_i^{CIP}(t)\Big|_{t=\tau} = R_S^+(\tau) - R_s(\tau) \cong 0{,}06 \tag{130}$$

$$R_S^+(\tau_1) \cong R_s(\tau_1) + 0{,}06 \tag{131}$$

$$R_s(\tau_1) = 0{,}71 \quad R_S^+(\tau_1) = 0{,}77 \tag{132}$$

$$R_S^+(\tau_2) \cong R_s(\tau_2) + 0{,}06 \tag{133}$$

$$R_s(\tau_2) = 0{,}27 \quad R_S^+(\tau_2) = 0{,}33 \tag{134}$$

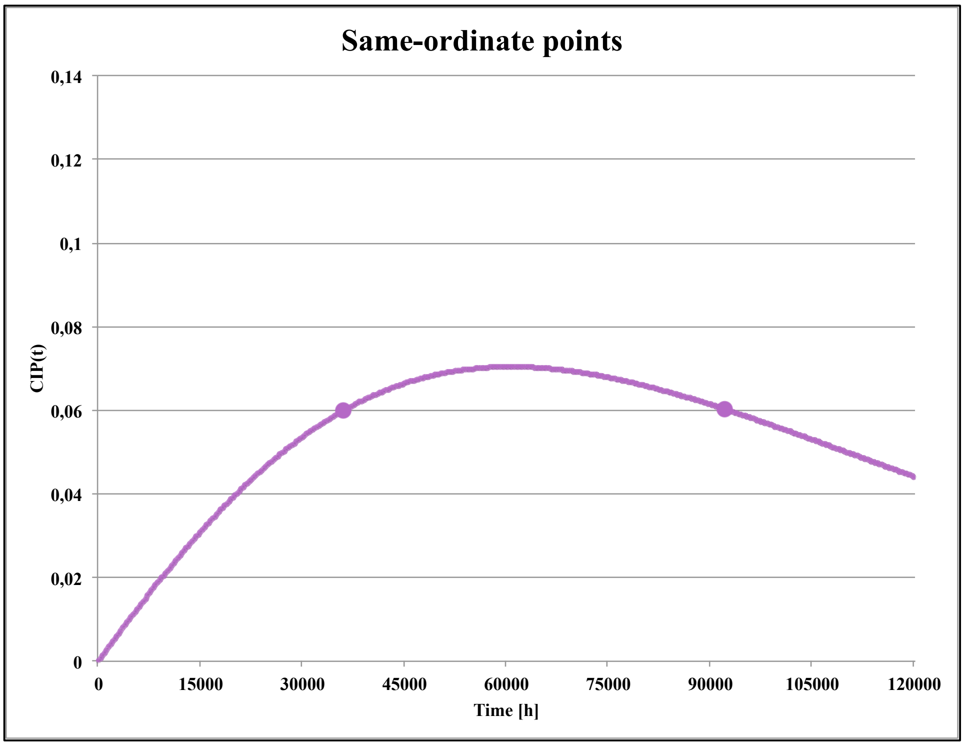


Fig. 22. CIP curve showing same-ordinate points

The CIP method described above was fully-integrated in the developed tool *RBDesigner*®: some test cases to validate the proposed method and show the potentiality of the tool are described in Chapter 6.

Chapter 4

Condition-based Maintenance & Markov Modelling

4.1 Availability and Maintainability Improvement

The interest for maintainability, availability and the techniques to evaluate it had an exponential growth in the last few years, in particular for systems involved in mission-critical environments: these applications demand both high performance and high availability.

Maintainability is directly associated to the concept of availability assessment since it concerns failure and recovery aspects of a system.

The best product design offers very high reliability performance but, however, all the products deteriorate over time: for this reason maintenance is fundamental to keep the system running during the useful life [53].

This thesis is focused on availability improvement and obviously it takes into account maintainability roles in order to achieve this kind of target: the goal is to develop a procedure for availability improvement that engineers may used during the early stages of product design.

The simplest maintenance technique is based on breakdowns and it is a run-to-failure procedure without any kind of plan. Advanced techniques are time-based preventive maintenance: in this case the maintenance operation is planned and performed at periodic intervals without considering the health status of the device. Anyway this method is quite expensive and, since the cost-saving in manufacturing is critical, more efficient maintenance approaches were developed such as condition-based maintenance (CBM).

CBM is a maintenance program referred to the information collected through condition monitoring. It consists of three steps: data acquisition to collect information, data processing to handle information and decision-making following maintenance policies.

In other words, condition-based maintenance is performed after one or more indicators show that the equipment under analysis is going to fail or that equipment performance is deteriorating. One of the best method to carry on availability analysis is the condition based maintenance modelling using Markov analysis [53-54].

In general there are two approaches to evaluate maintainability and, as a consequence, availability of the system: measurement-based and model-based.

Measurement-based evaluation is often quite expensive and it requires building a real system (or a prototype) to take measurements in order to statistically analyse the collected data.

On the other hand, model-based evaluation is less expensive and quite easy to perform even if some solving problems may arise to develop models for large and complex systems:

however in this study the model-based approach is considered and it is described in details in the following paragraphs.

4.2 Model-based Evaluation

As said before, model-based evaluation is the cost-effective solution as it allows system evaluation without having to build and measure a system.

It can be assessed through discrete-event simulation, analytic models or hybrid models (combining simulation and analytic procedures).

A discrete-event simulation is a program whose execution simulates the dynamic behaviour of the system and evaluates the required measures.

The main benefit of this procedure is the ability to characterize in detail the system behaviour through the model; at the same time the main drawback is the long execution time in particular when the solutions require tight confidence bounds.

An analytic model, instead, consists of a set of equations describing the system behaviour and the evaluation measures are obtained by solving these equations. In simple applications closed-form solutions are achievable otherwise numerical solutions are necessary.

In general, analytic models tend to be easier to develop and faster to solve than a simulation model. The main drawback of this procedure is the set of assumptions that are often necessary to use analytic models [54-55]

This study is focused on model-based maintainability and availability evaluation using analytic techniques with particular attention to Markov models.

4.3 Analytical Modelling

Analytical modelling techniques are usually divided in two categories, state space and non-state space models; the choice of the best-fitting technique is essential to generate a trustworthy model representing the system behaviour.

4.3.1. Non-state Space Models

These models can be solved without generating the underlying state space and they can be used to assess system availability, reliability and mean time to failure. Non-state space models require two assumptions: statistical independency of failures and independent repair units for components.

The techniques used to achieve system reliability and availability measures are Reliability Block Diagrams (RBDs) and Fault Trees (FTs).

4.3.2. State Space Models

These models are developed in complex systems with failure/repair dependencies and shared repair facilities; in these applications non-state space models such as reliability block diagrams and fault trees cannot be easily used.

A state-space representation is basically a mathematical model of the system under analysis based on a set of input, output and state variables that are related by first-order differential equations.

To abstract from the number of inputs, outputs and states, these variables are expressed as vectors; in case the dynamical system is linear, time-invariant, and finite-dimensional, the differential and algebraic equations may be written in matrix form [56].

In this case the state space representation of a system replaces an n-th order differential equation with a single first order matrix differential equation. The state space representation of a system is given by two equations, named respectively state equation and output equation:

$$\dot{x}(t) = Ax(t) + Bu(t) \qquad y(t) = Cx(t) + Du(t) \tag{135}$$

Where, for an *n*-th order system with *r* inputs and *m* outputs:

- x is the state vector (*nx1*), a function of time;
- A is the state matrix (*nxn*), a constant;
- B is the input matrix (*nxr*), a constant;
- u is the input vector (*rx1*), a function of time;
- C is the output matrix (*mxn*), a constant;
- D is the direct transition matrix (*mxr*), a constant;
- y is the output vector (*mx1*), a function of time.

For systems with a single input and single output, r=1 and m=1.

One of the best state space techniques is Markovian modelling and it is described in the following section.

4.4 Markov Modelling

The Markov property is the memory-less property of stochastic processes where the conditional probability distribution of future states depends only on the present state without taking into account the sequence of events that preceded it: the state of the system at future time t_{n+1} is function of the system state at the current time t_n and does not depend on the path that led the system to be in the present state (states at time instants t_1, … , t_{n-1}).

A Markov random field is a set of random variables having a Markov property and it extends this property to variables defined for an interconnected network of items.

In probability theory and statistics a stochastic process that satisfies Markov property is known as a Markov process. So a Markov process is a stochastic memory-less model and it can be used to model the changes of states of a random system according to a transition rule that only depends on the current state [54-56].

There are two basic components common to all the Markov models, a set of states and a set of transitions between the states: the system can be in only one state at time and from time to time it makes a transition from one state to another.
The states usually represent system configurations or operational status of the system components and they can represent instances where the system is operational/failed, undergoing recover/repair, operating in a degraded mode, etc.
Usually the states of any Markov model are divided in two sets: one set contains the states representing situations where the system is working properly (sometimes in degraded modality) and the other set containing states where the system is degraded too much that the system must be considered failed [56].
The standard set of possible states is the following:

- Operating: component is working properly;
- Standby: component is ready for operation on demand (cold, warm or hot standby);
- In-service inspection: component is under periodic inspection during operation (on-line);
- Maintenance: component is under maintenance or off-line inspections;
- Failure: component is not working due to a failure;
- Post-failure repair: component is under repair or is going to be replaced after failure.

The events associated to each state are mutually exclusive so the system cannot be in more than one state at time; furthermore they are collectively exclusive since the system always must be in at least one of the states.
The transitions follow rates that govern the length of time between transitions from one state to another: these rates may be constant or time dependent and they are usually related to failure and repair rates of components in the system.
The system availability varies together with the mission progress and this change is reflected in the probabilities of being in each state in the Markov model: these probabilities change over time too so the solution of the model is based on finding a procedure to determine the probability of the individual states at a particular point in time.
The change in the probability for a given state is the difference between the probability coming into the state from all other states and the amount of probability going out of the state to other states in the model: this behaviour is described with a system of simultaneous differential equations, one differential equation for each state and the solution is a vector of state probabilities at a specified time [57-58].

4.5 Maintainability and Availability Analysis using Markov Models

Markov modelling offers both advantages and disadvantages for maintainability and availability assessment; the main advantages of using Markov models are the following:

- Dynamic system behaviour: Markov models have great flexibility in expressing dynamic system behaviour and they are suitable for a wider range of systems than combinatorial models (e.g. Reliability Block Diagrams, Fault Tress, etc.); anyway they are limited by the Markov property and other assumptions on the distributions involved and for this reason they cannot model systems containing non-exponential component lifetimes.
- Complex repair: Markov models take into account repairs of individual components or groups, variable number of repair persons assigned to repair activities, sequential repair and partial repair procedures (components working in degraded conditions).
- Standby spares: Markov models include hot, warm and cold standby units; these different classes of spare components differ each other for the degrading process they endure during the inactive period and for the response time they offer to activate.
- Sequence dependent behaviour: Markov model takes into account the sequence in which events occur e.g. functional dependency (the failure of one component may cause other component to fail) and sequence enforcement (some events are not allowed to occur before certain others).
- Imperfect fault coverage: Markov model is used to deal with dynamic reconfiguration processes that may not be completely successful; in these cases the failure is said to be imperfectly covered.

These features are deeply analysed in paragraph 7.
On the other hand, the main disadvantages of Markov modelling are the following:

- Large number of states: realistic models can require large number of states and solving these huge models may challenge the computational resources of most computers.
- Difficulties in model construction and validation: in very complex models the analyst may face problems in specifying correctly states and transitions; furthermore it is difficult to built the model and verify its accuracy.
- Markov property and failure distribution assumptions: these assumptions may be invalid for the system under analysis.
- Techniques feasible for small systems: models of greatest complexity require huge execution time to solve and dedicated solution techniques that are currently feasible only for small systems.
- Physical or logical organization of the system: the structure of Markov model (based on states and transitions) may not have a great correspondence with system physical or logical organization.

For the reasons described above, sometimes Markov models are not the best choice and some other techniques are more advisable; this happens when the system can be satisfactorily modelled with simpler combinatorial methods (e.g. model may be smaller and more easily constructed or model solution may be computationally more efficient) or in case the system requires a very large number of states and its behaviour is too complex to be expressed in a Markov/semi-Markov model, so simulation approach is preferred [57-59].

4.6 Markov Models

For any generic system a Markov model is made of all the possible states, transitions and corresponding rate parameters of that system: in case of reliability analysis, the transitions usually consist of failures and repairs.

There are several Markov models depending on the application and typology of sequential states.

The simplest Markov model is the Markov chain: it models the state of a system using a random variable that changes through time. In other words, the sequence of transitions that moves the system from one state to another corresponds to a chain composed by different rings. This concept can be expressed with the following proportion:

$$\textit{Chain} : \textit{Sequence of transitions} = \textit{Ring} : \textit{State} \tag{136}$$

It is worth noting that Markov chains assume discrete states and a discrete time parameter while Markov processes require that states are continuous. In case the time step (e.g. Δt) is small enough the Markov process can be approximated by a Markov chain.

According to Markov property, the distribution of this variable depends only on the distribution of the present state [56-59].

Therefore a Markov chain is a stochastic process *X(t)* based on sequence of random variables (states) *x1*, *x2*, *x3*, ... ruled by the Markov property:

$$\Pr\left(X_{n+1} = x \mid X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n\right) = \Pr\left(X_{n+1} = x \mid X_n = x_n\right) \tag{137}$$

Where $\Pr\left(X_n = x_n\right)$ is the probability to find X_n in state x_n and $\Pr\left(X_{n+1} = x \mid X_n = x_n\right)$ is the probability of going from one state at time *n* to another state at time *n+1*. This feature shows the behaviour of Markov chains that are independent from the initial distribution $\Pr\left(X_1 = x_1\right)$.

As said before, Markov models describe the lifetime behaviour of systems in a state-time space. A number of distinct states of the system are identified and correspond to certain combination of component states and/or (environmental) conditions.

Transitions between these states are governed by events such as: component failure or repair, conditional events (such as loss of main power supply) and even trigger events like overload situations or short power peaks that invoke component failures. These transitions bring in the "time element" into the model [55].

There are two types of models to be considered depending on how the transitions are permitted to occur in the time domain: if transitions mandatory occur at fixed time interval (a transition at each interval), the model is called Discrete Time Markov Chain (DTMC).

Whereas transitions are permitted to occur at any time interval, the model is called Continuous Time Markov Chain (CTMC) and this class is split in two subcategories, homogeneous and non-homogeneous:

- Homogeneous Continuous Time Markov Chain

  For this type of model the Markov property holds at all times so the future behaviour of the model depends only on the present state of the system and not on the previous transitions and states.

  A second property of these models is that the state holding times are exponentially distributed and do not depend on previous or future transitions.

  Therefore if system is in state *i* at time *T*, the probability that the next transition leading out of state *i* will occur at or before a time *t* is given by:

  $$P = 1 - e^{-\lambda_i t} \tag{138}$$

  Otherwise the probability that the next transition will occur at or after time *T+t* is given by:

  $$1 - P = e^{-\lambda_i t} \tag{139}$$

  In both cases λ_i is the sum of all rates of the transitions going out from state *i*.

  As a consequence of this property, interstate transition rates are all constant and the time to the next transition is not influenced by the time already spent in the state.

  So for time-homogeneous Markov chains (or stationary Markov chains) holds the following equation for all *n* states and the probability of the transition is independent of *n*:

  $$\Pr\left(X_{n+1} = x \mid X_n = y\right) = \Pr\left(X_n = x \mid X_{n-1} = y\right) \tag{140}$$

- Non-homogeneous Continuous Time Markov Chain

  This type of model is obtained when a homogeneous CTMC is generalized to permit transition rates to be function of time. The Markov property still holds at all times so the transitions to the following states depend only on the present state; this is true also for the state holding times.

Another class of models is the Semi-Markov type where the Markov property does not hold at all times, rather it holds only when transitions occur. The behaviour of the Semi-Markov model is the same of CTMCs so the transition to the next state does not depend on the previous path leading to the present state; however it differs from the others since the holding times follow general distributions (non-exponential) and they may also depend on the next state [57-59].

Anyway in most common Markov models the transition rates are constant and the transition times are exponentially distributed; in case the model considers non-constant failure rates, the equations governing the states are the same but the failure rates are function of time *t* (corresponding to the age of the component).

This kind of models is very useful to describe wear-our or infant mortality but they are more complex to develop: each state is no longer determined by the health status of a finite number of components but it depends on the time passed since the last component replacement (or restoration). In this models units fail following their instantaneous failure rate and then return to the full-operative state at various times so this "reference" state is constituted by components in the same operational condition but with different ages. Therefore the operative life of each component must be analysed separately without taking into account interactions or repairs.

One of the best solutions to solve Markov models with time-dependent failure rates is the Monte Carlo simulation that is based on a large number of simulated systems to be analysed. Anyway this subject is not deepened in this study since the focus is on analytical modelling.

4.6.1. Single Item

The simplest Markov model is made of one component with two associated states, healthy and failed and the transition rate is time-independent.

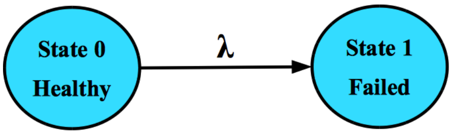


Fig. 1. Markov model with two states

Some useful symbols and corresponding definitions are described below:

- State 0: normal running, item is working properly;
- State 1: fault condition, item is not able to complete the mission;
- λ: rate parameter of the transition from State 0 to State 1;
- *$P_j(t)$*: probability of the system being in State *j* at time *t*; in case the device is known to be working at initial time t=0, the initial probabilities of the two states are $P_0(0) = 1$ and $P_1(0) = 0$.

The probability to find the system in State 0 decreases following the constant rate λ; so if the system is in State 0 at time *t*, the transition probability in the successive time interval *dt* is λdt.

Therefore the probability of transition occurrence during *dt* is given by the product of the probability of being in starting State 0 and the probability of the transition during the interval *dt.* This represents the incremental change dP_0 in probability of State 0 at any given time, so it can be written:

$$dP_0 = -(P_0)\lambda dt \tag{141}$$

Dividing both sides by *dt*, the following differential equation is obtained:

$$\frac{dP_0}{dt} = -\lambda P_0 \qquad (142)$$

Therefore the rate of change of the probability of the source state is reduced to the product between the transition rate and the probability of the source state itself.

Since the item should be in State 1 or in State 0, the total probability of both states must equal 1: so when the probability of State 1 increases, the probability of State 0 must decrease following the same rate:

$$\frac{dP_0}{dt} = -\lambda P_0 \qquad \frac{dP_1}{dt} = \lambda P_0 \qquad P_0 + P_1 = 1 \qquad (143)$$

Using the initial state conditions $P_0(0) = 1$ and $P_1(0) = 0$, the solutions are the following:

$$P_0(t) = e^{-\lambda t} \qquad P_1(t) = 1 - e^{-\lambda t} \qquad (144)$$

In this simple example the transition times are exponentially distributed; this way the total probability of all the states is conserved and the probability moves from one state to another.

In more complex systems the Markov model usually includes a state with all components fully operating and a set of intermediate states representing partially failed conditions.

Transition paths may include both failure and repair events and, in general, each transition path between two states of the system reduces the probability of the starting state and increases the probability of the successive one; the corresponding rate is the product of the transition parameter λ and the current probability of the source state $P_i(\tau)$.

Also in complex systems the rate of change of the probability of each state (dP/dt) corresponds to the probability of flowing into and out of that state; furthermore, the total probability to flow into a given state is achieved by the sum of the transition rates into that state, each multiplied by the probability of the state at the beginning of that transition. On the other hand, the probability to flow out of the given state is the sum of all transitions out of the state multiplied by the probability of that given state.

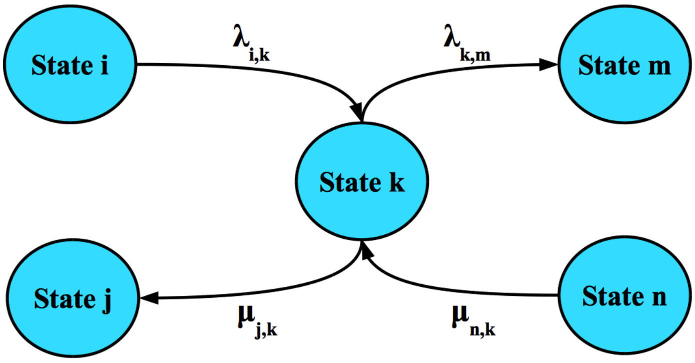


Fig. 2 – Markov model of a generic state k

Fig. 2 shows a generic single state k with some of surrounding states and corresponding transitions. The state equation for state k is shown below: the rate of the change of the probability dP_K/dt is equal to the sum of the probability flows associated to each transition:

$$\frac{dP_k}{dt} = \sum_i \lambda_{i,k} P_i + \sum_n \mu_{n,k} P_n - \left(\sum_j \mu_{k,j} + \sum_n \lambda_{k,n} \right) P_k \tag{145}$$

The system behaviour is determined by these equations, one for each state in the model.
In the simple model under analysis both failure and repair transitions are ruled by constant rates: anyway, in real-word applications, repairs may not be characterized by constant rates.
The following paragraphs show Markov models of more complex architectures.

4.6.2. 1oo2 Redundant Architecture

The effects of repairs are deepened below, taking into account a redundant system with two components in 1oo2 architecture.
There are four possible states for this system:

- State 0: Unit 1 healthy, Unit 2 healthy;
- State 1: Unit 1 unhealthy, Unit 2 healthy;
- State 2: Unit 1 healthy, Unit 2 unhealthy;
- State 3: Unit 1 unhealthy, Unit 2 unhealthy.

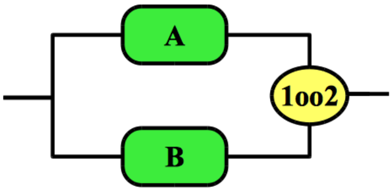


Fig. 3 – RBD of 1oo2 redundant system

The maintenance strategy changes depending on the application; in this case Unit 1 is monitored continuously and, in case of failure, it is repaired within 100 hours. In case of failure of Unit 1 or during maintenance period, Unit 2 ensures system continuity of operation.
The status of Unit 2 is not monitored continuously, the only requirement is an inspection every 1000 hours.
In case both units fail, the system is no longer able to carry on its required function and both units are immediately repaired, than the system returns to the full-up state.
The corresponding Markov model is shown in Fig. 4.

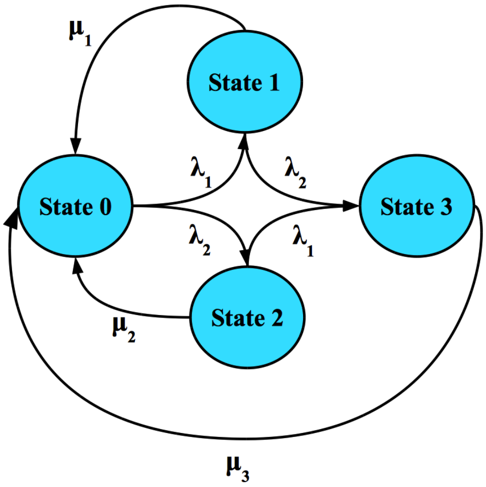


Fig. 4 – Markov model of 1oo2 redundant system

The repair transition in case of both failed units from State 3 to State 0 is usually ruled by a constant, and very large, rate; on the other hand, the repair transitions in case of single failure (i.e. State 1 and State 2) cannot be governed by a constant rate since the repair time of Unit 1 depends on the time of failure and the repair time of Unit 2 has a discrete distribution (repair actions on Unit 2 occur at discrete intervals).

Therefore, in practice, the distribution of failures is usually considered uniform and the expected time between failure and repair is actually half of the inspection interval; following these hypotheses, assuming T the periodic inspection/repair interval, the repair rate is fixed at 2/T (instead of 1/T).

In order to assess the average system failure rate it is necessary to consider only the steady-state condition, so the flow entering and exiting each state is the same. The corresponding equations for this system are the following:

$$\begin{aligned}
&(\lambda_1+\lambda_2)P_0=\mu_1P_1+\mu_2P_2+\mu_3P_3\\
&\lambda_1P_0=(\lambda_2+\mu_1)P_1\\
&\lambda_2P_0=(\lambda_1+\mu_2)P_2\\
&\lambda_2P_1+\lambda_1P_2=\mu_3P_3
\end{aligned} \tag{146}$$

Notice that the probability of State 3 is zero since its repair rate is infinite. The set of equations, taking into account also the conservation property, can be solved as follows:

$$\begin{aligned}
&(\lambda_1+\lambda_2)P_0=(\lambda_2+\mu_1)P_1+(\lambda_1+\mu_2)P_2\\
&P_1=\frac{\lambda_1}{\lambda_2+\mu_1}P_0\\
&P_2=\frac{\lambda_2}{\lambda_1+\mu_2}P_0\\
&P_0+P_1+P_2=0
\end{aligned} \tag{147}$$

$$P_0 = \frac{1}{1 + \frac{\lambda_1}{\lambda_2 + \mu_1} + \frac{\lambda_2}{\lambda_1 + \mu_2}} \tag{148}$$

$$\lambda_{SYS} = \lambda_2 P_1 + \lambda_1 P_2 = \frac{\lambda_2 \frac{\lambda_1}{\lambda_2 + \mu_1} + \lambda_1 \frac{\lambda_2}{\lambda_1 + \mu_2}}{1 + \frac{\lambda_1}{\lambda_2 + \mu_1} + \frac{\lambda_2}{\lambda_1 + \mu_2}} \tag{149}$$

The procedure described above is suitable for small systems, in case of larger models is more appropriate the matrix form (using the state equations for each state of the model):

$$\begin{aligned} &(\lambda_1 + \lambda_2) P_0 - (\lambda_2 + \mu_1) P_1 - (\lambda_1 + \mu_2) P_2 = 0 \\ &\lambda_1 P_0 - (\lambda_2 + \mu_1) P_1 = 0 \\ &\lambda_2 P_0 - (\lambda_1 + \mu_2) P_2 = 0 \end{aligned} \tag{150}$$

Since the first equation is the sum of the other two, it is preferred to replace it with the conservation property; therefore the resulting set of equations can be written in matrix form:

$$C = \begin{bmatrix} 1 & 1 & 1 \\ \lambda_1 & -(\lambda_2 + \mu_1) & 0 \\ \lambda_2 & 0 & -(\lambda_1 + \mu_2) \end{bmatrix} \quad P = \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix} \quad U = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{151}$$

The solution is:

$$P = C^{-1} U \tag{152}$$

Usually the system shutdown rate is a linear combination of the state probabilities and in this example it is:

$$\lambda_2 P_1 + \lambda_1 P_2 \tag{153}$$

So, defining the vector L, the average system failure rate is the following:

$$L = \begin{bmatrix} 0 & \lambda_2 & \lambda_1 \end{bmatrix} \tag{154}$$

$$\lambda_{SYS} = L C^{-1} U \tag{155}$$

4.6.2.1. System Failure Rate Transient Analysis

The transient (time-dependent) analysis is very interesting on either closed-loop or open-loop models, corresponding to models with or without repair.

In some applications it is necessary to know the worst-case instantaneous system failure rate as a function of time since the average system failure rate is not enough.

Taking into account the simple 1oo2 system described above, the time-dependent equations of failure and repair transitions are the following:

$$\begin{aligned}
\frac{dP_0}{dt} &= -\lambda\left(\lambda_1+\lambda_2\right)P_0+\left(\lambda_2+\mu_1\right)P_1+\left(\lambda_1+\mu_2\right)P_2 \\
\frac{dP_1}{dt} &= \lambda_1 P_0-\left(\lambda_2+\mu_1\right)P_1 \\
\frac{dP_2}{dt} &= \lambda_2 P_0-\left(\lambda_1+\mu_2\right)P_2
\end{aligned} \tag{156}$$

Using matrix notation:

$$\frac{dP(t)}{dt} = MP(t) \tag{157}$$

$$M=\begin{bmatrix} -\left(\lambda_1+\lambda_2\right) & \left(\lambda_2+\mu_1\right) & \left(\lambda_1+\mu_2\right) \\ \lambda_1 & -\left(\lambda_2+\mu_1\right) & 0 \\ \lambda_2 & 0 & -\left(\lambda_1+\mu_2\right) \end{bmatrix} \quad P(t)=\begin{bmatrix} P_0(t) \\ P_1(t) \\ P_2(t) \end{bmatrix} \tag{158}$$

The time-dependent solution of this system of equations is:

$$P(t)=e^{Mt}P(0)=\left[\sum_{j=0}^{\infty}\frac{(Mt)^j}{j!}\right]P(0) \tag{159}$$

$$\lambda_{SYS}=LP(t) \qquad L=\begin{bmatrix} 0 & \lambda_2 & \lambda_1 \end{bmatrix} \tag{160}$$

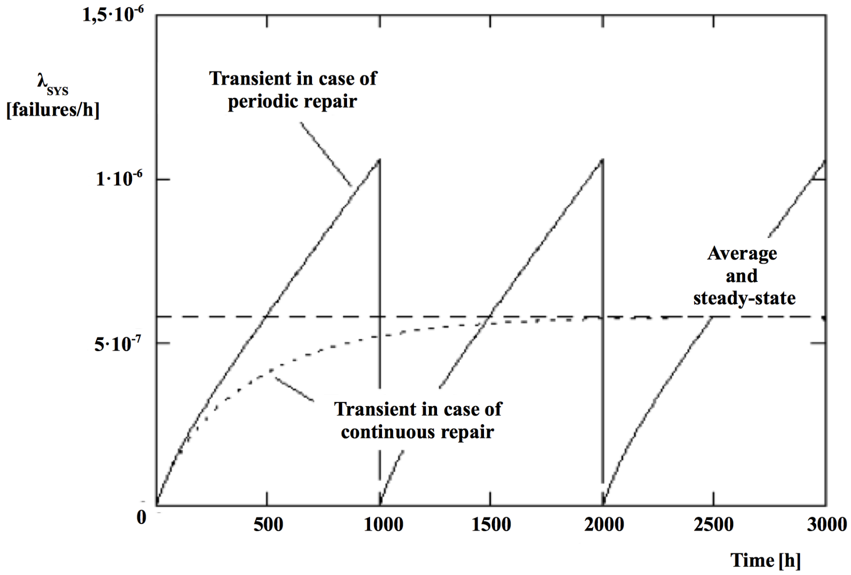


Fig. 5. Instantaneous failure rate in case of both continuous and periodic repair

Supposing continuous repair procedures (exponentially distributed), the instantaneous failure rate shown in Figure 5 goes close to the asymptotic steady-state rate in a short period of time (in this case around 1000h) but this is irrelevant comparing to the expected useful life period (e.g. millions of hours). This is the reason why sometimes the steady-state analysis is often preferred to the transient one to achieve the average system failure rate.

The trend of the system failure rate is completely different considering a discrete distribution for repair activities; the repair rate is zero during each inspection or repair interval (e.g. 1000h), and infinite for an instant at the end of each interval.

This way the full-up state is restored at the end of each interval and every interval begins in exactly the same full-up conditions; the result is a saw-tooth function repeating every cycle (in this case 1000h). Anyway the main value results the same of the steady-state analysis.

In critical applications long latency is rarely acceptable and for this reason these critical systems are frequently monitored and inspected: as a result, they are repaired and the system is restored in short time intervals. Therefore the main interest is to achieve the long-term average reliability over the whole period of maintenance cycles that corresponds to the steady-state solution of the closed loop model where all repair transitions are taken into account. The periodic variations within each maintenance cycle are not considered.

In conclusion, the transient analysis represents the average instantaneous failure rate over a single period of cyclic maintenance while the steady-state analysis is a good approximation of the long-term average failure rate over multiple maintenance intervals [60-63].

4.6.3. Complex Systems

The procedure described in the previous paragraphs is used for simple systems with just dual redundancy and repair rates that are much greater than failure rates.

For more complex systems or in case of repair/failure rates of the same order of magnitude (e.g. in case of long latency maintenance), in order to use Markov models some other factors must be taken into account [57].

4.6.3.1. Sporadic Periodic Repair For First-Order States

As said before, a component periodic repair of T hours can be modelled with a repair transition with constant rate 2/T since the failure distribution is approximately uniform during each interval; with this hypothesis, the duration between failure and repair is T/2 hours.
In critical applications it is often preferred a more conservative solution (repair rate of 1/T) but for this analysis the “standard” 2/T is accepted and the two following conditions are taken into account:

- The repair rate of 2/T works only for model states that are a single failure away from the fully operative condition and this is acceptable in dual redundant systems where the failure of both units leads to complete system failure;
- In case the inspection interval is large compared to MTBF, the average time between failure and repair approaches T hours and it makes sense to set the repair rate to 1/T; when inspection interval is the same order of magnitude of MTBF, instead, the corresponding repair rate is contained in the interval 1/T and 2/T and, for first-order states, the rate of inspection/repair transitions is the following:

$$\mu = \frac{1}{T}\left(\frac{1}{\frac{1}{1-e^{-(\Sigma\lambda)T}} - \frac{1}{(\Sigma\lambda)T}}\right) \tag{161}$$

Where $\Sigma\lambda$ is the sum of all the failure rates in the full-up state.

4.6.3.2. State Aggregation For High-Order Models

The number of states in the model may increase exponentially together with the increase of number of components. In a system of N components, in case each state is associated to a combination of failures, the model will contain 2^N states. In the following picture are shown the diagrams for N = 2, 3, and 4.

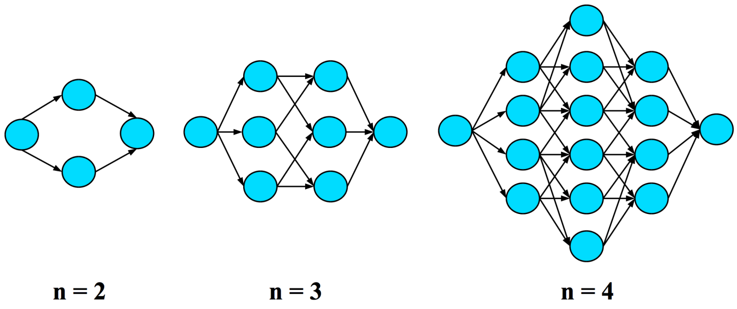


Fig. 6. State models in function of the number of components making up the system

Supposing a system made of N identical components, each failure transition has the same rate λ and the conditions with the same number of failures can be collected together in the same state; this aggregation reduces the model to N+1 states, as shown in Figure 7.

However state aggregation may be a concrete solution for any system with identical configurations that can be gather in a single state.

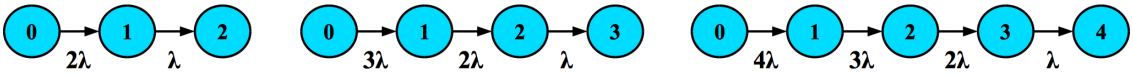


Fig. 7. State models after state aggregation

State aggregation is also useful in larger systems with a huge number of possible high-order failure combinations: in case only the single and double failure combinations have significant probabilities, all the other combinations are negligible and may be gathered in the total system failure state [57].

4.6.3.3. Periodic Repairs For High-Order Models

The solutions described in the previous sections show the procedure to model periodic inspection/repair of states that are one failure away from the fully-operative condition.

This solution is not applicable in higher-order models however it is possible to estimate with great accuracy the repair transition rates for T-hour inspection/repair intervals.

The repair rate has a minimum value (1/T) and it may assume values greater than 2/T in case of higher-order states: this is caused by the rate of going into higher-order states that is not constant, but increases over time in correspondence to the probabilities of the feeding states increase.

This is the reason why going into a higher-order state is not uniformly distributed over the interval and it is weighted toward the end of any given inspection interval.

Therefore the periodic repair rate of high-order models is usually difficult to assess except for some types of models such as systems consisting of N identical units with the same failure rate λ (Figure 8). The system fails in case of failure of all the N components and, in case of failure, the component is repaired immediately in order to bring the system back to the fully-operative state.

The maintenance policy is based only on periodic inspections every T hours and, in case a component is found failed, it is repaired at that time.

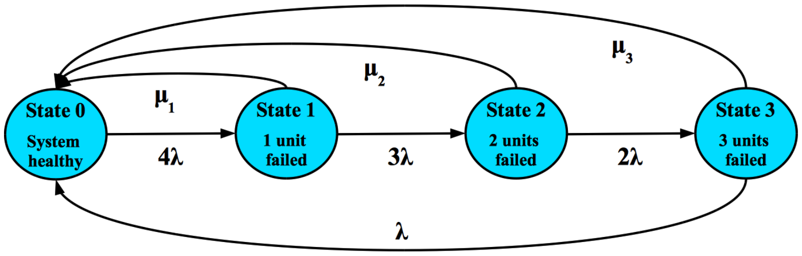


Fig. 8. Markov model for generic system with 4 identical components

For the sake of simplicity the transition rate of repair transitions is considered constant for each of the three partially-failed stated but actually the rates differ each other because the mean times to enter the corresponding states are all different.

Usually for systems with N identical components the periodic inspection/repair of a state that is k failures away from the fully-repaired condition can be represented by a transition with the following constant rates:

- $\mu = \frac{(k+1)}{T}$ in case the inspection interval is small compared to MTBF;
- $\mu = \frac{1}{T}$ in case the inspection interval is large compared to MTBF.

A general formula to cover both these conditions is:

$$\mu = \frac{1 + ke^{-\lambda T}}{T} \tag{162}$$

This is true for the simple system described before real applications force to face with more complex architectures with different components and different maintenance policies for each unit; in these systems it is necessary to carry out an exact analysis of repair transitions to achieve suitable repair rates instead of using the approximate solution.

4.6.3.4. Model Reduction And Simplification

As said before, one of the greatest problems using Markov technique in reliability analysis is the number of states required to model systems with large number of components; a system of N components has 2^N possible states supposing each component has two states (working and failed). This issue may be mitigated with reduction techniques that can be applied to reduce the number of states and maintain the accuracy of the model.

The simplest and smartest practice is to implement a design that should eliminate common mode failures and isolate redundant functional paths.

Furthermore, in case the model shows multiple levels of failure contribution, it is possible to implement a truncation procedure. The failure rates of most real components are many orders of magnitude smaller than 1 so their combination may be negligible and, as a consequence, system should be simplified (e.g. in case the combination of three or more failures are negligible, it is possible to consider in the model only single and double failure combinations and consider the transitions after double failure states directly to the complete system failure state).

It's not rare in complex systems to find different elements that have the same impact in case of failure so the result of these elements can be combined without any loss of accuracy in the model [64].

Another reduction technique to achieve the probability of failure of the target event considers the division of this top-level event into n sub-events that are modeled separately. The probability of the sub-events is then combined to assess the probability of the top-event.

4.7 Hidden Markov Models

Hidden Markov Models (HMM) are statistical models for system modeling through a finite number of states.

The states of the system are hidden to the observer but using HMMs it is possible to determine which state the system is currently in.

A Hidden Markov Model consists of several states with corresponding initial probability value, a transition probability matrix that indicates the likelihood of moving from one state to another and an output probability distribution.

Hidden Markov Models are divided in two main categories, discrete and continuous, differing for the input they are able to accept and the way the input is processed. Discrete HMMs are based on a limited number of observations while continuous models are able to handle input (e.g. real numbers) that is not part of a predefined list.

The Hidden Markov Model of a complex system can be represented with a single HMM or several HMMs: in the first case, each state of the model corresponds to a different health state of the system while using multi Markov Models, each health state is described by a dedicated model. The last solution is preferable in case little data is available since new health states may be identified and new HMMs can be added to the system when other data becomes available.

Hidden Markov models are widely used for failure coverage and their application in Condition-Based Maintenance is described in the following paragraph.

4.7.1. Failure Coverage With Markov Modelling

In redundant systems the fault coverage capability of the system must be taken into account and the possible states of Markov models may be reduced at three states: good, failed covered and failed uncovered. The fail states are mutually exclusive since a component cannot fail both covered and uncovered.

The failure coverage deals with detection, isolation and reconfiguration and can be associated to fault diagnosis that is one of the cornerstone of Condition-Based Maintenance [57].

Assuming the system under analysis is a Markov process with unobserved states, fault diagnostics can be achieved using Hidden Markov Models (HMMs).

Each state in the HMM represents a different health-state of the system and, with the use of HMMs, it is possible to assess different tasks that are described in the following list:

- Detection and identification of anomalies in the process (anomaly detection)

  This procedure refers to discovering data that the HMMs have not been trained to recognize. In case the model is defined by an observation vector, it is possible to calculate the probability that the observation vector belongs to the HMM; it responds to the presented observations and a log-likelihood estimate is generated depending on the recognition of the measurements. The log-likelihood values are used to detect potential anomalies.

  For example, if is taken into account the HMM corresponding to a state of good health of the system, the only training data available are those corresponding to the system in good health. The observations are taken from the measurements that were not used to train the HMM; if the log-likelihood of the data received is closer to 0 it means that there is high probability that data belongs to the HMM corresponding to good health.

  In case something in the system goes wrong, the likelihood drops until it goes below a predefined value: at this point it is not possible to know the state of the system but only that it is no longer in the good health condition.

  For this kind of applications the expert knowledge of the system together with the detection assessment is necessary to find errors and train the HMM and detect these errors in future measurements.

- Determine the current health of the system (current state detection)

  With this procedure it is possible to determine the current state of the system among the several HMMs the system has been trained to.

  For example, if two HMMs are taken into account, one “good” and one “bad”, they are competing each other since they are presented with the same observation vector.

  The log-likelihood is used to determine the winner that will be the HMM producing the value closest to 0; as a consequence, the state system will be (probably) in that state (there is no 100% probability since the winner represents the most likely state the system is in).

Similarly for anomaly detection applications, also for the determination of the current system state an expert knowledge can help to define more error states and train HMMs to improve the classification of the states of the system.

Furthermore huge variations in the training data values may affect the capacity of HMMs to properly recognize the data: in these applications to detect errors more accurately it is necessary to reduce the amount of training data and consider only the measurements with similar values.

- Predict the future health of the system (future state prediction)

  The aim of this procedure is to estimate when the state of the system is going to change; there aren't procedures to know the exact moment of the change but it is possible to estimate when the system is expected to move from one state to another.

  In order to estimate state transition points it is necessary to analyze the training data to find the state transition points: they are determined following the same procedure based on log-likelihood trajectories described above for current state prediction.

  When the state transition points of the training data have been established, the joint and conditional distributions can be modelled to perform future state predictions.

  This prediction of the state transitions becomes more accurate together with the growth of information on the component under analysis [57].

4.8 Applications of Markov Models

In this section different system behaviours are presented using Markov modelling: repair, standby spares, sequence dependency, transient/intermittent faults and imperfect fault coverage.

4.8.1. Repair

Markov modelling is very well suited to model repair activities; repair involves the restoration of the functionality lost due to a failure in order to bring the system back to normal running. For this reason modelling repair usually adds cycles to a Markov models.

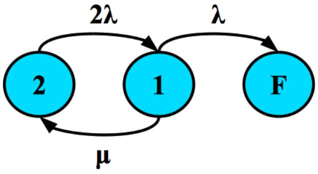


Fig. 9. Markov model for system with two components

The Markov chain in figure represents a system with two active redundant components. In the starting state (2) both components are properly functioning and, when a failure occurs in one of the two components (rate 2λ), the system goes in a degraded state (1). In case of occurrence of a second failure (rate λ) the system goes to a failure state (F).

The repair procedure is represented by the transition from state (1) to state (2) and it follows the repair rate μ; in case of perfect repair, full system functionalities are restored and system returns to the initial state (2).

As a result, the addition of the repair activity adds a cycle between states (2) and (1).

This basic procedure for repair modelling can be generalized to represent a wide range of repair resources and procedures e.g. dimension of the maintenance team, sequence of component repairs, number of required failures before initiating the repair activities, etc.

4.8.2. Standby Spares

Markov models are suited to reproduce the three types of standby spares: hot, warm and cold. Standby spares are considered to be similar or identical components to the main unit for both functionality and performance; they are supposed to take over when a failure arises in order to give continuity to the process. In this condition the system keeps working in a degraded mode until the maintenance service repairs the failure and restore redundancy.

In a hot standby architecture, the second unit is continuously powered and ready-to-use during normal running in order to take over as quickly as possible in case of main equipment failure. As a consequence, the standby component is assumed to have the same failure rate of the main unit; furthermore all the redundant items can fail at any time and the failure rate at which a failure occurs is the sum of the failure rates for all active components (e.g. in a redundant system with main unit and two standby spares with failure rate λ each, the transition rate is 3λ).

Note that the presence of standby units requires the presence of detection and reconfiguration processes: as a consequence failure coverage probabilities must be taken into account [55-58].

In a cold standby configuration the support unit is powered down until the main unit experiences a failure: at that time the standby component is switched in order to take over the process load. For this reason the cold spare is usually considered not vulnerable to failure before the activation, whereas it can fail at any time during fully active state.

If the redundant system is composed by three units, the starting state is defined by one active functioning device and two cold standby spares available.

A transition resulting from a failure obviously implies that the main unit has failed (since the standby spares are unpowered and cannot fail during quiescent period); when the destination state of the transition is reached, one of the unpowered spare units will have been activated in order to take over the process.

The new state will indicate that one fewer spare component is available than before the failure: the transition rate is λ rather than 3λ of hot standby architecture since only the main unit can fail and for this reason all transitions representing failures in the model have a transition rate of λ, regardless the number of spare units remain available [57].

The last architecture is the warm standby configuration where spare components are powered during normal operation of the main unit although they are not fully operative:

these devices can fail during quiescent period because they are subjected to the same environmental stress of the main unit. As a consequence, the warm spares are vulnerable to failure with a lower failure rate than the fully active functioning and these failure rates contribute to the rate of the transition representing the failure of the main component.

So the transition rate representing the failure of a component is the sum of the failure rates of all active components including warm spares.

The resulting state after the transition (failure of the main unit) is defined by one fully active device and one remaining warm spare.

4.8.3. Sequence Dependency

Since Markov models are based on sequences of states connected by transitions, they are a perfect means to represent sequence dependent behaviour.

Some processes are based on the sequence in which events occur and some examples are described below:

- Processes with events that cannot take place until other events have occurred; e.g. cold spare activation after main unit failure;
- Processes where certain events cause other events to occur, or preclude other events from occurring (functional dependency);
- Processes where future system behaviour can change depending on the order in which some other events occur; this situation was typically modelled with fault trees using Priority-AND (AND gate in which the output event occurs only if all input events occur in a specific sequence).

4.8.4. Transient and Intermittent Faults

Markov models are a natural instrument to model processes involving transient and/or intermittent faults.

Transient faults are faults that arise and can cause malfunctioning for a finite time; then they disappear or turn in a friendly state (no longer cause malfunctioning).

Intermittent faults, instead, are faults that randomly oscillate between active and quiescent states [56].

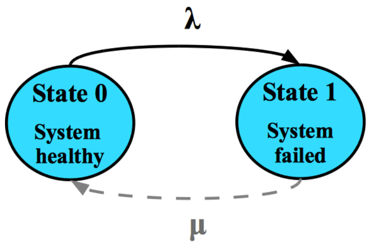


Fig. 10. Markov model for system with transient faults

4.8.5. Imperfect Fault Coverage

When a failure arises, the process of failure detection and reconfiguration sometimes can fail itself: in this case the fault is considered imperfectly covered and the probability that detection/reconfiguration process is successful is called coverage probability.

Imperfect fault coverage can be modeled using Markov processes using two outgoing transitions for each imperfectly covered fault that can occur while the system is in a certain operational state:

- The first transition outlines the success of detection/reconfiguration processes and it leads the system to an operative state. It is define by a transition rate achieved as the product of the successful reconfiguration probability and the imperfect coverage occurrence.
- The second transition represents an unsuccessful reconfiguration process that leads the system to a faulty condition (caused by the uncovered failure). The transition rate is the product of the unsuccessful reconfiguration probability and the imperfect coverage occurrence [57].

4.9 Case Studies

In this section are described some examples of Markov models involving systems differing for application and number of components: this is necessary to prove if Markov modelling is a suitable procedure to assess availability improvement during the early stages of product design.

4.9.1. 1oo2 Redundancy With Dedicated External Fault Monitoring

The system is shown in Figure 11 and is composed by three devices: main unit, standby unit and external monitoring unit.

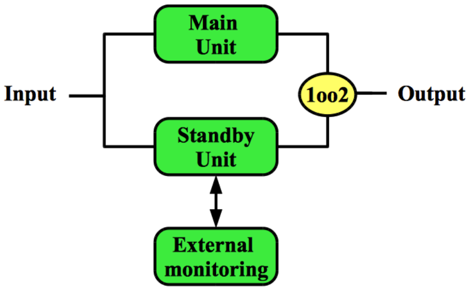


Fig. 11. 1oo2 system with external fault monitoring

The main unit is equipped with continuous on-board fault monitoring and it is also controlled before each mission that lasts 10 hours on average. In case a failure or a partial efficiency is detected during maintenance procedure, the unit is repaired before the following start. If the failure displays during normal running, the on-board monitoring has to detect the problem and the backup device is activated to guarantee the continuity of the process. The failure rate of main device is λ_1=4,9*10^{-5} failure/hour.

On the other hand, the standby system is not equipped with on-board diagnostics but it is continuously monitored by the dedicated external monitoring device so, in case of backup unit failure, it is repaired before the successive missions depart. Since the standby unit can fail latently, it is checked every 50 hours.

Clearly also the external unit can fail and in this circumstance the detection coverage is lost; in order to detect latent failures this device is checked every 500 hours.

If the backup unit is found faulty at one of these checks without any kind of warning from the control device, they are both considered failed and they are both repaired.

The failure rates of standby and external monitoring systems are λ_2=2,2*10^{-5} failure/hours and λ_3=2,6*10^{-5} failure/hours.

Mean time to failures is some order of magnitude greater than the intervals of scheduled inspection/repair and most of the states making up the Markov model are first-order states since they are just one failure far from full-operation so there is no loss of accuracy if these repairs are modeled with continuous transitions following constant rates μ=2/T for the respective intervals.

This hypothesis is quiet conservative for the state defined by both monitor and backup device failures but anyway it is acceptable and, above all, convenient to use.

Failure and repair rates for the three devices are listed in Table I while the Markov model is shown in Figure 12.

Tab. I – Failure and repair rates, MTBFs and MTTRs

Device	λ (failure/h)	μ (repair/h)	MTBF (h)	MTTR (h)
Main Unit	4,9E-05	2,0E-01	20400	10
Standby Unit	2,4E-05	4,0E-02	41700	50
External Monitoring	2,6E-05	4,0E-03	38500	500

State 6 in the model corresponds to the total failure condition where all components are not working and, for the sake of simplicity, the repair rate from this state is considered infinite; this assumption allow to remove State 6 from the system equations.

On the other hand, the system failure rate is the rate of going into that state:

$$\lambda_{SYS} = (P_1 + P_4)\lambda_2 + (P_3 + P_5)\lambda_1 \tag{163}$$

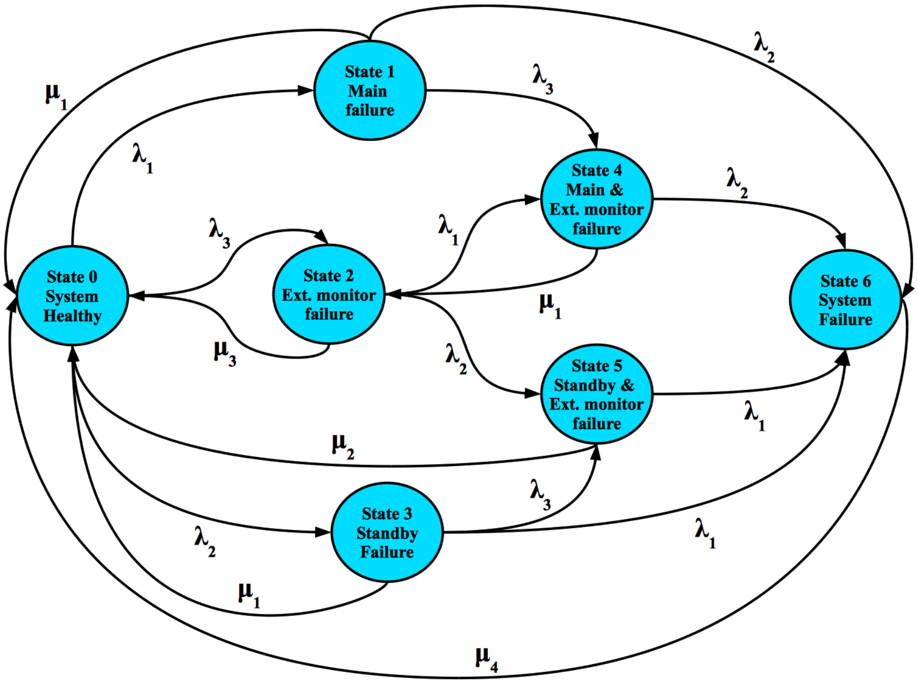


Fig. 12. Markov model of 1oo2 system with external fault monitoring

The equation system of the Markov model developed is shown below and the sixth equation is the conservation equation:

$$\begin{aligned}
&\lambda_1 P_0 - (\lambda_2 + \lambda_3 + \mu_1) P_1 = 0 \\
&\lambda_3 P_0 - (\lambda_1 + \lambda_2 + \mu_3) P_2 + \mu_1 P_4 = 0 \\
&\lambda_2 P_0 - (\lambda_1 + \lambda_3 + \mu_1) P_3 = 0 \\
&\lambda_3 P_1 + \lambda_1 P_2 - (\lambda_2 + \mu_1) P_4 = 0 \\
&\lambda_2 P_2 + \lambda_3 P_3 - (\lambda_1 + \mu_2) P_5 = 0 \\
&P_0 + P_1 + P_2 + P_3 + P_4 + P_5 = 1
\end{aligned} \tag{164}$$

The corresponding matrix notation is the following:

$$\lambda_{SYS} = LC^{-1}U \tag{165}$$

Where:

$$C = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ \lambda_1 & -(\lambda_2+\lambda_3+\mu_1) & 0 & 0 & 0 & 0 \\ \lambda_3 & 0 & -(\lambda_1+\lambda_2+\mu_3) & 0 & \mu_1 & 0 \\ \lambda_2 & 0 & 0 & -(\lambda_1+\lambda_3+\mu_1) & 0 & 0 \\ 0 & \lambda_3 & \lambda_1 & 0 & -(\lambda_2+\mu_1) & 0 \\ 0 & 0 & \lambda_2 & \lambda_3 & 0 & -(\lambda_1+\mu_2) \end{bmatrix}$$

$$U = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$L = \begin{bmatrix} 0 & \lambda_2 & 0 & \lambda_1 & \lambda_2 & \lambda_1 \end{bmatrix}$$

The failure rate of the modelled system is λ_{SYS}=6,3*10^{-9} failure/hours.

The chart in Figure 13 represents a sensitivity analysis of system failure rate in case of modulation of the monitoring inspection interval of the backup system (ranging from 50 to 800 hours). This plot offers a great support to design engineers to optimize the maintenance interval and the corresponding cost.

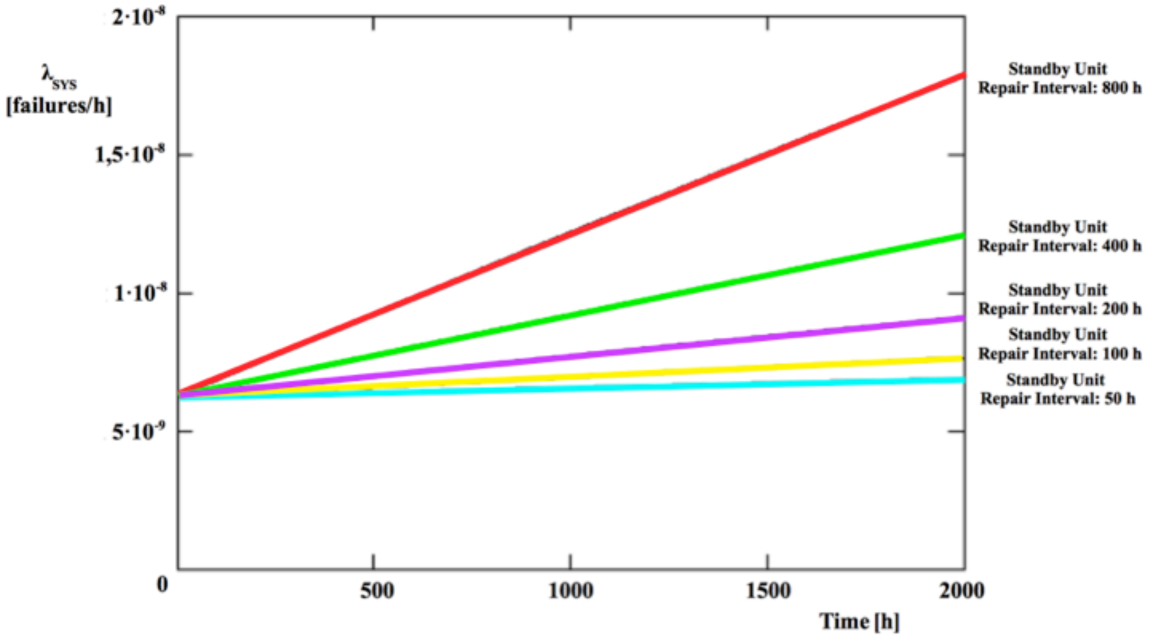


Fig. 13. System failure rate vs. inspect/repair interval backup unit

4.9.2. Redundant Safety System

A system involved in safety applications often have one or more intermediate operating states since it is required to keep running during the repair of failed components and during maintenance intervals.

In these applications maintenance intervals and duration of maintenance become important because the system is designed to guarantee continuous operation and may become vulnerable during maintenance activity, also in redundant architectures.

Markov modelling is performed considering three different system states: all components operating normally, one or more components in failure while the system is still running with reduced capacity and the third state in which one or more components are in failure and the system is no longer operative.

The system under analysis is shown in figure: there are two sections in series (A and B) each with redundant architectures.

In both sections A and B there are two branches, main and standby (warm or hot), with two and three items respectively. A controller is in common between the sections and it activates the standby train if a failure occurs in the main one; this operation is possible only in case the standby device is available, otherwise a system failure occurs.

One of the branches may become inoperable due to a failure or in case of maintenance and also the controller may produce a system failure too since it is a single item.

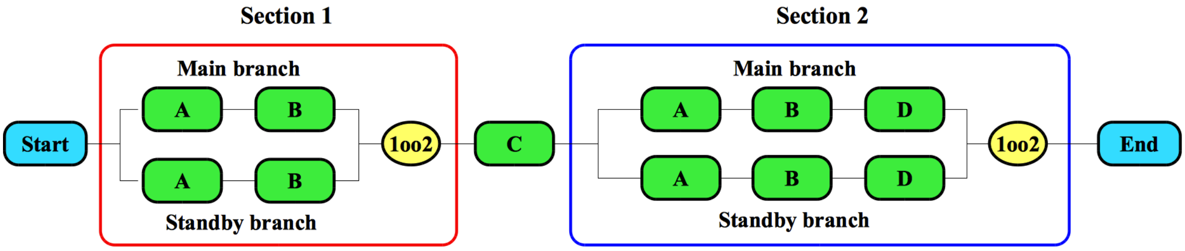


Fig. 14. Reliability Block Diagram of system under analysis

Fig. 15 shows the Markov model of the system. There is one normal operating state (NR), three half operating states in which one or more trains are not working (S1, S2, and S3), and five states of system failure (FC, F1, F2, F3, and F4).

The failure rates and repair/restore rates of the components in the system are listed below:

- Failure rate item A: λ_A=3,3*10^{-6} failure/hours
- Failure rate item B: λ_A=3,1*10^{-6} failure/hours
- Failure rate item C: λ_C=2,2*10^{-5} failure/hours
- Failure rate item D: λ_B=2,9*10^{-5} failure/hours
- Maintenance rate: μ=5,4*10^{-3} repair/hours
- MTTR: variable between 4, 8, 12, 24 and 48 hours

The states considered in the system are described in the following list:

- NR: normal running state with one (i.e. main) branch operating in each section while the other one is in standby (but ready to supplant).
- S1: one branch in Section 1 is not working due to a failure or maintenance operation while the other one is running.
- S2: one branch in Section 2 is not working due to a failure or maintenance operation while the other one is running.
- S3: one branch in Section 1 and one in Section 2 are not working due to a failure or maintenance operation, the remaining components in each section are normally running;
- FC: the controller fails and produces a system failure (absorbing state);
- F1: one branch in Section 1 or 2 is not working and the controller fails with resulting system failure (absorbing state);
- F2: one branch in Section 1 or 1 is not working and another component in the same section fails producing system failure (absorbing state);
- F3: one branch in Section 1 and one in Section 2 are not working and also the controller fails with resulting system failure (absorbing state);
- F4: one branch in Section 1 and one in Section 2 are not working and another component in a running branch fails with consequent system failure (absorbing state).

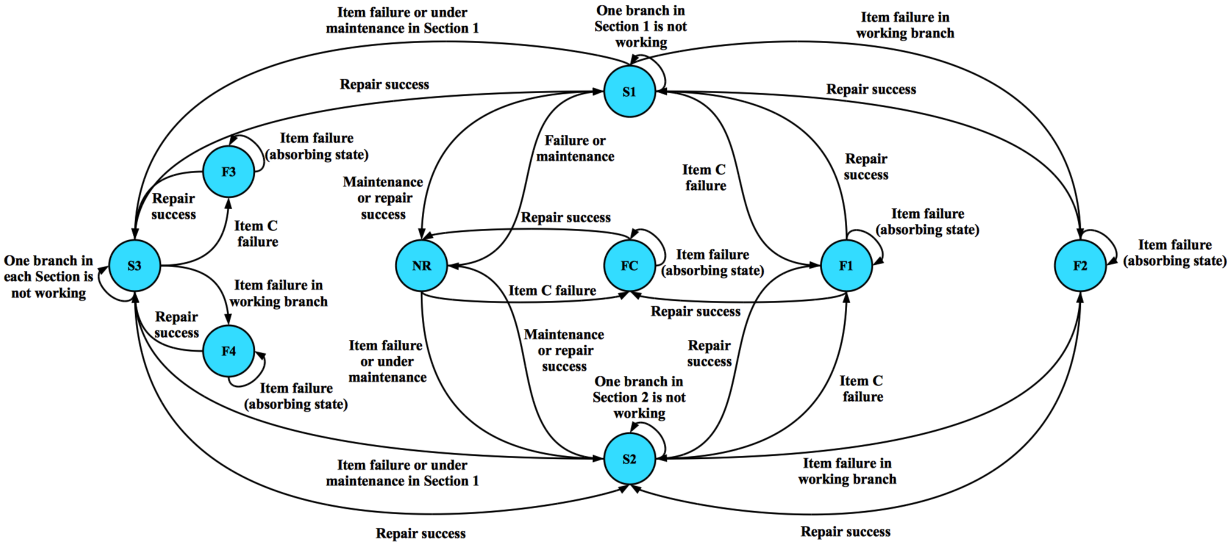

Item failure or under maintenance in Section 1
One branch in Section 1 is not working
Item failure in working branch
Repair success
S1
Repair success
Item failure (absorbing state)
F3
Failure or maintenance
Item C failure
Repair success
Maintenance or repair success
Repair success
Item C failure
Repair success
Item failure (absorbing state)
One branch in each Section is not working
S3
NR
FC
Item failure (absorbing state)
F1
F2
Item failure (absorbing state)
Item failure in working branch
Item C failure
Repair success
Repair success
F4
Maintenance or repair success
Repair success
Item failure or under maintenance
Item failure (absorbing state)
One branch in Section 2 is not working
Item C failure
S2
Item failure or under maintenance in Section 1
Item failure in working branch
Repair success
Repair success

Fig. 15. System Markov model

After the identification of system states and corresponding transitions, the transition probability matrix is developed in order to estimate the failure rate of the system.

The resulting equation is the following:

$$M = \left[I - P \right]^{-1} \tag{166}$$

Where M is the matrix composed by the elements m_{ij} indicating the time spent in the state j supposing the system started in state i, I is the identity matrix and P is the truncated transition probability matrix (without rows and columns containing absorbing states).

$$MTTF = \sum_{j=1}^{n} m_{ij} \qquad R_S = \frac{1}{MTTF} \tag{167}$$

Where MTTF is the Mean Time To Failure, m_{ij} are the elements of matrix M.

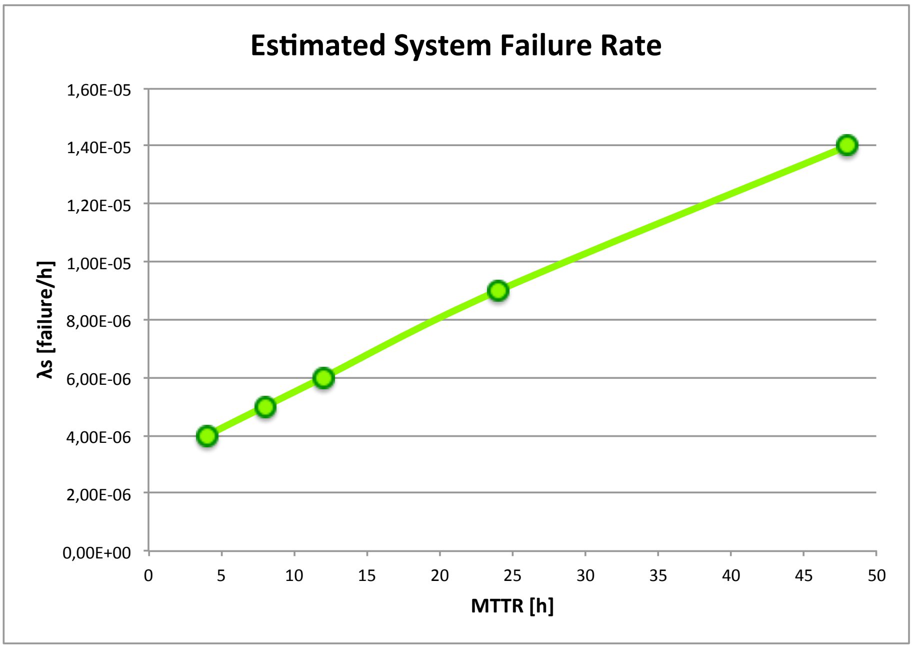


Fig.16. System failure rate vs. MTTR

Markov modelling is a powerful method to assess system maintainability and this procedure can be a valid support for availability assessment but it is usable only in case of low-complex systems: as seen above, the complexity of the system directly translates into the complexity of the corresponding Markov model and a system of ten components may produce a model with hundred states and even more transitions.

Furthermore, this approach requires a manual construction of the model and this practice is not so familiar to design engineers so this implies cumbersome and error-prone modelling. For these reasons Markov processes are not the most suitable practice for the purpose of this study.

Chapter 5

Diagnostics And Condition Monitoring

5.1 Diagnostics And Maintenance Policy

Diagnostics plays a fundamental role in industrial engineering and nowadays is an essential part of performance requirements. Fault diagnosis and condition monitoring are almost mandatory in particular for Oil&Gas applications where products are forced to endure extreme process and environmental conditions.

With the introduction of fault diagnosis design engineers are allowed to improve standard scheduled maintenance methods based on planned actions and severe timetables: thanks to diagnostics both corrective and predictive maintenance procedures can be put onto practice.

These methods are a great step forward comparing with scheduled strategies that can be a waste of time, money and resources if maintenance is made when not effectively needed and, on the other hand, it can miss some impending issues. Corrective and predictive maintenance are real-time diagnostics and optimize efforts in terms of costs and time [27-31].

The main added value of predictive maintenance is to allow convenient scheduling of corrective maintenance and prevent unexpected equipment failures or system emergency shutdowns [65-66].

Corrective maintenance is assessed in case of fault show to repair the failed equipment and restore the system to “normal running” state.

This maintenance practice is performed in three steps: fault identification, isolation and correction.

Predictive maintenance, instead, checks the condition of on-line equipment and decides when maintenance should be performed.

In this way tasks are performed only when warranted so designers achieve cost savings over routine or time-based preventive maintenance. In other words, predictive maintenance allows convenient scheduling of corrective maintenance and prevents unexpected equipment failures or system emergency shutdowns [67-68].

Diagnostic information is used to:

- increase equipment lifetime;
- increase plant safety;
- guarantee continuity of productive process;
- reduce loss of productivity due to system accidental shutdown;
- optimized spare parts handling.

Diagnostic assessment can be fulfilled at different levels depending on the complexity of the system under analysis: if diagnosis is restricted to a single item, usually that device is equipped with dedicate on-board circuit that supplies information about working/failure state to the logic solver. This is a local check involving only that device.
Otherwise system diagnostics is achievable on more complex devices (e.g. lube oil console) using information from many sensors placed in the machinery; these instruments monitor system development and promptly activate a dedicated loop when required [68-69].

5.2 Fault Detection, Isolation And Reconfiguration Methods

Faults can be classified following different criteria and the most common ones are listed below:

- Failure based on equipment involved (e.g. actuator, plant or sensor);

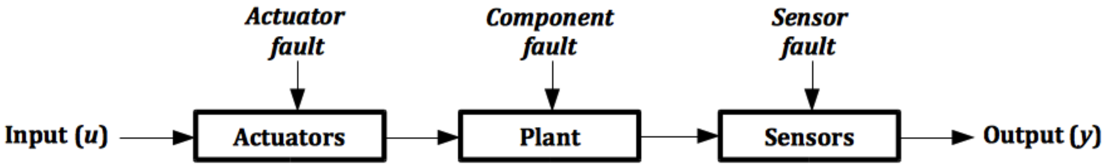


Fig. 1. Failure based on equipment

- Failure based on fault form e.g. abrupt (stepwise), incipient (drift-like) and intermittent faults (with interrupts);

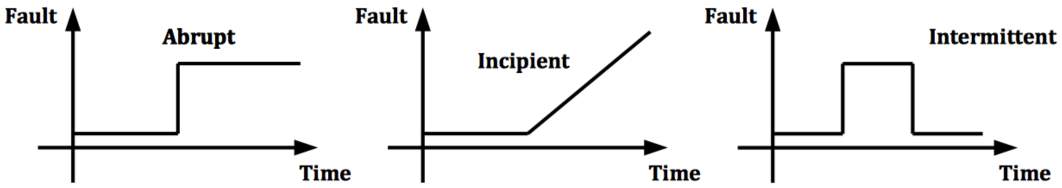


Fig. 2. Failure based on fault form

- Failure based on the modality of fault addition (additive or multiplicative).

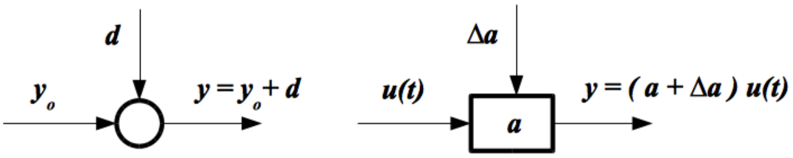


Fig. 3. Failure based on fault addition modality

Fault management is based on three steps: detection, isolation and reconfiguration (FDIR).

The first step is fault detection and obviously it is the most critical of the whole loop; in literature there are many methods and also several classification standards.
In this study the following is adopted: Data and Signal Methods, Process Model-based Methods, Knowledge-based Methods.

5.2.1. Data Methods

Data based methods use only available experimental and historical data and the two reference procedures are Limit/trend checking and Data analysis (PCA).
Limit checking requires two limit values (thresholds), a maximal value y_{max} and a minimal value y_{min}. In normal running conditions:

$$y_{\min} \leq y(t) \leq y_{\max} \tag{168}$$

Otherwise, trend checking method uses first derivative:

$$y'_{\min} \leq y'(t) \leq y'_{\max} \tag{169}$$

These methods are very simple and reliable but they can detect only large change of feature.

5.2.2. Signal Methods

Signal analysis can be applied only in case a fault in a process produces a change in a signal. By assuming mathematical models for the measured signal, suitable features such as amplitude, phase and spectrum are assessed. Normal behaviour values are used as term of reference for each feature (analytical symptoms).
In Spectrum Analysis, the extraction of fault-relevant signal characteristics can be restricted to the amplitudes or amplitude densities within a certain signal bandwidth. Fast Fourier transform (FFT) can be used to calculate frequency content of signal in time domain x(t). During normal operation component amplitudes A_i fall within particular range:

$$x(t) = A_0 + \sum_{i=1}^{N} A_i \sin(\omega_i t + \theta_i) \tag{170}$$

$$A_{i,\min} \leq |A_i| \leq A_{i,\max} \tag{171}$$

Another method is Parametric Signal Model such as ARMA - AutoRegressive Moving Average: *ARMA(p,q)* refers to the model with p autoregressive terms φi and q moving average terms i, constant c and error terms εt, εt-i:

$$X_i = c + \varepsilon_t + \sum_{i=1}^{p} \varphi_i X_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} \tag{172}$$

Parametric models are very sensitive to small frequency changes [66-68].

5.2.3. Knowledge-based Methods

The most used knowledge-based methods are found on expert systems and Fuzzy logic.
Rule-based expert systems obviously require expertise and experience of the system under analysis but full understanding of system physical properties and operating principles is frequently unavailable or too costly.
The main knowledge source is the experience of domain specialists and the main advantages of expert system are easy-add (removal) of rules, explanation of the reasoning process and induction (deduction) of processes.
Otherwise, some disadvantages are expensive development and maintenance, lack of generality and insufficient familiarity with novel situations.

The other procedure widely utilized to assess fault detection is Fuzzy logic that is used in case a simple binary decision (based on two states, fault/not fault) is not enough to control the system.
Fuzzy controller (Fig. 4) is based on a linguistically interpretable rule-based model built on expert knowledge and measured data [83].

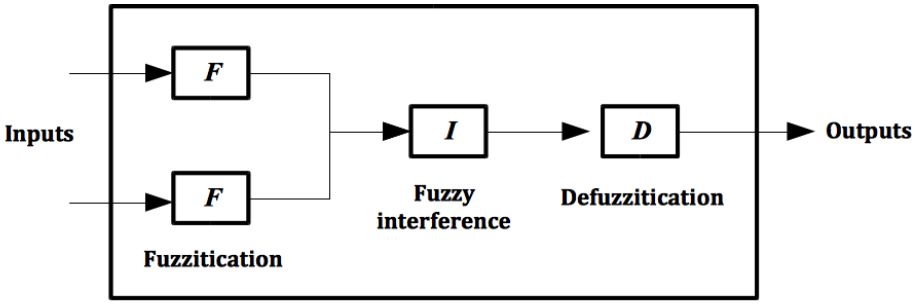


Fig. 4. Fuzzy logic controller logic diagram

The Fuzzy inference process involves the following steps:

- Fuzzification: inputs pass through the fuzzification process using membership functions that are a graphical representation of the magnitude of participation of each input.
- Rule-based inference: all rules are evaluated in parallel using fuzzy logic and the process of Fuzzy uses membership functions, logical operations and if-then rules.

- Defuzzification: fuzzy information are converted to neat ones using the inference process and computing the "fuzzy centroid" of the area:

$$x^* = \frac{\int \mu_i(x) x\, dx}{\int \mu_i(x)\, dx} \tag{173}$$

where x* is the defuzzified value, $\mu_i(x)$ is the aggregated membership function and x is the output variable.

5.2.4. Process Model-based Methods

Process model-based methods for fault detection often utilize the concept of redundancy that can be either analytical or hardware-based: analytical redundancy compares the system outcomes with a mathematical model and usually it doesn't require additional hardware. It can be divided in quantitative model-based method (using explicit mathematical models and control theories to generate residuals) and qualitative model-based ones (using artificial intelligence techniques to capture differences between observed and predicted behaviour). On the other hand, hardware-based redundancy compares the same signal measurements generated by various hardware.

Process model-based method involves two stages: residual generation and residual evaluation. This approach assumes that the structure and the parameters of the model are precisely known [83-84].

This study is focused on these fault detection methods and the whole fault management procedure is described in the following paragraphs.

5.2.4.1. Fault Management Steps

Fault management is based on three steps: detection, isolation and reconfiguration (FDIR).

The first step is to generate the residuals, a set of variables achieved using one or more residual generation procedures: these residuals should be zero in absence of faults and totally insensitive to noise and model uncertainties. In some applications two or more residual generation filters (designed to be sensitive only to a selective set of faults) are used in parallel for fault isolation assessment.

The second step is failure isolation that concerns the identification of the type of failure occurred: it is usually achieved using statistical tools to test the residual deviation from zero.

After the detection and the isolation of fault, the final step is the system reconfiguration.

In the following paragraph four basic concepts of FDIR are described: system and fault modelling, residual generation, fault isolation, decision making [84].

5.2.4.2. System and Fault Modelling

Figure 5 illustrates a general model for fault detection and isolation. The plant dynamics are modelled as follows:

$$x(t+1) = (A + \Delta A)x(t) + (B + \Delta B)u(t) + E_1 n_1(t) \tag{174}$$

$$y(t) = (C + \Delta C)x(t) + (D + \Delta D)u(t) + E_2 n_2(t) \tag{175}$$

Where x is the state vector, u is the plant's input vector, y is the output vector measured by the sensors and n_1 and n_2 are noise and unknown disturbance vectors; ΔA, ΔB, ΔC and ΔD are model uncertainties.

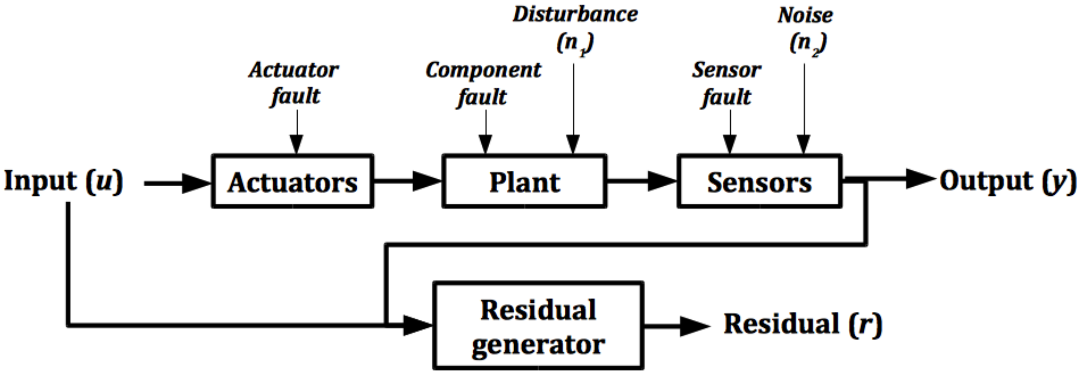


Fig. 5. System and fault modelling

The faults in the considered system can involve actuators, sensors and components: actuator and sensor faults are usually modelled as additive faults while component faults typically lead to changes in the parameters of the system dynamics and they are commonly modelled as multiplicative faults.

A general fault model for the system is the following:

$$x(t+1) = (A + \Delta A + \Delta A_C)x(t) + (B + \Delta B + \Delta B_C)u(t) + E_1 n_1(t) + B f_a(t) \tag{176}$$

$$y(t) = (C + \Delta C + \Delta C_C)x(t) + (D + \Delta D + \Delta D_C)u(t) + E_2 n_2(t) + f_s(t) \tag{177}$$

Where $f_a(t)$ represents the actuator faults, $f_s(t)$ the sensor faults, ΔA_C and ΔB_C the component faults.

Obviously the main objective of FDI procedures is to generate residuals which are insensitive to noise, disturbances, and model uncertainties.

Some algorithms are designed to be robust to additive noise and disturbances while more difficulties arises in case of multiplicative faults such as component faults and model uncertainties. The simplest way to overcome this problem is to model ΔA, ΔB etc. as additive disturbances (using time-varying disturbance-to-state system matrices) and to replace component faults ΔA_C and ΔB_C with an additive fault vector $f_c(t)$. The corresponding model is shown below:

$$x(t+1) = Ax(t) + Bu(t) + E_1(t)n_1(t) + Bf_a(t) + F_1(t)f_c(t) \tag{178}$$

$$y(t) = Cx(t) + Du(t) + E_2(t)n_2(t) + f_s(t) + F_2(t)f_c(t) \tag{179}$$

The state-space model can be transformed into the input-output framework considering $n(t) = \left[n_1(t)\ n_2(t)\right]^T$ as the noise vector and $f(t) = \left[f_a(t)\ f_s(t)\ f_s(t)\right]^T$ as the fault vector:

$$y(t) = G(z)u(t) + F(z)f(t) + E(z)n(t) \tag{180}$$

Where:

$$G(z) = C(zI - A)^{-1}B + D \tag{181}$$

$$F(z) = \left[(zI - A)^{-1}E_1E_2\right] \tag{182}$$

$$E(z) = \left[(zI - A)^{-1}BI(zI - A)F_1 + F_2\right] \tag{183}$$

5.2.4.3. Residual Generation

The residual is defined as the difference between the measured output *y(t)* and the estimated output *y'(t)* in the plant's model:

$$r(t) = y(t) - y'(t) \tag{184}$$

The residual must satisfy two properties to accurately assess fault detection: invariance relation and fault detectability.

Invariance relation requires that in absence of failures the mean of the residual is zero; fault detectability, instead, requires that in case of failure arising the residual deviates from zero.

In many applications residuals are conditioned by the presence of noise, unknown disturbances and system model uncertainties: the main target of FDI procedures is to realize a residual model that is sensitive to faults and, at the same time, insensitive to disturbances.

There are different methods to achieve robust residuals, the most important are listed below:

- Observer-based approach;
- Parity relations approach;
- Optimization-based approach;
- Kalman filter-based approach;
- System identification approach;
- Discrete event systems/hybrid systems approaches;
- Parameter estimation approach;
- Non-linear approach.

Observer-based methods generate residuals from an observer: in absence of fault, the observer tracks the actual plant and the residual should be zero.

In case of fault the observer is usually designed to produce residuals that facilitates fault isolation.

The basic idea of the observer approach is to reconstruct the outputs of the system from the measurements with the aid of observers using the estimation error as residual for the detection of the fault.

In terms of the residual characteristics, the various observer-based methods (e.g. eigen-structure assignment, fault detection filters, and unknown input observers) give identical results of an equivalent Parity relation method: this method compares the process behaviour with a process model describing nominal (and non-faulty) behaviour. The key idea is to check the consistency (parity) of the mathematical equations of the system [83].

Parity equations method and state observers differ in filtering of a residual but they have similar equations and they both convert input-output transfer function or plant state-space models to generate directional/structural residual vectors directly. In more, these two approaches do not consider multiplicative faults or model uncertainties and they are also limited to linear time-invariant systems.

Furthermore in the presence of disturbances not modelled or model uncertainties, the fault detection algorithm would fail to generate a zero-mean residual in absence of faults. This error is usually compensated with the choice of a higher detection threshold but obviously it increases the detection delays.

The Optimization-based methods provide solutions to the FDI problems with improvement of some mathematical objective functions: these methods are widely used for nonlinear and time-varying systems and they are designed to minimize the sensitivities of the residuals to noise, disturbance, and model uncertainties. Anyway their application could be complex and there is no guarantee about the usefulness and performance of solutions.

Also Kalman filter-based methods are complex if compared to observer based or parity relation methods, particularly when the number of failure modes is high; however they provide optimal filtering under normal operating conditions (in absence of faults) and, thanks to the filter parallel structure, they provide accurate state estimation after a fault occurs.

The multiple model approach could be extended to nonlinear or time-varying systems to detect multiplicative (or component) faults as well as additive ones.

The main drawback of this approach is the assumption that fault parameters have a discrete value belonging to finite set.

In case of fault with continuous values, few fault models may be used to describe it: however this method is higher in complexity and may increase false alarm rate [84].

The presence of both continuous state dynamics and discrete state dynamics in hybrid systems introduces additional difficulty and complexity to the FDI problem [**fonte**]. Residual assessment can be considered robust in many FDI methods in case of unknown disturbance or system uncertainties but in hybrid systems the discrete state transitions generate other uncertainties that can't be ignored.

System identification methods are applicable to both linear and non-linear systems and they are also useful to detect small or incipient faults. However, the efficiency of these methods is closely related to the statistical decision techniques used to detect the change in parameters.

In case the process parameters are partially not known or not known at all, they can be determined with Parameter estimation methods by measuring the input and output signal (it is required to know the basic model structure).

Faults of a dynamical system are reflected in physical parameters (friction, mass, resistance, capacitance, inductance etc.). The idea of the parameter identification approach is to detect the faults with the estimation of parameters of the mathematical model.

In many industrial processes it is not possible to use conventional modelling approaches due to the lack of knowledge of the system and also due to non-linear behaviour. In cases a mathematical process model is not available, a non-linear model can be employed to generate residuals. One of the most used ways to build a non-linear model is to use neural networks: this procedure does not require specific knowledge of process structure. Neural networks can serve as black-box models for nonlinear multivariable static and dynamic systems.

Neural networks are based on many parameters but these parameters are generally not suitable for physical interpretation of the modelled system: after process modelling, however, fault detection can be assessed with parity equations.

The choice of the fault detection method is an important step of the fault management strategy and many factors must be considered: type of failures, process structure, process dynamics, available process signals, process complexity, available amount of process input-output data and process suitability for description in terms of rules [65-70].

The simplest approach is the direct check of variable thresholds, whereas large scale processes such as can benefit from multivariate statistical analysis, in particular PCA. Some processes generate periodic or stochastic signals that can be used for fault detection if changes in signal models are caused by process faults.

Pattern recognition methods (e.g. neural nets) can be use when large amount of process input-output data can be obtained, but process structure is unknown or too complex to be modelled.

Process model-based fault detection includes process dynamics and non-measurable state variables, and it is easy to use in well-defined processes (e.g. electrical and mechanical) but requires accurate model.

In case basic relationship between faults and symptoms is known, knowledge based methods is probably the best choice [83-84].

5.2.4.4. Fault Isolation

The isolation of faults is based on the one-to-one correspondence between process deviation and fault cause: each residual should be sensitive to faults and distinguish between different types of faults.

There are two methods to generate residuals oriented to fault isolation, directional and structured residuals.

Directional residual approach is based on generation of residual vectors being part of the residual subspace: each vector has a specified direction that corresponds to a specified type of fault. This way fault isolation concerns determining the direction of the residual vector.

Structured residuals, instead, are sensitive to a single fault (or selective set of faults) and completely insensitive to the rest: these residuals are usually associated to incidence matrixes in which the row corresponds to residuals and column to faults (value "1" in the matrix represents the correspondence between a residual and a fault). In order to guarantee fault isolation, all columns must be different.

5.2.4.5. Decision-Making

The following step after residual assessment is to determine if a fault is occurred and its corresponding location and type; this decision is usually taken on statistical tests of the residuals.

The simplest procedure to detect a fault is to monitor if the instantaneous value of a residual vector exceeds a fixed threshold.

Sometime stochastic system models are used and the residuals generated are associated to probability distributions: this way it is possible to design decision tests based on adaptive thresholds.

More robust decision procedures use the history and trend of the residuals, and utilize powerful or optimal statistical test techniques.

5.2.4.6. Reconfiguration

The final step is reconfiguration that involves the change of the controller/equipment after fault occurrence and detection in order to ensure safe or satisfactory operation of the system.

There are different methods of reconfiguration such as those based on online learning or system identification.

The most important reconfiguration methods based on FDI are multiple model approach and adaptive control approach.

In the multiple-model approach, "n" parallel models are used to describe the system during normal operation and under different fault conditions.

Each model is associated to a corresponding controller; a switching mechanism is designed to determine the mode of the system and select the corresponding controller designed for that mode.

This results in robust and improved performance under various operating conditions.

The second approach uses an adaptive control to ensure robust or acceptable performance in case of fault and it is classified into two methods: the indirect adaptive control method (based on a parameter isolation process) and the direct adaptive control method (where explicit parameter isolation is not required).

5.2.4.7. Discussion And Remarks

Fault Detection, Isolation and Reconfiguration is a procedure to ensure the desired performance of a dynamic system both in the absence and presence of faults.

The previous paragraphs show the procedures and approaches for each step of FDIR process that are different in terms of performance, complexity, residual assessment and robustness.

The performance of fault detection procedures is a trade-off between the false alarm rate and the mean detection delay.

Fault isolation, instead, depends on the structural and directional characteristics of the residuals.

The robustness concerns the algorithm sensitivity to noise, disturbances, and model uncertainties.

5.3 System and Process Diagnostics: Condition Monitoring

Condition monitoring (CM) is the process of monitoring one or more condition parameters in machinery to identify some changes that are indicative of an incipient fault or equipment health degradation [70-71].

In the past, condition monitoring was applied simply through routine manual diagnostic actions but, with the introduction of low cost sensors and automated monitoring systems, online condition monitoring was adopted.

Condition monitoring systems select and survey parameters from the sensors placed in the system in order to detect a change in the health machine condition.

This technique is a major component of predictive maintenance: the use of CM provides all the information to schedule maintenance activities and prevent failures. Cost reduction is guaranteed due to maximum uptime and optimal production efficiency.

The added values of online condition monitoring rather than offline and manual data collection are listed below:

- Workforce optimization: manual diagnostics requires time and resource allocation to analyse collected data and assess required maintenance targets.
- Increase data storage: online monitoring guarantees continuous measurements for any piece of machinery, avoids mistakes in the registration of values and creates a trustworthy database.
- Improved diagnostics: more accuracy in failure prediction is achievable thanks to unique database for historical trend and baseline data.

Condition monitoring consists of data collection, signal processing and analysis; these steps are necessary to provide a complete overview of machine health and predict remaining useful life (RUL) of each component.

Condition monitoring techniques are widely used on rotating equipment and other machinery (e.g. pumps, electric motors, engines); the most popular methods used in modern industries are vibration analysis, oil analysis, thermal analysis and ultrasound analysis [71].

5.3.1. Limit Alarm Trip

A limit alarm trip is one of the simplest and most important applications of condition monitoring in industrial systems: process limit alarm monitors the signals provided by temperature, pressure, level or flow sensors and compares it against a pre-set limit. In case the process signal goes to an undesirable high or low condition, the limit alarm activates a relay output to warn of trouble, provide on/off control or command an emergency shutdown.

Limit alarm trip allows process monitoring in points of the system that are considered critical for both control and safety purposes.

There are two types of alarm:

- Hard alarm: it is an independent limit alarm trip hard-wired into the process and it is usually founded on relay output;
- Soft alarm: it is a software-implemented alarm based on DCS (distributed control systems) or PLC (programmable logic controllers).

In many applications alarm functions are performed by "soft" practices, anyway "hard" ones are widely used to implement low-cost redundancy, simple control and backup of DCS and PLC strategies in critical emergency shutdown and Safety Related Systems [71-72].

Soft alarms, in fact, are susceptible to common-mode failures while hard alarms are not exposed since they are independent from the DCS or PLC.

Furthermore hard alarms provide continuous supervision of the individual monitored process while soft practices performs intermittent scanning.

Limit alarm trip can provide many different actions from a simple annunciation of process unexpected behaviour to a system emergency shutdown.

The alarm trip receives input signals from monitoring or control instrument and, in case the monitored process variable moves outside the set-point, the alarm trip command a pre-set action. There are two thresholds, high and low, associated to the pre-set high and low alarm points.

Usually the alarm condition is maintained until the process signal moves back to "normal running" values and passes out the dead-band (it is the measurement range to reset the device and restore the "non-alarm" state). There are two thresholds, high and low, associated to the pre-set high and low alarm points.

A limit alarm trip can have one, two or even four relay outputs: usually each relay output corresponds to a dedicated trip point [72].

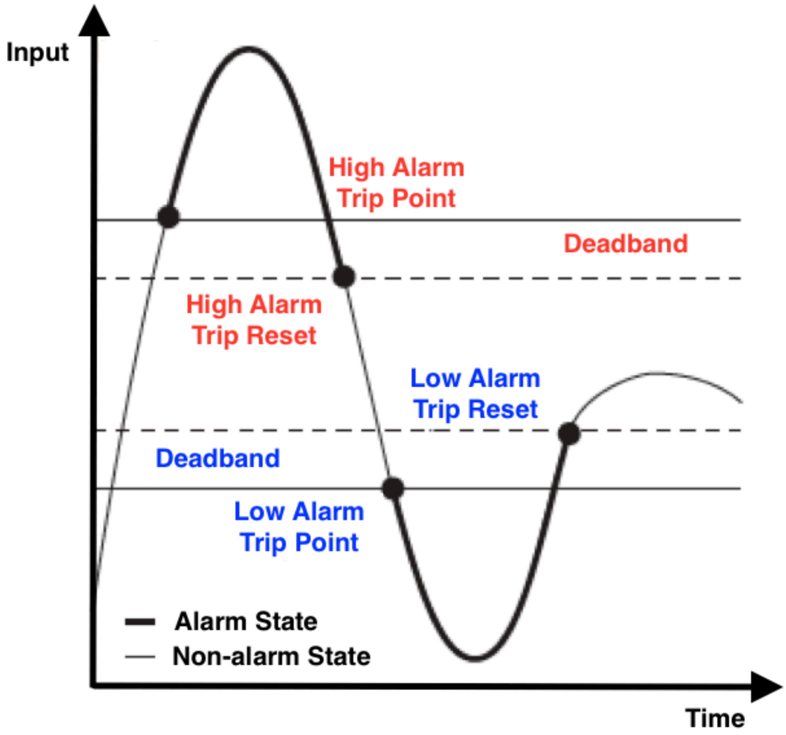


Fig. 5. Input signal vs. Time

5.4 Local Diagnostics

Many devices used in manufacturing applications e.g. Oil&Gas are a two wire 4-20mA sensor assembly made of one or more sensing device (i.e. thermocouples or RTDs in temperature instruments) and one dedicated transmitter to communicate with system control panel.
The outcome of a field sensor can vary in response of changes in the monitored physical quantity or in case of failure.
Diagnostics clearly play an essential role to distinguish between these two conditions.

5.4.1. On-board Diagnostics

If the sensor is provided with a dedicated on-board circuit, the device itself communicates its health status to the control logic using out-of-range outputs or dedicated communication channels.
There are two main communication protocols: *Highway Addressable Remote Transducer* (HART®) and *Foundation™ Fieldbus* (FF).
HART and FF both bring significant benefits to process industry using intelligent field devices with the difference that HART is a hybrid protocol fully compatible with 4–20 mA wiring while FOUNDATION fieldbus is a distributed control system based on a multi-drop bus. The focus of HART protocol is to bring digital information maintaining compatibility with 4-20

mA signal; on the other hand, the focus of Foundation fieldbus is to bring the control architecture to the bus and bring down the control to device level [73-74].
The resulting protocols have different layout and capabilities that are described in the following paragraphs.

5.4.1.1. Highway Addressable Remote Transducer (HART)

HART is a master-slave communication protocol so each slave device (or field instrument such as transmitters, actuators, and controllers) during normal operation is initiated by a master device (e.g. distributed control system, programmable logic controller, personal computer).
This communication protocol is widely used in Oil&Gas applications because can communicate over legacy 4-20mA analog instrumentation wiring sharing the pair of wires used by the standard system.
For this reason HART is considered a "smart" protocol since it doesn't require any change in the wiring and can be implemented in a pre-existing system.
HART Protocol indeed makes use of the Bell 202 Frequency Shift Keying (FSK) standard to superimpose digital communication signals (containing additional device information such as device status and diagnostics) on top of the 4-20mA (containing the primary measured value); as the digital FSK signal is phase continuous, there is no interference with the 4-20mA signal. Together, the two communication channels provide a low-cost and robust solution that is easy to use and implement.
HART devices can operate in two network configurations, point-to-point or multi-drop. In point-to-point mode, the analog 4–20 mA signal is used to communicate one process variable and the additional info (e.g. process variables, configuration parameters, device data) are transferred digitally using the HART protocol. These secondary variables can be used for operations, commissioning, maintenance, and diagnostic purposes.
The multi-drop modality, instead, requires only a single pair of wires and, in some applications, safety barriers and auxiliary power supply; all process values are transmitted digitally. Multi-drop connection is usually used for supervisory control installations that are widely spaced [73-76].
The benefits of HART protocol are listed below:

- Backward compatibility: HART implementation is compatible with the installed base of instrumentation in use in the pre-existing system so it doesn't require any change in the wiring.
- Improved plant operations: HART protocol improves plant performance and provides savings in commissioning and installation (in particular for wiring).
- Improvement of plant operation quality: HART protocol provides access to all information in multi-variable devices which can be used for verification and control of the whole plant.

- Maintenance: cost-saving can be assessed also by reducing downtime; in fact HART diagnostic guarantees to minimize the time required to identify failures and take corrective action.

5.4.1.2. Foundation Fieldbus (FF)

FOUNDATION Fieldbus is a digital, bi-directional and multi-drop Local Area Network (LAN) for process control sensors, actuators, and control devices; furthermore it is an open standard that allows the field devices to run both input/output and control. This is one of the main differences with PROFIBUS and HART protocols that do not implement control and require a separate controller: field devices take input measurements and send the information to a control unit for processing. In case of failure of the control device, field devices go into some pre-defined fail-safe mode, leaving actuators (e.g. pumps, valves, etc.) without any interactive control until logic solver restores [77].

With FF control is brought down to the device level so many operations can be assessed even if the monitoring computer is disconnected; furthermore, field device operations such as calibration and testing can be done directly from the control room without manual operations.

The benefits of FOUNDATION fieldbus are listed below:

- Distributed control: the control unit computer does not do it but it is assessed at device level.
- Open standard: customers can choose interchangeably products from different vendors.
- The Fieldbus Foundation standardized the way the user can bring new devices into the network, set and configure them.
- The building block in this system is the Device Description (DD) which tells everything about the device and its functionality
- Wiring and controller cost reduction: with FF users need only one twisted wire pair that will carry multiple signals and power, and they can drop devices off the network at any point. 4-20 mA systems require one pair of wires per device while FF requires only a single set of wires to connect multiple devices.

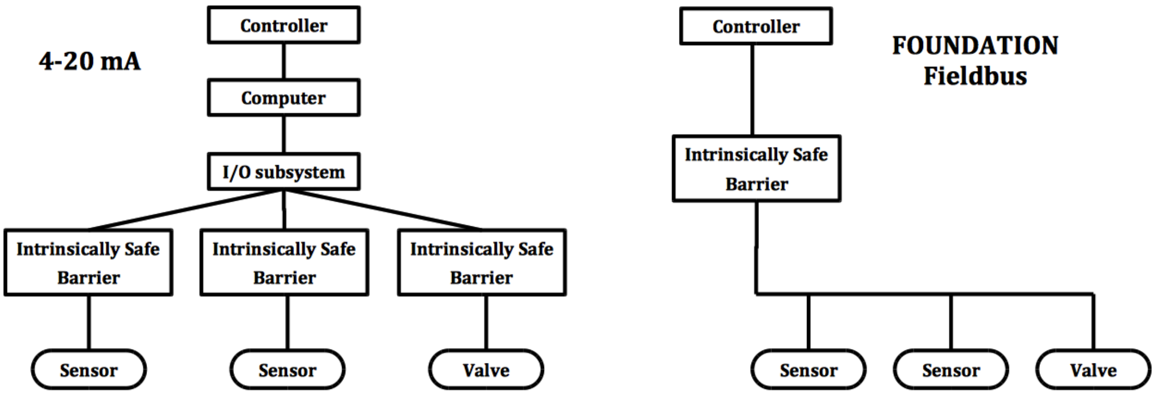

4-20 mA
Controller
Computer
I/O subsystem
Intrinsically Safe Barrier
Intrinsically Safe Barrier
Intrinsically Safe Barrier
Sensor
Sensor
Valve
Controller
FOUNDATION Fieldbus
Intrinsically Safe Barrier
Sensor
Sensor
Valve

Fig. 6. Wiring comparison of FOUNDATION Fieldbus and 4-20mA

5.4.1.3. HART vs. FF

HART and FF are good protocols for configuration, calibration, diagnostics, and viewing internal variables. Furthermore FF is used for real-time closed loop control. This is the major difference between these two protocols (in HART applications, control system must use 4-20 mA for closed loop real-time control).

FF is completely digital end-to-end, from sensor to actuator and it has several benefits over loops using hardwired 4-20 mA and on/off signals.

Other pros of FF comparing with HART are the following:

- Balanced (non-grounded) signal with high amplitude for noise immunity;
- Multiple devices on the same pair of wires reducing cable, tray, junction boxes and associated manpower;
- Reduction of I/O cards reducing system footprint and weight;
- Elimination of I/O card selection, safety barrier selection and signal marshalling simplifying engineering;
- Easy addition of devices and signals in devices;
- Time synchronized control and fast control response period.

These points are really FF advantages over 4-20 mA and on/off signals.

Anyway a plant using 4-20 mA with HART is far better than a plant using only 4-20 mA or other proprietary smart protocols [74-77].

5.4.2. Logic Solver Diagnostics

In case field sensors are not equipped with on-board diagnostics or HART protocol is not put into practice, condition monitoring is submitted to the logic solver that analyses measure trends or compares different data coming from multiple devices (in case of redundant architectures).

If the process signal moves to an undesirable high or low condition, the logic solver performs the safety loop e.g. warning relay output activation, on/off control or emergency shutdown.

Usually the control implementation is associated to different thresholds that are upper and lower limits of the physical quantity under analysis. During normal operation these thresholds are used to define the operative range; in case the measurement crosses those values, the control panel triggers the loop following the implemented logic.

Temperature, pressure, level and flow monitoring in Oil&Gas application is usually achieved with a four threshold strategy, shown in Fig. 7: "H, high", "HH, high-high", "L, low" and "LL, low-low".

"H" and "L" threshold-crossing usually leads to a visual alarm on the control panel to make the operator aware of the problem; "HH" and "LL" values, instead, are associated with more dangerous conditions. They lead to a progressive load reduction and gradual system shutdown or, in case of extremely critical loop, to an emergency shutdown that instantly stops the machine (this kind of event is always associated to a concrete risk for environment and health and safety of operators).

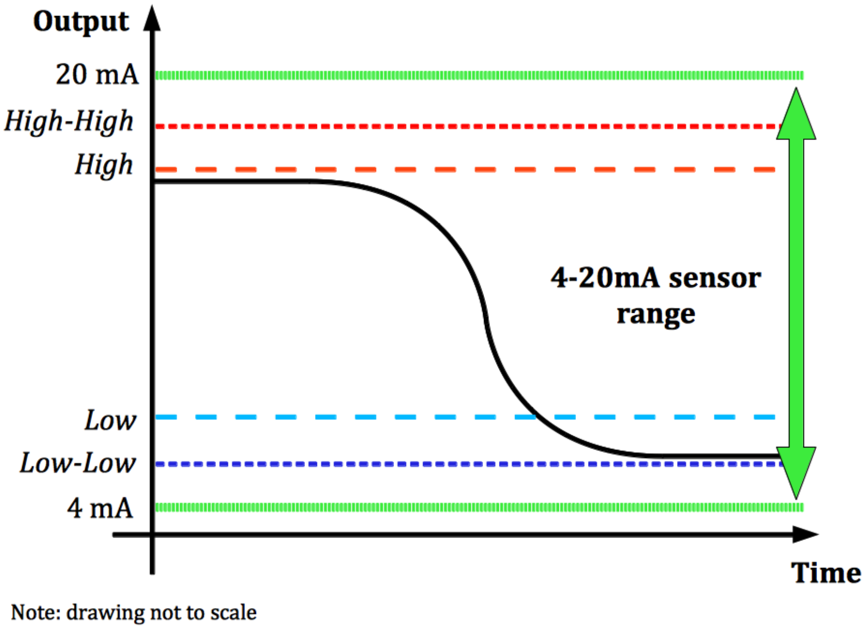


Fig. 7. Output range and safety thresholds for 4-20mA analog sensor

Field sensors can be affected by different type of failures that underline the importance of diagnostics to detect failures and guarantee safe system operation [78-80].

The most common failure modes of field sensors are described as follows:

- Out of calibration: field sensors must be calibrated against a known standard but only short-term stability is checked during calibration; long-term stability should be monitored and determined by the user. This kind of failure can be observed instantly after installation (faulty assembly) or during device operation.
- Out of range: sensor failure is usually detected by performing a range check of device outcomes: all incoming values are checked against a given range by the logic solver.

Sensor values that are outside that range are assumed to be incorrect and the device is considered out of order. High, low and no output fall into this category.

- Stack in-range: sensors are usually designed to fail out of range and this failure mode is quite rare but anyway cannot be ignored. When stack in-range occurs, sensor outcome is fixed inside the standard range of operation; for this reason this failure mode is undetectable by the control logic without a dedicated on-board diagnostic.
- Drift: signal drift follows a gradual and incremental trend towards the upper/lower limit of operative range; for this reason drift failures are critical since control logic cannot be aware of failure occurrence until device output goes out of range (in absence of on-board diagnostics). Drift can occur very slowly and that period of time is rather critical because control logic is using wrong values coming from a faulty sensor.

Out of calibration and stack in-range are detectable only in presence of on-board diagnostics since the logic solver by itself only compares sensor outcomes with predefined thresholds. Anyway out-of-range and drift are the most common failure modes of field sensors [81-83].

In the following paragraphs two applications of logic solver diagnostics are shown: single item (Pressure Indicator Transmitter) and safety loop consisting of sensors in redundant architecture (2oo3 Temperature Indicator Transmitter), logic solver and actuators.

5.5 Case study 1: Pressure Indicator Transmitter

Rosemount developed an Advanced Diagnostics Suite for Pressure Indicator Transmitter 3051S: this device has two distinct diagnostic functions, *Statistical Process Monitoring* (SPM) and *Plugged Impulse Line Detection* (PIL).

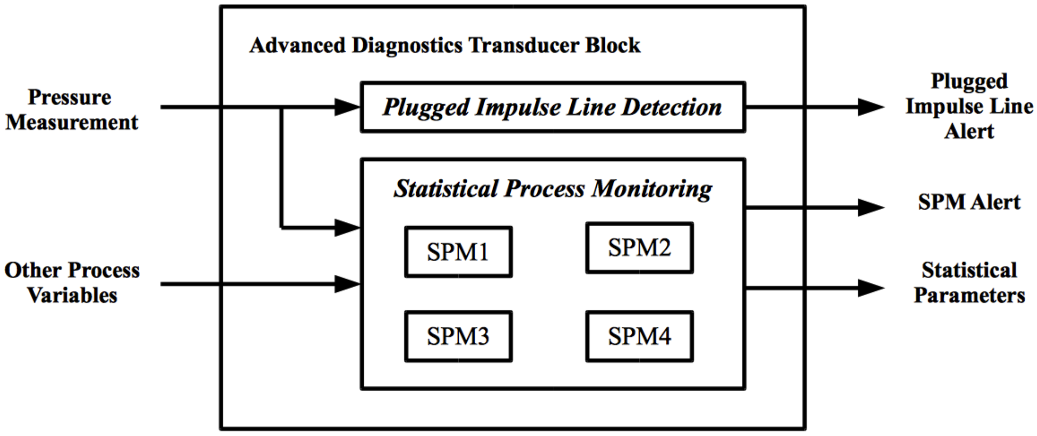

Advanced Diagnostics Transducer Block
Pressure Measurement
Other Process Variables
Plugged Impulse Line Detection
Statistical Process Monitoring
SPM1
SPM2
SPM3
SPM4
Plugged Impulse Line Alert
SPM Alert
Statistical Parameters

Fig. 8. Advanced Diagnostic block diagram

Statistical Process Monitoring (SPM) is used to detect changes in processes, in process equipment or installation conditions of the device: this technology is based on modelling the process noise signature using statistical values such as mean and standard deviation.

The key assumption required for this method is the following: all dynamic processes have a unique noise signature during normal conditions so any change in these signals is symptomatic of a significant change in the process.
The baseline corresponds to the values assumed by the statistical parameters during normal operation (normal running) and these values are compared to current values over time; in case a significant change is detected, the transmitter can generate an alert.
Statistical Process Monitoring perform this statistical processing on either the primary value of the field device (e.g. pressure measurement) or any other process variable available (up to four variables simultaneously with SPM1-SPM4).
For Rosemount 3051S Pressure Indicator Transmitter, the statistical parameters monitored to detect any process variation are mean and standard deviation of the input pressure. Fig. 9 shows how the standard deviation value (σ) is affected by changes in noise level while the mean or average value (μ) remains constant.

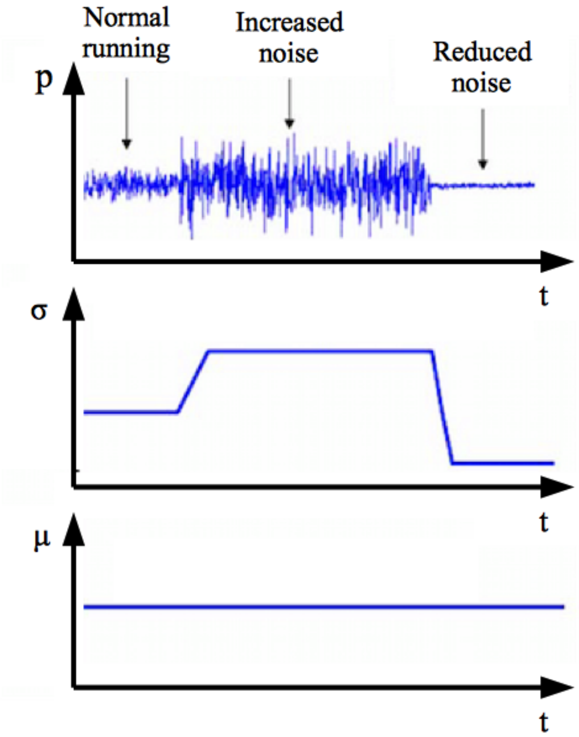


Fig. 9. Input pressure with noise level changes, standard deviation and mean values vs. time

Statistical Process Monitoring devices are equipped with a learning module and a decision module. The first one fixes the process baseline values for comparison, baselines are set in normal running conditions under user control and are made available to the secondo module that compares them with current values of the mean and standard deviation.
The statistical parameters can be provided to the user in two ways, with communication protocols or internal software.
Foundation fieldbus communication protocol can be used to transfer statistical information to the control room and, once available, these data are used to detect a change in process conditions [65].

Otherwise, the device may be equipped with internal software used to baseline the process noise or signature via a learning process. Once the learning process is completed, the device itself can detect significant changes in the noise or variation, and produce an alarm.

The device has three states: learning, verifying and monitoring [81-83].

During learning period, the baseline mean and standard deviation are calculated over a period of time controlled by the user (default is 15 minutes).

Afterwards, a second set of values is calculated and compared to the original set to validate the stability and repeatability of the process: if also this stage is completed (the process is stable), the monitoring status can be activated.

During monitoring, new mean and standard deviation values are continuously calculated, with new values available every few seconds.

Both the mean value and standard deviation are compared with baseline standards: if the difference exceeds the established threshold, likely a failure occurred and an alert is generated.

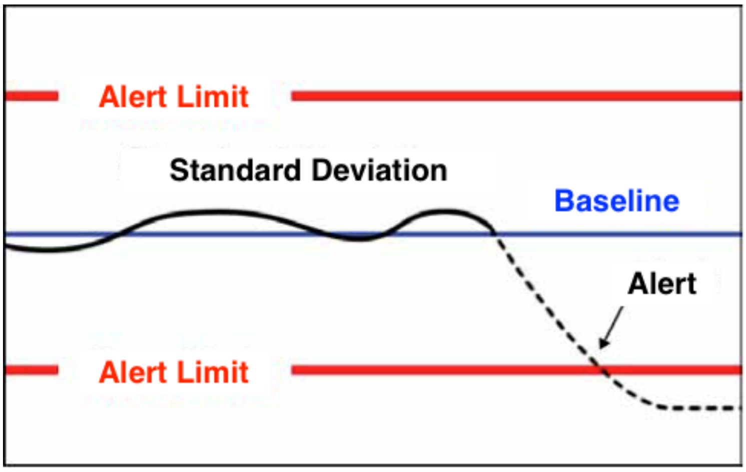


Fig. 10. Input pressure trend and alert thresholds

Plugged Impulse Line (PIL) is the second algorithm implemented in the Rosemount Advanced Diagnostics Suite: PIL diagnostics can detect the presence of a plug in pressure measurement impulse lines which are small diameter pipes used to transmit the pressure signal from the process to the transmitter. Pressure transmitters are rarely connected directly to the pipe or vessel and, in some applications, these lines can become plugged with solids or frozen fluid blocking the pressure signals [65].

Without a dedicated diagnostics, the user can’t become aware that the blockage has occurred; this event is rather critical because the transmitter may provide wrong values to control logic.

This problem is solved with PIL diagnostics that produces an alarm after plugging detection; furthermore PIL automatically relearn new baseline values if the process condition changes.

Since the plug effectively disconnects the transmitter from the process, it changes the noise pattern received by the transmitter.

Both pressure and differential pressure signals usually show fluctuations or noise: fluctuations are produced by the fluid and they are a connected to system layout and physical features. Noise may be produced by pump section or control system however it is generally little compared to average pressure value.

The noise signature monitoring is the key to recognize system changes and it is one of the best way to assess plugged impulse line detection since it isn't affected by small changes of the average pressure value.

When impulse lines start to plug, noise signatures in both time and frequency domain start to take distance from normal running conditions: the transmitter may not receive the noise signal anymore or the noise may decrease significantly while the average pressure value remains the same. This is true also for differential pressure devices that are equipped with two impulse lines placed at high and low pressure sides of the equipment-under-test; in this case the noise signature decreases when both impulse lines start plugging.

During normal operations both lines are open and the sensor calculate the difference between high and low-pressure measurements.

When a plug occurs in one of the lines, there is no more common mode cancellation and a corresponding noise increase in the differential signal.

PIL diagnostics calculate the mean and the standard deviation of the pressure measurement and, in case standard deviation exceeds the established threshold, an alert is generated.

An important requirement for the process under analysis is its stability: a trustworthy baseline is necessary to compare the process trend and recognize the presence of a plug. An unstable process is a poor candidate to assess Plugged Impulse Line diagnostics and may produce frequent re-learning procedures and false trips.

The length of the impulse line is another important feature: a long line can generate additional noise signals (due to resonances) overlapping on process noise. This way, in case of plug, the transmitter does not detect a significant change in noise level and the dangerous condition is undetectable [81-83].

5.6 Case Study 2: Safety Loop

Logic solver diagnostics may be involved in safety applications: in this study we'll focus on diagnostic procedures of 2oo3 TIT architecture that is a sensor assembly dedicated to thermal analysis. It plays the role of sensing stage of a Safety Instrumented System (SIS), a particular control system capable to take the process to a safe state when hazardous condition are detected.

Safety Instrumented Systems (Fig. 11) are typically constituted by a combination of three fundamental blocks [28]:

- Sensor(s) detects a physical quantity and provides a corresponding electrical output. Field sensors are used to collect information and determine an incipient danger; these sensors evaluate process parameters (e.g. temperature, pressure, flow, etc.) in order to determine if single equipment or the whole process or plant is working properly and it is in a safe state. Such sensors do not monitor the normal process but they are usually dedicated to SIS.

- Logic solver(s) receives the information collected by the sensor and elaborates it to take the best response. It is typically a controller that takes actions according to the defined logic in order to prevent hazardous conditions.
- Final element(s) implements the outcomes of the logic solver. This actuator is the last element of the loop and in many industrial applications is represented by a pneumatic valve.

The aim of SIS is to implement one or more Safety Instrumented Functions (SIF): these functions control critical processes and avoid unacceptable or dangerous conditions for health and environment. Each SIF is associated with a safety loop that is the process involving all the three stages described above (sensor, logic solver and final element) in order to detect a failure, elaborate the collected data and perform the corrective action [28].

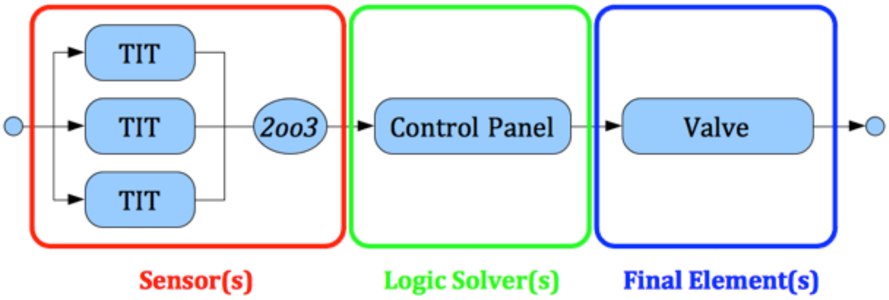


Fig. 11. SIS – Safety Instrumented System

5.6.1. Safety Loop Operation

The loop activation is usually associated with two processes, low and high trip corresponding to lower and upper threshold monitoring:

- Low trip level: the risk is associated to measurements below a predefined value and, in case the sensor output crosses this threshold, the safety function is activated. The best sensor for this kind of application is a "fail low" device that, in case of failure, goes to the predefined fail-safe state and produces a current < 3.6mA to activate the safety loop (supposing a standard 4-20mA analog instrumentation).
- High trip level: the risk is associated to measurements above the threshold and, in case the sensor output crosses it, the safety function is activated. For this kind of application "fail high" sensors are required since, when a failure arises, the device goes to the predefined fail-safe state and produces a current > 21.5mA (supposing a standard 4-20mA analog instrumentation).

The logic solver is the second stage of the safety loop and its detection strategy can influence the effectiveness of the safety function.
Table I shows the under-range and over-range detection capability of the logic solver, where "λ_{low}" is the failure rate of a fail-low device (in case of failure it goes to the predefined fail-safe

state producing a current < 3.6mA) while "λ_{high}" is the failure rate of a fail-high device (in this case the fail-safe state generates a current > 21.5mA).

The other failure rates (λ_{SD}, λ_{SU}, λ_{DD}, λ_{DU}) are distinguished by failure consequences and detection likelihood in according to IEC 61508 where the first subscript letter is referred to safe/dangerous failure, the second letter concern detection:

- Safe (S): a safe failure is a failure that causes the system to go to the defined fail-safe state without a demand from the process so it does not compromise the system safety integrity (e.g. a failure leading to a safe shut-down). This kind of failures impact only availability and productivity, not safety.
- Dangerous (D): a dangerous failure is a failure leading to a safety-related system failing to function and compromise system safety integrity.
- Detected (D): a failure that will be detected by diagnostic tests.
- Undetected (U): a failure that will be undetected by diagnostic tests.

In "low trip" practice a fail-low is always associated with a safe condition while a fail-high produces a dangerous scenario because the failure will prevent the device from indicating that the safety action needs to be performed; in case of "high trip" applications the opposite is true.

The difference between detected/undetected, instead, is due to under/over range monitoring by the logic solver: a sensor failure is detectable in case the device output goes fail-safe in the same direction of the monitored threshold (Fig.12).

In this analysis it is assumed that the logic solver is able to detect under and over range currents so both fail-low and fail-high conditions are detectable; the supposed behaviour of the logic solver is highlighted in Table I.

Table I - Logic Solver behaviour and corresponding failure rates

Application	Logic Solver Behaviour	λ_{low}	λ_{high}
Low Trip	< 4mA	λ_{SD}	λ_{DU}
	> 20mA	λ_{SU}	λ_{DD}
	< 4mA and > 20mA	λ_{SD}	λ_{DD}
	x	λ_{SU}	λ_{DU}
High Trip	< 4mA	λ_{DD}	λ_{SU}
	> 20mA	λ_{DU}	λ_{SD}
	< 4mA and > 20mA	λ_{DD}	λ_{SD}
	x	λ_{DU}	λ_{SU}

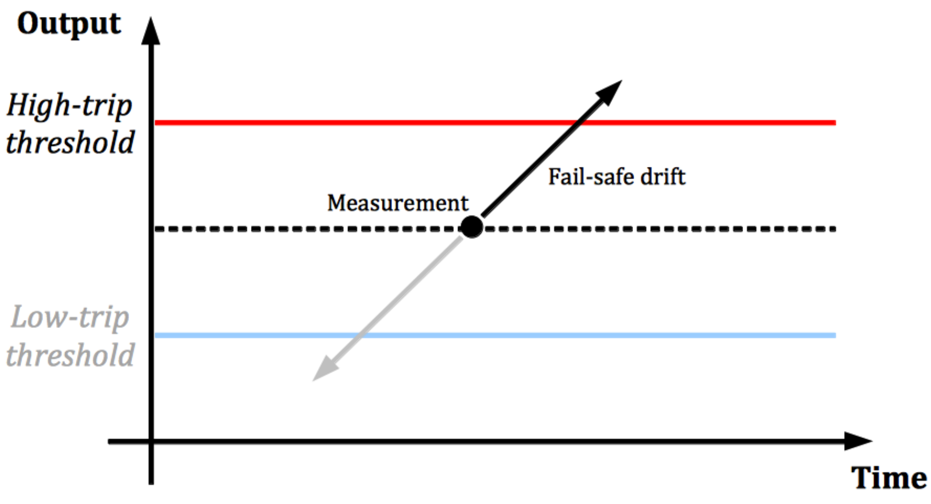


Fig. 12. Fail-safe drift in high-trip threshold direction

5.6.2. Reliability Block Diagram Approach for PFD

In order to assess Probability of Failure on Demand (PFD), Safe Failure Fraction (SFF) and Diagnostic Coverage (DC) Reliability Block Diagram approach is used for PFD assessment in different system architectures.

The necessary assumption, in compliance with IEC 61508, are shown below:

- Component failure rates are constant over the life of the system;
- The sensor (input) subsystem comprises the actual sensor(s) and wiring but not includes voting or other processing devices;
- The final element (output) subsystem comprises all the components and wiring from the logic solver to final actuating component(s);
- For each safety function, there is perfect proof testing and repair so all failures that remain undetected are detected by the proof test;
- The proof test interval is at least an order of magnitude greater than the Mean Repair Time (MRT);
- For each subsystem there is a single proof test interval and MRT;
- The expected interval between demands is at least an order of magnitude greater than the proof test interval.

Legend:

T_1: Proof Test Interval (hour)
T_2: interval between demands (hour)
MTTR: Mean Time To Restoration (hour)
MRT: Mean Repair Time (hour)
DC: Diagnostic Coverage
β: Fraction of undetected failures that have a common cause

β_D: Fraction of detected failures that have a common cause
PFD_{avg}: Average Probability of Failure on Demand
PFD_{SE}: Sensing element Probability of Failure on Demand
PFD_{LS}: Logic solver Probability of Failure on Demand
PFD_{FE}: Final element Probability of Failure on Demand
PFD_{SYS}: System Probability of Failure on Demand
t_{CE}: Channel equivalent MDT (combined down time for all the component in the channel of the subsystem; hour)
t_{GE}: Voted group equivalent MDT (combined down time for all the channels in the voted group; hour)

The average probability of failure on demand of a safety function for the safety-related system is determined by the combination of the average probability of failure on demand for all the subsystems involved in the safety function. Average PFD can be expressed as follows:

$$PFD_{SYS} = PFD_{SE} + PFD_{LS} + PFD_{FE} \tag{185}$$

5.6.3. Sensor Stage

SIS sensors monitor process conditions and provide process parameters in order to recognize a potential hazard. Usually the monitored variables are the same used for control. The fundamental requirement for sensors in safety applications is accuracy and reliability.

In this study the first stage of the safety loop under analysis is a 2-out-of-3 (2oo3) redundant architecture of temperature sensors widely used for Oil&Gas applications, Rosemount® 3144P HART Temperature Indicator Transmitter (TIT).

5.6.3.1. Temperature Indicator Transmitter

TIT is a two wire 4-20mA temperature sensor assembly made of one or more temperature-sensing devices (e.g. Thermocouples or RTDs) and one dedicated transmitter to communicate with system control panel: for Safety Instrumented Systems the 4-20mA output is used as the primary safety variable by the safety logic solver.

The outcome of a field sensor can vary in response of changes in the monitored physical quantity or in case of failure. Diagnostics clearly play an essential role to distinguish between these two conditions that is mandatory in particular for Safety Instrumented Systems; if the sensor is provided with a dedicated on-board circuit, the device itself communicates its health status to the control logic using out-of-range outputs or dedicated communication channel. Highway Addressable Remote Transducer (HART) Communication Protocol is widely used in Oil&Gas applications because it can communicate over legacy 4-20mA analog instrumentation wiring and share the pair of wires used by the standard system. For this reason HART is considered a “smart” protocol since it doesn’t require any change in the wiring and can be implemented in a pre-existing system. HART Protocol indeed makes use of the Bell 202 Frequency Shift Keying (FSK) standard to superimpose digital communication signals (containing additional device information such as device status and diagnostics) on top of the 4-20mA (containing the primary measured value); as the digital FSK signal is phase continuous, there is no interference with the 4-20mA signal. Together, the two communication channels provide a low-cost and robust solution that is easy to use and implement [73-74].

In case field sensors are not equipped with on-board diagnostics or HART protocol is not put into practice, condition monitoring is submitted to the logic solver that analyses measure trends or compares different data coming from multiple devices (in case of redundant architectures). The limit of this application is that the control logic can detect a failure only if the sensor output goes out of range.

Rosemount® 3144P HART Temperature Transmitter is made of a transmitter and a temperature-sensing device (e.g. thermocouple or RTD) and the analysis of this sensor must be separated depending on the nature of the sensing device.

Table II shows the failure rates of different sensing devices in different stress environments while Table IV shows the percentage of each failure mode for thermocouples and thermistors. The 3144P HART Temperature Transmitter with TC will detect thermocouple burnouts and drive its output to the specified failure state; wire short and drift failures are considered dangerous undetected. In RTD architecture, otherwise, both open and short circuits are detectable.

Table II - Failure rates of temperature sensing devices depending on the stress environment

Temperature Sensing Device	Failure Rate (FITs)
TC low stress environment	5000
TC high stress environment	20000
RTD low stress environment	2000
RTD low stress environment	8000

Table III - Failure mode percentages for thermocouples and RTDs

Failure Mode	TC Percentage	RTD Percentage
Open circuit	95%	70%
Wire short/ Short circuit	1%	29%
Drift	4%	1%

Table IV shows the failure rates for 3144P Temperature Transmitter (in TC and RTD configuration respectively) according to IEC 61508 and assuming that the logic solver can detect both over-scale and under-scale input (so both fail-high and fail-low are detectable).
According to IEC 61508 "no effect" (failure of a component part of the safety function but that has no effect on the safety function) and "annunciation undetected" (failure that does not directly impact the safety but influence the ability to detect a future fault) are classified as safe undetected.

Table IV - Failure rates for Temperature Transmitter in TC and RTD configuration

Failure Category	TC Failure Rate (FITs)	RTD Failure Rate (FITs)
Fail high (detected by logic solver)	28	28
Fail low (detected by logic solver)	302	295
Fail Dangerous (undetected)	66	63
No effect	104	104
Annunciation undetected	5	5

As said before, fail-low and fail-high can either be safe or dangerous depending on the application and detected or undetected depending on the programming of the logic solver.
In this study the Temperature Transmitter is programmed to drive its output low on detected failure (fail-safe state is under-range).
Since the temperature transmitter and the sensing device are in series, corresponding failure rates can be added; the failure rate contribution for the RTD in a low stress environment is:

$$\lambda_L = (2000)\cdot(0,70+0,29) = 1980 FITs\,; \quad \lambda_{DU} = (2000)\cdot(0,01) = 20 FITs \qquad (186)$$

The failure rate contribution of temperature transmitter when used with a thermistor is:

$$\lambda_L = 295 FITs\,; \quad \lambda_H = 28 FITs\,; \quad \lambda_{DU} = 63 FITs \qquad (187)$$

The total failure rates of the temperature sensor assembly are:

$$\lambda_L = 2275 FITs\,; \quad \lambda_H = 28 FITs\,; \quad \lambda_{DU} = 83 FITs \qquad (188)$$

Two important parameters for safety assessment are Diagnostic Coverage (DC) and Safe Failure Fraction (SFF).

DC is the ratio of the probability of detected failures to the probability of all the dangerous failures and it is a measure of system ability to detect failures; SFF, instead, indicates the probability of the system failing in a safe state so it shows the percentage of possible failures that are self-identified by the device or are safe and have no effect [28].

$$DC = \frac{\sum \lambda_{DD}}{\sum \lambda_{DD} + \sum \lambda_{DU}} \qquad SFF = \frac{\sum \lambda_S + \sum \lambda_{DD}}{\sum \lambda_S + \sum \lambda_{DD} + \sum \lambda_{DU}} \qquad (189)$$

Table V shows DC and SFF assessment for the Rosemount® 3144P HART Temperature Transmitter in RTD configuration: in this architecture, high vibration and recurrent temperature cycling are the key stress variables to cause cranks in the substrate leading to failures.

Table VI shows DC and SFF assessment for the whole temperature sensor assembly made of sensing element and transmitter.

Safe Failure Fraction in both cases is the same for high and low trip applications; undetected failures are always the same while detected ones switch themselves, so the SFF result doesn't change within the architecture.

A noticeable improvement in the SFF value is visible, instead, from the single transmitter to the sensor system assessment: this is due to the huge growth of safe detected failures and, on the other hand, thanks to the reduced increase of dangerous undetected [28].

On the other hand, Diagnostic Coverage results are not so intuitive and they require some in-depth analysis. DC in fact is higher in high trip applications rather than in low trip ones: the difference between these two practices is restricted to detected failures. As said before, both safe and dangerous undetected failures doesn't change and detected ones switch themselves; since DC assessment takes into account only dangerous failures, the number of dangerous undetected failures $\sum \lambda_{DU}$ has little incidence on the total dangerous failures amount $\sum \lambda_{DD}$ in high trip mode of operation so the Diagnostic Coverage value is higher.

So in terms of DC the use of a fail-low device in high trip applications is recommended since it produces a fewer safe detected failures and more dangerous detected ones.

Table V - Failure rates and SFF for 3144P Temperature Transmitter in RTD configuration

Failure Categories	λ_{SD} (FIT)	λ_{SU} (FIT)	λ_{DD} (FIT)	λ_{DU} (FIT)	DC	SFF
Low Trip	295	109	28	63	30,77%	87,27%
High Trip	28	109	295	63	82,40%	87,27%

Table VI - Failure rates and SFF for Temperature sensor assembly in RTD configuration

Failure Categories	λ_{SD} (FIT)	λ_{SU} (FIT)	λ_{DD} (FIT)	λ_{DU} (FIT)	DC	SFF
Low Trip	2275	109	28	83	25,23%	96,67%
High Trip	28	109	2275	83	96,48%	96,67%

In compliance with IEC 61508, 2oo3 architecture consists of three channels connected in parallel with a major voting strategy: the safety function is required in case at least two channels demand it and the system state is not changed if only one channel gives a different result which disagrees with the other two channels.
The necessary assumption for PFD assessment are shown below:

- Logic solver can detect both over-scale and under-scale currents;
- Sensor assembly made of sensing device and temperature transmitter;
- Sensing device is a resistance temperature detector (RTD) with fail safe state set as fail-low;
- 2-out-of-3 redundant architecture;
- Low stress environment;
- Low demand mode of operation;
- $\lambda_D = 0,5 \cdot 10^{-7}$;
- $\beta = 10\%$, $\beta_D = 5\%$;
- $DC = 90\%$;
- $MTTR = 8h$;
- 1 year proof test interval.

$$\lambda_D = \lambda_{DD} + \lambda_{DU} = \left(28+83\right)\cdot 10^{-9} = 1{,}11\cdot 10^{-7}\, failure\,/\,h \tag{190}$$

$$t_{CE} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{2} + MTTR\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR = 3{,}28\cdot 10^3 h \tag{191}$$

$$t_{GE} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{3} + MTTR\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR = 2{,}19\cdot 10^3 h \tag{192}$$

The average PFD for 2oo3 architecture is:

$$PFD_{avgSE} = 6\left(\left(1-\beta_D\right)\lambda_{DD} + \left(1-\beta\right)\lambda_{DU}\right)^2 t_{CE} t_{GE} + \beta_D \lambda_{DD} MTTR + \beta \lambda_{DU}\left(\frac{T_1}{2} + MTR\right) = 3{,}68\cdot 10^{-5} \tag{193}$$

5.6.3.2. Redundancy in SIS

In the safety loop taken into account, the first stage is composed by three sensors in redundant architecture following 2-out-of-3 logic.

Single Rosemount 3144P Temperature Indicator Transmitter has a dangerous undetected failure rate of ~0.001 /year. This means that 1 in 1000 devices in this application will experience a dangerous and hidden failure every year.

The introduction of a redundant architectures (such as 1-out-of-2) can mitigate the risk associated to this dangerous event in case failures have no common cause: in 1oo2 configuration the process keeps operating only if process conditions are considered safe by both transmitters.

With redundancy risk reduction is achieved but, at the same time, the risk of spurious or unnecessary trip increases: in fact each transmitter can cause a spurious trip so 1oo2 architecture improves safety but reduces availability.

In a 2oo2 arrangement with devices in series the process keeps operating if either transmitter considers the system in a safe state; the risk of a dangerous failure redoubles due to the introduction of the second device but at the same time availability improves.

2oo3 is the best architecture for this kind of applications because both safety and availability improve.

However the transmitter itself is only one of all the causes that contribute to total risk, some other are the following: electrical noise (due to coating), material compatibility, environmental conditions, extreme processes and temperatures, installation or maintenance errors.

The real enemy of redundancy is common cause failures: if any of the conditions listed above affect more than one sensor, that is a common cause condition that nullifies redundancy benefits.

So redundancy may be a great enhancement for field devices but before introducing additional components all common causes must be taken into account during design phase. Therefore diversity is better than quantity.

Safety improvement should follow these steps:

- Improve common cause strength;
- Use diversity;
- Use diagnostics;
- Add redundancy.

Each step is associated with improvements in technology and installation/maintenance practices; for example, in order to improve resistance to common cause failures in high stress environments due to high temperatures, a designer has different solutions to take:

- Improve strength using more robust devices or installing them far from the heat source;
- Use diagnostics using transmitters capable to predict impending failures;
- Add redundancy with one or more backup devices;
- Use diversity to choose backup technology more resistant to high stresses.

Obviously each choice is a trade-off between safety and costs: to take these decisions designers should select the best safety improvement taking into account not only data provided by suppliers (usually validated in a laboratory environment) but considering real-world installed safety (which is always much worse).

Only by quantifying installed safety designers can evaluate the real world safety and cost impact of specific technology.

5.6.3.3. Benefits of Diagnostics on SIS

Both redundant and non-redundant repairable control systems have improved availability and safety in case on-line diagnostic is provided. Other benefits are the reduction of time the system operates in dangerous and degraded (not completely operational) mode.

Safety is improved by diagnostic coverage even in a non-redundant architecture.

In a normally energized safety protection application, if a standard 1oo1 PLC architecture fails with outputs de-energized, the process is inadvertently shut down (false trip). Usually to detect a process shut down is not required on-line diagnostics because a false trip is usually quite apparent. However, if 1oo1 PLC fails with output energized, it cannot respond to demand in case of danger. The process keeps operating with no safety protection and there is no indication that something is faulty.

The main added value of diagnostics is the detection of dangerous failures to allow a quick repair and restore of the system [81].

In case of failure in a redundant architecture (e.g. 1oo2 PLC configuration) diagnostics reduces the time spent in the degraded mode: the output of PLC modules is wired in series so if one module fails, the other can still provide a safety protection function e.g. energizing the load (in a normally energized protection application).

So diagnostics improve the safety of this architecture because if one module fails dangerously, the system is degraded and a second dangerous failure is required to cause the system to fail. At the same time, diagnostic capability will also allow quick repair and minimize the amount of time the system operates in a degraded mode [81-86].

5.6.3.4. A2M and A3M Architectures

The most important redundant architectures used in Oil&Gas applications are A2M and A3M that differs for the number of devices involved (two or three respectively).

A2M architecture with on-board diagnostics is based on two signals: the process measurement and a dedicated boolean variable "unhealthy".

The measurement assumes values inside the 4-20mA range, otherwise it is considered out-of-range (this condition is associated to the corresponding state "OUT OF RANGE"). "UNHEALTHY" signal assumes two values, false or true, and it communicates sensor status (working or fault respectively) to the logic solver.

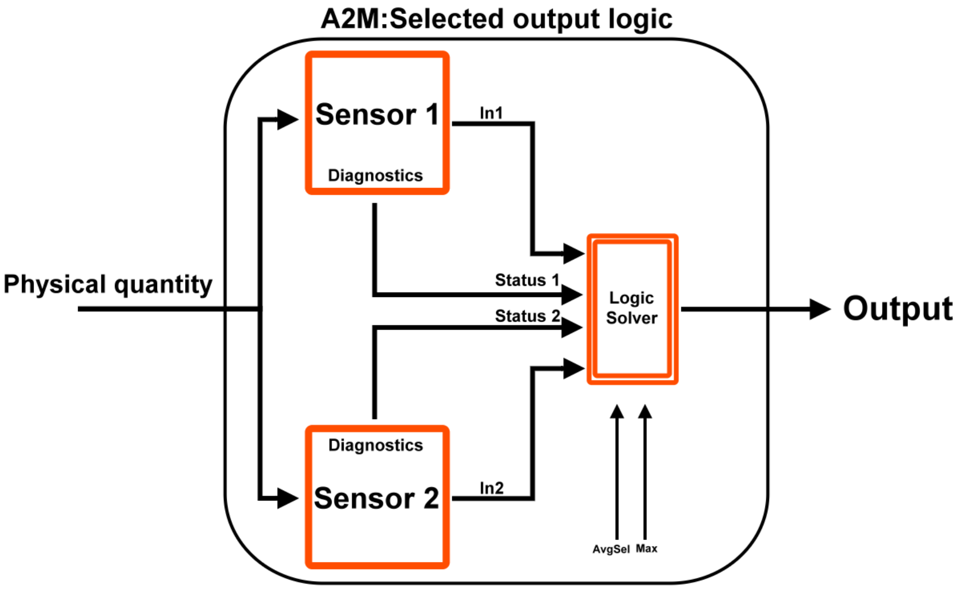


Fig.13. A2M architecture

Table VII - Sensor status accordingly to unhealthy and out-of-range signals

UNHEALTHY	OUT OF RANGE	STATUS
FALSE	FALSE	**OK**
FALSE	TRUE	**FAIL**
TRUE	FALSE	**FAIL**
TRUE	TRUE	**FAIL ALL**

The logic solver calculates average, maximum or minimum of the measurements received from field sensors depending on the values assumed by AVGSEL (average) and MAX (maximum) pins. Furthermore LS checks the spread between the two measurements (Spread=|In1-In2|): "high spread" is a boolean variable that is true in case the difference between input signals exceeds a fixed threshold.

Table VIII - Logic solver output for A2M

Fault management					**Fault tolerant**	**Reliability model after 1° failure**	**Reliability model**
Number of working sensors	**AVGSEL**	**MAX**	**HIGHSPREAD**	**OUTPUT**			
2	no	ND	False	AVG(Ini,Inj)	**No**	**ND**	**2oo2**
	1	1	True	MAX(Ini,Inj)			
		0	True	Min(Ini,Inj)			
	0	1	True/False	MAX(Ini,Inj)			
		0	True/False	Min(Ini,Inj)			
1	si	1oo1	ND	Ini	**Yes**	**1oo2**	**1oo2**

High spread signal is "ND – not defined" in case the measurements are not valid and the spread is not achievable.
"Default" status produces a pre-set output in case both measurements are not trustworthy.
A2M logic may be used in either 1oo2 or 2oo2 architecture depending on the number of sensors required by the control panel to execute the safety loop.

In A2M architecture without on-board diagnostics, out of calibration and stack in-range are detectable only in presence of on-board diagnostics since the logic solver by itself only compares sensor outcomes with predefined thresholds. Anyway out-of-range and drift are the most common failure modes of field sensors.

Drift is a gradual and incremental signal trend towards the upper (or lower) limit of operative range; in absence of on-board diagnostics and corresponding HEALTY/UNHEALTY signal (only OUT OF RANGE is provided), control logic cannot be aware of failure occurrence until device output goes out of range. Drift can occur very slowly and that period of time is rather critical because control logic is using wrong values coming from a faulty sensor.
Obviously drift is more critical in case it develops moving away from the monitored threshold.

A3M architecture with on-board diagnostics is based on three sensors; "UNHEALTHY" and "OUT OF RANGE" signals, mode of operation and measurement management are the same used in A2M.

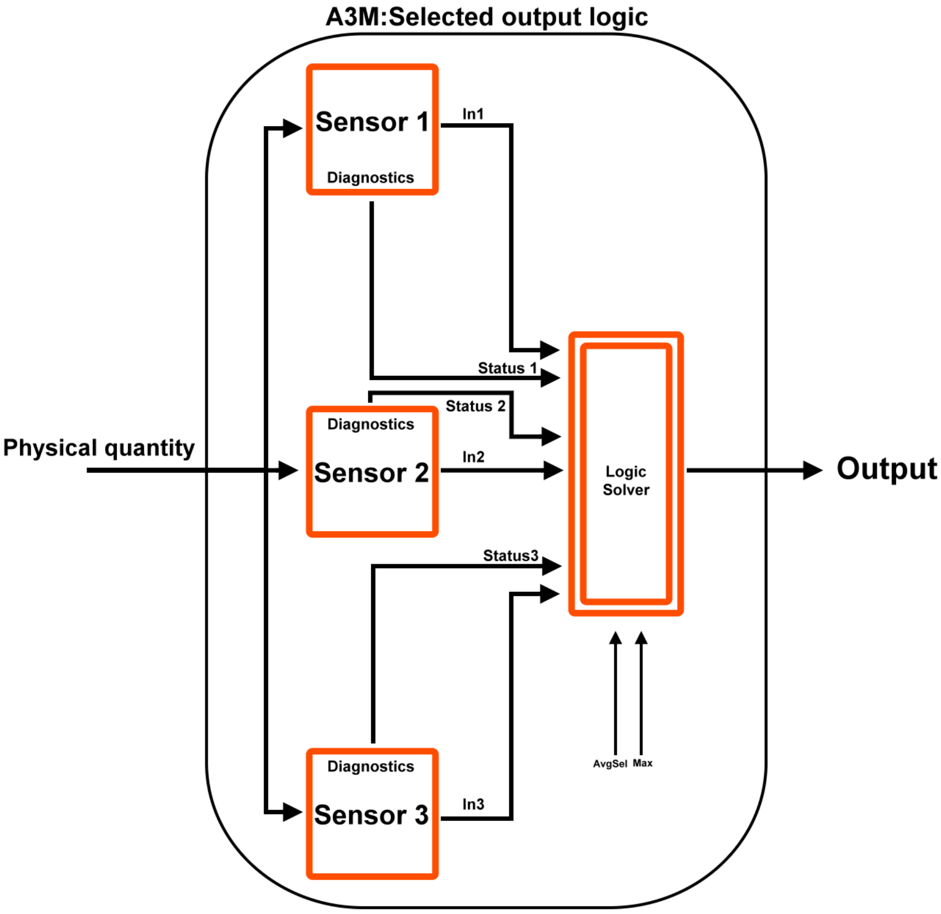


Fig.14. A3M architecture

Table IX - Logic solver output for A3M

Fault management					Fault tolerant	Reliability model after 1° failure	Reliability model
Number of working sensors	**AVGSEL**	**MAX**	**HIGHSPREAD**	**OUTPUT**			
3	1/0	1/0	True/False	MEDIAN(In1,In2,In3)	**No**	**ND**	**3oo3**
2	1	1/0	False	AVG(Ini,Inj)	**Yes**	**2oo2**	**2oo3**
	1	1	True	Max(Ini,Inj)			
		0	True	Min(Ini,Inj)			
	0	1	True/False	Max(Ini,Inj)			
		0	True/False	Min(Ini,Inj)			
1	1/0	1/0	ND	Ini	**Yes**	**1oo2**	**1oo3**

The mode of operation of A3M architecture without on-board diagnostics is the same of A2M without diagnostics: in fact both out of calibration and stack in-range are undetectable and in absence of on-board diagnostics only "OUT OF RANGE" signal is provided so control logic cannot be aware of failure occurrence until device output goes out of range.

A3M architecture without on-board diagnostics is more robust in terms of drift failures if compared with A2M: median calculation takes the central value excluding the drifting input that comes from the faulty sensor.

5.6.4. Logic Solver Stage

The logic solver is the second stage of the safety loop. In a safety instrumented system it provides the intelligence to take any decision and perform other functions such as comparison, filtering, averaging, etc.

There are three safety logic architectures that can be used to assess safety control:

- Discrete safety using dedicated relay and micro controller: it is a simple solution that requires simple installation but it is suitable for small applications (single control zone) and it offers a low safety level; diagnostics on the relay configuration is easy.
- Modular and programmable safety using modular relay or controller: modular and expandable relay systems are capable of multiple zone control whereas safety controllers use a dedicated network; these are low to medium cost solutions, they requires simple installation and offers mid safety level.

- Integrated safety using safety PLC: safety PLC solutions are large, complex and distributed; they are flexible, expandable, they offers easy diagnostics using HMI, multiple zones of control and high safety level.

The best safety system solution can be chosen following the safety life cycle:

- Hazard or risk assessment (identify hazards and estimate the associated risk);
- Functional safety system requirements (based on risk assessment, system performance, applicable standards);
- Design and verification (system architecture and safety critical circuit deisgn, validation protocol);
- Installation and validation (final site assembly, commissioning and final risk assessment validation);
- Maintain and improve (verify that system requirements operate within specified parameters for production and safety purposes; preventive maintenance set up and system upgrades).

In complex systems used Oil&Gas applications the best solution is obviously an integrated safety system using PLCs.

In these systems the logic solver task is to evaluate input signals coming from field sensors, determine if a potentially hazardous condition exists and energize or de-energize the actuators depending on the application [28].

For example, in a de-energized to trip safety system, the output de-energizes to move the process to a safe state: if any of the components in the single path fail and the output can't be de-energized, the PLC won't be able to provide the safety protection function.

Programmable Logic Controllers (PLCs) are devices that use microprocessors to handle logic control. PLCs used in safety applications are named "Safety PLCs". These devices are part of the safety system and they are designed to satisfy two important requirements: the former is avoiding to fail and, in case this condition cannot be prevented, failing only in a predictable and safe way.

Anyway there are many similarities between safety and standard PLCs: they both perform logic and math calculations, they have I/O modules to interpret signals from process sensors and actuate control final elements, furthermore they typically have digital communications ports.

The main difference is that common PLCs are not initially designed to be fault tolerant and fail-safe.

Fault tolerance is obviously fundamental in safety applications: as said before, a fault in the logic solver system must not create erroneous inputs or outputs nor prevent the system from functioning as designed.

Also the fault detection is an essential requirement to aware operators about fault location and allow on-line repair in order to avoid interruption in operation and consequent availability reduction.

Data acquisition is the first task of logic solver stage; it is the process of measuring a physical phenomenon (e.g. voltage, current, temperature, pressure, etc.) with sensors converting its physical parameters in electrical signals that are ready to be manipulated by a computer.

Therefore an acquisition system (DAQ) consists of sensors, measurement hardware and a computer with programmable software.

DAQ hardware acts as the interface between the computer and physical phenomena. The three key components of a DAQ device are signal conditioning circuitry, analog-to-digital converter and computer bus. Signal conditioning improve signal quality and manipulates them into a form that is suitable for analog-to-digital conversion (including amplification, attenuation, filtering, and isolation); since analog signals continuously vary over time, an analog-to-digital converter (ADC) is necessary to take periodic "samples" of the signal at a predefined rate and transfer them to a computer over a dedicated bus. The computer bus serves as the communication interface between the DAQ device and computer for passing instructions and measured data.

There are different logic solver architectures suitable for each application. For example, in a dual redundant architecture, the two sensors can be connected to the same data acquisition system (fanned input) or to dedicated boards (spread simplex input).

The logic solver output is usually a command to energize or de-energize the actuators and the procedure is specular to the input one described before.

In this study the logic solver is the Moore Industries® Safety Trip Alarm (STA) logic solver. This device acts on potentially hazardous process conditions in order to:

- Warn of unwanted process conditions;
- Provide emergency shutdown;
- Provide on/off control in both Safety Instrumented Systems and traditional alarm trip applications.

STA accepts signal input from transmitters, temperature sensors and a wide array of other monitoring and control instruments (e.g. current and voltage signals, resistance and potentiometer devices, etc.).

With regard to the output, the logic solver is equipped with two programmable relays used as process trip alarms and one SPDT (single pole double throw) relay used as a faulty relay.

This logic solver is used in Safety Instrumented Systems to implement one or more SIF such as shutdown fuel supply to a furnace, open a valve to relieve excess pressure, close a feed valve to prevent tank overflow, initiate release of a fire suppressant and initiate an evacuation alarm.

Component failures that cause the output relays to be de-energized are considered safe failures whereas failures that leave the relays energized are to be considered dangerous.

There are three standard architectures for the logic solver under test: high integrity, high availability and 1oo2 redundancy.

High integrity architecture offers the highest trip integrity in a non-redundant

Application: three relays are wired in series so any trip or fault alarm will execute the safety loop.

In high availability architecture the Safety Trip Alarm provides higher process or system availability; the fault alarm is wired separately from the trip relays so the safety system may be informed that a component is not able to carry out its portion of the SIF without performing the safety loop. This way the fault should be removed before the STA can provide proper safety coverage [81].

This configuration is widely used in applications where process continuity is essential and for this purpose the output process trip relays are connected in a 1oo2 configuration to trip in order to prevent from single relay failure.

The last STA architecture is a 1oo2 redundant framework: in this case if a sensor input reaches a trip condition or a fault relay is activated, the safety loop is activated. This architecture offers improved reliability of trip action and reduced vulnerability to a single failure compare to a 1oo1 architecture.

In this study three STA logic solvers are required because the first stage is a 2oo3 Temperature Indicator Transmitter architecture and one Safety Trip Alarm is required for each sensor (using 4-20mA loop); the 2oo3 vote is then performed on the STA relay output.

The necessary assumption for PFD assessment are shown below:

- Type B
- SFF of 90-99%
- HFT=1
- 2-out-of-3 redundant architecture;
- $\beta = 10\%$, $\beta_D = 10\%$;
- $MTTR = 8h$;
- 1 year proof test interval

Table X - Failure rates for Safety Trip Alarm

Device	λ_{SD} (FIT)	λ_{SU} (FIT)	λ_{DD} (FIT)	λ_{DU} (FIT)	DC	SFF
Safety Trip Alarm	**0**	**660**	**170**	**86**	**66,41%**	**90-99%**

The average PFD assessment for 2oo3 architecture is the following:

$$\lambda_D = \lambda_{DD} + \lambda_{DU} = (28 + 83) \cdot 10^{-9} = 2{,}56 \cdot 10^{-7} \, failure / h \tag{194}$$

$$t_{CE} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{2} + MTTR\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR = 1{,}48 \cdot 10^3 h \tag{195}$$

$$t_{GE} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{3} + MTTR\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR = 9{,}89 \cdot 10^2 h \tag{196}$$

(197)

$$PFD_{avgLS} = 6\left(\left(1-\beta_D\right)\lambda_{DD} + \left(1-\beta\right)\lambda_{DU}\right)^2 t_{CE}t_{GE} + \beta_D\lambda_{DD}MTTR + \beta\lambda_{DU}\left(\frac{T_1}{2} + MTR\right) = 3{,}83\cdot 10^{-5}$$

5.6.5. Final Element Stage

SIF applications can be very different depending on the monitored processes. Each SIF application requires a dedicated type of actuator and some examples of applications are the following: system shutdown in a hazardous chemical plants, valve opening due to over pressure, control switch on/off to prevent tank overflow, furnace fuel supply shutdown, evacuation alarm initialization, fire suppressant release, etc.

For this reason different devices are used as final elements in safety instrumented functions: applications may require annunciation devices (e.g. horns, flashing lights or sirens), simple devices such as relays, motor controllers and solenoid valves or more complex systems [Safety Instrumented Systems Verification: Practical Probabilistic Calculations].

However in the process industries the most common final element is a remote actuated valve consisting of:

- Pneumatic or hydraulic control assembly such as three-way solenoid, a smart partial valve stroke box or a complex electro-pneumatic assembly;
- Actuators that are defined by the power source (electric, hydraulic or pneumatic) and range of motion (linear, partial turn or multi-turn).
- Valves such as ball, butterfly, offset butterfly, gate, globe and other special designs.

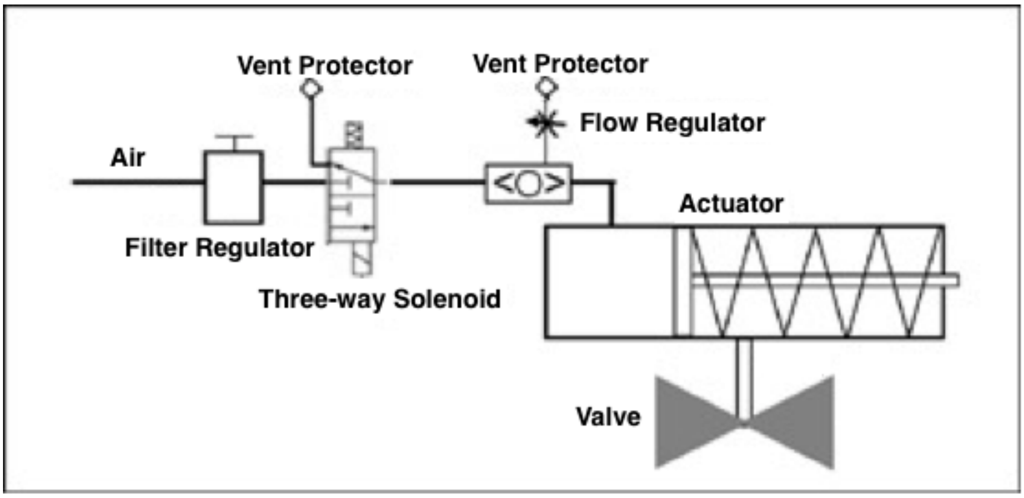


Fig.14. Remote Actuated Valve Assembly

Process material, pressures, temperatures and flow rates obviously have a deep impact during the selection of the type of valve. Design engineers must choose remote actuated valves very carefully to match process requirements taking into account materials of construction, valve seat material, valve type, actuator type and controls characteristics. Actuators are dedicated to convert power to motion and, as said before, they are defined by the power source and range of motion.

For example, hydraulic and pneumatic piston and diaphragm actuators provide a

linear output and may be integrated with a crank arm mechanism (quarter turn output) or a rack and pinion mechanism (full turn output).

Electric actuators have output ranges similar to the hydraulic and pneumatic motor drives but, on the other hand, they are more complex due to the additional functionality provided.

In any case, the most important feature of actuators involved in safety applications is their fail-safe attitude: diaphragm and piston actuators driven by hydraulic and pneumatic power have usually spring return valves so in case of failure the pressure source automatically drives the actuator to its safe position.

With electrically powered actuators there is no solution to provide fail-safe functionality: nowadays there are some partial turn electric actuators that have a spring return in case of total loss of power [86].

Anyway during design stage the choice of the best combination od actuator and valve is critical to achieve the best optimization of fail-safe characteristics.

5.6.5.1. Redundant Control System

In this study the final element under analysis is a pneumatically actuated block valve controlled by ASCO® Redundant Control System (RCS): it is an electro-mechanical and pneumatic system consisting of two solenoid valves and one pneumatic valve.

Three pressure switches are provided on each valve for diagnostic purpose to monitor the pneumatic pressures at critical points of the RCS assembly: switches are required to confirm the proper position of the valve.

In this device both automatic and manual diagnostic tests can be implemented to achieve the safety ratings.

RCS achieves a high level of process safety and reliability thanks to a fault tolerant architecture, high diagnostic coverage, and automated testing procedures.

The RCS is connected to the safety rated logic solver that is actively performing the safety function as well as automatic diagnostics designed to diagnose potentially dangerous failures within the RCS.

In compliance with IEC 61508 for safety assessment Redundant Control System is considered part of the final element together with the controlled block valve.

Depending on the protected process, the safety action of a block valve can either be spring return open or close. The spring block valve actuator may receive air supply or be vented in order to move the block valve to the safe-state “normally open” (NO) or “normally closed” (NC). In “double acting” (DA) valves the piston receives air to one side and it is vented on the other to move the block valve to the safe-state.

So there are different configurations for RCS assembly: in normally-closed version, RCS is used to vent air from a spring-forced actuator if the solenoids are de-energized while in normally-closed version it is used to supply air to a spring-forced actuator if the solenoids are de-energized.

The choice between these versions directly impacts the PFD of the entire SIS: a failure in the Redundant Control System will prevent the proper working of the block valve and decrease safety integrity.

The selection of normally open/closed version is based on the spring forced state of the controlled actuator. Many safety applications require vented condition (spring forced position) as the block valve actuator safe-state while other ones require safe-state in pressurized condition and unforced spring position.

In this application the safe-state is achieved with de-energized signals so at least one of the two solenoid valves has to be energized to prevent the block valve from moving to the safe state. The pressure switch contacts are normally open so they are closed in presence of pressure.

So when de-energized, RCS moves to the fail-safe position (NC or NO) and air will be supplied or vented depending on the application; in DA version, de-energy command will simultaneously supply air to one side of the cylinder and vent the opposite one [88-89].

Figure 15 shows the RCS functional block diagram in NC configuration: SOV_1 and SOV_2 are solenoid valves and B/P is the bypass valve (pneumatically controlled).

Bypass valve is used to apply pneumatic supply directly to the block valve in order to force it to remain in the normal condition (not safe state, maintenance override), while isolating and venting solenoid valves and all three pressure switches.

In Fig. 15 both solenoid valves are de-energized so air is vented from the block valve actuator and the spring return actuator moves the block valve to the safe state.

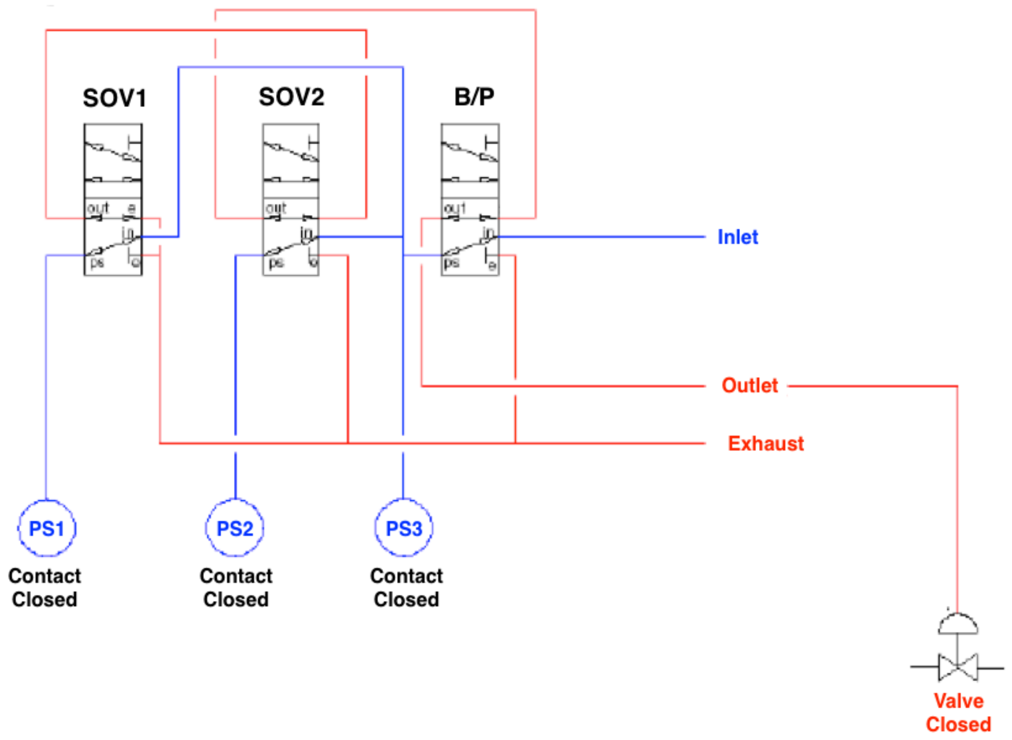


Fig.15. SIS – Functional Block Diagram of RCS in NC configuration

Figure 16 shows the RCS functional block diagram in NO configuration: SOV_1 and SOV_2 have to remain de-energized in order to keep block valve in the safe state position.
In Fig. 16 both solenoid valves are de-energized so air is supplied to the block valve actuator and actuator spring return is overcame to move the block valve to the safe state.

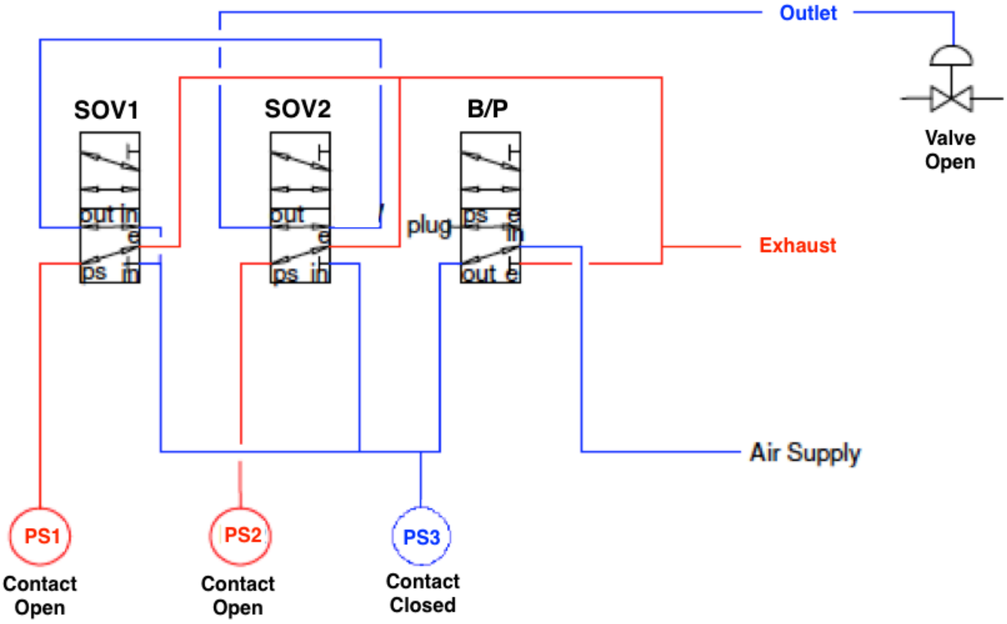


Fig. 16. SIS – Functional Block Diagram of RCS in NO configuration

RCS system has two different operational mode of operation:

- 2oo2D: in this mode both solenoids must de-energize for shutdown; the pressure switches are used to individually alarm in case one of the solenoid valves goes to the vent state when not commanded.
- 1oo1HS:I n this mode only one solenoid valve is on-line during normal operation. Any spurious trip of the on-line solenoid valve is detected by the logic solver using signals coming from the associated pressure switches; in response to spurious trip the logic command to energize the second solenoid valve in order to maintain air supply to the block valve.

  With this configuration, RCS achieves the safety availability of a 1oo1 solenoid valve and the reliability of a 2oo2 voted solenoid operated valve configuration.

Since the RCS architecture is not sufficient to achieve the required diagnostic coverage for devices used in critical environments, it is equipped with three pressure switches in order to verify the system transitions into the safe state (on demand), detect illegal and degraded states of the system and detect the bypass (forced) state of the safety function.
Any failure detected by the ADT shall be annunciate by the safety rated logic solver.

Tables I shows failure rates for the ASCO RCS with Automated Diagnostic Tests: the failure rates are valid for the entire useful life of the devices expected to be 10 years in accordance with manufacturer endurance tests and general field failure data.

Table XI - Failure rates for ASCO RCS with Automated Diagnostic Tests

Device	**λ_{SD} (FIT)**	**λ_{SU} (FIT)**	**λ_{DD} (FIT)**	**λ_{DU} (FIT)**
Solenoid Valve	**594**	**216**	**502**	**10**
Bypass Valve	**57**	**88**	**7**	**0**
Pressure switch	**444**	**5**	**0**	**0**

Supposing RCS as the only final element, the design can met SIL 3 with HFT = 0 based on SFF > 90%; anyway in this study the final element subsystem includes also the controlled block valve.

The necessary assumption for PFD assessment are shown below:

- $\beta = 1\%$
- ADT = 24h
- MTTR = 24h
- Fast switch between solenoid valves not to cause a trip of the block valve
- 1oo1HS mode of operation

Using Markov modelling the average PFD for RCS with ADT is the following:

$PFD_{avg} = 1{,}24 \cdot 10^{-4}$ considering 1 year proof test interval;

$PFD_{avg} = 2{,}12 \cdot 10^{-4}$ considering 2 years proof test interval;

$PFD_{avg} = 3{,}00 \cdot 10^{-4}$ considering 3 years proof test interval.

Tables XII shows failure rates for the ASCO RCS with Manually Initiated Diagnostic Tests: the failure rates are valid for the entire useful life of the devices expected to be 10 years in accordance with manufacturer endurance tests and general field failure data.

Table XII - Failure rates for ASCO RCS with Manually Initiated Diagnostic Tests

Device	λ_{SD} (FIT)	λ_{SU} (FIT)	λ_{DD} (FIT)	λ_{DU} (FIT)
Solenoid Valve	**0**	**855**	**0**	**512**
Bypass Valve	**0**	**145**	**0**	**7**
Pressure switch	**0**	**449**	**0**	**0**

Supposing RCS as the only final element, the design can met SIL 2 with HFT = 0 based on SFF > 60%.
The necessary assumption for PFD assessment are shown below:

- $\beta = 1\%$
- Manual diagnostic interval = 24h
- Fast switch between solenoid valves not to cause a trip of the block valve
- 1oo1HS mode of operation

Using Markov modelling the average PFD for RCS with Manually Initiated Diagnostic Tests is the following:

$PFD_{avg} = 1{,}11 \cdot 10^{-4}$ considering 1 year proof test interval;

$PFD_{avg} = 1{,}99 \cdot 10^{-4}$ considering 2 years proof test interval;

$PFD_{avg} = 2{,}87 \cdot 10^{-4}$ considering 3 years proof test interval.

In this study RCS subsystem is provided with Automatic diagnostic tests so considering 1 year of proof test interval the average probability of failure on demand is: $PFD_{avg} = 1{,}24 \cdot 10^{-4}$.

5.6.5.2. Controlled Valve

The second item to take into account in the final element stage is the controlled block valve.
A valve involved in SIS applications has different features from a standard valve used in basic process control systems where life cycle and number of repetitive operations are the only requirements; furthermore the type of actuator impacts the distinction between safe and dangerous failures.
The dangerous failure modes can be divided in two categories: failure to move to safe position and failure to seal upon reaching safe position [87-90].
Failure to move to safe position can occur due to two mechanisms:

- Binding: this failure may occur between the closure member and the seat (depending on the amount of contact the two surfaces maintain) or between the stem and the stem bore (stems with large surface contact area will have significantly more binding failures). Different types of valves suffer some failure mechanisms more than others: for example, binding failure have great incidence on ball valves, fewer on butterfly ones and no effect in globe valves. On the other hand, the linear stem of a globe valve has a higher binding risk than the quarter turn stem of a ball one.
- Breakage: some valve weakness may manifest when there are increased operating loads in the valve. This is more frequent in valves with long stems or more point loads in the design (e.g. butterfly valves).

Failure to seal upon reaching safe position corresponds to a leakage on completion of stroke typically caused by damage to the seat or by solids holding the seat and closure member apart.

This varies by design and application and on the type of solids involved; obviously valves with more closure member to seat contact have more opportunity to manifest this kind of damage.

In this study the final element is a ball valve with floating ball design (Abc. X Series Ball Valve): the safety function is to move to the designated safe position within the required time.

Table XIII and Table XIV show in clean and severe service respectively the failure rates for the equipment under test with and without Partial Valve Stroke Tests (this topic is developed in Appendix II).

The valve operation is divided as follows:

- Close on trip: the valve is closed (full stroke) or the valve is closed and sealed with leakage no greater than the defined leak rate (tight-shutoff);
- Open on Trip: the valve is open.

Table XIII - Failure rates for ball valve w/o PVST in clean service

Failure category	Failure rate (FIT) without PVST			Failure rate (FIT) with PVST		
	Close on trip		Open on trip	Close on trip		Open on trip
	Full stroke	Tight-shutoff		Full stroke	Tight-shutoff	
λ_{SD}	0	0	0	0	0	172
λ_{SU}	0	0	172	0	0	0
λ_{DD}	0	0	0	149	149	149
λ_{DU}	479	1370	307	330	1221	158
Residual	931	40	931	931	40	931

Table XIV - Failure rates for ball valve w/o PVST in severe service

Failure category	Failure rate (FIT) without PVST			Failure rate (FIT) with PVST		
	Close on trip		Open on trip	Close on trip		Open on trip
	Full stroke	Tight-shutoff		Full stroke	Tight-shutoff	
λ_{SD}	0	0	0	0	0	317
λ_{SU}	0	0	317	0	0	0
λ_{DD}	0	0	0	259	259	259
λ_{DU}	858	2615	541	599	2356	282
Residual	1797	40	1797	1797	40	1797

In according to IEC 61508 [88] the residual failures are not included in the SU category: in fact these failures will not affect system reliability or safety, and should not be included in spurious trip calculations.

A word of clarification, the distinction "detected/undetected" is associated just with diagnostic tests (and corresponding DC factor); PVST coverage concerns only dangerous failures that are undetected by the diagnostics and that may be revealed by stroke tests. In other words, PVST is a procedure of detection of a part of normally undetected dangerous failures in absence of partial valve stroke testing. Since there are no other diagnostic tests implemented in the equipment under analysis, in Table I and Table II the failure rate of failures detected by Partial Valve Proof Tests fall into dangerous detected class [87-90].

Since the PVST is put into practice, it is necessary to follow a different procedure to assess the average PFD.

Partial Valve Stroke Test is assumed to be automatically performed at least an order of magnitude more frequent than the Full Valve Proof Test.

As a result, when PVSTs are performed at regular intervals, the ball valve contributes less to the overall PFD_{avg} of the Safety Instrumented Function .

A complete functional test of the valve can be viewed as consisting of two parts: the partial-stroke (PVST) and the full-stroke (FVST); for detail about this practice see Appendix II.

Since the ball valve under analysis is provided of partial stroke testing, and the necessary assumptions and the procedure to assess the average PFD are the following:

- Mission time of 10 years;
- MTTR = 96 hours;
- 2 months partial proof test interval (1460h);
- 6 months full proof test interval (4380h);
- Close on trip;
- Full stroke operation in clean service.

In this study the fraction of dangerous undetected failures that are detected by the Partial Valve Proof Test is considered as a contribution to the rate of dangerous detected failures: for this reason in Table I the Dangerous Detected failure rates in case of PVST implementation correspond to the fraction of dangerous undetected failures detected by the partial valve stroke test.

No other diagnostic tests are implemented so in case of close on trip operation (full stroke) in clean service:

$$\lambda_{DU,PVST} = \lambda_{DD} \tag{198}$$

$$PC = \frac{\lambda_{DU,PVST}}{\lambda_{DU}} = \frac{\lambda_{DD}}{\lambda_{D}} = 31\% \quad 1 - PC = \frac{\lambda_{D} - \lambda_{DD}}{\lambda_{D}} = 69\% \tag{199}$$

$$\lambda_D = \lambda_{DD} + \lambda_{DU} = (149 + 330) \cdot 10^{-9} = 4{,}79 \cdot 10^{-7} \mathit{failure/h} \tag{200}$$

$$\begin{aligned} PFD_{avgBV} &= PFD_{FVST} + PFD_{PVST} \cong (1-PC)\frac{\lambda_D \cdot \tau_{FVST}}{2} + PC\frac{\lambda_D \cdot \tau_{PVST}}{2} = \\ &= \frac{(\lambda_{DU} - \lambda_{DD}) \cdot \tau_{FVST}}{2} + \frac{\lambda_{DD} \cdot \tau_{PVST}}{2} \end{aligned} \tag{201}$$

So the average PFD for 1oo1 architecture is:

$$\begin{aligned} PFD_{avgBV} &= PFD_{FVST} + PFD_{PVST} \cong (1-PC)\frac{\lambda_D \cdot \tau_{FVST}}{2} + PC\frac{\lambda_D \cdot \tau_{PVST}}{2} = \\ &= \frac{(\lambda_{DU} - \lambda_{DD}) \cdot \tau_{FVST}}{2} + \frac{\lambda_{DD} \cdot \tau_{PVST}}{2} = 5{,}05 \cdot 10^{-4} \end{aligned} \tag{202}$$

The average probability of failure on demand of the whole final element stage, considering one-year proof test interval is:

$$PFD_{avgFE} = PFD_{avgRCS} + PFD_{avgBV} = 6{,}29 \cdot 10^{-4} \tag{203}$$

5.6.6. Safety Loop PFD Assessment

This paragraph contains a summary to recap all the requirements and specifications of the complete safety loop under analysis.

1st stage contains the sensing elements Rosemount® 3144P HART Temperature Indicator Transmitter (TIT) and the necessary assumptions for the safety assessment are listed below:

- RTD temperature-sensing device;
- 2-out-of-3 redundant architecture;
- Temperature transmitter and sensing device are in series (failure rates can be added);
- Temperature Transmitter is programmed to drive its output low (low-trip) on detected failure, fail safe state set as fail-low (under-range);

- Logic solver can detect both over-scale and under-scale input (so both fail-high and fail-low are detectable);
- "No effect" failures (with no effect on the safety function) and "annunciation undetected" failures (not directly impact the safety but influence the ability to detect a future fault) are classified as safe undetected;
- Low stress environment;
- Low demand mode of operation;
- Low stress environment;
- Low demand mode of operation;
- $\lambda_D = 0{,}5 \cdot 10^{-7}$;
- $\beta = 10\%$, $\beta_D = 5\%$;
- $DC = 90\%$;
- $MTTR = 8h$;
- 1 year proof test interval.

The average PFD for 2oo3 architecture is:

(204)

$$PFD_{avgSE} = 6\left(\left(1-\beta_D\right)\lambda_{DD} + \left(1-\beta\right)\lambda_{DU}\right)^2 t_{CE} t_{GE} + \beta_D \lambda_{DD} MTTR + \beta \lambda_{DU}\left(\frac{T_1}{2} + MTR\right) = 3{,}68 \cdot 10^{-5}$$

2nd stage contains the logic solver Moore Industries® Safety Trip Alarm (STA) and the necessary assumptions for the safety assessment are listed below:

- 2-out-of-3 redundant architecture;
- Type B;
- SFF of 90-99%;
- HFT=1;
- 2-out-of-3 redundant architecture;
- $\beta = 10\%$ $\beta_D = 10\%$
- $MTTR = 8h$;
- 1 year proof test interval.

The average PFD for 2oo3 architecture is:

(205)

$$PFD_{avgLS} = 6\left(\left(1-\beta_D\right)\lambda_{DD} + \left(1-\beta\right)\lambda_{DU}\right)^2 t_{CE} t_{GE} + \beta_D \lambda_{DD} MTTR + \beta \lambda_{DU}\left(\frac{T_1}{2} + MTR\right) = 3{,}83 \cdot 10^{-5}$$

3rd stage contains the final elements ASCO® Redundant Control System (RCS) and Abc. X Series Ball Valve and the necessary assumption for RCS safety assessment are shown below:

- Automated Diagnostic Tests implemented;
- Safe-state with de-energized signals;
- RCS fail-safe position is "normally closed";

- Pressure switch contacts normally open, close in presence of pressure;
- Fast switch between solenoid valves not to cause a trip of the block valve;
- $\beta = 1\%$;
- ADT = 24h;
- MTTR = 24h;
- SFF > 90%;
- 1 year proof test interval;
- 1oo1HS mode of operation.

The average PFD for 1oo1 architecture is:

$$PFD_{avg} = 1{,}24 \cdot 10^{-4} \tag{206}$$

The necessary assumption for Ball Valve safety assessment are shown below:

- Mission time of 10 years;
- Clean service;
- Close on trip and full stroke operation without tight shutoff requirements;
- Partial Valve Stroke Test implemented;
- 2 months partial proof test interval (1460h);
- 6 months full proof test interval (4380h);
- MTTR = 96 hours.

The average PFD for 1oo1 architecture is:

$$\begin{aligned} PFD_{avgBV} &= PFD_{FVST} + PFD_{PVST} \cong (1 - PC)\frac{\lambda_D \cdot \tau_{FVST}}{2} + PC\frac{\lambda_D \cdot \tau_{PVST}}{2} = \\ &= \frac{(\lambda_{DU} - \lambda_{DD}) \cdot \tau_{FVST}}{2} + \frac{\lambda_{DD} \cdot \tau_{PVST}}{2} = 5{,}05 \cdot 10^{-4} \end{aligned} \tag{207}$$

The average probability of failure on demand of the whole final element stage is:

$$PFD_{avgFE} = PFD_{avgRCS} + PFD_{avgBV} = 1{,}03 \cdot 10^{-3} \tag{208}$$

The average probability of failure on demand of a safety function for the safety-related system is determined by the combination of the average probability of failure on demand for all the subsystems involved in the safety function. Average PFD can be expressed as follows:

$$PFD_{SYS} = PFD_{SE} + PFD_{LS} + PFD_{FE} \tag{209}$$

$$PFD_{avg} = PFD_{avgSE} + + PFD_{avgSL} + PFD_{avgFE} = 7{,}04 \cdot 10^{-4} \tag{210}$$

So for the system under analysis, SIL 3 range is achieved.
A reasonable division that seems to be widely accepted is 35-15-50% to the sensor, logic and final element subsystems respectively: in this application 89% to final element due to 1oo1 architecture.

Table XV – PFD safety loop assessment

Device			λ_{SD} (FIT)	λ_{SU} (FIT)	λ_{DD} (FIT)	λ_{DU} (FIT)	Common Cause Beta Factor	Diagnostic Test Coverage	Proof Test Coverage	Proof Test Interval	Mean Time To Repair [h]	Safe Failure Fraction	Type	Architecture	HFT	PFD
SE	Temperature Transmitter Assembly	RTD	1980	0	0	20	10%	x	x	6 months	8	99,00%	B	2oo3	1	3,68
		Transmitter	295	109	28	63		30,77%	90%	6 months	8	87,27%	B			
LS	Safety Trip Alarm		0	660	170	86	10%	66,41%	100%	1 year	8	90-99%	B	2oo3	1	3,83
FE	Redundant Control System	Solenoid Valve	594	216	502	10	x	x	99%	1 year	24	> 90%	A	1oo1	0	6,29
		Bypass Valve	57	88	7	0				1 year	24		A			
		Pressure Switch	444	5	0	0				1 year	24		A			
	Ball Valve		0	0	149	330	x	x	31%	2 months	96	X	A			

Chapter 6

Reliability Assessment Loop

The reliability parameter prediction is one of the most common methods to evaluate the performance of the system under analysis and this practice is broadly used in industrial applications in particular to assess design feasibility, compare design choices, identify potential failure areas, trade-off system design factors and track reliability improvements [32]. This thesis is focused on availability improvement and takes into account maintainability and, in particular, reliability roles in order to achieve this kind of target.

The goal, as said before, is to develop a procedure for availability improvement that engineers may used during the early stages of product design.
RBDesigner® is a brand new dedicated tool for Oil&Gas applications and it was developed to achieve reliability prediction in the early product design stages of thermal-hydraulic systems: these machineries (e.g. gas turbine auxiliary systems) are very complex and contain both mechanical equipment and electronic devices. The tool takes into account all these features and provides reliability feedbacks to design engineers to reduce re-design costs and time for system upgrades.

For this reasons *RBDesigner®* represents the central phase of the Reliability Assessment Loop (see Figure 1): it starts from the sketches of the thermal-hydraulic system and following three steps the Reliability Block Diagram is built and available for reliability parameters assessment. The three stages of this procedure are: automatic model generation and net-list production (XML format), semi-automatic RBD design and final reliability assessment [32].

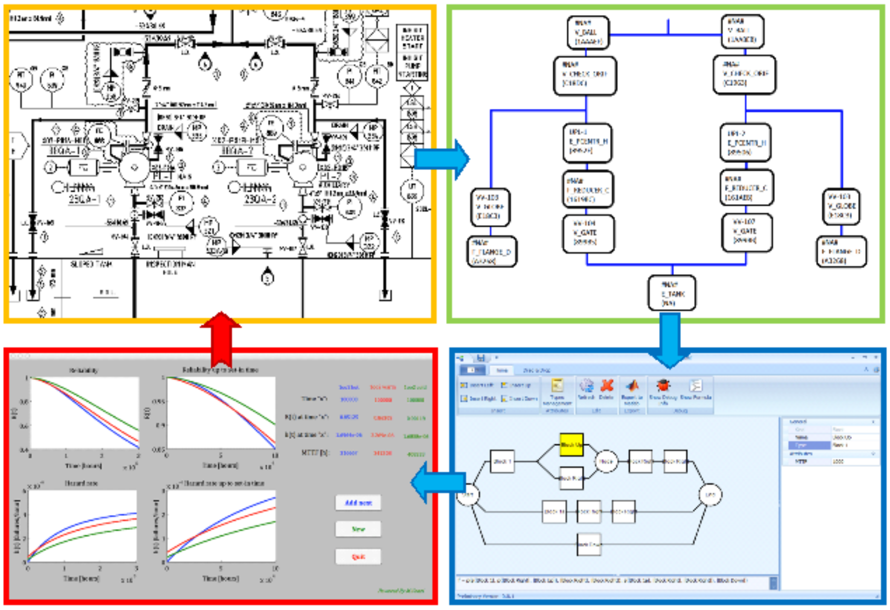


Fig. 1. Reliability assessment loop

6.1. Automatic Model Generation

The Reliability Assessment Loop starts with the “Automatic model generation”: the starting point of the procedure is a P&ID which is a diagram reproducing a thermal-hydraulic system and containing equipment, instrumentation and piping of the process flow. A P&ID is the mandatory input to achieve reliability assessment with the proposed procedure since the first stage is the generation of a Functional Block Diagram (FBD) starting from the P&ID itself.

The outcome of this procedure is a net-list in XML format containing all blocks and connections making up the system. The exported net accurately reproduces the topology of the thermal-hydraulic system and, at the same time, each block is directly associated with the corresponding on the starting sketch keeping any attribute and feature (e.g. technical information, position within the system and generic block properties where expected).

XML format was chosen for the fitting perspective to many different purposes. A supporting tool named XML Drawer was implemented to display and edit the XML diagram: this software is essential to fit possible mistakes and inaccuracies.

Finally the automatic model generation process reduces potential errors introduced by data transcriptions or human mistakes and translates P&ID projects into a universal language.

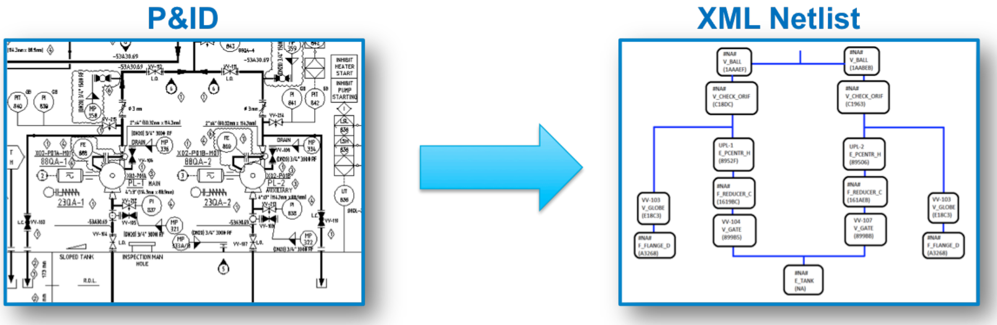


Fig. 2. Automatic net-list generation

6.2. Semi-automatic RBD Design

The second step is the semi-automatic RBD design: a Reliability Block Diagram can be created with a drag-and-drop procedure of the blocks in the net-list on XML Drawer to the diagram in *RBDesigner®*. This is the required procedure:

- Select a block on XML Drawer window;
- Drag the block to *RBDesigner®* input window;
- Drop the block in the RBD desired position.

The RBD generation is guided with structural restrictions and all the rules to raise a correct diagram are fully integrated in order to permit only suitable block arrangement and connections; in case different structural solutions are possible, multi-architectural suggestions are shown during the diagram assembly so the user would have a complete overview of design feasibility.

These features are a great support, in particular for users with little reliability experience to avoid mistakes and facilitate improvements.

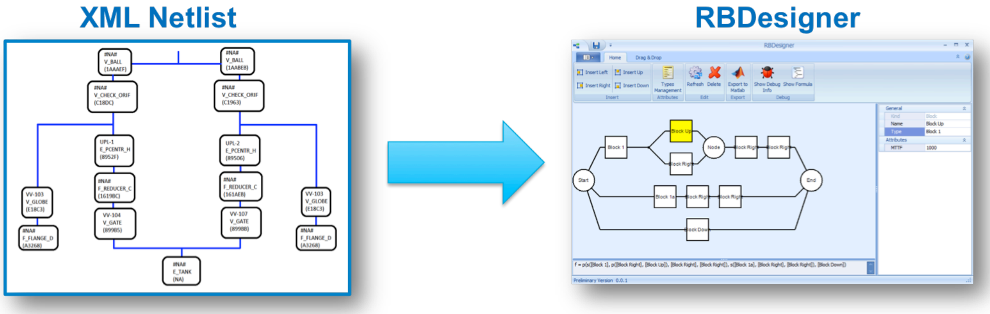


Fig. 3. Semi-automatic RBD design

Similarly as the net-list blocks, also RBD components keep all the attributes and the features of the starting sketch but at this stage the user must add some parameters (such as failure/hazard rate, MTTF, etc.) necessary to achieve the reliability prediction: this information can be manually added by users with suitable field experience or loaded from two integrated reliability databases (OREDA Handbook – Offshore Reliability Data and NSWC Handbook – Naval Surface Warfare Center [50]). In any case a default database is available and user can edit it as he pleases.

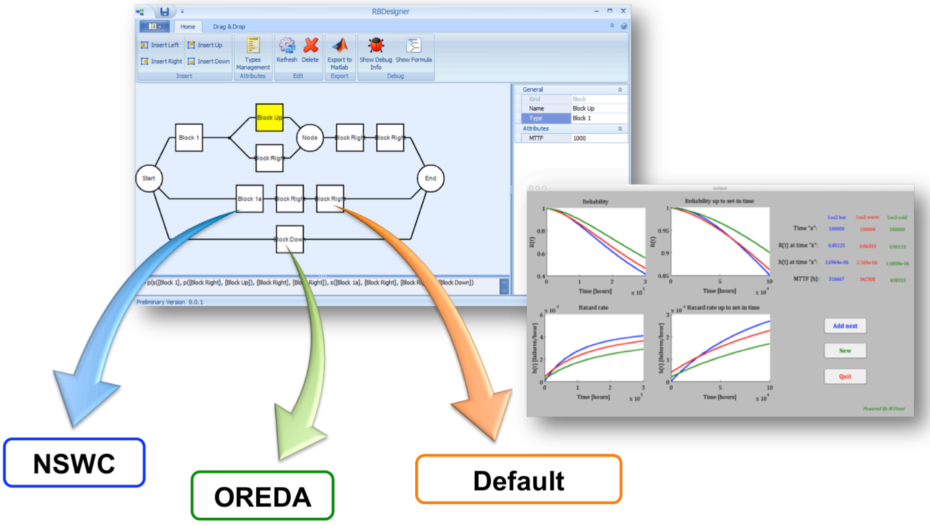


Fig. 4. Multiple database connection & Reliability Assessment

Considering the advantages above mentioned, the generation of the reliability diagram is extremely intuitive and even users without a huge knowhow can enjoy it, otherwise advanced editing procedures are provided for expert users that don't need structural restrictions or architectural suggestions.

6.3. Reliability Assessment

The last step consists in the Reliability Assessment. Once RBD generation is completed, all the information concerning diagram structure and component reliability is used to obtain a reliability prediction.

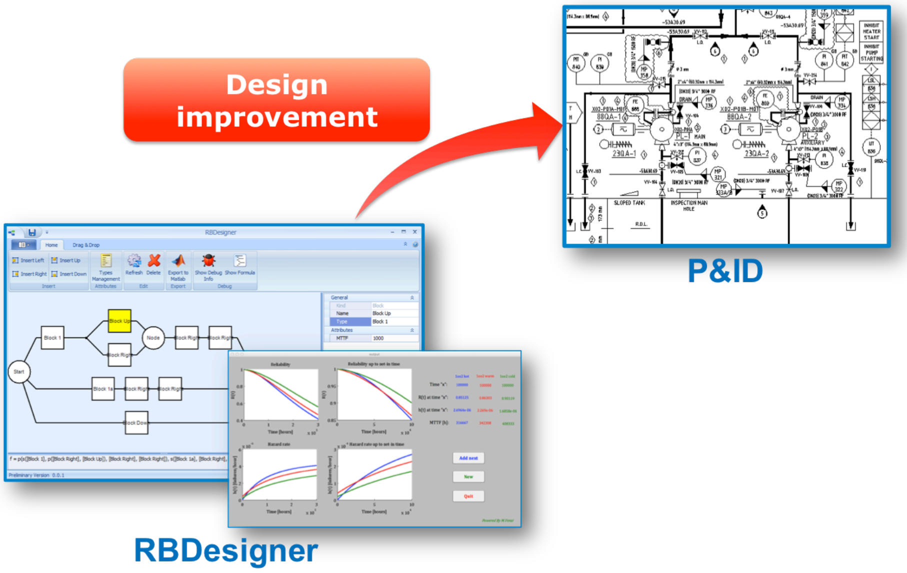


Fig. 5. Design improvement throw reliability feedback

Processing phase and output generation are achieved on *Matlab*® platform and the outputs are listed below:

- Reliability vs. time plot (up to default time, 3•10^{5}h);
- Reliability vs. time plot (up to user set-in time);
- Failure/hazard rate vs. time plot (up to default time, 3•10^{5}h);
- Failure/hazard rate vs. time plot (up to user set-in time);
- Reliability value calculated at user set-in time value;
- Failure/hazard rate value calculated at user set-in time value;
- System MTTF - Mean Time To Failure.

Once reliability parameters are achieved, this feedback is useful to take re-design actions or prove system robustness [32].

6.4. Case Study A: Mineral Lube Oil Console

A gas turbine is a “turbo-machinery”, term used in mechanical engineering to describe machinery that transfers energy between a rotor and a fluid. A gas turbine is a turbo-machinery that converts thermal energy in mechanical energy.

The standard set-up of a gas turbine is an upstream rotating compressor coupled to a combustion chamber and a downstream turbine (Figure 6). Gas turbines work in a continuous thermodynamic cycle and the basic operation is described below.

Atmospheric air flows through a compressor that brings it to higher pressure, than a fuel is added into the air to create a high-temperature flow after ignition in combustion chamber. This way the chemical energy of the air mixture (air and fuel) is converted in thermal energy.

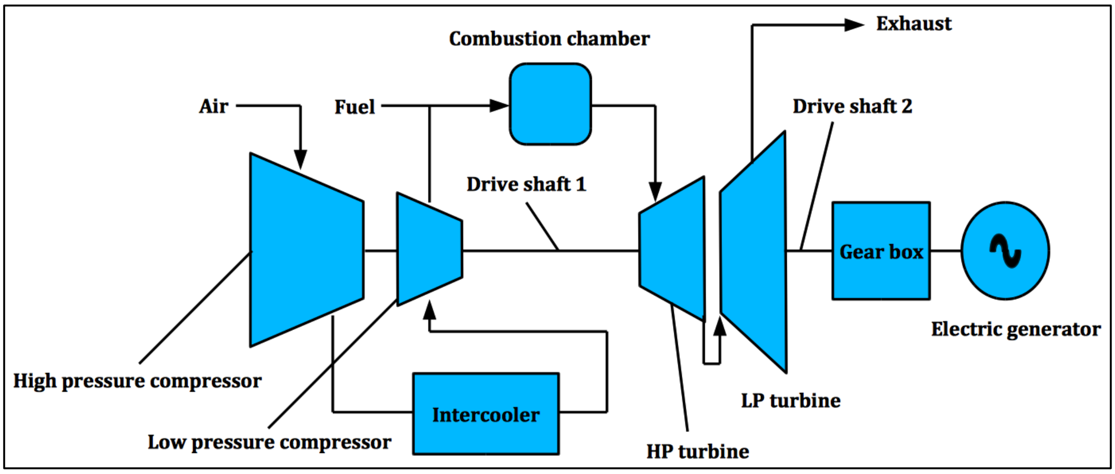


Fig. 6. Gas turbine framework

The high-temperature and high-pressure gas enters the turbine: here it expands down to the exhaust pressure and produces mechanical energy. The output of the process is the turbine shaft work that is used to drive the compressor and other devices coupled to the shaft (e.g. electric generator). The remaining energy that is not used for shaft work comes out in the exhaust gases.

The proper working of the turbo-machinery is ensured by the gas turbine auxiliary systems such as starting system, lubrication system and control system (Figure 7). One of the most important is the lubrication system, and in particular the mineral lube oil console: the mineral oil is used to reduce friction and fatigue between moving surfaces (e.g. bearings) and for this reason the efficiency of the console is critical for the proper workability of the whole gas turbine.

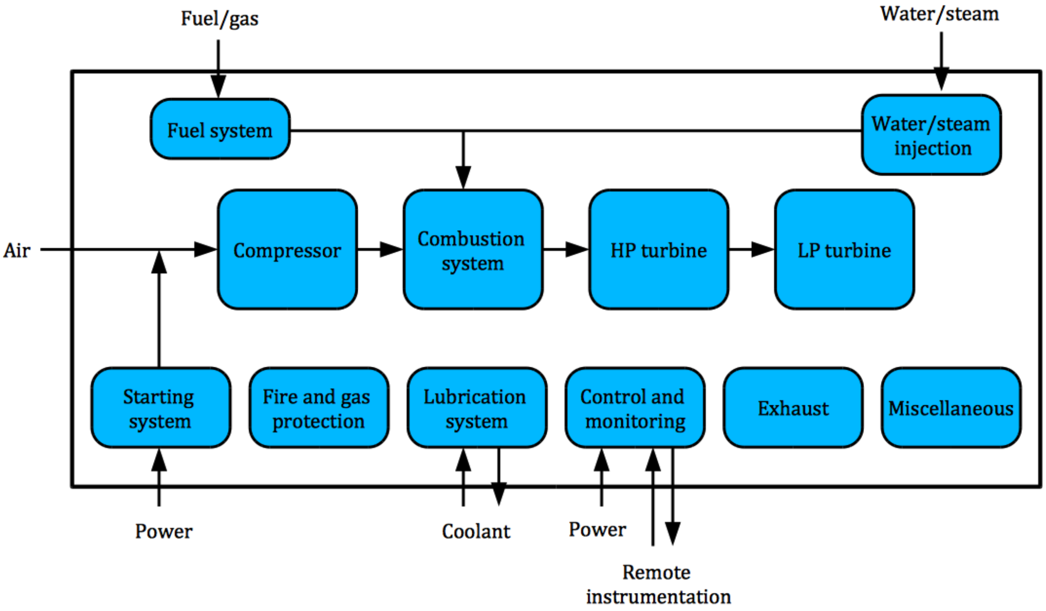


Fig. 7. Gas turbine Functional Block Diagram

All the acronyms of the blocks used in the Reliability Block Diagram (Figure 8) are listed below:

- PDIT: Pressure Differential Indicating Transmitter;
- LIT: Level Indicating Transmitter;
- PSV: Pressure Safety Valve;
- PIT: Pressure Indicating Transmitter;
- PCV: Pressure Control Valve;
- TCV: Temperature Control Valve.

In the mineral oil console under analysis the sub-system that mostly affects the whole system reliability is the pumps section containing two pumps (main and auxiliary respectively) and each pump is supplied by two electrical motors (main and standby).

These motors are in cold stand-by configuration being only main one operative; the other motor is disconnected from power supply and is activated when the main unit fails.

The pump branches, instead, can be considered both in cold or warm architecture, on the basis of the application and the response-time required in case of failure. This choice is matter of project engineer that must take this decision depending on specifications and requirements.

In order to compare reliability performance achievable with these two different architectures, "Reliability vs. time" chart (see Figure 9) was built using data from referring to aero-derivative gas turbines; the failure rates of all items and the corresponding MTTF=1/λ are shown in Table I. In Table II, instead, is shown the system reliability calculated at fixed time interval (1 year ≈ 8700h).

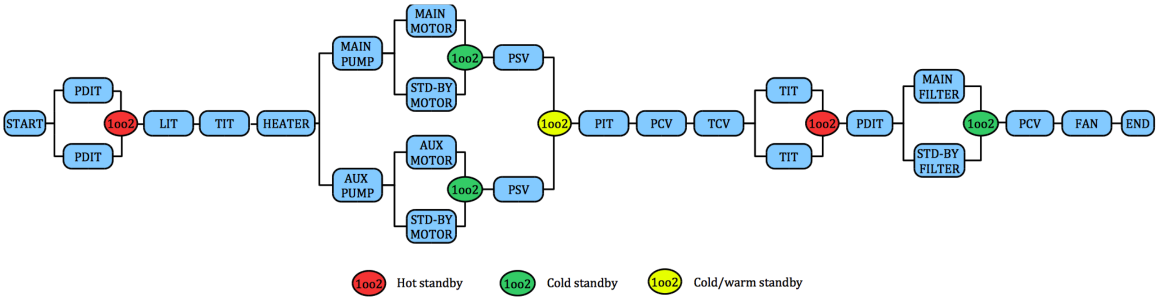


Fig. 8. Gas turbine Reliability Block Diagram

Table I – Failure Rates and MTBF

Item	Failure rate [failures/10^6h]	MTTF [h]
PDIT	0,66	1515728
PIT	0,66	1515728
LIT	2,45	408163
TIT	3,42	292398
Pump	1,54	648021
Motor	1,76	567944
PSV	0,66	1515728
TCV	3,48	287356
PCV	3,48	287356
Filter	1,98	505243
Fan	3,63	275200
Heater	1,63	613496

Table II – System Reliability vs. Time

Time [h] (1 year ≈ 8700h)	Rs(t) with warm standby pump branches	Rs(t) with cold standby pump branches
0	1	1
8700	0,8183	0,8735
17400	0,6678	0,7609
26100	0,5436	0,6612
34800	0,4414	0,5731
43500	0,3576	0,4957
52200	0,2891	0,4278
60900	0,2333	0,3684
69600	0,1879	0,3167
78300	0,1510	0,2717
87000	0,1212	0,2327
95700	0,0971	0,1990
104400	0,0776	0,1699

Reliability as a function of time in the chart (Figure 9) has the expected trend: two negative exponential with different bending due to different system failure rates. Significant differences are already highlighted starting from the first year of use and reach the maximum at 60900 hours (7 years).

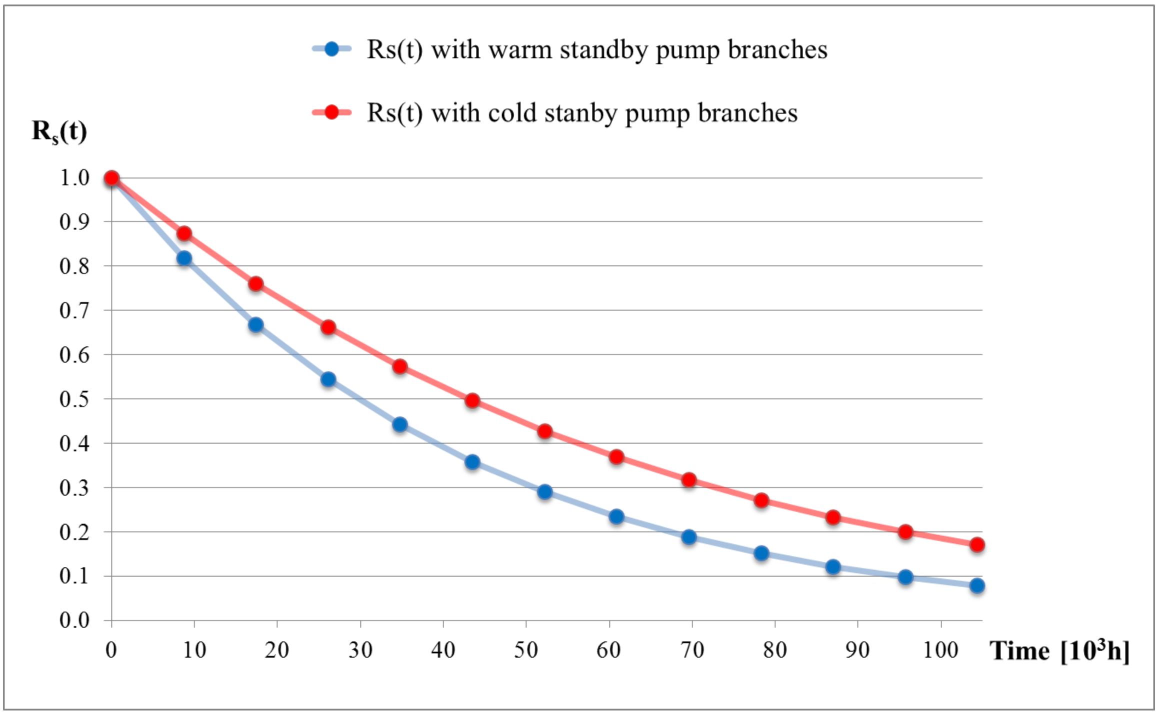


Fig. 9. Reliability vs. Time chart

The results validate the proposed method to assess reliability of redundant architectures in case of stand-by items: the developed procedure reduces to zero the limits in the complexity of RBDs under analysis.

The implemented tool was cross-validated with other commercial software. This comparison was performed considering several plants and architectures and results achieved using OREDA and NSWC [50] failure rates are very close compared to the expected ones and therefore show the validity of the proposed approach.

6.5. Case Study B: Synthetic Oil Console

Aero-derivative gas turbines usually have two lubricating systems and the synthetic one (Fig. 10) is the second test case used in this study in order to shows the potential of *RBDesigner*®.

The synthetic oil is used for its fire-resistant property and it is used in a dedicated lubricating system for the aero gas generator: in this system the lubricating oil plays a fundamental role for lubricating rotors that are carried on ball-and-roller antifriction bearings.

The system for the aero gas generator uses an oil cooler to reject the heat removed from the engine to the atmosphere. Sometimes in liquid-fueled installations the synthetic oil is cooled in a shell-and-tube heat exchanger by the incoming fuel.

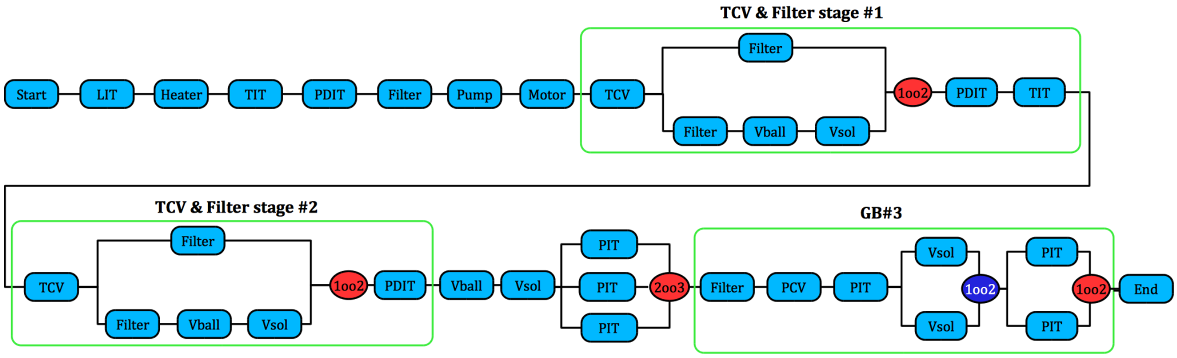


Fig. 10. Synthetic Oil Console RBD – Baseline layout

In Table III are shown the failure rates, Mean Time To Failures and Time To Repairs of all the components in the system.

RBDesigner® and Reliability Importance practice show the components that most affect the reliability of the whole system: Temperature Control Valve in both sub-systems 1 and 2, Solenoid Valve, Filter and Pressure Controlled Valve in sub-system 3.

Following these pieces of information the console design was changed (Fig. 11) and the comparison between the reliability performances of the two configurations is shown in Fig. 12.

The actions taken to improve the system are listed below:

- 1oo2 hot standby redundancy was introduced for the Temperature Controlled Valves using a Ball Valve in both sub-systems 1 and 2;
- Solenoid Valve downstream sub-system 2 was removed;
- 1oo2 cold standby redundancy was introduced for the Filter and Pressure Controlled Valve stage in sub-system 3;
- Level Indicator Transmitter (sub-system 1) and Pressure Indicator Transmitter (sub-system 3) were replaced with high-quality components that offer higher reliability and maintainability performance.

Table III – Synthetic Oil Console

Item	Failure rate [failure/h]	MTTF [h]	TTR [h]
LIT	2,45E-06	4,08E+05	8
Heater	1,63E-06	6,13E+05	2
TIT	2,45E-06	4,08E+05	1
PDIT	6,59E-07	1,52E+06	4
Filter	1,98E-04	5,05E+03	6
Ball valve	6,59E-06	1,52E+05	8
TCV	3,48E-06	2,87E+05	2
3-way valve	6,59E-06	1,52E+05	1
TIT	2,45E-06	4,08E+05	4
PIT	6,59E-06	1,52E+05	6
Solenoid valve	1,63E-05	6,13E+04	2
PCV	3,48E-06	2,87E+05	3

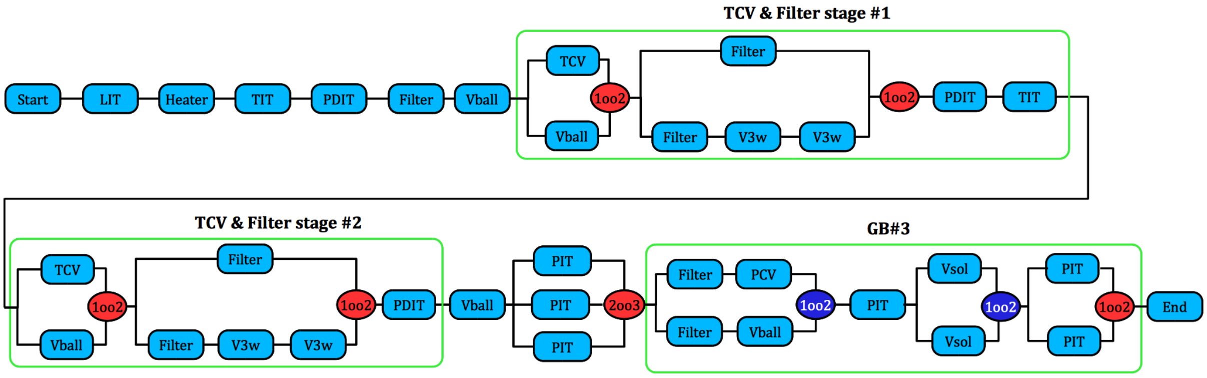


Fig. 11. Synthetic Oil Console RBD – Improved layout

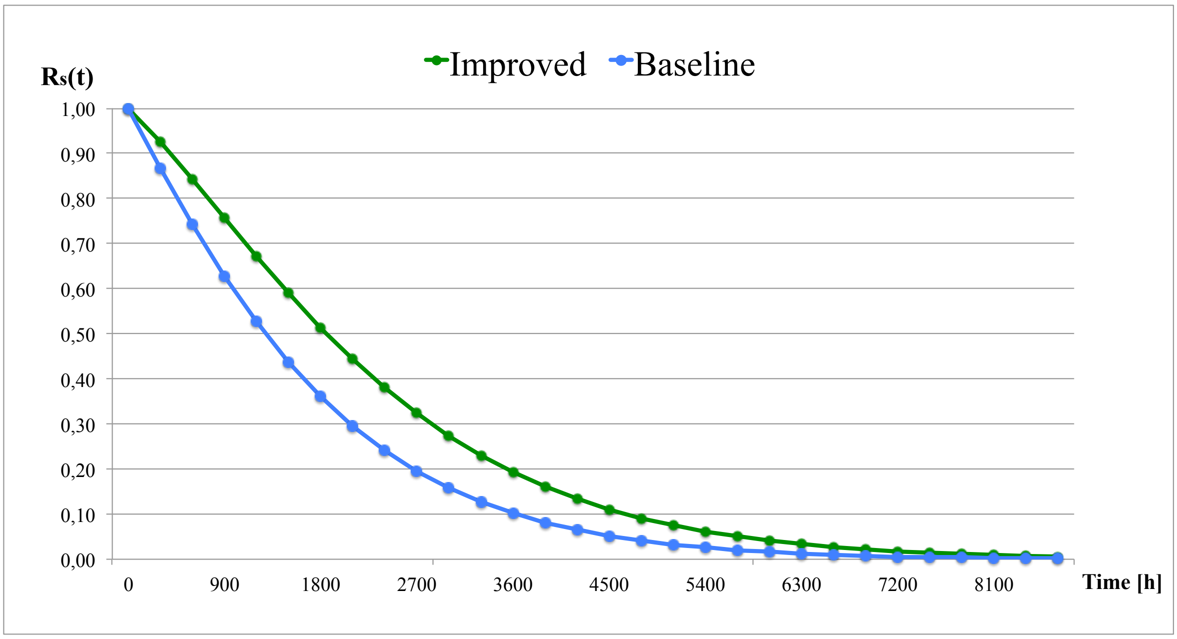


Fig. 12. Reliability vs. Time chart

Fig. 12 shows the growth of System Reliability: the baseline, at 2500 hours, offers $R_S (t) = 0{,}24$ while the improved layout guarantees $R_S (t) = 038$, a 14% increase.
System MTBF of the standard layout is 1823 hours while the improved layout offers 2414 hours that is a huge gain in terms of reliability (approximately 600 hours increase).
System MTTR of the standard layout is 3,9 hours while the improved layout offers 2,4 hours; finally System Availability increases from 0,9979 to 0,9990.

The results show the great impact that *RBDesigner®* and Reliability Importance have during design stage since they are fundamental to compare different design solutions, validate design choices and achieve reliability and availability target.

6.6. Discussion and Remarks

The results of the two case studies validate the proposed method for the reliability assessment of complex systems containing stand-by redundant architectures: the developed procedure reduces to zero the limits in the complexity of RBDs under analysis.
Furthermore using *RBDesigner®* project engineers are able to achieve a reliability prediction of very complex systems in the early stages of product development. This feature is a huge improvement in industrial applications because design can be based on reliability assessment.
The results presented in this work prove the main advantages achievable with the use of *RBDesigner®*, that is: reduction of time-delivery and time for improvements, confidence in achievable reliability targets and reliability performance guarantee to costumers.
The implemented tool was cross-validated with other commercial software. This comparison was performed considering several plants and architectures and results were always in compliance with expected ones.
The added values offered by *RBDesigner®* if compared with commercial software are listed below:

- Customized architecture library for Oil&Gas applications;
- Real-time multi-architecture comparison;
- User-friendly interface and guided RBD generation;
- No limit to RBD architecture complexity (in particular for standby redundancy configurations);
- Multi-database library for component reliability parameters;
- Full-integration with piping and instruments diagram definition.

Conclusions and Final Remarks

The number and variety of failures are growing due to modern technologies and business requirements with a corresponding increase of variety and complexity in the manufacturing production.

This study outlines how system downtime and unplanned outages massively affect plant productivity, in particular for Oil&Gas applications where an emergency shutdown produces an interruption of normal running operation with resulting reduction of productivity and loss of thousands of dollars.

This is the reason why this study is focused on RAMS disciplines together with fault diagnosis and condition-monitoring that nowadays are almost mandatory in Oil&Gas applications.

This thesis analyses the best methods to improve system availability and takes into account reliability and maintainability roles in order to achieve this kind of target.

The goal of improving system availability is to detect incipient failures, minimize downtime and minimize the time needed to restore the system to normal working conditions.

The first method described to achieve high-availability is redundancy, in particular in case of standby architectures that are widely used in gas turbine auxiliary systems and many other Oil&Gas applications: cold and warm standby were analysed, compared and two new reliability models were developed to describe reliability function vs. time.

This innovative approach was proposed to assess reliability in complex systems containing redundant architectures and the outcomes of the test cases outline the different reliability performance achievable with these two configurations and, at the same time, their wide pertinence to complex system modelling: the main added value is that there is no limit to RBD complexity and this feature is essential to achieve reliability prediction without restrictions for the system under analysis.

The second solution is reliability improvement using Reliability Allocation (RA): after a comparison of all the RA methods described in literature, a new procedure based on MEOWA technique was developed and successively implemented in a brand new dedicated Reliability Allocation tool. This software was developed on MathWorks "Matlab r2015a" platform and it calculates the reliability and the failure rate to be allocated to each component of the system.

Reliability Allocation tool was also tested on two complex case studies in order to validate it and extend its applicability to more complex architectures.

Reliability Importance methods, instead, are the third solution to improve system reliability: this study shows the generic procedure of Reliability Importance and focuses on a particular method named Credible improvement Potential (CIP) that has turned out to be the most flexible and efficient index to measure the impact of each component on the overall system reliability.

The trend of component Reliability Importance measures underlines the time-dependency behavior and the significance of RI assessment that is central to focus on the "right" components and take the best design decisions for money and time saving.

CIP method was modified and adjusted to best-fit the application required and it turned out to be particular effective in applications involving complex systems with different and composite redundant architectures.

Reliability Importance assessment confirmed to be particularly useful if hold with continuity during all the design stages of a product and this feature is mandatory to allow design engineers to take structural decisions and select the best items to complete the mission.

The fourth method to improve system availability was focused on maintenance, in particular on condition-based maintenance and Markov models: this technique is very helpful in case of complex repair solutions, standby spares, sequential dependency and imperfect fault coverage but showed some limits in modelling complex systems such as gas turbine auxiliary systems.

In fact Markov modelling is a powerful method to assess system maintainability and this procedure can be a valid support for availability assessment but it is usable only in case of low-complex systems: the complexity of the system directly translates into the complexity of the corresponding Markov model and a system of ten components may produce a model with hundred states and even more transitions.

Furthermore, this approach requires a manual construction of the model and this practice is not so familiar to design engineers so this implies cumbersome and error-prone modelling.

For these reasons Markov processes are not the most suitable practice for the purpose of this study.

Finally, diagnostics and condition monitoring were the last solution to achieve high-availability performance. This study contains both on-board and logic solver diagnostics with a detailed application on a gas turbine safety loop and corresponding Probability of Failure on Demand (PFD) assessment.

This target was achieved defining a new system of thresholds to rule the logic solver (e.g. A2M and A3M architectures) that was calibrated taking into account the different failure modes of the devices involved in the loop.

In conclusion one of the best procedures to achieve high-availability taking into account reliability improvement is Reliability Importance, in particular using CIP analysis: this method was implemented in a dedicated tool, *RBDesigner®*, which plays a fundamental role during the design stage of gas turbine auxiliary systems.

Thanks to *RBDesigner®*, project engineers are able to achieve a reliability prediction of very complex systems in the early stages of product development. This feature is a huge improvement in industrial applications because design can be based on reliability assessment.

The results presented in this work prove the main advantages achievable with the use of *RBDesigner*®: reduction of time-delivery and time for improvements, confidence in achievable reliability targets and reliability performance guarantee to costumers.

Furthermore the results of the two described case studies validate the proposed method for the reliability assessment of complex systems containing stand-by redundant architectures and prove that the developed procedure reduces to zero the limits in the complexity of RBDs under analysis.

The implemented tool was also cross-validated with other commercial software considering several plants and architectures and the added values offered are the following: customized architecture library for Oil&Gas applications, real-time multi-architecture comparison, user-friendly interface and guided RBD generation, limitless RBD architecture complexity, multi-database library for component reliability parameters and full-integration with piping and instruments diagram definition.

Appendix I

Sensing Devices

The most used sensor classes for Oil&Gas applications are temperature, pressure, flow and level: a brief description of each type is shown below.

The most commonly type of sensors is the temperature or heat ones: these sensors measure the amount of heat energy that is generated by the system, allowing the user to detect any physical change to that temperature. There are many different classes of temperature sensors and the main distinction is between contact devices that require to be in physical contact with the object under sensing and uses conduction to monitor changes in temperature, and non-contact devices that uses convection and radiation to monitor changes [65-69].

The three main types of temperature sensor are the following:

- Thermostat - TS: contact type electro-mechanical temperature sensor made of two different metals (e.g. nickel, copper, tungsten or aluminium) bonded together to form a bi-metallic strip. The different linear expansion rate of the two dissimilar metals produces a mechanical bending of the strip corresponding to temperature change.
- Thermistor - RTD: special type of resistor that changes its physical resistance when exposed to changes in temperature. They are passive resistive devices made of ceramic materials (e.g. oxides of nickel, manganese or cobalt coated in glass) so it is necessary to pass a current through it to produce a measurable voltage output; their main advantage is the fast response to and accuracy of measures.
- Thermocouple - TC: thermoelectric sensors made of two junctions of different metals (e.g. copper and constantan) welded together. The reference junction is kept at a constant temperature (cold junction) while the other is the measuring (hot) junction. When the two junctions are at different temperatures, a voltage is developed across the junction. Thermocouples have the widest temperature range of all the temperature sensors from below -200°C to over 2000°C and are popular due to simplicity, ease of use and speed of response.

The measurement of pressure is generally associated with fluids, either liquids or gases; pressure is defined as force per unit area. There are different types of pressure measurements: absolute pressure (measurement referred to perfect vacuum e.g. atmospheric pressure), gauge pressure (measurement referred to ambient pressure) and differential pressure (difference between two points of measurement). Pressure sensor, depending on the reference pressure used, indicate absolute, gauge or differential pressure.

There are three basic categories of pressure sensors:

- Piezoelectric: uses the piezoelectric effect distinctive of some materials (e.g. quartz) to measure the strain upon the sensing area due to pressure. This technology is widely employed for the measurement of highly dynamic pressures with fast change in pressure.
- Piezoresistive: uses the piezoresistive effect of materials such as silicon to detect strain due to applied pressure; deformation causes a change in the band structure of the material leading to a change in the resistivity of the material. This change can be an increase or a decrease according to the orientation of the resistors.
- Capacitive: two different plates form a parallel plate capacitor, a fixed element having a rigid conductive surface (base plate) and a deformable conductive member (diaphragm). The capacitance of the sensor is inversely proportional to the gap between the central portion of the diaphragm and the conductive surface of the fixed plate; pressure deforms the diaphragm and induces the capacitance variation.

Another class of sensors used in Oil&Gas applications is flow meter: flow is the rate (volume or area per unit time) at which a fluid travels through a given cross section. Flow sensors use acoustic waves and electromagnetic fields to measure the flow through a given area using physical quantities (e.g. acceleration, frequency, pressure and volume).

There are various kinds of flow sensors and flow meters: the flow rate is assessed by the flow sensor and derived from other physical properties. The relationship between the physical properties and the flow rate is derived from fundamental fluid flow principles, such as Bernoulli's equation [65-69].

The three main class of flow sensor are the following:

- Differential pressure: sensors work according to Bernoulli's principle so the pressure dropping across the meter is proportional to the square of the flow rate. The use of pressure drop across e.g. a pipe's cross section is one of the most common manners to determine a flow measurement. Some kind of differential pressure flow sensors are orifice meters, Pitot tubes, Venturi tubes, flow nozzles.
- Direct force: flow meters governed by balancing forces within the system; some of the most important are rotameters, turbine meters and Coriolis mass flow meters.
- Ultrasonic: there are two types of ultrasonic meters, Doppler and transit time. The former use the frequency shift of an ultrasonic signal when it is reflected by suspended particles or discontinuities (e.g. gas bubbles) in motion. The Doppler effect flowmeters uses reflected ultrasonic sound to measure the fluid velocity. Transit time meters, instead, have two opposing transducers outside the pipe to measure the time of a signal sent from a transducer upstream to a transducer downstream and vice versa.

Level sensors detect the level of liquids and fluids that become horizontal in their containers because of gravity. This measurement can be either continuous or point values: continuous level devices measure the exact amount of substance (within a range) in a certain place while point-level sensors indicate if the fluids is above or below the sensing point. There are many different classes of level sensors because a lot of variables are involved (e.g. phase, temperature, pressure, density…). Also level sensors can be designed using variety of sensing principles. Some examples are:

- Magnetic and mechanical: direct contact or magnetic operation control the opening or closing of a mechanical switch.
- Ultrasonic: level sensors used for contactless level sensing in particular for highly viscous liquids. The sensor transmits an ultrasonic beam to the surface level and the returned echo from the surface is detected by the sensor and converted into a digital representation of the distance between the sensor and the surface level.
- Hydrostatic: used for measuring liquid levels in open or closed vessels. The level measure is detected measuring the liquid column pressure; with the pressure measure and the fluid specific gravity is possible to calculate the fluid column height.

In the following tables are shown all the failure modes (Table I) and mechanisms (Table II) related to both mechanical and electronic items in Oil&Gas applications.

Table I – Failure modes in Oil&Gas applications in compliance with UNI EN ISO 14224

Failure Mode	Description
Abnormal instrument reading	False alarm, faulty instrument indication
Breakdown	Serious damage (seizure, breakage)
External leakage	Oil, gas, condensate, water, lubricant, cooling water leakage
Erratic output	Oscillating, hunting, instability of outcomes
Failure to connect	Failure to connect when required
Failure to disconnect	Failure to disconnect when demanded
Faulty output frequency	Wrong/oscillating output frequency
Faulty output voltage	Wrong/unstable output voltage
Failure to rotate	Failure to rotate whrn required
Failure to close on demand	Doesn't close on demand
Failure to function on demand	Doesn't start on demand
Failure to function as intended	General operation failure
Failure to lock/unlock	Doesn't lock or unlock when demanded
Failure to open on demand	Doesn't open on demand
Failure to regulate	Absence of proper setting
Failure to start on demand	Doesn't start on demand
High output	Overspeed/output above acceptance
Insufficient heat transfer	Cooling/heating below acceptance
Internal leakage	Leakage internally of process or utility fluids
Loss of buoyancy	Loss of buoyancy in idle position
Low oil supply pressure	Oil supply pressure below acceptance
Leakage in closed position	Leak through e.g. valve in closed position
Load drop	Load drop
Loss of barrier	One or more barriers against oil/gas escape lost
Low output	Delivery/output below acceptance
Loss of performance	Performance below specifications
Loss of redundancy	One or more redundant units not functioning
Mooring failure	Mooring failure
Noise	Abnormal/excessive noise
No immediate effect	No effect on function
No output	Absence of output
Overheating	Overheating of machine parts, exhaust, cooling water
Parameter deviation	Monitored parameter exceeding limits, e.g. high/low alarm
Plugged / Choked	Partial or full flow restriction due to contamination, objects, etc.
Insufficient power	Too low power supply
Power/signal transmission failure	Power/signal transmission failure
Minor in-service problems	Loose items, discoloration, dirt
Failure to set/retrieve	Failed set/retrieve operations
Spurious high alarm level	High alarm level when not necessary
Spurious low alarm level	Low alarm level when not necessary
Slippage	Wire slippage
Spurious operation	Unexpected operation, fails to operate as demanded
Spurious stop	Unexpected shut down
Structural deficiency	Material damages (cracks, wear, fracture, corrosion, rupture)
Failure to stop on demand	Doesn't stop on demand
Spurious stop	Unexpected shutdown
Vibration	Abnormal vibration
Delayed operatioin	Expected action with delay
Unknown	No information available
Other	Failure modes not covered above

Table II – Failure mechanisms in Oil&Gas applications in compliance with UNI EN ISO 14224

	Failure Mechanism	Description
Mechanical failure	General mechanical Failure	A failure related to some mechanical defect, no further details are known
	Leakage	External and internal leakages, either liquids or gases
	Vibration	Abnormal vibration
	Clearance/alignment failure	Failure caused by faulty clearance or alignment
	Deformation	Distortion, bending, buckling, denting, yielding, shrinking, blistering, creeping, etc.
	Looseness	Disconnection, loose items
	Sticking	Sticking, seizure, jamming due to reasons other than deformation or clearance/alignment failures
Material failure	General material failure	A failure related to a material defect, no further details known
	Cavitation	Relevant for equipment such as pumps and valves
	Corrosion	All types of corrosion, both wet (electrochemical) and dry (chemical)
	Erosion	Erosive wear
	Wear	Abrasive and adhesive wear, e.g. scoring, galling, scuffing, fretting
	Breakage	Fracture, breach and crack
	Fatigue	In case the cause of breakage can be traced to fatigue
	Overheating	Material damage due to overheating/burning
	Burst	Item burst, blown, exploded, imploded, etc.
Instrument failure	General instrument failure	Failure related to instrumentation, no details known
	Control failure	No or faulty regulation
	No signal/indication/alarm	No signal/indication/alarm when expected
	Faulty signal/indication/alarm	Signal/indication/alarm is wrong in relation to actual process. Can be spurious, intermittent, oscillating, arbitrary
	Out of adjustment	Calibration error, parameter drift
	Software failure	Faulty or no control/monitoring/operation due to software failure
	Common cause/mode failure	Several instrument items failed simultaneously e.g. redundant fire and gas detectors
Electrical failure	General electrical failure	Failures related to the supply and transmission of electrical power, no further details known
	Short circuiting	Short circuit
	Open circuit	Disconnection, interruption, broken wire/cable
	No power/voltage	Missing or insufficient electrical power supply
	Faulty power/voltage	Faulty electrical power supply, e.g. overvoltage
	Earth/isolation fault	Earth fault, low electrical resistance
External influence	General external influence	Failure caused by some external events or substances outside the boundary, no further details known
	Blockage/plugged	Flow restricted/blocked due to fouling, contamination, icing, flow assurance (hydrates), etc.
	Contamination	Contaminated fluid/gas/surface, e.g. lubrication oil contaminated, gas detector head contaminated
	Miscellaneous external influences	Foreign objects, impacts, environmental influence from neighbouring systems
Miscellaneous	General miscellaneous	Failure mechanism that does not fall into one of the categories listed above
	No cause found	Failure investigated but cause not revealed or too uncertain
	Combined causes	Several causes
	Other	No code applicable
	Unknown	No information available

In the following tables are shown the failure mechanisms and relative failure modes for each class of sensors.

Table III – Failure mechanisms and modes for temperature sensors

Failure Mechanism	Description	Failure Mode
Vibration	Abnormal vibration	Erratic output
Corrosion	All types of corrosion, both wet (electrochemical) and dry (chemical)	External leakage
		Low output
		Erratic output
Breakage	Fracture, breach and crack	No output
Overheating	Material damage due to overheating/burning	High output
		Low output
		External leakage
Faulty signal/indication/alarm	Signal/indication/alarm is wrong in relation to actual process. Can be spurious, intermittent, oscillating, arbitrary	High output
		Low output
		No output
Out of adjustment	Calibration error, parameter drift	Spurious operation
Software failure	Faulty or no control/monitoring/operation due to software failure	High output
		Low output
		No output
		Failure to function on demand
Open circuit	Disconnection, interruption, broken wire/cable	No output
Contamination	Contaminated fluid/gas/surface, e.g. lubrication oil contaminated, gas detector head contaminated	High output
		Low output
Miscellaneous external influences	Foreign objects, impacts, environmental influence from neighbouring systems	High output
		Low output
No cause found	Failure investigated but cause not revealed or too uncertain	Unknown
Combined causes	Several causes	Unknown

Table IV – Failure mechanisms and modes for pressure sensors

Failure Mechanism	Description	Failure Mode
Deformation	Distortion, bending, buckling, denting, yielding, shrinking, blistering, creeping, etc.	Erratic output
		Spurious operation
Corrosion	All types of corrosion, both wet (electrochemical) and dry (chemical)	External leakage
		Low output
		Erratic output
Breakage	Fracture, breach and crack	No output
Fatigue	In case the cause of breakage can be traced to fatigue	High output
		Low output
		Spurious operation
Burst	Item burst, blown, exploded, imploded, etc.	No output
		External leakage
Faulty signal/indication/alarm	Signal/indication/alarm is wrong in relation to actual process. Can be spurious, intermittent, oscillating, arbitrary	High output
		Low output
		No output
Out of adjustment	Calibration error, parameter drift	Spurious operation
Software failure	Faulty or no control/monitoring/operation due to software failure	High output
		Low output
		No output
		Failure to function on demand
Open circuit	Disconnection, interruption, broken wire/cable	No output
No power/voltage	Missing or insufficient electrical power supply	No output
Earth/isolation fault	Earth fault, low electrical resistance	No output
Contamination	Contaminated fluid/gas/surface, e.g. lubrication oil contaminated, gas detector head contaminated	High output
		Low output
Miscellaneous external influences	Foreign objects, impacts, environmental influence from neighbouring systems	High output
		Low output
No cause found	Failure investigated but cause not revealed or too uncertain	Unknown
Combined causes	Several causes	Unknown

Table V – Failure mechanisms and modes for level sensors

Failure Mechanism	Description	Failure Mode
Vibration	Abnormal vibration	Erratic output
Deformation	Distortion, bending, buckling, denting, yielding, shrinking, blistering, creeping, etc.	Erratic output
		Spurious operation
Looseness	Disconnection, loose items	Erratic output
		Spurious operation
Corrosion	All types of corrosion, both wet (electrochemical) and dry (chemical)	External leakage
		Low output
		Erratic output
Breakage	Fracture, breach and crack	No output
No signal/indication/alarm	No signal/indication/alarm when expected	No output
Faulty signal/indication/alarm	Signal/indication/alarm is wrong in relation to actual process. Can be spurious, intermittent, oscillating, arbitrary	High output
		Low output
		No output
Out of adjustment	Calibration error, parameter drift	Spurious operation
Software failure	Faulty or no control/monitoring/operation due to software failure	High output
		Low output
		No output
		Failure to function on demand
Open circuit	Disconnection, interruption, broken wire/cable	No output
No power/voltage	Missing or insufficient electrical power supply	No output
Faulty power/voltage	Faulty electrical power supply, e.g. overvoltage	High output
		Low output
Earth/isolation fault	Earth fault, low electrical resistance	No output
Blockage/plugged	Flow restricted/blocked due to fouling, contamination, icing, flow assurance (hydrates), etc.	No output
		Erratic output
Contamination	Contaminated fluid/gas/surface, e.g. lubrication oil contaminated, gas detector head contaminated	High output
		Low output
Miscellaneous external influences	Foreign objects, impacts, environmental influence from neighbouring systems	High output
		Low output
No cause found	Failure investigated but cause not revealed or too uncertain	Unknown
Combined causes	Several causes	Unknown

Table VI – Failure mechanisms and modes for flow sensors

Failure Mechanism	Description	Failure Mode
Deformation	Distortion, bending, buckling, denting, yielding, shrinking, blistering, creeping, etc.	Erratic output
		Spurious operation
Looseness	Disconnection, loose items	Erratic output
		Spurious operation
Clearance/alignment failure	Failure caused by faulty clearance or alignment	Erratic output
Sticking	Sticking, seizure, jamming due to reasons other than deformation or clearance/alignment failures	Erratic output
Cavitation	Relevant for equipment such as pumps and valves	Erratic output
Corrosion	All types of corrosion, both wet (electrochemical) and dry (chemical)	External leakage
		Low output
		Erratic output
Breakage	Fracture, breach and crack	No output
Out of adjustment	Calibration error, parameter drift	Spurious operation
Software failure	Faulty or no control/monitoring/operation due to software failure	High output
		Low output
		No output
		Failure to function on demand
Open circuit	Disconnection, interruption, broken wire/cable	No output
No power/voltage	Missing or insufficient electrical power supply	No output
Faulty power/voltage	Faulty electrical power supply, e.g. overvoltage	High output
		Low output
Blockage/plugged	Flow restricted/blocked due to fouling, contamination, icing, flow assurance (hydrates), etc.	No output
		Erratic output
Contamination	Contaminated fluid/gas/surface, e.g. lubrication oil contaminated, gas detector head contaminated	High output
		Low output
Miscellaneous external influences	Foreign objects, impacts, environmental influence from neighbouring systems	High output
		Low output
No cause found	Failure investigated but cause not revealed or too uncertain	Unknown
Combined causes	Several causes	Unknown

Appendix II

Functional Safety & Safety Instrumented Systems

I. Functional Safety

The operation of many industrial processes, especially in chemical and oil & gas fields, involves inherent risk to persons, property, and environment.

The goal of functional safety is to design, built, operate and maintain systems in such a way to prevent dangerous failures or, at least, to be able to control them in case of hazardous conditions.

A risk-based approach is mandatory to determine the required performance of safety systems.

The risk for a system is associated with an initiating event that leads the system into a degraded state in which the integrity of the system itself is more or less severely impacted. To mitigate a risk the solution is to reduce its frequency or its severity or both.

Table I shows the risk matrix in terms of severity and frequency in compliance with IEC 61508: this is a generic standard that provides the framework and core requirements for functional safety of safety related systems that use Electrical/Electronic/Programmable Electronic (E/E/PE) technologies in industrial applications [86-88].

Table I - Risk matrix in compliance with IEC 61508

Risk matrix			**Severity**			
			Negligible	*Marginal*	*Critical*	*Catastrophic*
			Minor injuries at worst	Major injuries to one or more persons	Loss of a single life	Multiple loss of live
Frequency	*Frequent*	$> 10^{-3}$	Undesirable	Unecceptable	Unecceptable	Unecceptable
	Probable	10^{-3} to 10^{-4}	Tolerable	Undesirable	Unecceptable	Unecceptable
	Occasional	10^{-4} to 10^{-5}	Tolerable	Tolerable	Undesirable	Unecceptable
	Remote	10^{-5} to 10^{-6}	Acceptable	Tolerable	Tolerable	Undesirable
	Improbable	10^{-6} to 10^{-7}	Acceptable	Acceptable	Tolerable	Tolerable
	Incredible	$\leq 10^{-7}$	Acceptable	Acceptable	Acceptable	Acceptable

There are many sources of safety failures: incorrect specifications of the system, omissions in the safety requirements specification, random/systematic hardware failures, common cause failures, human errors, unplanned system changes after commissioning and environmental influences.

In order to reduce the risk arising from industrial plants, it might be necessary to automatically activate safety measures when required to avoid dangerous situations:

functional safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems is achieved with Safety Instrumented Systems (SIS). These systems are specifically designed to protect personnel, equipment, and the environment by reducing the likelihood or the impact severity of hazardous events.

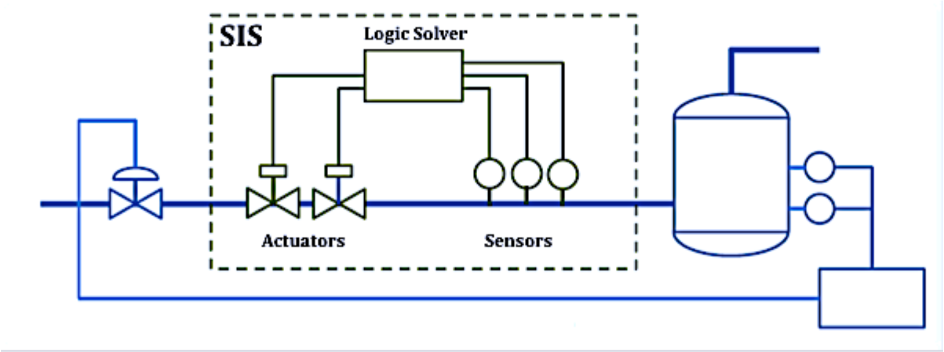


Fig. 1. SIS - Safety Instrumented System

Safety Instrumented Systems (see Fig. I and Fig. II) are typically constituted by a combination of three fundamental blocks [88-89]:

- Sensor(s) detects a physical quantity and provides a corresponding electrical output. Field sensors are used to collect information and determine an incipient danger; these sensors evaluate process parameters (e.g. temperature, pressure, flow, etc.) in order to determine if single equipment or the whole process or plant is working properly and it is in a safe state. Such sensors do not monitor the normal process but they are usually dedicated to SIS.
- Logic solver(s) receives the information collected by the sensor and elaborates it to take the best response. It is typically a controller that takes actions according to the defined logic in order to prevent hazardous conditions.
- Final element(s) implements the outcomes of the logic solver. This actuator is the last element of the loop and in many industrial applications is represented by a pneumatic valve.

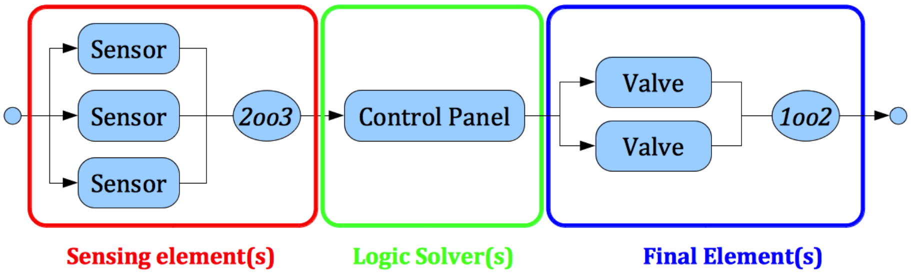


Fig. 2. Safety Instrumented System Functional Block Diagram

The aim of SIS is to implement one or more Safety Instrumented Functions (SIF) in order to guarantee a Safety Integrity Level (SIL): SIFs control critical processes and avoid unacceptable or dangerous conditions for health and environment. Each SIF is associated with a safety loop that is the process involving all the three stages described above (sensor, logic solver and final element) in order to detect a failure, elaborate the collected data and perform the corrective action.

SIL is determined by the Risk Reduction Factor (RRF) provided by the SIS to the equipment under control. The inverse of the RRF is the Probability of Failure on Demand (PFD) that is a value that indicates the probability of a system failing to respond to a demand.

Average Probability of Failure on Demand (PFDavg) is the average probability of a system failing to respond to a demand in a specified time interval, usually called Proof Test Interval.

$$RFF = {}^{1}\!/_{PFD} \tag{211}$$

There are two modes of operation for a safety function, low and high (or continuous) demand mode.

In a low demand mode the safety function is only performed on demand in order to lead the EUC to a specified safe state; in this case the frequency of demands is no greater than one per year or twice the proof test frequency (frequency setting how often the safety system is completely tested and insured to be fully operational).

In a high demand mode the safety function is always performed on demand but more than twice the proof check frequency; in continuous mode of operation, instead, the safety function is part of normal operation.

Low demand mode is defined by PFD target while high demand and continuous mode follow the Probability of (dangerous) Failure per Hour (PFH).

Table II - SIL and corresponding PFD and PFH targets

SIL	Low demand mode of operation PFD_{avg}	High demand or continuos mode of operation PFH [h^{-1}]
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 $< 10^{-5}$

The probability to fail on demand can be calculated using the dangerous failure rate λ_D and the testing interval T_1 (assuming that systematic failures are minimized) as follows:

$$PFD = \lambda_D \cdot \frac{T_1}{2} \tag{212}$$

The equation shows that the relationship between PFD and TI is linear so longer test intervals lead to larger PFDs.

II. Definitions

Common terms related to Safety Instrumented Systems are listed below [88-89]:

- Functional safety: part of the overall safety relating to the equipment under control (EUC) and the EUC control system that depends on the correct functioning of the E/E/PE Safety-Related System and other risk reduction measures.
- Safety-Related System: a system that implements the required safety functions necessary to achieve or maintain a safe state for the EUC and achieves safety integrity for the required safety functions.
- Process Hazard Analysis (PHA): it requires identifications of hazards, causes of accidents, possible outcome of accidents, safeguard to prevent and recommendation to implement measures to reduce process risk.
- Safety Instrumented Function (SIF): Safety function with a specified safety integrity level, which is necessary to achieve functional safety.
- Safety Instrumented System (SIS): Instrumented system used to implement one or more safety instrumented functions. A SIS is composed of three sub-systems, sensors, logic solvers, and final elements.
- Safety Integrity Level (SIL): A quantifiable measurement of risk used to establish safety performance targets for SIS systems. It is a classification of Safety Function ability to reduce the risk for accidents in industrial processes.
- Spurious Trip: Refers to the shutdown of the process not for safety reasons
- Safe State: State that the equipment under control, or the process, shall attain as defined by the Process Hazard Analysis.
- Demand: A condition or event that requires the safety instrumented system to take appropriate action to prevent an arising hazardous event or mitigate the consequence of a hazardous event.
- Hardware Fault Tolerance (HFT): it is the maximum number of hardware faults that will not lead to a dangerous failure, so a hardware fault tolerance of zero means that a single fault can cause loss of the safety function. It is a measure of the quality of a safety function.
- Probability of Failure on Demand (PFD): A value that indicates the probability of a system failing to respond to a demand. PFDavg is the average probability of a system failing to respond to a demand in a specified time interval, usually called Proof Test Interval.
- Redundancy: Use of multiple elements or systems to perform the same function.
- Random hardware failure: failure that occurs at random time and produces a degradation mechanism in system hardware.
- Systematic failure: failure that cannot be accurately predicted (such as random ones) and requires modification of the design to be solved.
- Common Cause Failures (CCF): single failure that affects the operation of multiple (usually identical) devices, produces concurrent failures and leads to system failure. Common cause failures can result in the SIS failing to function when there is a process demand.

- Safe/Dangerous failure: a safe failure causes the system to go to the defined fail-safe state without a demand from the process so it does not compromise the system safety integrity and reduces availability and productivity. A dangerous failure, instead, leads to a safety-related system failing to function and compromise system safety integrity.
- Detected/undetected failure: a failure that will be detected/undetected by diagnostic tests.

III. Diagnostic Coverage And Safe Failure Fraction

There are four types of random hardware failures depending on safe/dangerous scenarios and detectability:

- Safe undetected (λ_{SU}): SIF can always be performed;
- Safe detected (λ_{SD}): SIF can always be performed;
- Dangerous detected (λ_{DD}): SIF cannot be performed but system will quickly go into the safe state;
- Dangerous undetected (λ_{DU}): failure occurs without notice and in case of demand the safety system cannot perform SIF.

Two important parameters for safety assessment are Diagnostic Coverage (DC) and Safe Failure Fraction (SFF) [28].

DC is the ratio of the probability of detected failures to the probability of all the dangerous failures and it is a measure of system ability to detect failures; SFF, instead, indicates the probability of the system failing in a safe state so it shows the percentage of possible failures that are self-identified by the device or are safe and have no effect [86].

The steps required for DC and SFF assessment are listed below:

- Starting from the Reliability Block Diagram (RBD) of the system (which is a sketch containing all the items making-up the system and their interconnections) – the first step is to assess Failure Mode and Effect Analysis (FMEA) to determine the effect of each failure mode of the components on the behaviour of the whole system.
- The second step is the categorization of failure modes according to its consequences: safe or dangerous failures.
- The third step is the estimation of failure rate (λ) of each component or group of components and the probability of safe (λ_S) and dangerous (λ_D) failures. The failure rate of each component or group of components can be estimated using data from a recognized industry source, taking the application environment into account.
- For each component or group of components, estimate the fraction of safe/dangerous failures that will be detected ("D") or undetected ("U") by diagnostic tests. The corresponding probabilities are λ_{SD}, λ_{SU}, λ_{DD} and λ_{DU} so the first subscript letter is referred to safe/dangerous failure, the second letter concern detection likelihood.

$$DC = \frac{\sum \lambda_{DD}}{\sum \lambda_{DD} + \sum \lambda_{DU}} \qquad SFF = \frac{\sum \lambda_S + \sum \lambda_{DD}}{\sum \lambda_S + \sum \lambda_{DD} + \sum \lambda_{DU}} \tag{213}$$

In order to assess DC and SFF the analyst has to include all the electrical, electronic, electromechanical and mechanical items necessary to allow the system to process the required safety functions.

Similarly it is mandatory to consider all of the possible dangerous modes of failure that could lead to an unsafe state, prevent a safe response on demand or compromise the system safety integrity.

Within the dangerous failures, it is necessary to estimate for each component the fraction of failures that are detected by the diagnostic tests: these tests (e.g. comparison checks in redundant architectures, additional built-in test routines and continuous condition monitoring) are a huge contribute to the diagnostic coverage.

IV. Architectural Constrains

Architectural constrains on hardware safety integrity are a set of architectural requirements that influence the SIL assessment for each subsystem. These constraints are associated with three parameters: Hardware Fault Tolerance, Safe Failure Fraction and "A/B-type" classification [28].

Hardware Fault Tolerance (HFT) is the maximum number of hardware faults that will not lead to a dangerous failure. HFT of "n" means that "n+1" faults cause a loss of the SIF.

This type of fault tolerance can be increased by means of system architecture: the limit of each configuration is the number of working devices required to perform the safety function.

As for all redundant architectures, common-cause failures (CCF) can nullify redundancy.

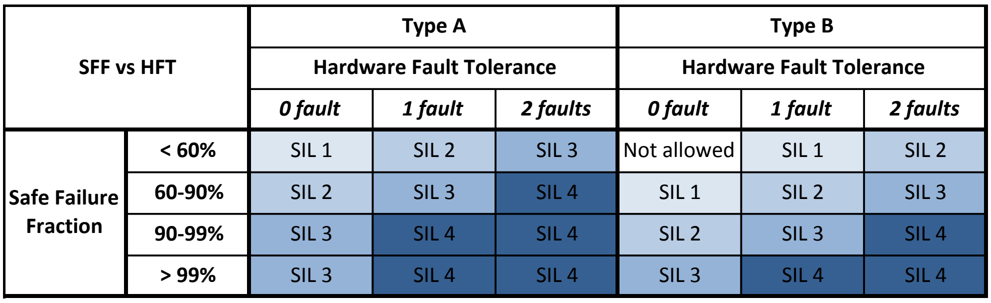
There are three different stages of hardware fault tolerance:

- HFT=0 In a single channel architecture (1oo1) only in case of no failure the safety function can be performed.
- HFT=1 In a dual redundancy (1oo2 or 2oo3) even in case of one failure in the sensing elements or logic solvers the safety function can still be performed.
- HFT=2 In a triple redundancy (1oo3) up to two failures can be tolerated in order to perform the safety function.

The second parameter is Safe Failure Fraction described in the previous paragraph that represent the fraction of failures which can be considered "safe" since they are detected by diagnostic tests or do not cause a loss of the safety function.

The last architectural constrain is the subsystem classification in "A/B-type"; type A subsystems have consolidated design and the behaviour in case of error is well known. For type B subsystems, instead, the behaviour in case of failure is not completely known [88].

Table III. SIL depending on SFF and HFT



SFF vs HFT		Type A			Type B		
		Hardware Fault Tolerance			Hardware Fault Tolerance		
		0 fault	*1 fault*	*2 faults*	*0 fault*	*1 fault*	*2 faults*
Safe Failure Fraction	< 60%	SIL 1	SIL 2	SIL 3	Not allowed	SIL 1	SIL 2
	60-90%	SIL 2	SIL 3	SIL 4	SIL 1	SIL 2	SIL 3
	90-99%	SIL 3	SIL 4	SIL 4	SIL 2	SIL 3	SIL 4
	> 99%	SIL 3	SIL 4	SIL 4	SIL 3	SIL 4	SIL 4

V. Probability Of Failure On Demand And Probability Of Failure Per Hour

Probability of Failure on Demand (PFD) and Probability of Failure per Hour (PFH) can be assessed by understanding how the components of the SIS can fail: there are two basic ways for a SIS to fail.

The first way is commonly called a nuisance or spurious trip which usually results in an unplanned process shutdown without safety impact. So there is no danger associated with this type of SIS failure and it only reduces availability and productivity (safe failure).

The second type of failure does not cause a process shutdown or nuisance trip. In this case the failure remains undetected, permitting continued process operation in an unsafe and dangerous condition. If an emergency demand occurred, the SIS would be unable to respond properly (dangerous failure).

In low demand mode of operation, IEC 61508 requires the PFDavg assessment starting from Mean Down Time (MDT).

MDT(T) is the mean down time of the safety system over the period [0,T]; for components in series, MDT is the sum of MDTs of each part [88].

$$MDT_{SYS} = MDT_A + MDT_B \tag{214}$$

For redundant architectures:

$$MDT_{SYS} = \int_0^T PFD_A(t) \cdot PFD_B(t) \cdot dt \tag{215}$$

For a Safety Instrumented System:

$$MDT_{SYS} = MDT_{SE} + MDT_{LS} + MDT_{FE} \tag{216}$$

Dividing by T:

$$PFD_{avgSYS} = PFD_{avgSE} + PFD_{avgLS} + PFD_{avgFE} \tag{217}$$

In continuous or high demand mode of operation, IEC 61508 requires the PFH assessment: it is the average of the unconditional failure intensity (also called failure frequency) w(t) over the period of interest:

$$PFH(T) = \frac{1}{T}\int_{0}^{T} w(t) \cdot dt \tag{218}$$

VI. Reliability Block Diagram Approach For PFD

Reliability Block Diagram approach is used for PFD assessment in different system architectures. The necessary assumption, in compliance with IEC 61508 [88], are shown below:

- Component failure rates are constant over the life of the system;
- The sensor (input) subsystem comprises the actual sensor(s) and wiring but not includes voting or other processing devices;
- The final element (output) subsystem comprises all the components and wiring from the logic solver to final actuating component(s);
- For each safety function, there is perfect proof testing and repair so all failures that remain undetected are detected by the proof test;
- The proof test interval is at least an order of magnitude greater than the mean repair time (MRT);
- For each subsystem there is a single proof test interval and MRT;
- The expected interval between demands is at least an order of magnitude greater than the proof test interval.

Legend:

- T_1: Proof Test Interval (hour)
- T_2: Interval between demands (hour)
- MTTR: Mean Time To Restoration (hour)
- MRT: Mean Repair Time (hour)
- DC: Diagnostic Coverage
- β: Fraction of undetected failures that have a common cause
- β_D: Fraction of detected failures that have a common cause
- PFD_{avg}: Average Probability of Failure on Demand
- PFD_{SE}: Sensing element Probability of Failure on Demand
- PFD_{LS}: Logic solver Probability of Failure on Demand
- PFD_{FE}: Final element Probability of Failure on Demand
- PFD_{SYS}: System Probability of Failure on Demand
- t_{CE}: Channel equivalent MDT (combined down time for all the component in the channel of the subsystem; hour)
- t_{GE}: Voted group equivalent MDT (combined down time for all the channels in the voted group; hour)

The average probability of failure on demand of a safety function for the safety-related system is determined by the combination of the average probability of failure on demand for all the subsystems involved in the safety function [88]. Average PFD can be expressed as follows:

$$PFD_{SYS} = PFD_{SE} + PFD_{LS} + PFD_{FE} \tag{219}$$

A. 1oo1

The simplest structure is a single-item architecture (1oo1) that consists of a single channel where any dangerous failure leads to a failure of the safety function (when a demand arises).
The dangerous failure rate for the channel is given by:

$$\lambda_D = \lambda_{DU} + \lambda_{DD} \quad \lambda_{DU} = \lambda_D(1 - DC) \quad \lambda_{DD} = \lambda_D DC \tag{220}$$

The channel can be considered made of two components, one with a dangerous failure rate λ_{DU} (resulting from undetected failures) and the other with a dangerous failure rate λ_{DD} (resulting from detected failures).
It is possible to calculate the channel equivalent mean down time t_{CE} by the sum of the individual down times of the two components, t_{c1} and t_{c2}, taking into account the contribution of each component to the probability of failure of the channel:

$$t_{CE} = t_{C1} + t_{C2} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{2} + MRT\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR \tag{221}$$

The PFD with down-time t_{CE} resulting from dangerous failures is:

$$PFD = 1 - e^{-\lambda_D t_{CE}} \cong \lambda_D t_{CE} \tag{222}$$

The average PFD for 1oo1 architecture is:

$$PFD_{avg} = (\lambda_{DU} + \lambda_{DU}) t_{CE} \tag{223}$$

B. 1oo2

This architecture consists of two channels connected in parallel and each channel can process the safety function. So there would have to be a dangerous failure in both channels before a safety function failed on demand.

$$t_{GE} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{3} + MRT\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR \tag{224}$$

The average PFD for 1oo2 architecture is:

$$PFD_{avg} = 2\left(\left(1-\beta_D\right)\lambda_{DD} + \left(1-\beta\right)\lambda_{DU}\right)^2 t_{CE}t_{GE} + \beta_D\lambda_{DD}MTTR + \beta\lambda_{DU}\left(\frac{T_1}{2} + MTR\right) \tag{225}$$

C. 2oo2

This architecture consists of two channels connected in parallel but both channels need to demand the safety function before it can take place. The value of t_{CE} is the same of 1oo1 architecture:

$$t_{CE} = t_{C1} + t_{C2} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{2} + MRT\right) + \frac{\lambda_{DD}}{\lambda_D}MTTR \tag{226}$$

The average PFD for 2oo2 architecture is:

$$PFD_{avg} = (\lambda_{DU} + \lambda_{DU})t_{CE} \tag{227}$$

D. 1oo2D

The detected safe failure rate for every channel is given by:

$$\lambda_{SD} = \lambda_S DC \tag{228}$$

$$t_{CE} = \frac{\lambda_{DU}\left(\frac{T_1}{2} + MRT\right) + \left(\lambda_{DD} + \lambda_{SD}\right)MTTR}{\lambda_{DU} + \left(\lambda_{DD} + \lambda_{SD}\right)} \qquad t_{GE} = \frac{T_1}{3} + MRT \tag{229}$$

The average PFD for 1oo2D architecture is:

$$\begin{aligned} PFD_{avg} = {} & 2\left(1-\beta\right)\lambda_{DU}\left(\left(1-\beta\right)\lambda_{DU} + \left(1-\beta_D\right)\lambda_{DD} + \lambda_{SD}\right)t_{CE}t_{GE} + \\ & + 2\left(1-K\right)\lambda_{DD}t_{CE} + \beta\lambda_{DU}\left(\frac{T_1}{2} + MTR\right) \end{aligned} \tag{230}$$

E. 2oo3

This architecture consists of three channels connected in parallel with a major voting strategy: the safety function is required in case at least two channels demand it and the system state is not changed if only one channel gives a different result which disagrees with the other two channels.

$$t_{CE} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{2} + MTTR\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR \tag{231}$$

$$t_{GE} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{3} + MTTR\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR \tag{232}$$

The average PFD for 2oo3 architecture is:

$$PFD_{avg} = 6\left(\left(1-\beta_D\right)\lambda_{DD} + \left(1-\beta\right)\lambda_{DU}\right)^2 t_{CE} t_{GE} + \beta_D \lambda_{DD} MTTR + \beta \lambda_{DU}\left(\frac{T_1}{2} + MTR\right) \tag{233}$$

F. 1oo3

This architecture consists of three channels connected in parallel with a voting device in series. The output state follows 1oo3 voting.

It is assumed that any diagnostic testing would only report the faults found and would not change any output states or change the output voting.

$$PFD_{avg} = 6\left(\left(1-\beta_D\right)\lambda_{DD} + \left(1-\beta\right)\lambda_{DU}\right)^3 t_{CE} t_{GE} t_{G2E} + \beta_D \lambda_{DD} MTTR + \beta \lambda_{DU}\left(\frac{T_1}{2} + MTR\right) \tag{234}$$

Where:

$$t_{G2E} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{4} + MRT\right) + \frac{\lambda_{DU}}{\lambda_D} MTTR \tag{235}$$

G. Case study for low demand mode of operation

Consider a safety loop composed by three sensors in 2oo3 architecture, two logic solvers voting 1oo2D and two final elements both in 1oo1.

The final elements are a single shut-down valve plus a single vent valve. Both the shut-down and vent valves need to operate in order to achieve the safety function so the actuator architecture is 2oo2.

The safety function requires a SIL 2 system and it is assumed a proof test period of one year and 8 hours MTTR [88].

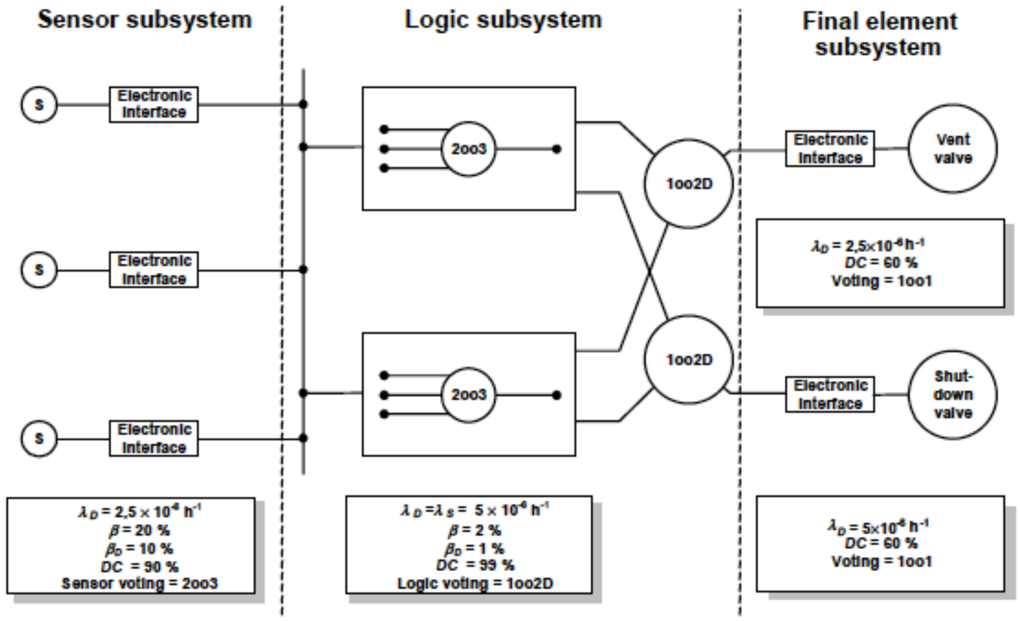


Fig. 3. Safety loop IEC 334/2000

Average probability of failure on demand for the subsystems in case of low demand mode of operation is achievable by dedicated tables in 61508, considering 2oo3 voting for sensors, 1oo2D voting for logic solvers and finally the sum of 1oo1 PFD$_{avg}$ for the final elements [88].

$$PFD_{avgSE} = PFD_{avg2oo3} = 2{,}3 \cdot 10^{-4} \tag{236}$$

$$PFD_{avgLS} = PFD_{avg1oo2D} = 4{,}8 \cdot 10^{-6} \tag{237}$$

$$PFD_{avgFE} = PFD_{avg1oo1} + PFD_{avg1oo1} = 4{,}4 \cdot 10^{-3} + 8{,}8 \cdot 10^{-3} = 1{,}3 \cdot 10^{-2} \tag{238}$$

$$PFD_{SYS} = PFD_{SE} + PFD_{LS} + PFD_{FE} = 1{,}3 \cdot 10^{-2} \tag{239}$$

This way the system meet SIL 1, so in order to improve safety integrity level, there are two possibilities:

- Reduce the proof test interval to six months;

$$\begin{aligned} PFD_{SYS} = PFD_{SE} + PFD_{LS} + PFD_{FE} = 1{,}1 \cdot 10^{-4} + 2{,}6 \cdot 10^{-6} + \\ +(2{,}2 \cdot 10^{-3} + 4{,}4 \cdot 10^{-3}) = 6{,}7 \cdot 10^{-3} \end{aligned} \tag{240}$$

- Change the 1oo1 shutdown valve (which is the output device with the lower reliability) to 1oo2 architecture;

$$PFD_{SYS} = PFD_{SE} + PFD_{LS} + PFD_{FE} = 2{,}2 \cdot 10^{-4} + 4{,}8 \cdot 10^{-6} + \\ +(4{,}4 \cdot 10^{-3} + 9{,}7 \cdot 10^{-4}) = 5{,}6 \cdot 10^{-3} \tag{241}$$

In both cases, SIL 2 requirement is achieved.

VII. Reliability Block Diagram approach for PFH

Reliability Block Diagram approach is used also for PFH: the method for calculating the probability of failure of a safety function for an E/E/PE safety related system operating in high demand or continuous mode of operation is the same of the one seen before for PFD in low demand mode of operation. Average frequency of dangerous failures can be expressed as follows [88]:

$$PFH_{SYS} = PFH_{SE} + PFH_{LS} + PFH_{FE} \tag{242}$$

A. 1oo1

The dangerous failure rate for the channel is given by:

$$\lambda_D = \lambda_{DU} + \lambda_{DD} \quad \lambda_{DU} = \lambda_D(1 - DC) \quad \lambda_{DD} = \lambda_D DC \tag{243}$$

The channel equivalent mean down time t_{CE} is the sum of the individual down times of the two components, t_{c1} and t_{c2}:

$$t_{CE} = t_{C1} + t_{C2} = \frac{\lambda_{DU}}{\lambda_D}\left(\frac{T_1}{2} + MRT\right) + \frac{\lambda_{DD}}{\lambda_D} MTTR \tag{244}$$

If it is assumed that the safety system puts the equipment under control into a safe state on detection of any failure, for a 1oo1 architecture the following PFH_{avg} is obtained:

$$PFH_{avg} = \lambda_{DU} \tag{245}$$

A. 1oo2

If it is assumed that the safety system puts the equipment under control into a safe state once there is detection of a failure in both channels and taking a conservative approach, the following PFH_{avg} is obtained:

$$PFH_{avg} = 2\big((1-\beta_D)\lambda_{DD} + (1-\beta)\lambda_{DU}\big)(1-\beta)\lambda_{DU} t_{CE} + \beta\lambda_{DU} \tag{246}$$

A. 2oo2

Assuming that each channel is put into a safe state on detection of any fault, for a 2oo2 architecture, the following PFH$_{avg}$ is obtained:

$$PFH_{avg} = 2\lambda_{DU} \tag{247}$$

A. 1oo2D

The detected safe failure rate for every channel is given by:

$$\lambda_{SD} = \frac{\lambda}{2} DC \tag{248}$$

$$t_{CE}' = \frac{\lambda_{DU}\left(\frac{T_1}{2} + MRT\right) + \left(\lambda_{DD} + \lambda_{SD}\right) MTTR}{\lambda_{DU} + \lambda_{DD} + \lambda_{SD}} \tag{249}$$

The average PFH for 1oo2D architecture is:

$$PFH_{avg} = 2\left(1-\beta\right)\lambda_{DU}\left(\left(1-\beta\right)\lambda_{DU} + \left(1-\beta_D\right)\lambda_{DD} + \lambda_{SD}\right)t_{CE}' + 2\left(1-K\right)\lambda_{DD} + \beta\lambda_{DU} \tag{250}$$

A. 2oo3

If it is assumed that the safety system puts the equipment under control into a safe state once there is detection of a failure in any two channels and taking a conservative approach, the following PFH$_{avg}$ is obtained:

$$PFH_{avg} = 6\left(\left(1-\beta_D\right)\lambda_{DD} + \left(1-\beta\right)\lambda_{DU}\right)\left(1-\beta\right)\lambda_{DU}t_{CE} + \beta_D\lambda_{DU} \tag{251}$$

A. 1oo3

If it is assumed that the safety system puts the equipment under control into a safe state once there is detection of a failure in any two channels and taking a conservative approach, the following PFH$_{avg}$ is obtained:

$$PFH_{avg} = 6\left(\left(1-\beta_D\right)\lambda_{DD} + \left(1-\beta\right)\lambda_{DU}\right)^2\left(1-\beta\right)\lambda_{DU}t_{CE}t_{GE} + \beta_D\lambda_{DU} \tag{252}$$

A. Case study for high demand mode of operation

Consider the safety loop described in the previous paragraph composed by two sensors in 1oo2 architecture, three logic solvers voting 2oo3 and one final elements in 1oo1.

The final elements is a single shut-down contactor.

The safety function requires a SIL 2 system and it is assumed a proof test period of 6 months and 8 hours MTTR.

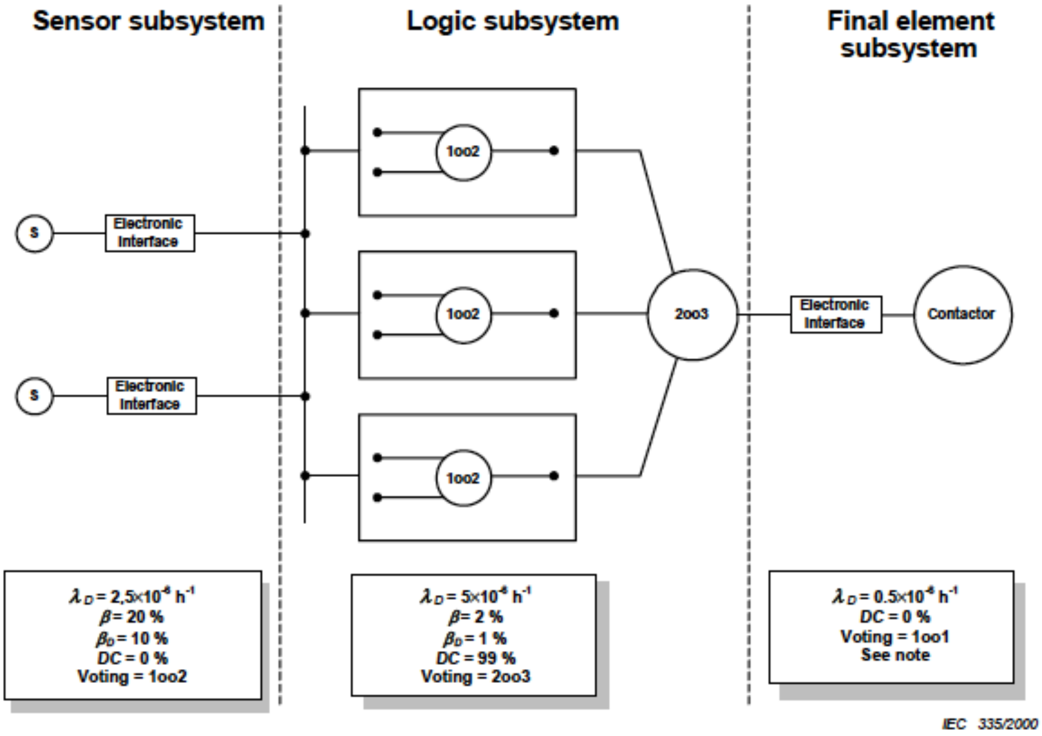


Fig. 4. Safety loop IEC 335/2000

Average probability of failure on demand for the subsystems in case of low demand mode of operation is achievable by dedicated tables in 61508, considering 1oo2 voting for sensors, 2oo3 voting for logic solvers and finally 1oo1 PFD$_{avg}$ for the final element [88].

$$PFD_{avgSE} = PFD_{avg1oo2} = 5{,}2 \cdot 10^{-7} / h \tag{253}$$

$$PFD_{avgLS} = PFD_{avg2oo3} = 1{,}0 \cdot 10^{-9} / h \tag{254}$$

$$PFD_{avgFE} = PFD_{avg1oo1} = 5{,}0 \cdot 10^{-7} / h \tag{255}$$

$$PFD_{SYS} = PFD_{SE} + PFD_{LS} + PFD_{FE} = 1{,}02 \cdot 10^{-6} / h \tag{256}$$

This way the system meet SIL 1, so in order to improve safety integrity level, there are two possibilities:

- Change the input sensor type in order to improve the defenses against CCF (e.g. improving β from 20% to 10% and β_D from 10% to 5%);

$$PFD_{SYS} = PFD_{SE} + PFD_{LS} + PFD_{FE} = \left(2,7\cdot10^{-7} + 1,0\cdot10^{-9} + 5,0\cdot10^{-7}\right)/h = 7,7\cdot10^{-7}/h \tag{257}$$

- Change the single output device to 1oo2 architecture (β = 10% and β_D = 5%);

$$PFD_{SYS} = PFD_{SE} + PFD_{LS} + PFD_{FE} = \left(2,7\cdot10^{-7} + 1,0\cdot10^{-9} + 5,0\cdot10^{-7}\right)/h = 7,7\cdot10^{-7}/h \tag{258}$$

In both cases, SIL 2 requirement is achieved.

A. ISA-TR84.0.02 Standard

This paragraph shows the PFD-calculation according to another standard, ISA-TR84.0.02, that includes a specific guidance in the application of SIS, providing methodologies for evaluating SIF and corresponding SIL.

In ISA-TR84.0.02 [86] PFD equations are divided in two categories: with and without common cause factor.

Equations for 1oo1, 1oo2 and 2oo3 architectures are listed below, the first equation takes into account CCFs and MTTR, the second is the simplified one:

- 1oo1

$$PFD_{avg} = \lambda_{DU}\cdot\frac{T_1}{2}\ ; \quad PFD_{avg} = \lambda_{DU}\cdot\frac{T_1}{2} \tag{259}$$

- 1oo2

$$PFD_{avg} = \left[\left(\lambda_{DU}\right)^2\cdot\frac{T_1^2}{3}\right] + \left[\lambda_{DU}\cdot\lambda_{DD}\cdot MTTR\cdot T_1\right] + \left[\beta\cdot\lambda_{DU}\cdot\frac{T_1}{2}\right]; \quad PFD_{avg} = \left(\lambda_{DU}\right)^2\cdot\frac{T_1^2}{3} \tag{260}$$

- 2oo3

$$PFD_{avg} = \left[\left(\lambda_{DU}\right)^2\cdot T_1^2\right] + \left[3\lambda_{DU}\cdot\lambda_{DD}\cdot MTTR\cdot T_1\right] + \left[\beta\cdot\lambda_{DU}\cdot\frac{T_1}{2}\right]; \quad PFD_{avg} = \left(\lambda_{DU}\right)^2\cdot T_1^2 \tag{261}$$

In ISA-TR84.0.02 there is no definition of SFF and DC parameters, there is not differentiation between Type-A and Type-B devices and between β and β_D either.
In the IEC 61508 all these factors are considered comparing to the ISA-standard.
However using both standards PFD values are in the same ranges, as long as not the simplified calculations of the ISA-standard are applied.

VIII. SIS Proof and Diagnostic Tests

A Safety Instrumented System (SIS) is usually subjected to periodical diagnostic measures.
Many SISs are only activated when a process demand occurs in the equipment under control and as a result some dangerous failures in SISs cannot be found until the systems are activated or tested.
There are two main categories of tests: proof test and diagnostic test.

A. Proof Tests

A Proof Test (PT) is a periodic test performed to detect dangerous hidden failures in a safety-related system so that, if necessary, a repair can restore the system to an "as good as new" or "as good as possible" depending on practical conditions [87].
In other words, a proof test is a form of stress test with the aim of demonstrating the fitness of equipment. Usually it is achieved on a single unit and the structure is subjected to loads above that expected in actual use, demonstrating safety and design margin. Anyway proof testing is nominally a non-destructive test if both design margins and test levels are well-chosen.
Proof tests are always conducted at regular intervals, e.g. once per year, therefore the EUC is not protected by the SIS from the moment when a dangerous undetected failure occurs until the subsequent proof test; redundant structures are often applied in SISs in order to achieve a high reliability.
For safety valves, a functional test means to perform a full stroke operation (Full Valve Stroke Test) however many automatic procedure were introduced to replace the need for offline testing.
The most common procedure to test final elements is the partial valve stroke test (PVST). This is a technique used in control systems to cover a percentage of the possible failure modes of a shut down valve without closing the valve.
Conducting a full system proof test is usually possible only when the process is shut down. In order to reduce the frequency of such complete tests, the PVST are the best solution.
Many dangerous undetected failure modes can be detected just with a small valve movement that would not affect system availability and productivity.
Partial stroke test is necessary to guarantee that the safety function will operate on demand but it is not sufficient so the need of proof testing (fully stroke valves) is still mandatory.

Partial valve proof tests must be performed periodically because they are the only method to discover dangerous undetected failures and avoid severe consequences with sufficient high probability.

The frequency to conduct these tests depends on the component's average probability of failure on demand (PFDavg): higher test frequency means lower PFDavg and higher risk reduction factor (RRF).

In the statistical mean the dangerous undetected failures occur at half of the interval between two tests (proof test interval), so:

$$PFD = \lambda_D \cdot \frac{T_1}{2} \tag{262}$$

There are several different types of partial valve stroke testing.

The simplest method involves a PLC and a dedicated switch: supposing that a remote actuated valve is energized during normal operation, during the test the digital output of the corresponding PLC is de-energized and the dedicated switch is used to detect any valve movement as a result of the momentary off pulse.

A more sophisticated approach pulses the solenoid and measures the pressure response waveform: default patterns in the pressure response indicate a failure condition of the valve.

Analog methods usually offer better results to detect failures rather than simple digital feedback techniques; the effectiveness of the detection depends on many factors e.g. PVST methodology, application conditions, and shut-off requirements.

A partial stroke test is a delicate procedure in particular in high energy and high flow applications where the PVST could generate a response (and instabilities) in the process control system or in the safety instrumented system leading to a spurious trip.

Table V shows coverage factors for the more sophisticated partial valve stroke techniques for different product types; these results were achieved with FMEDA analysis [86-89].

Obviously the application of a valve affects failure rates, failure modes and coverage factor: energize or de-energize to trip, open or close to trip.

Two primary categories of service have been used, clean service and severe service. The first means that the fluid involved in the process is a clean gas or a fluid without particulate or droplets of water; the second means that the valve is exposed to particulates, abrasives and/or corrosive gases or fluids that produces higher stress levels [87].

FVPT procedure consist of a full stroke of the valve following these steps:

- Bypass the safety function and take appropriate action to avoid a false trip;
- Interrupt (or change) the signal to the actuator to force the valve to achieve the fail-safe condition with the required time;
- Re-store the standard signal to the actuator in order to achieve the normal operating state;
- Inspect the valve for any visible damage, leaks or contamination;
- Remove the bypass;

- Confirm the valve movement;
- Monitor the travel of the valve and slew rate comparing the results with expected outcomes to validate the test.

Table V. Partial Valve Stroke capability

Product Type	Application	Partial Valve Stroke Dangerous Coverage Factor
Solenoid	De-energize to trip	99.0%
Pnuematic Piston Actuator, clean service	De-energize to trip	99.3%
Pneumatic Piston Actuator, severe service	De-energize to trip	99.6%
Pneumatic Rack & Pinion Actuator, clean service	De-energize to trip	81.9%
Pneumatic Rack & Pinion Actuator, severe service	De-energize to trip	88.0%
Scotch Yoke Actuator, clean service	De-energize to trip	92.6%
Scotch Yoke Actuator, severe service	De-energize to trip	94.0%
Gate Valve, clean service	Close to trip	87.9%
Gate Valve, severe service	Close to trip	84.9%
Ball Valve, severe service, full stroke only	Close to trip	45.2%
Ball Valve, severe service, tight shut-off	Close to trip	22.2%
Resilient Butterfly Valve, clean service	Open to trip	63.6%
Resilient Butterfly Valve, clean service	Close to trip	53.8%

Full Valve Stroke Tests (FVST) offer greater levels of diagnostic coverage. The standard setup for on-line achievement of this kind of test requires two valves piped in a 2oo2 architecture: one valve can be completely closed while the other remains open to assure continued process operation. However FVST are usually done during scheduled system shutdowns.

The three diagrams that follow illustrate how more-frequent testing can reduce PFDavg or extend the intervals between full proof tests.

Probability of failure on demand (PFD) increases over time but returns to its original level when a full proof test is done to prove that everything works as expected.

Running the same partial test twice as often lowers the average PFD: this strategy allow design engineer to meet higher SIL requirement using the same equipment or choose cheaper one to achieve the same SIL.

Another approach is to run partial stroke tests frequently in order to double the full proof test interval and maintain the same average PFD [87].

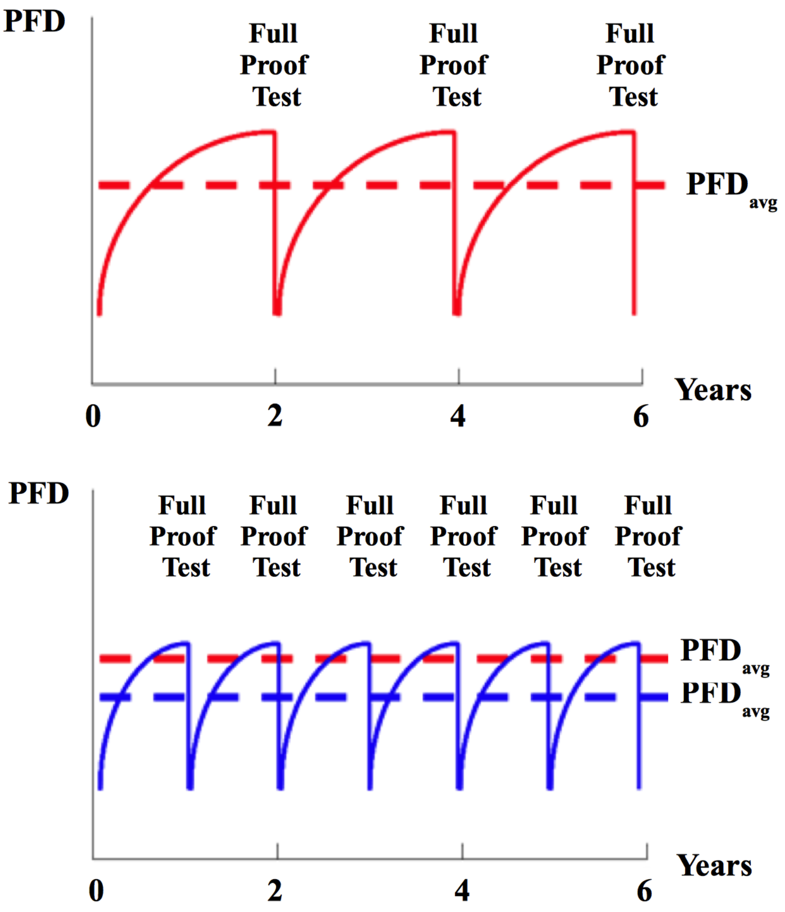


Fig. 5. PFD trend in case of full proof test procedure

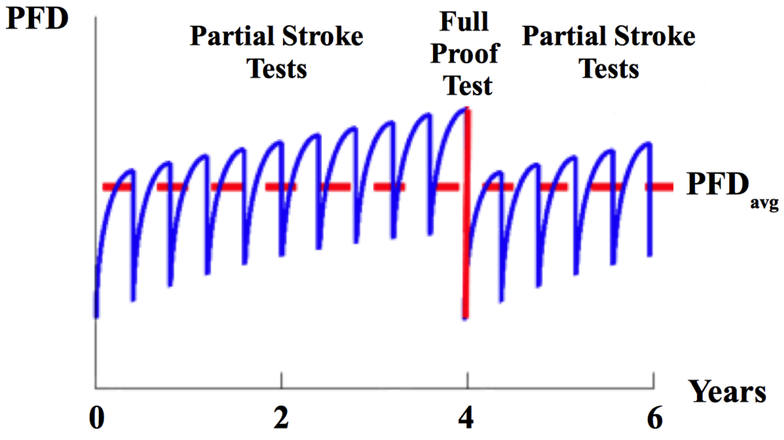


Fig. 6. PFD trend in case of full proof test procedure and partial stroke test

Table VI. Dangerous valve failure modes, effects and corresponding test strategy

Failure Modes	Effects	Test Strategy
Actuator sizing is insufficient to actuate valve in emergency conditions	Valve fails to close (or open)	Not tested
Valve packing is seized	Valve fails to close (or open)	Test valve – Partial or full-stroke
Valve packing is tight	Valve is slow to move to closed or open position	Not tested unless speed of closure is monitored
Air line to actuator crimped	Valve is slow to move to closed or open position	Not tested unless speed of closure is monitored. Physical inspection
Air line to actuator blocked	Valve is slow to move to closed or open position	Test valve – Partial or full-stroke
Valve stem sticks	Valve fails to close (or open)	Test valve – Partial or full-stroke
Valve seat is scarred	Valve fails to seal off	Full-stroke test with leak test
Valve seat contains debris	Valve fails to seal off	Full-stroke test
Valve seat plugged due to deposition or polymerization	Valve fails to seal off	Full-stroke test

In IEC 61508 [88] and IEC 61511 [89] dangerous failures (λ_D) are divided in detected (λ_{DD}) and undetected (λ_{DU}) depending on the effectiveness of the diagnostic equipment; the Diagnostic Coverage (DC) factor is the fraction of dangerous failures that are detected by the diagnostics beside all dangerous failures and, at the same time, it is the conditional probability that a dangerous failure is detected by the diagnostics (in presence of a dangerous failure).

$$DC = \frac{\lambda_{DD}}{\lambda_D} \tag{263}$$

As said before, in order to reveal potential dangerous undetected failures, final elements (in particular safety valves) are usually tested with partial stroke tests and, to a lower extend, with functional tests (or full stroke tests): the "standard" failure rate λ_{DU} may be split into $\lambda_{DU,PVST}$ and $\lambda_{DU,FVST}$.

Since PVST is not always successful it is necessary to introduce a dedicated coverage factor, the Partial Valve Stroke Test Coverage (PC).

Similarly for Diagnostic Coverage, the PVST coverage has two different interpretations: it is defined as the fraction of dangerous undetected failures detected by the partial valve stroke test and the total number of dangerous undetected failures and, at the same time, it is the conditional probability that a dangerous undetected failure is detected by the PVST (once a dangerous undetected failure is present).

$$PC = \frac{\lambda_{DU,PVST}}{\lambda_{DU}} \tag{264}$$

A word of clarification, as said before, the distinction “detected/undetected” is associated just with diagnostic tests (and corresponding DC factor); PVST coverage concerns only dangerous failures that are undetected by the diagnostics and that may be revealed by stroke tests. In other words, PVST is a procedure of detection of a part of normally undetected dangerous failures in absence of partial valve stroke testing.

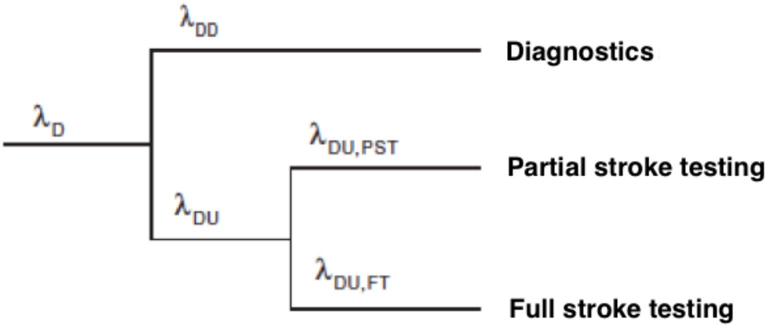


Fig. 7. Dangerous failure rate and test procedures

When PVST is not implemented, the average probability of failure on demand of the safety valve (supposing a low demand application) is the following:

$$PFD = PFD_{FVST} + PFD_{DT} \cong \frac{\lambda_{DU} \cdot \tau_{FVST}}{2} + \frac{\lambda_{DD} \cdot \tau_{DT}}{2} \qquad (265)$$

Where τ_{FVST} and τ_{DT} are the full stroke test interval and the diagnostic test interval respectively.
IEC 61508 and IEC 61511 recommend that the diagnostic test interval is taken into account through the MTTR.

In case a PVST is implemented, it is a valid supplement to full-stroke testing to reduce the block valve PFD. The amount of the reduction is dependent on the valve and its application; PVST may detect a fraction of the dangerous undetected failures corresponding to the PVST coverage:

$$PFD = PFD_{FVST} + PFD_{PVST} + PFD_{DT} \cong (1 - PC)\frac{\lambda_{DU} \cdot \tau_{FVST}}{2} + PC\frac{\lambda_{DU} \cdot \tau_{PVST}}{2} + \frac{\lambda_{DD} \cdot \tau_{DT}}{2} \qquad (266)$$

Where τ_{PVST} is the Partial Valve Stroke Test interval. The PFD with and without PVST is shown in Fig. 8.

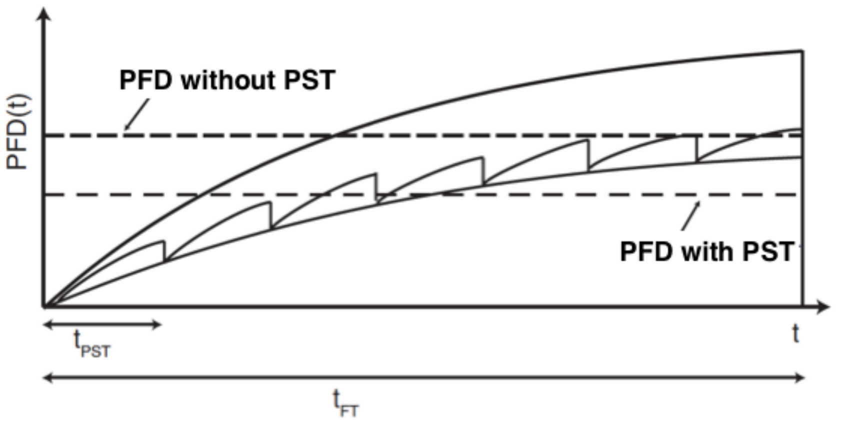


Fig. 8. PFD comparison with/without PST

As expected the average PFD is lower when partial valve stroke tests are implemented because a portion of dangerous undetected failures may be detected in a shorter time interval rather than by functional or full valve stroke testing. The PFD improvement depends on how frequent the PVST is performed compared to the functional test interval.

The probability of failure on demand is the unknown unavailability of the safety valve. Unavailability may be “known” in case a failure has been detected by diagnostics, PVST, functional test or a real demand; this is known unavailability or downtime unavailability (DTU):

$$\begin{aligned} DTU &= DTU_{FVST} + DTU_{PVST} + DTU_{DT} \cong \\ &= (1 - PC)\lambda_{DU} MTTR_{FVST} + PC\lambda_{DU} MTTR_{PVST} + \lambda_{DD} MTTR_{DT} \end{aligned} \tag{267}$$

Where MTTRFT, MTTRPVST and MTTRDT are the Mean Time To Repair in case of failures detected by functional testing, partial valve stroke testing, and diagnostic testing, respectively.

A. Diagnostic Tests

A diagnostic test, instead, is performed periodically to detect some of the dangerous faults that prevent the SIS from responding to a demand.

Some SISs may conduct self-diagnostic testing during operation in order to detect some dangerous failures immediately when they occur.

Diagnostic tests may be accomplished using a variety or combination of methods, such as:

- Monitoring hardware integrity (e.g., impedance monitoring in thermocouples)
- Selecting devices that have internal diagnostic capability (e.g., input/output module self-tests)

- Incorporating external diagnostic capability through design (e.g., automated testing of solenoid valves; partial stroke testing of isolation valves; or comparison of redundant analog signals)
- Using watchdog timers
- Using end-of-line monitoring

Diagnostic tests can detect only a fraction of all dangerous failures (detected dangerous). Assume therefore that at the end of a time interval a Proof Test is performed: the strength of the proof test will be dependent both on failure coverage and repair effectiveness. In any case 100% detection of hidden dangerous failures is easily achievable only for low-complexity E/E/PE safety-related systems.

For this reason the dangerous failure rate (λ_D) is split in detected (λ_{DD}, e.g. failures detected by a partial stroke test on a valve) and undetected (λ_{DU}, failures not detected by a partial stroke test but detected by a roof test at the end of the scheduled interval).

As seen before, the Diagnostic Coverage factor DC is defined as the fraction of dangerous failures detected by automatic on-line diagnostic tests:

$$DC = \frac{\lambda_{DD}}{\lambda_{DD} + \lambda_{DU}} \tag{268}$$

Diagnostic tests are usually divided in three categories: manual, automatic and semiautomatic.

Fig. 9 shows the formula for System Diagnostic Coverage consisting of N subsystems: system DC depends on the DC factor of each subsystems and on the parameters characterizing the manual and the automatic parts of the diagnostic test [86].

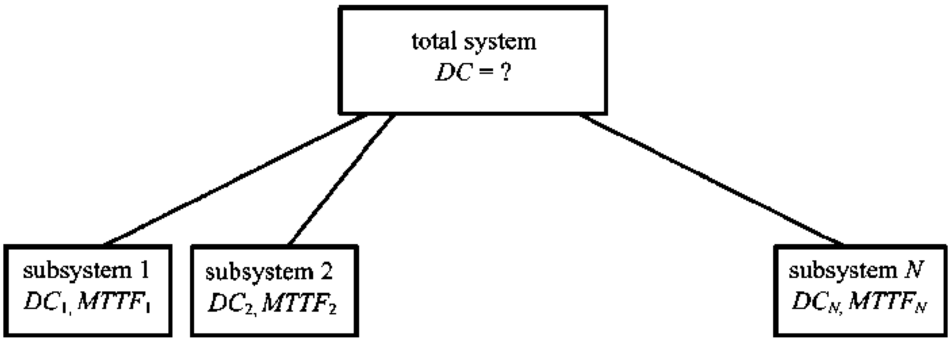


Fig. 9. System Diagnostic Coverage assessment

The choice between manual and automatic tests is a trade-off in terms of costs and DC factor requirements.

In compliance with EN ISO 13849-1, in a system with n-subsystems with known diagnostic coverage factor (DC_i) and mean time to dangerous failure ($MTTF_{Di}$), the average diagnostic coverage factor (DC_{avg}) of the total system may be calculated as the weighted mean of the diagnostic coverage factors of subsystems with weight factors 1/ $MTTF_{Di}$:

$$DC_{avg} = \frac{\frac{DC_1}{MTTF_{D1}} + \frac{DC_2}{MTTF_{D2}} + \ldots + \frac{DC_n}{MTTF_{Dn}}}{\frac{1}{MTTF_{D1}} + \frac{1}{MTTF_{D2}} + \ldots + \frac{1}{MTTF_{Dn}}} \tag{269}$$

$$MTTF = \frac{1}{\lambda_{DD} + \lambda_{DU} + \lambda_{SD} + \lambda_{SU}} ; \quad MTTF_D = \frac{1}{\lambda_{DD} + \lambda_{DU}} \tag{270}$$

Supposing that for each subsystem a set of different diagnostic measures is available (manual, automatic or semi-automatic). In order to quantify these diagnostic modes mathematically, weighting factors are now introduced:

α_{comp}: weight of automatically tested components;

$\mu_{comp} = 1 - \alpha_{comp}$: weight of manually tested components.

If the subsystem is assembled by n components, this means that "$\alpha_{comp} \cdot n$" components are tested automatically (n_a) and "$\mu_{comp} \cdot n$" components are tested manually (n_m).

This way there are three different diagnostic modes:

- automatic diagnostic mode, all components are tested automatically
  $\alpha_{comp} = 1$, $\mu_{comp} = 0$;
- manual diagnostic mode, all components are tested manually
  $\alpha_{comp} = 0$, $\mu_{comp} = 1$;
- semi-automatic diagnostic mode, some components are tested automatically, some manually
- $0 < \alpha_{comp} < 1$, $0 < \mu_{comp} < 1$.

In SIS applications, diagnostics can be divided in two groups: reference diagnostics for non-redundant architectures and comparison diagnostics for redundant ones [88-89].

Former method is based on the comparison between an operating value and a predetermined reference; it can be assessed by a single unit and the coverage factor varies widely between 0.6 to 0.999 depending on the application.

The second method is used in redundant architectures because it requires the comparison between outcomes of different operational units. The coverage factor depends on the implementation but standard range goes from 0.9 to 0.999.

In a dual configuration, any disagreement may identify a fault. In a triple configuration, a voting circuit is usually used to identify drifts or disagreements of one device (a disagreement likely corresponds to a failure).

A. Field Device Coverage

Diagnostics must be extended to all field devices and associated wiring: this is mandatory for process sensors and other field devices (e.g. valves) in order to preserve redundancy benefits and not impair system failure rate.

For this reason, a complete system safety and reliability analysis must include valves, sensors, field transmitters, limit switches, solenoids, and other devices, along with associated wiring, junction boxes and connections.
In case a field device is equipped with on-board microprocessor is called “smart device” and it allows diagnostic capabilities.
Usually in SISs the finale element is a remote-actuated valve that is activated when a potential dangerous condition arises. In many processes this is quiet a rare event and the valve may sit motionless for a very long time (e.g. years).
There are many different failures that can cause the valve to stick, for example cold welding of O-rings and seals and corrosion between moving parts. A partial stroke test can be set up to test the valve e.g. moving the valve a small amount) and indicate the failure: this procedure can detect most of dangerous failures in the final elements.

Failure detection can be assessed in some field devices with on-board diagnostics using sensor output current: for instance, if the average output current exceeds an upper threshold for too long, this trend usually corresponds to a short circuit failure of the load device or a faulty field wiring. Otherwise, if a channel is operative and a minimum current is not being detected, this indicates an open circuit failure of device or associated wiring.

For example, in solenoid-operated valves (electro-valve) a common failure is the coil burn-out that arises especially in normally energized SIS applications where the coil is energized 24h/day. The coil can reach very high temperatures: this stress can break down the insulation and produce a short out adjacent windings in the coil.
These circumstances produce an increase of the current consumption and temperatures giving rise to an eventual burn-out of the coil.
Diagnostic equipment on the coil can detect the failure before the burn-out and, in case the repair can be assessed soon enough, a system false trip can be avoided [86-89].

A. Testing Policy

The testing policy defines the way to test components in redundant architectures in order to minimise PFD_{avg}. The most important policies are the following:

- Simultaneous
  With the simultaneous policy all components are put off-line and tested following a defined scheduling; during the test the safety function is not available.
- Sequential
  With the sequential policy the components are tested at fixed intervals but one after the other in such a way that only one component at a time is off-line for testing.
- Staggered

In the staggered policy components are tested regularly in overlapping sequence e.g. given n components in parallel configuration, each component is tested every “k” hours, but the time between two tests is “n/k” hours.

B. Proof Test and Diagnostics on 1oo2 Shutdown Valves

If two shutdown valves are installed in 1oo2 architecture, the flow can be stopped in case of emergency when any of them is able to close on demand.

Given one valve has dangerous undetected failure, the system is still functional if the other one can work well.

Both of two components in a 1oo2 system can have dangerous detected failures, and if such a failure is detected, the system becomes a 1oo1 SIS during the repair: the other component is assumed to perform SIF until the failed one is restored.

After restoration activities the maintenance team can select to conduct a proof test on this newly fixed component and check whether there is a hidden failure.

Then there are maintainability options available for the other component: test the component for proof as soon as possible or test the component following the regular proof test interval [87-88].

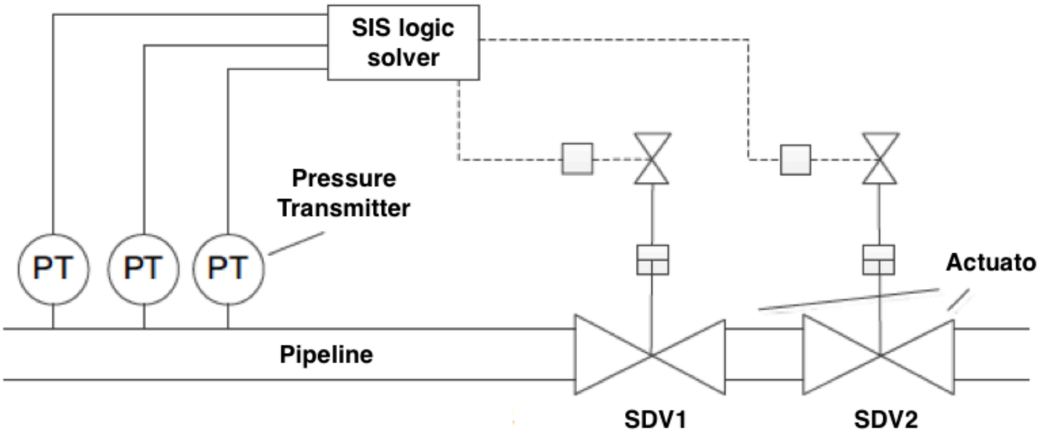


Fig. 10. 1oo2 shutdown valves

There are different strategies in case of dangerous detected failure arising in redundant architectures: each potential test strategy needs to be identified and modelled with proper methods, so as to measure their impacts on the reliability of SIS.

For a 1oo2 SIS, such as the two shutdown valves in a high integrity pressure protection system for a pipeline, two components in the system are normally tested one by one in proof tests.

If a dangerous detected failure occurs in one valve, the maintenance team can fix the failure and have a proof test on the valve: then they can decide whether or not to conduct a proof test on another valve. This decision is usually taken on the working conditions and available maintenance resources.

There are three main strategies possible to be adopted by the SIS operators in case of dangerous detected failure in 1oo2 architecture:

- Do not test the other component until the subsequently regular proof test, keep following the current test strategy;

- Test the other component and keep following the current proof test schedule;
- Test the other component and change the test scheduling, usually postponing the following proof test.

In all these three strategies, the successive proof test will cover both components.

In case a proof test is conducted on the valve without dangerous detected failures, it can be considered as an additional proof test between two regular ones; since such test also guarantee to find all hidden failures in the tested valve, it is necessary to decide whether the original proof test schedule needs some adjustments [86].

For example, supposing that proof tests are initially planned once a year at June 1st. If an additional proof test is done at March 1st, maintenance team should decide to schedule the next proof test on June 1st or March 1st next year.

C. Benefits of Diagnostics on SISs

Both redundant and non-redundant repairable control systems have improved availability and safety in case on-line diagnostic is provided. Other benefits are the reduction of time the system operates in dangerous and degraded (not completely operational) mode.

Safety is improved by diagnostic coverage even in a non-redundant architecture.

In a normally energized safety protection application, if a standard 1oo1 PLC architecture fails with outputs de-energized, the process is inadvertently shut down (false trip). Usually to detect a process shut down is not required on-line diagnostics because a false trip is usually quite apparent. However, if 1oo1 PLC fails with output energized, it cannot respond to demand in case of danger. The process keeps operating with no safety protection and there is no indication that something is faulty.

The main added value of diagnostics is the detection of dangerous failures to allow a quick repair and restore of the system.

In case of failure in a redundant architecture (e.g. 1oo2 PLC configuration) diagnostics reduces the time spent in the degraded mode: the output of PLC modules is wired in series so if one module fails, the other can still provide a safety protection function e.g. energizing the load (in a normally energized protection application).

So diagnostics improve the safety of this architecture because if one module fails dangerously, the system is degraded and a second dangerous failure is required to cause the system to fail. At the same time, diagnostic capability will also allow quick repair and minimize the amount of time the system operates in a degraded mode [86-89].

Bibliography

[1] M. Catelani, L. Ciani, S. Rossin, M. Venzi, "Failure rates sensitivity analysis using Monte Carlo simulation", Proc of 13th IMEKO TC10 Workshop on Technical Diagnostics - "Advanced measurement tools in technical diagnostics for systems' reliability and safety, June 26-27, 2014, Warsaw, Poland, pp.195-200.

[2] M. Catelani, L. Ciani, M.Venzi, "Improved RBD analysis for reliability assessment in industrial application", Proc. Of IEEE - International Instrumentation And Measurement Technology Conference (I2MTC) - Montevideo (Uruguay) - May 2014, pp. 670-674

[3] M. Venzi, S. Rossin, C. Michelassi, C. Accillaro, M. Catelani, L. Ciani, "Improved FBD and RBD generation for system reliability assessment" , Proc of 12th IMEKO TC10 Workshop on Technical Diagnostics: New Perspective in Measurements, Tools and Techniques for Industrial Applications, Florence (Italy), June 2013, pp. 266-270.

[4] Y. Zhang, C. Bingham, Z. Yang, B. Wing-Kuen Ling, M. Gallimore, "Machine fault detection by signal denoising—with application to industrial gas turbines", Measurement, Volume 58, December 2014, Pages 230-240, ISSN 0263-2241,

[5] Zhijing Yang, B. Wing-Kuen Ling, C. Bingham, "Fault detection and signal reconstruction for increasing operational availability of industrial gas turbines", Measurement, Volume 46, Issue 6, July 2013, Pages 1938-1946, ISSN 0263-2241,

[6] M. Catelani, L. Ciani, V. Luongo, "Safety Analysis in Oil &Gas Industry in compliance with Standards IEC61508 and IEC61511: Methods and Applications" Proc. Of IEEE - International Instrumentation And Measurement Technology Conference (I2MTC) – Minneapolis (USA) – May 2013, pp. 686-690

[7] M. Catelani, L. Ciani, V. Luongo, "A new proposal for the analysis of Safety Instrumented Systems" Proc. Of IEEE - International Instrumentation And Measurement Technology Conference (I2MTC) - Graz (Austria) - May 2012, pp. 1612-1616.

[8] M. Catelani, L. Ciani, V. Luongo, "A simplified procedure for the analysis of Safety Instrumented Systems in the process industry application", Microelectronics Reliability, Volume 51, Issues 9-11, September-November 2011, Pages 1503-1507, ISSN 0026-2714, 10.1016/j.microrel.2011.07.044.

[9] M. Rausand, A. Hoyland, "System Reliability Theory", John Wiley & Sons Inc. Publication, 2004

[10] J. Møltoft, "Behind the "bathtub"-curve A new model and its consequences", The Engineering Academy of Denmark, Department of Electrical Engineering, Building 451, 2800 Lyngby, Denmark, February 2003

[11] Barlow, R. and Proschan, F. Statistical theory of reliability and life testing", Holt, Rinehart & Winston, 2005

[12] A. Castellani, "Teoria Dell'Affidabilità", Pearson Education Italia 2009

[13] Beard, R. "Failure accommodation in linear systems through self reorganization", Technical Report MVT-71-1, Man Vehicle Laboratory, Cambridge, MA, 2011.

[14] Chen, J. and Patton, R. Robust model-based fault diagnosis for dynamic systems, Kluwer, Boston, 1999

[15] W. Wessels, "Use of the Weibull versus Exponential to Model Part Reliability", University of Alabama, January 2007

[16] S. Beretta, "Affidabilità nelle costruzioni meccaniche, strumenti e metodi per l'affidabilità di un progetto", Springer, Milan 2009

[17] Balakrishnan N., Kateri M., On The Maximum Likelihood Estimation of parameters of Weibull distribution based on complete censored data, December 2008

[18] PTC Incorporation, "Reference Guide PTC Windchill Quality Solutions™ 10.2M060", Needham, USA, 2015

[19] I. Pobočíková and Z. Sedliačková, "Comparison of Four Methods for Estimating the Weibull Distribution Parameters", Žilina, Slovakia, June 2014

[20] M. Catelani, L. Ciani, M. Venzi, G. Guidi, "Parameter estimation methods for failure rate distributions", 14th IMEKO TC10 Workshop Technical Diagnostics New Perspectives in Measurements, Tools and Techniques for system's reliability, maintainability and safety, Milan, Italy, June 27-28, 2016

[21] A. Albertini, G. Mazzanti, L. Peretto, R. Tinarelli, "Development of a Life Model for Light Emitting Diodes Stressed by Forward Current", Bologna, June 2014

[22] A. Thiraviam, T. Foley, L. Malone, "Development of an Acceleration Model for Subsea Pressure", Florida , 2010

[23] MIL-HDBK-217F, Department Of Defense Washington DC, December 2001

[24] IEC 61078 (2006). Analysis techniques for dependability - Reliability block diagram and boolean methods.

[25] M. Catelani, L. Ciani, M. Venzi, "Component Reliability Importance assessment on complex systems using Credible Improvement Potential", Microelectronics Reliability, Volume 64, September 2016, Pages 113-119, ISSN 0026-2714,

[26] K. Fowler, "Dependability [reliability]", IEEE Instrumentation & Measurement Magazine, vol.8, no.4, pp.55,58, Oct. 2005.

[27] MIL-HDBK 338B (1998) - Electronic Reliability Design Handbook, Department of defense Washington DC 20301.

[28] M.Rausand, A. Høyland, "Reliability of Safety-Critical Systems", New Jersey: J.Wiley & Sons, Inc., Hoboken, 2014

[29] Blanke, M., Kinnaert, M., Lunze, J., and Staroswiecki, M. "Diagnosis and fault tolerant control", Springer, Berlin, 2nd edition, 2006.

[30] ReliaSoft Corporation, "Life Data Analysis Reference", Tucson USA, May 2015

[31] M. Catelani, L. Ciani, M. Venzi, "Sensitivity analysis with MC simulation for the failure rate evaluation and reliability assessment", Measurement, Volume 74, October 2015, Pages 150-158, ISSN 0263-2241,

[32] M. Catelani, L. Ciani, M. Venzi, "Credible Improvement Potential measure for Reliability Importance assessment", 14th IMEKO TC10 Workshop Technical Diagnostics New Perspectives in Measurements, Tools and Techniques for system's reliability, maintainability and safety, Milan, Italy, June 27-28, 2016

[33] M. Catelani, L. Ciani, M. Venzi "Reliability assessment for complex systems: A new approach based on RBD models" Proc of (2015) 1st IEEE International Symposium on Systems Engineering, ISSE 2015 - Proceedings, art. no. 7302771, pp. 286-290.

[34] M. Catelani, L. Ciani, M. Venzi, L. Cristaldi, M. Faifer, M. Khalil, “A condition monitoring tool based on a FMECA and FMMEA combined approach in Oil&Gas applications”, 2016 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) 23 May - 26 May 2016 Taipei International Convention Center, Taipei, Taiwan

[35] Biswal, G.R.; Maheshwari, R.P.; Dewal, M.L., "System Reliability and Fault Tree Analysis of SeSHRS-Based Augmentation of Hydrogen: Dedicated for Combined Cycle Power Plants," IEEE Systems Journal, vol.6, no.4, pp.647-656, Dec. 2012

[36] Tsilipanos, K.; Neokosmidis, I.; Varoutas, D., "A System of Systems Framework for the Reliability Assessment of Telecommunications Networks," IEEE Systems Journal, vol.7, no.1, pp.114-124, March 2013

[37] D. Kececioglu - Reliability Engineering Handbook, vol.1, vol.2 - DEStech Publications, 2004

[38] Thorsen, O. and Dalva, M. “A survey of the reliability with an analysis of faults on variable frequency drives in industry”, In Proc. European Conference on Power Electronics and Applications EPE ’95 , pages 1033–1038, 1995

[39] Chang Y. C., Chang K.H., Liaw C.S. - Innovative reliability allocation using the maximal entropy ordered weighted averaging method

[40] ReliaSoft Corporation - Reliability Allocation using Lambda Predict - Tucson, AZ Reliability HotWire, Issue 98, April 2009

[41] G. Yang - Life cycle reliability engineering - Ford motor company - John wiley&soons, inc 2007

[42] R. R. Yager - On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decision making (1988)

[43] R. Fuller, P. Majlnder – An analytic approach for obtaining maximal entropy OWA operator weights (2001)

[44] M. Catelani, L. Ciani, M. Venzi, G. Patrizi “Reliability Allocation assessment using MEOWA method in complex redundant systems”, IEEE International Symposium on Systems Engineering 2016, October 3-5, 2016 | Edinburgh, Scotland

[45] Y. Lei, J. Lin, Z. He, Ming J. Zuo, “A review on empirical mode decomposition in fault diagnosis of rotating machinery”, Mechanical Systems and Signal Processing, Volume 35, Issues 1–2, February 2013, Pages 108-126, ISSN 0888-3270

[46] L. Hou; Bergmann, N.W., "Novel Industrial Wireless Sensor Networks for Machine Condition Monitoring and Fault Diagnosis," , IEEE Transactions on Instrumentation and Measurement, vol.61, no.10, pp.2787,2798, Oct. 2012

[47] C. Wang; Gao, R.X., "A virtual instrumentation system for integrated bearing condition monitoring," IEEE Transactions on Instrumentation and Measurement, , vol.49, no.2, pp.325,332, Apr 2000

[48] D. Galar, Adithya Thaduri, Marcantonio Catelani, Lorenzo Ciani, “Context awareness for maintenance decision making: A diagnosis and prognosis approach”, Measurement,

[49] C.H. Lauro, L.C. Brandão, D. Baldo, R.A. Reis, J.P. Davim, “Monitoring and processing signal applied in machining processes – A review”, Measurement, Volume 58, December 2014, Pages 73-86, ISSN 0263-2241,

[50] M. Lazzaroni, L. Cristaldi, L. Peretto, P. Rinaldi, M. Catelani, “Reliability Engineering:

Basic Concepts and Applications in ICT", 2011 Springer-Verlag, Berlin Heidelberg.

[51] NSWC, Naval Surface Warfare Center, Handbook of reliability prediction procedures for mechanical equipment, Carderock Division. Logistics Engineering Technology Branch, NSWC-10, January 2010.

[52] SINTEF Industrial Management, OREDA Offshore Reliability Data Handbook 2002, 4th edition, 2002.

[53] M. Catelani, L. Ciani, M. Venzi, "Component Reliability Importance assessment using Credible Improvement Potential" Proc of (2015) 1st IEEE International Symposium on Systems Engineering, ISSE 2016

[54] Applied R&M Manual for Defence Systems GR-77, Part C – R&M Related Techniques, issue 2011.

[55] IEC-61165, 2007. Application of Markov techniques, 2nd Ed.

[56] R. Kumar, A. Jackson, 2009 "Accurate reliability modeling using Markov Analysis with non-constant hazard rates". Proceeding of 2009 IEEE Aerospace conference. pp.1-7.

[57] G. Haifeng, 2010. Maintenance Optimization for Substations with Aging Equipment (2010).Electrical Engineering Theses and Dissertations. Paper 7.

[58] M. Boyd, An Introduction to Markov Modeling: Concepts and Uses, Annual Reliability and Maintainability Symposium; Anaheim, CA; United States

[59] W. Blischke, Murthy DNP (2000). Reliability: modeling, prediction, and optimization, Wiley.

[60] S. Demir (2009). Reliability of Combined kn -out-of-n and Consecutive kc-out-of-n Systems of Markov Dependent Components. IEEE Trans. Reliab., 58(4): 691-693.

[61] K. Gaeid, Ping HW (2011).Wavelet fault diagnosis and tolerant of induction motor: A review. Int. J. Phys. Sci., (IJPS) 6(3): 19.

[62] A. Habib, Yuge T, Al-Seedy RO, Ammar SI (2010). Reliability of a consecutive (r, s)-out-of-(m, n):F lattice system with conditions on the number of failed components in the system. Appl. Math. Model., 34(3): 531-538.

[63] A. Karimi A, Zarafshan F, Jantan AB, Ramli ARB, Saripan MIB (2010). An Optimal Parallel Average Voting For Fault-Tolerant Control Systems. 2010 International Conference on Networking and Information Technology (ICNIT2010), Manila, Philippines, IEEE.

[64] A- Karimi, Zarafshan F, Jantan AB, Al-Haddad SAR (2009). A Novel N-Input Voting Algorithm for Real-Time Fault-Tolerant Control Systems. J. Circuits, Syst. Comput., Under Review.

[65] A. Karimi, Zarafshan F, Jantan AB, Ramli ARB, Saripan MIB (2010). Accurate and Efficient Reliability Markov Model Analysisi of Predictive Hybrid M-out-of-N Systems. 2010 3rd IEEE Int. Conf. Comput. Sci. Inform. Technol., (ICCSIT 2010), Chengdu, China, IEEE Press.

[66] J. Fraden; Handbook of modern sensors, Springer, Third edition, 2003.

[67] Magnani, Ferretti, Rocco; Tecnologie dei sistemi di controllo,McGraw-Hill, Seconda edizione, 2007

[68] Isermann, R. "Fault-diagnosis systems – An introduction from fault detection to fault tolerance", Springer, Heidelberg, 2006

[69] Kiencke, U. Diagnosis of automotive systems. In Proc. IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS), Hull, UK, August 1997. Pergamon Press.

[70] K. Gaynes, Smith,Darrow; "Electrical contact failure mechanism relevant to electronic packages", IBM Corporation, IEEE 1991

[71] Filbert, D. "Technical diagnosis for the quality control of electrical low power motors" Technisches Messen , 70(9):417–427, 2003

[72] M. Catelani, L. Ciani, M. Venzi, "TTH Library: metodo innovativo per la valutazione di parametri diagnostici in applicazioni Oil&Gas"; Atti del XXXII Congresso nazionale GMEE, Area Metrologica, Misure e metodi per la qualità e la gestione dei processi, Lecco, 2015

[73] Isermann, R. Mechatronic systems – fundamentals . Springer, London, 2nd printing edition, 2005

[74] W. Globe, I. Van Beurden, J. Grebe; "Failure modes, effects and diagnostic analysis", Minnesota, USA

[75] Grimmelius, H., Meiler, P., Maas, H., Bonnier, B., Grevink, J., and Kuilenburg, R. van. Three state-of-the-art methods for condition monitoring. IEEE Trans. on Industrial Electronics , 46(2):401–416, 1999

[76] "An end user functional comparison of Hart and Foundation Fieldbus Protocol", Emerson, 2007

[77] C. Bonivento, L. Gentili, A. Paoli; "Sistemi di automazione industriale-Architetture e controllo", McGraw-Hill, 2011

[78] An Guochen, Meng Zhiyong, Ma Hongtao, Sui Bingdong; "Design of Intelligent Transmitter based on HART Protocol", Institute of information science and engineering, Henbei university of science and technology, Shi Jiazhuang, China, 2010

[79] H. K. Goh, R. Devanathan; "Fieldbus for control and diagnostic", Seventh International Conference on control, Automation, Singapore, Dicembre 2002

[80] C. Soares; "Gas Turbines. A Handbook of Air, Land and Sea Applications", Cap.7, Butterworth-Heinemann, 2008

[81] M. Catelani, L. Ciani, V. Luongo, "Functional safety assessment: an issue for technical diagnostics", Proc. XX IMEKO World Congress – Metrology for Green Growth, 2012, Busan, Rep. of Korea

[82] S. Nunns; "Principles of proof testing of safety instrumented systems in the chemical industry", ABB Ltd, 2002

[83] H.D. Wacker, P. Holub, J. Borcsok ; "Optimization of diagnostics with respect to the diagnostic coverage and the cost function", XXIV ICAT, 2013

[84] G. Rogoll; "Advanced online physical layer diagnostics", Pepperl+Fuchs, White paper, 2009

[85] I. Hwang, "A Survey of Fault Detection, Isolation, and Reconfiguration Methods", IEEE Transactions on Control Systems Technology, June 2010

[86] D. Miljković, "Fault detection methods: A literature survey", Proceedings of the 34th International Convention, Opatija, Croatia, 23-27 May, 2011, At Opatija, Croatia

[87] ISO 14224:2016, "Petroleum, petrochemical and natural gas industries — Collection and exchange of reliability and maintenance data for equipment"

[88] "Safety Instrumented Systems (SIS) -- Safety Integrity Level (SIL) Evaluation Techniques", ISA-TR84.0.02, ISA, Research Triangle Park, NC, 1999.

[89] M. A. Lundteigen, “The effect of partial stroke testing on the reliability of safety valves”, Conference: ESREL 2007, At Stavanger

[90] IEC 61508-1:2010, Functional safety of electrical/electronic/programmable electronic safety-related systems

[91] IEC 61511-1:2016, Functional safety — Safety instrumented systems for the process industry sector — Part 1: Framework, definitions, system, hardware and software requirements

[92] A. P. Naumenko A. P. “Methodology of Vibroacoustic Diagnostics of Reciprocating Machines” The Bulletin of the Moscow State Technical University of a Name of N.E. Bauman, Special Release, A Series Mechanical engineering 2007 pg. 85–95