Automatic Synthesis of Fault Trees from Process Modelling with Application in Ship Machinery Systems

Gabriele Manno[a*], Alexandros S. Zymaris[b], and Nikolaos M.P. Kakalis[b]
[a] DNV GL, Strategic Research & Innovation, Høvik, Norway
[b] DNV GL, Strategic Research & Innovation, Piraeus, Greece

Abstract: System safety analysis provides assurance that the system satisfies safety-constrains even in the presence of components failures. Traditionally, safety analyses are performed based on various formal and informal requirements and design documents. These analyses can often be subjective and are dependent on the skills/expert knowledge of the practitioner. Moreover, the construction of Fault Trees (FTs) is generally a time consuming and tedious activity especially when complex systems are assessed. In this paper we propose a methodology in which FTs are generated automatically from the resolution of formal system engineering models of the system under investigation. The process models are able to capture both the steady-state and dynamic behavior of components in their nominal and failure states. For the development of the process models the DNV COSSMOS library/platform was used, that is the first formal process modelling platform developed for marine energy systems. Since functional dependencies are captured by the input-output relations implemented in the system model, the methodology allows for a separation of concerns when modelling different components. Moreover, a library of components can be generated for reuse in other applications. In the view of the authors, not only does the proposed methodology allow for automatic synthesis of FTs and state-space exploration, but it also bridges the gap existing between safety and process engineering analyses.

Keywords: Fault Trees, Flow Sheet models, Hybrid Automata, Dynamic Reliability, Markov Chains.

1. INTRODUCTION

In safety-critical systems incorrect operation could lead to loss of life, substantial material or environmental damage and large monetary losses, thus these systems are designed with stringent safety requirements. For instance, in the maritime industry, new standards are continuously developed, by intergovernmental institutions and classification societies in order to guarantee safe operations at sea. Besides the introduction of new regulations involving e.g., structural issues, fire barriers, etc., efforts have converged in the definition of a formal methodology to conduct safety analyses, the Formal Safety Assessment (FSA) [1].

In the current practice, safety analyses are based on a range of well-established methodologies and techniques which include Fault Trees (FT) [2-4], Failure Modes and Effect Analysis (FMEA) [2, 5, 6] and HAZard IDentification (HAZID) and HAZard & OPerability (HAZOP) analysis [7, 8]. A common denominator of these methodologies is that they are manual processes based on expert analysis. For relatively simple systems, this is a manageable process, but with increasing system complexity, such analysis could become expensive and complex. Moreover, issues like model update arise not only during the design process but also during the life of a system whenever changes, such as new components and/or technologies are introduced. Furthermore, expert judgment cannot always answer questions related to complex system operation under failures, especially when quantification of the failure consequence is needed for decision-making.

Recent work has investigated the automation of system safety analyses using failure models in order to make the process more formal, automated, and consistent. Hierarchically Performed Hazard Origin and Propagation Studies (HiP-HOP) [9, 10] is one technique that belongs to this category and that was developed during the SAFEDOR project with the aim of supporting risk-based ship design for safe operations [11]. Several applications of HiP-HOP in other industries, like the automotive, have also

been reported [12]. HiP-HOP largely automates the development of FTs and FMEAs from the topology of the engineering system enhanced with local component failure modes. The "link" between the system model and the FT is based on qualitative information introduced to the failure models by the user. No physics are modelled within this framework and thus the failure propagation is not deduced by any formal mathematical model of the system.
Another approach for automating the construction of FTs, based on the Architecture Analysis and Design Language (AADL), was proposed in [13]. In [13] a mapping algorithm was described capable of deriving a static FT directly from an AADL model of the system. According to the authors of [13] methods based on formal behavioral models offer little support to representing the system architecture in comparison to architecture description languages. Other proposed methodologies, based on similar approaches, have been presented in literature [15]. Earlier approaches can be found in [16, 17] while methodologies based on formal methods like Model checking can be found in [18]. In [14], Altarica, a high level language for event driven modeling based on Guarded Transition Systems, was presented.

The method proposed here differentiates from the ones described above by employing quantitative evaluations through formal process modelling of the system studied. This analysis is done considering the possible system failure configurations. In order to realize this, a Hybrid Transition System (HTS) approach was used, where each state of the transition system describes a system/component failure mode (and/or degradation state) and is given a set of differential and algebraic equations describing the system/component physical behavior [19, 20]. The approach uses communicating Transition Systems so that the modeling of physical units can be done separately. A Reachability graph approach is used to build the system state-space.
This kind of representation of the system is linked to a Piecewise Deterministic Markov Process (PDMP) [21]. In fact, when a system sojourns in a state it executes a specific dynamic process from which the evolution over time of certain process variables can be tracked. Finally, this Markov chain can also be seen as a Reward process [22], where the reward variables accumulated in each state are a function of the variables output of the dynamic process executed in the state and the bounds (i.e., safety margins) imposed on these variables. It is from the choice of opportune reward variables, and their bounds, that a Fault tree can be automatically generated by a set of (deterministic) simulation campaigns and a state-aggregation algorithm.

The remainder of this paper is organized as follows: Section 2 gives a brief introduction of the methodology; Section 3 summarizes the framework implementing the method; Section 4 describes the tool implementation; in Section 5 a contrived case study based on a marine boiler system with a hot stand-by redundancy is analyzed; Section 6 reports related work in the field of Dynamic Reliability; and, finally, in Section 7 conclusions and future work are summarized.

2. METHODOLOGY

An engineering system can be considered as an event-driven dynamic system where the variables of the process are random and their behavior regulated by physical mechanisms, operating conditions, failure modes, etc. This system can be seen as two mutually dependent processes that take place during the system operation: the Deterministic and the Stochastic processes (Figure 1).

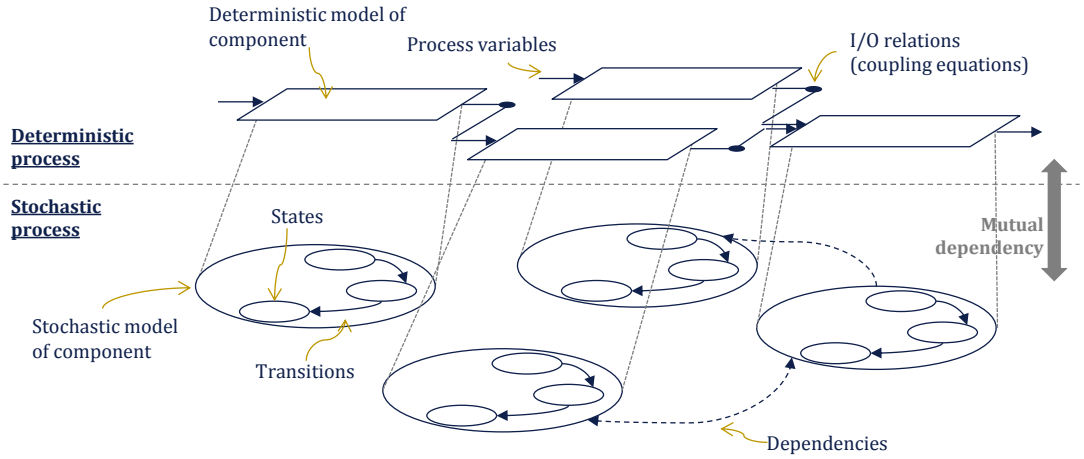
In the deterministic process, opportune continuous variables whose evolution in time is defined by a set of algebraic and partial differential equations (i.e., the process variables), are used to describe the physical behavior of the system (e.g., mass equations, thermodynamics). This process can be simulated by employing modern Model-based Systems Engineering (MBSE) approaches [23-25], especially when complex systems comprising of many components and subsystems are under consideration. A component-oriented approach is followed: for each component of the system an individual mathematical model is created (component model) and the overall system model is constructed through interconnecting the component models with input/output connectivity equations (connections). The connectivity between blocks is usually directly related to the connectivity of the actual system configuration found in Pipe and Instrumentation Diagrams (P&IDs). The component

models can be programmed in a reconfigurable way so as to form a generic component library for future use.

The other process is the stochastic one, which governs the failure mechanism of components. In this process, events like the failure of a component or the activation of a spare component take place at specific points in time, and their effect is to trigger a change in the state of the stochastic process. In turn, a change of state might lead to a change in the deterministic behavior of a component. In other words, a change of state might lead to a change in the parameters of the dynamic deterministic system, or, more generally, to a change of the model structure equations, variables and parameters.

The component-oriented architecture of the process system, described above, allows for assigning specific failure modes and the related physical behavior at the component level (in the form of a Hybrid Transition System). In the approach followed in this work, a FT graph is created from the HTSs of various components by analyzing the consequences of their failures at the system level. This graph can then be solved for obtaining probabilistic measures of the system reliability. Therefore, within this framework risk-based (stochastic) approaches and the analysis of dynamic deterministic systems are combined together.

Fig. 1. Modelling a Dynamic Reliability problem as two mutually interacting processes.



2.1. Generic mathematical component model

A stochastic process can be effectively represented in terms of Transition systems (TSs), a formalism that allows parallel and hierarchical compositional modelling [19]. The class of Transition systems of interest in this work is the Hybrid-TS (HTS). A HTS is a TS where for each state is defined a set of differential and algebraic equations describing the component physical behavior.
For a component with $k = 1, \dots, N$ failure/degradation modes it is possible to define its dynamic (deterministic) behavior as a Partial Differential Algebraic Equation (PDAE) system as below:

$$\frac{d\vec{Y}}{dt} = \vec{F}_k\left(\vec{Y}(t), \frac{d\vec{Y}}{dx_i}(t), \vec{u}(t), \vec{b}_k(t), t\right),$$

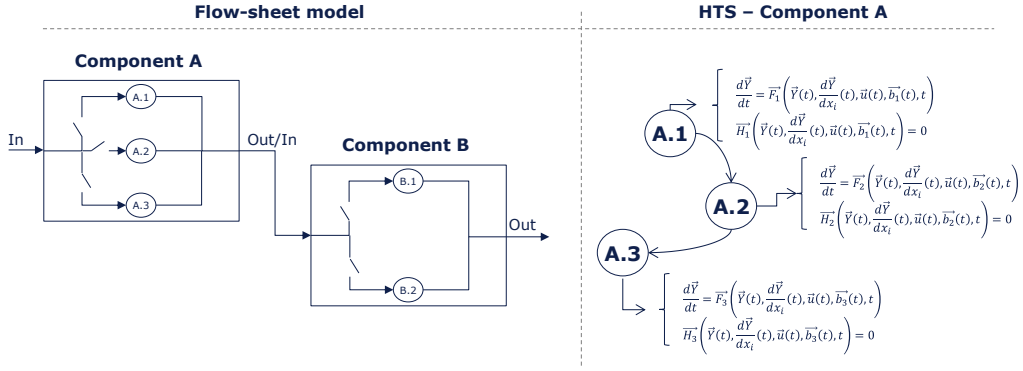
and (1)

$$\vec{H}_k\left(\vec{Y}(t), \frac{d\vec{Y}}{dx_i}(t), \vec{u}(t), \vec{b}_k(t), t\right) = 0,$$

where t denotes the time, $\vec{Y} = \left(Y_1, \dots, Y_{N_Y}\right)$, $\vec{u} = \left(u_1, \dots, u_{N_u}\right)$ and $\vec{b} = \left(b_1, \dots, b_{N_b}\right)$ are the vectors of differential variables, algebraic variables and parameters, respectively. The vectors $\vec{Y}$ and $\vec{u}$ comprise the *process variables* of the system for the specific component. $\vec{F}$ and $\vec{H}$ are vector functions. The

partial derivative base vector, $\vec{x}$, is an appropriate distribution domain, usually expressing geometry dimensions (e.g. length, width, radius, etc.). The PDAE system is completed by the necessary initial and boundary conditions.
The mathematical model of each component is coded into a "block of code" that can be used in the flow-sheet model of the system (see Figure 2 as an example). Each component model has a set of input and output variables as of Equation 1. The collection of these equations for all the components of the system defines the set of equations describing the physical (deterministic) dynamic behavior of the whole system in different failure configurations.

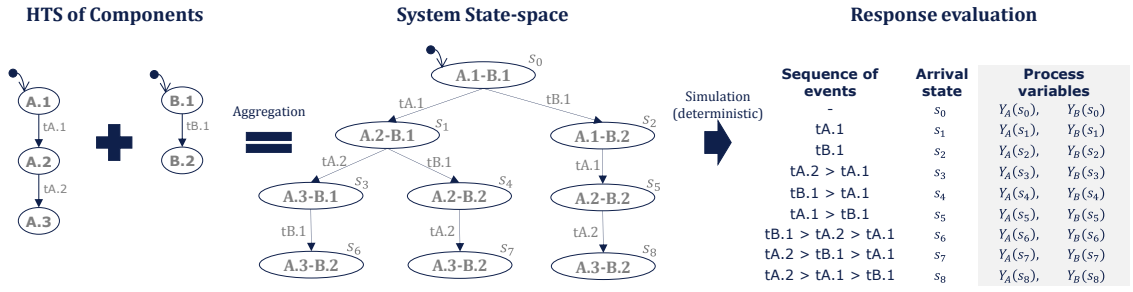
Fig. 2. The model of a component as a HTS embedded into a flow-sheet component model.



2.2. System state-space exploration

Given the HTSs of components it is possible to build the system state-space as in Stochastic Petri Nets (SPNs) through the method of the Reachability Graph [26]. However, differently from SPN, an extended concept of state is used that includes also the ordering of events (a similar approach can be found in Stochastic Activity Networks [27]). The advantage of this approach is the possibility of discerning between different sequences of failures. Moreover, each state of the generated system state-space will inherit the aggregated deterministic behavior of components, defining the set of equations that describes the dynamic (deterministic) behavior of the whole system in a state.
An example of state-space model of a system consisting of two components, A and B, is shown in Figure 3. In the figure, it is possible to see how the extended concept of state influences the construction of the state-space, e.g., the state <A.2, B.2> is considered twice depending on which transition, between tA.1 and tB.1, has fired first.
Each identified state carries the physical dynamic behavior of the whole system in the state. Therefore, it is possible to simulate the system in each state and retrieve the evolution over time of the deterministic process after a failure sequence occurs. Outputs of this procedure are twofold: (*i*) support hazard identification by exploring the deviations of process variables from their nominal values; and (*ii*) can be used as inputs to the FT synthesis stage.

Fig. 3. The system state-space construction and exploration approach.



Sequence of events	Arrival state	Process variables
-	s_0	$Y_A(s_0)$, $Y_B(s_0)$
tA.1	s_1	$Y_A(s_1)$, $Y_B(s_1)$
tB.1	s_2	$Y_A(s_2)$, $Y_B(s_2)$
tA.2 > tA.1	s_3	$Y_A(s_3)$, $Y_B(s_3)$
tB.1 > tA.1	s_4	$Y_A(s_4)$, $Y_B(s_4)$
tA.1 > tB.1	s_5	$Y_A(s_5)$, $Y_B(s_5)$
tB.1 > tA.2 > tA.1	s_6	$Y_A(s_6)$, $Y_B(s_6)$
tA.2 > tB.1 > tA.1	s_7	$Y_A(s_7)$, $Y_B(s_7)$
tA.2 > tA.1 > tB.1	s_8	$Y_A(s_8)$, $Y_B(s_8)$

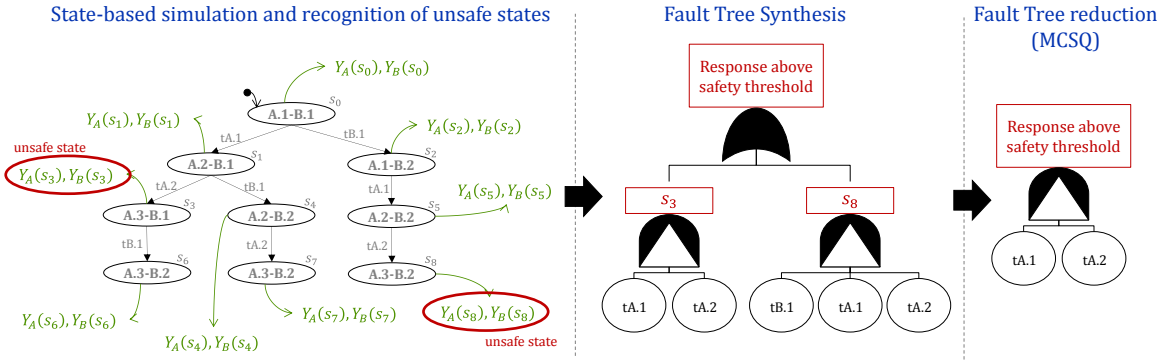
2.3. Unsafe region identification and Fault Tree construction

A hazard identification analysis is performed to define which of the performance variables are safety-critical (e.g., a variable whose deviation may lead to an explosion) and the acceptable safety boundaries (i.e., thresholds). A set of safety critical variables, with given acceptance ranges, may be used to define the Top Event of a Fault Tree. The process variables whose values are beyond safety limits indicate if a state (in the system state-space) is to be regarded as safety-critical or not. These states can then be used to generate the FT of the system with Top Event defined as the deviation of safety-critical variables from their safe region.

The class of Fault Trees considered in this work can be defined as a subclass of the Temporal Fault Tree (TFT) [28]. In fact, due to the way the FT is built, only AND, OR and PAND gates will be present in the generated FT. This type of FT describes more accurately a system, rather than static-FT, because it also captures temporal relations between events. (e.g., the case of the Priority-AND gate where the Boolean logic of the AND gate is extended with temporal relations). Another example of FT with temporal relations is the Dynamic FT [28, 30].
The FT is simply generated as an OR relation between all the conditions expressed in each of the identified critical states (Figure 4). Reduction laws apply also for a TFT when PAND relations between pairs of events are treated as "atomic" units. As introduced in [18], these atomic units are called "doublets" and are represented by the notation "(A < B)". To decompose temporal relations in doublets, the law of extension can be used. After application of the absorption law the tree is reduced in its redundant parts. Finally, the law of completion enables the condensation of PAND relations into AND relations, when possible. The output is then a TFT in the reduced form of Minimal Cut Sequences (MCSQs).

If a quantitative reliability assessment is needed, distributions of the time to failures of components must be defined. However, the amount of quantitative input data can be reduced. In fact, this task can be done after the construction of the FT, thus, considering only those failure modes of components leading to the occurrence of the top event FT.

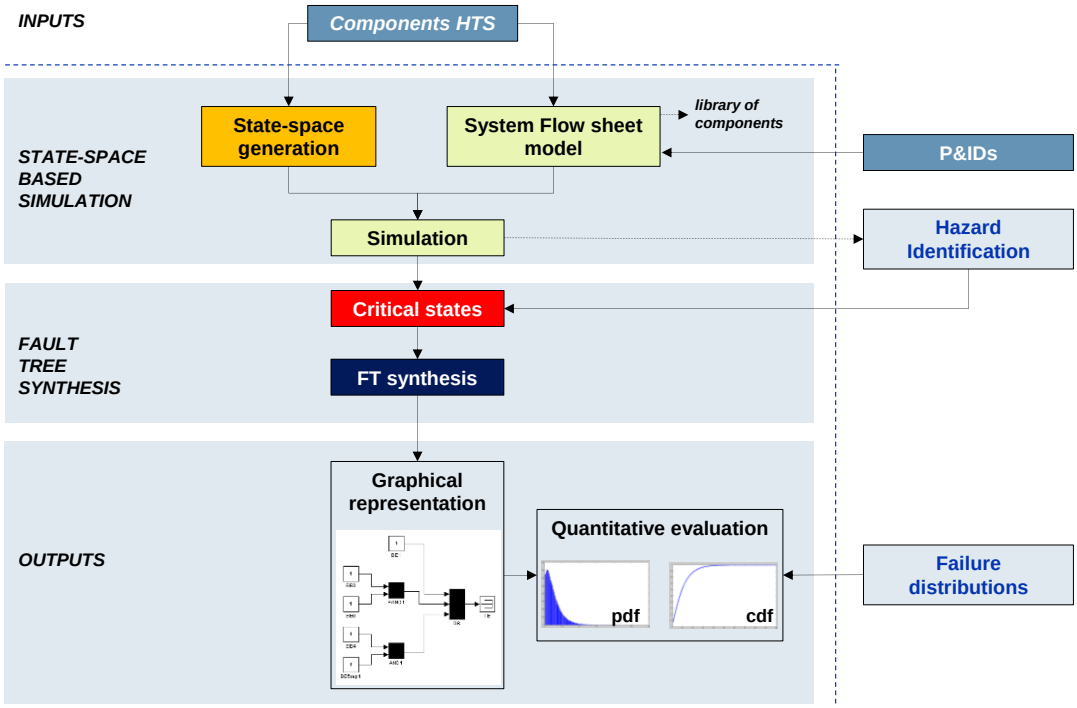
Fig. 4. Synthesis of the Temporal Fault Tree from process model simulation.



3. FRAMEWORK

The framework for a model-based safety analysis is shown in Figure 5. The inputs to the methodology are: (*i*) the HTSs of components; (*ii*) the information retrieved from P&IDs that is used to connect the models of components in the unified system model; (*iii*) the declaration of safety critical variables and their safety boundaries (outcome of hazard identification); and, (*iv*) the failure distributions of the basic events of the FT in case of probabilistic quantitative assessment.
These inputs are then used to build the system state-space and for the simulation of the system (i.e., the flow sheet model). Successively, given the outputs of the campaign of deterministic simulations, the FT is automatically generated. Finally, the results are graphically represented by a TFT in the form of Minimal Cut Sequences and can be further used to retrieve reliability and risk measures of interest.

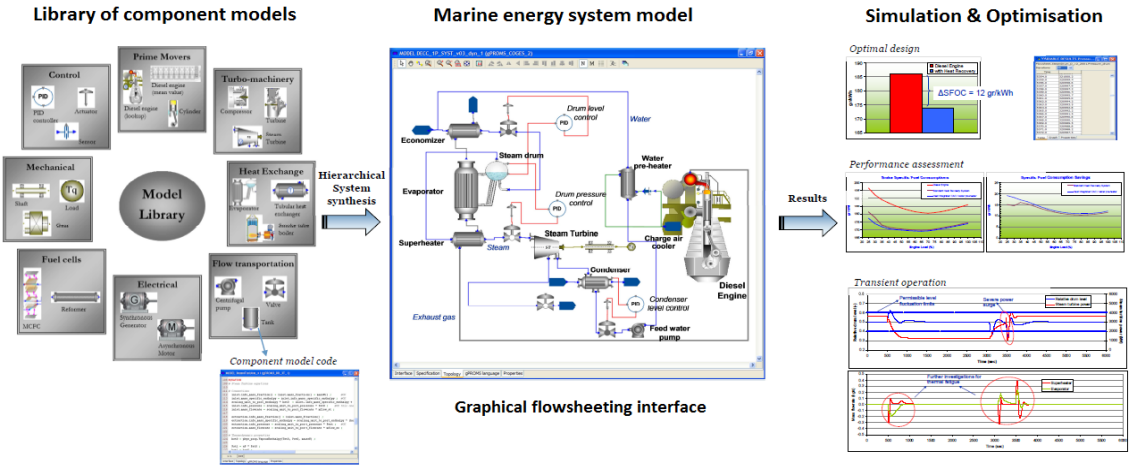
Fig. 5. Framework for Model-based Safety Analysis.



4. TOOL IMPLEMENTATION

The development of the methodology presented herein is based on the DNV COSSMOS library of components [31-36]. COSSMOS stands for COmplex Ship Systems MOdelling and Simulation and it is an ongoing activity within DNVGL Strategic Research and Innovation that has resulted at the development of a modular platform for thermofluid/electrical dynamic process modeling and simulation of complex ship machinery systems capable of assessing multiple configuration and technology alternatives, i.e., machinery components time varying conditions and ship mission envelopes (Figure 6).

Fig. 6. The DNV COSSMOS modelling framework.



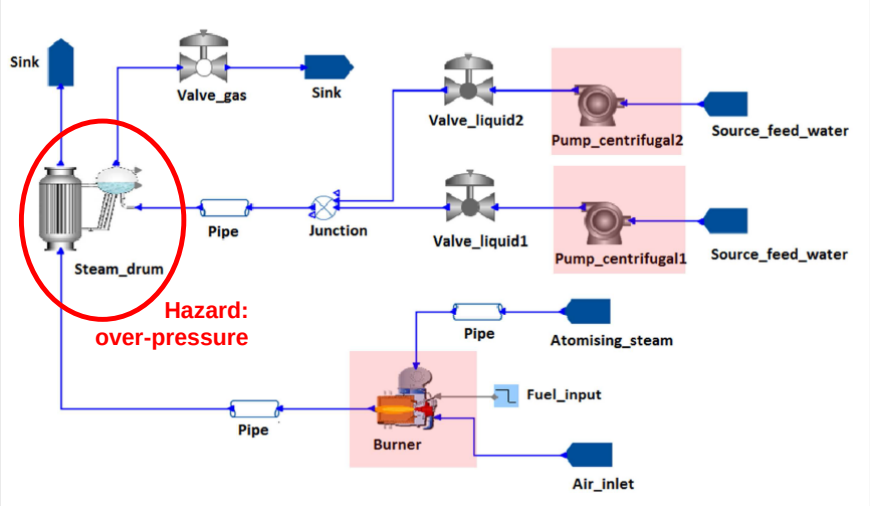
The implementation of COSSMOS is done in gPROMS [37], an equation-oriented modelling language specially designed for process system modelling and simulation that can efficiently handle the numerical solution of highly complex non-linear PDAE systems in a variety of processes. As discussed earlier, although the COSSMOS platform is developed primarily for applications in ship

machinery systems, it can be utilized in other industrial sectors, e.g. oil & gas, offshore. Matlab® is used for the synthesis of the FT.
The DNV COSSMOS Model Library (MDL) contains the basic building blocks for constructing marine energy system models. For each component of the system a mathematical model was developed, as described in [31-33]. The models have been specifically designed in order to: a) address dynamic and steady-state simulation, b) be suitable for optimization studies, and c) be suitable for both design and operation studies. Each model has been designed as an object-oriented, modular and re-configurable library component. The MDL contains an extendable list of models covering primary energy conversion, heat exchange processes, steam systems, flow transport and control, as well as control and automation; this list can be found in [33].

5. CASE STUDY

A marine boiler system is analyzed with the proposed methodology. The DNV COSSMOS model of the system is presented in Figure 7 and consists of the following library components: (i) 1 boiler burner; (ii) 1 steam-drum; (iii) 2 feed-water centrifugal pumps (one operating while the other is in hot standby); (iv) 2 feed-water valves (one for each pump); and (v) 1 valve to control the steam outflow from the drum. The system comprises of approximately 500 equations, 16 of which are PDAEs. For the needs of this example only the steady state operation of the system is examined, reducing the system to nearly 400 algebraic equations. The system components that are subjected to failures are the two pumps and the burner, highlighted within red boxes in Figure 7.

Fig. 7. Flow-sheet model of the marine boiler system (DNV COSSMOS snapshot).



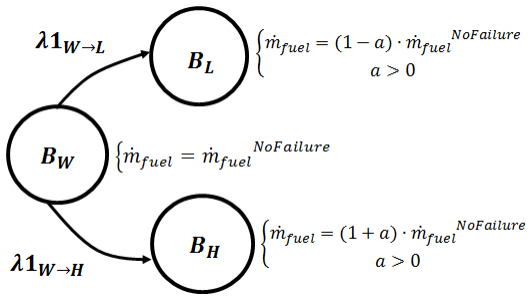
The pressure at the steam-drum is the process variable of interest in this case-study. Failing to maintain its value within safety margins can lead to metal rupture or even a blast, and consequently it was chosen to be the critical variable to be monitored.
The failure modes of the components subjected to failures are listed below (see Table 1):

- The two pumps can fail either by having high or low output. The pump failures are realized by imposing an increase (or decrease, respectively; Table 1) in the water flow-rate through the pump.
- Pump 2 is in stand-by configuration with respect to Pump 1 and it is subjected to the same failures as Pump 1. Pump 2 switches to the operating mode only if Pump 1 has failed and Pump 2 is not in the failed state characterized by a lower output than the nominal value.
- The burner can fail either by having a high or low output. This is modeled by an increase (or decrease, respectively) in the fuel supply to the burner (Figure 8, left). Figure 8 (right) shows a part of the DNV COSSMOS programmable process schedule that is used to implement the HTS.

Table 1. Failure modes of components.

Component	Failure mode ID	Failure mode description	Preceding state	Physical description
Pumps (x2)	$P1_W$, $P2_W$	Working	–	$\dot{m}_{feedwater}$
	$P1_L$, $P2_L$	Low output	$P1_W$, $P2_W$	$(1-a)\dot{m}_{feedwater}$
	$P1_H$, $P2_H$	High output	$P1_W$, $P2_W$ or stand-by	$(1+a)\dot{m}_{feedwater}$
Burner	B_w	Working	–	$\dot{m}_{fuel}$
	B_L	Low fuel	B_w	$(1-a)\dot{m}_{fuel}$
	B_H	High fuel	B_w	$(1+a)\dot{m}_{fuel}$

Fig. 8. HTS of a fuel burner (Left) and DNV COSSMOS process schedule (Right).




```
#  Burner failures: (1) low fuel flowrate; (2) high fuel flowrate
#  iterate through all the system operation
  FOR current_time = 0 to system_operation_time
#    check if burner has a fuel low failure at the current_time
    IF burner has a fuel low failure AT current_time
#       if yes, then alter the model parameter "burner_fuel_flowrate"
#       to its failed low value
        REASSIGN
            burner_fuel_flowrate = burner_fuel_flowrate_failed_low
        END REASSIGN
#     check if burner has a fuel high failure at the current_time
    ELSE IF  burner has a fuel high failure AT current_time
#       if yes, then alter the model parameter "burner_fuel_flowrate"
#       to its failed high value
        REASSIGN
            burner_fuel_flowrate = burner_fuel_flowrate_failed_high
        END REASSIGN
    END IF
  END FOR
```

In total 79 failure sequences are found during the state space generation. A time-ordered sequence of steady state problems is used to simulate each failure sequence. At each step of this procedure a failure event is introduced to the system according to the ordering each failure has within each failure sequence. The thermodynamics and mass/heat transportation equations as well as other physical equations comprising the process model are all solved concurrently within this procedure taking into account the effect of the possible failures. The simulation proceeds until the end of each failure sequence is reached and the "final" values of process variables are the ones used for the FT construction. The case was run on a PC with an Intel i7-2.80 GHz processor (utilizing one core) and 8Gb of RAM within approx. 2.5 CPU minutes. In Figure 9 (left) the pressure at the steam-drum of the boiler for the 79 failure sequences that comprise the state-space is presented. Each bar corresponds to one failure sequence. The red colored bars stand for no (or very small) variation from the nominal value of the process variable, the blue colored bars stand for an increase in process variable value and green colored bars stand for a decrease in process variable value.

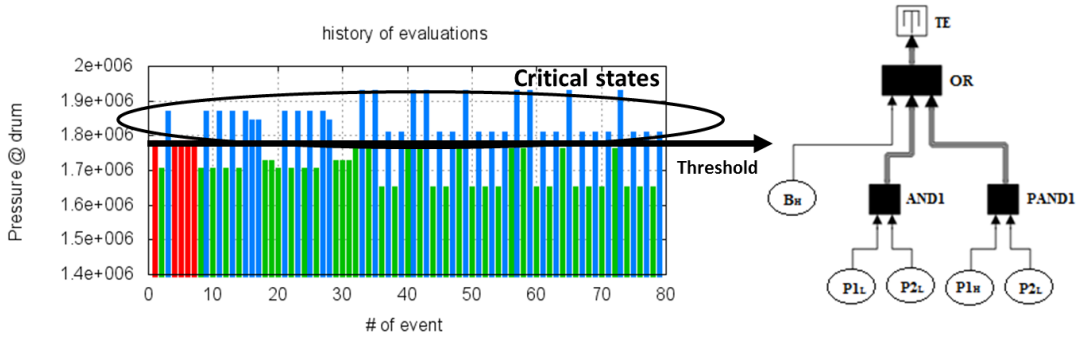
The data presented in Figure 9(a) are used for the generation of the FT that is presented in Figure 9(b). The top-event in this FT is any increase in the pressure at the steam-drum. Consequently all the failures that lead to such a system excitation can be found under the top event. In this case it is:

(i) the Burner's B_H failure;
(ii) the failure of Pump 1 (either low, $P1_L$, or high, $P1_H$); and
(iii) the failure $P2_H$ of the redundant pump.

In Figure 9(b), the branch to the left of the FT (B_H) shows that any increase in the fuel consumption leads to higher steam production and for a given feed-water flowrate this leads to pressurization of the system. With the proposed methodology, is capable of accounting for the physics that govern the system, such effect and consequence analysis is derived automatically. The analyst is enabled to identify consequence effects even in the case that the system is governed by complex underlying mechanisms. Furthermore, the branches in the middle and to the right of the FT are related to the

redundancy of the system's pumps and the fact that any decrease in the sub-cooled feed-water flowrate will lead to further evaporation and thus pressure increase. In more detail, the gate in the middle, $1 = AND(P1_L, P2_L)$, depicts the following failure sequence: the primary pump and the redundant pumps have failed low. This leads to the TE occurrence irrespective of the time orderings of the two failures. Finally, the gate $PAND1 = PAND(P1_H, P2_L)$ corresponds to the case where *Pump_1* has failed high and subsequently the pump in redundancy has failed low. The Priority-AND gate shows that only this sequence ordering of the two failures leads to the TE. It is interesting to note that due to the redundancy strategy used, when these two failure events occur with the opposite time ordering then no change (i.e., activation) to the already failed Pump_2 is observed and hence the TE is prevented. As described in this case-study, the proposed methodology is capable to account for operational strategies (e.g. here redundancy issues) and process physics in a holistic and concurrent way enabling an automated analysis of the system. By utilizing the DNV COSSMOS platform and its programmable interface, the analyst is capable to develop and simulate any functional relationship and operation strategy that he is interested in.

Fig. 9. State-space-based deterministic simulation and FT generation.

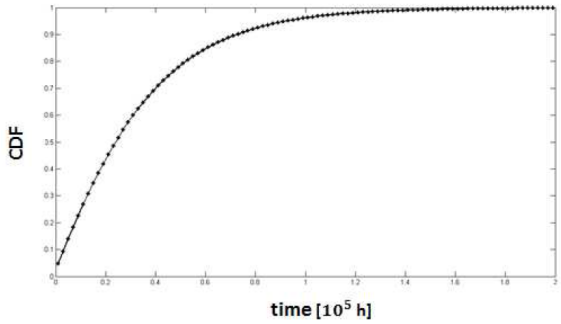


(a) **(b)**

Finally, a probabilistic assessment can be performed by assigning failure rates to each of the basic events in the retrieved FT. In the table in Figure 10 the failure rates for each failure mode of components are presented; values were retrieved from the OREDA handbook [38]. The overall failure rate values reported for each component in OREDA is divided by two in order to account for the two types of failures of each component (low and high). The figure also shows the cumulative density function for the top event of the FT.

Fig. 10. Quantitative probabilistic analysis.

Component name	*Failure Modes*	*Failure rates*
Boiler	B_L	$\lambda_{B_L} = 0.2365 \times 10^{-04}$
	B_H	$\lambda_{B_H} = 0.2365 \times 10^{-04}$
Pump_1	$P1_L$	$\lambda_{P1_L} = 0.1080 \times 10^{-04}$
	$P1_H$	$\lambda_{P1_H} = 0.1080 \times 10^{-04}$
Pump_2	$P2_L$	$\lambda_{P2_L} = 0.1080 \times 10^{-04}$
	$P2_H$	$\lambda_{P2_H} = 0.1080 \times 10^{-04}$



5. RELATED WORK IN DYNAMIC RELIABILITY

It was mentioned in the introduction that the work presented in this paper is linked to the Dynamic Probabilistic Risk Assessment approach (DPRA), [39], developed in the field of nuclear engineering during the last decades. However, DPRA does not involve the (automatic) generation of FTs, while being focused more on the evaluation of reliability measures of interest for the system at hand. In general, this class of problems, i.e. problems where stochastic transitions (e.g., component failures) are coupled with deterministic physical models (that also may lead to transitions), belong to the general

class of Hybrid Stochastic (HS) systems [20, 40, 41]. In the relevant literature, DPRA is also referred to simply as "Probabilistic dynamics", [42, 43] or "Dynamic reliability", [44, 45, 46]. Although a formal Markovian mathematical framework is established (that also incorporates the physical behavior) [19, 21], the direct solution of these equations is a computationally demanding task even for simple systems and thus biased and analogue Monte-Carlo simulation techniques [47, 48] are usually employed. Some more recent works in this field include possibilistic clustering classification of the failure event scenarios [49, 50].
In [51, 52] Fluid Stochastic Petri Net (FSPN) is introduced in order to handle continuous variables governed by PDAE s. This is an alternative methodology to simulate and assess a hybrid system. A more extended presentation of FSPNs along with a solution method based on Discrete Event Simulation can be found in [53]. In these approaches the problem is treated in a holistic way where the physics are solved along with the probabilistic aspects of the system (e.g. component failures) during the stochastic simulations. Another approach where a Discrete Event Simulation algorithm for HTS is proposed can be found in [54].
Similar approaches to the one proposed here can be found in the literature related to industrial applications in the Oil & Gas [55] and Aeronautics fields [56]. These approaches are based on the decomposition between the stochastic component failures and the deterministic process variable trajectories. However, these approaches do not consider the automatic generation of Fault Trees and have limitations in the number and types of failures considered.

6. CONCLUSIONS

In this paper a framework for the automatic synthesis of FTs that uses formal process system models including system dynamic behavior was presented. Compared to past work, a more thorough and realistic description of the true behavior of the system is achieved through the use of the governing physics. The involvement of the physical models improved the state-of-the art since limited analyst engagement in the definition of the propagation of failures through the system is required. Moreover, the possibility of optimization studies with respect to system parameters (e.g., sizing and type of components, redundancies, control logics) for safety-cost optimization can be performed (since these parameters are incorporated in the component hybrid models). The benefits of this method can be summarized as follows:

- Explicit modeling of effects of components failures on other components is not needed because functional relations (and, thus, failure propagation) are captured by the dynamic model of components and propagated through the system using process simulation.
- Qualitative analysis of the FT is based on an exhaustive quantitative evaluation of the true dynamic behavior of the system subjected to failures;
- Possibility of highlighting critical events that are not seen by the analyst during the preliminary hazard identification process;
- Intuitive visualization of the failure-region via of a high level model such as a FT;
- Decoupling of the analysis of the system dynamics and the system stochastic behavior, with gains in terms of computational time and modelling efforts; and
- The quantitative probabilistic evaluation can be run considering only those events that have been identified as leading to the system safety critical state.

Finally, although the proposed paper is focused on the machinery system of a ship, the proposed framework is general enough to be employed in other kinds of systems such as process plants, and other electro-mechanical engineering ones. Moreover, the model could be used as a basis for real time reliability evaluations as a part of a (model-based) diagnostics and prognostics framework.

Future work will address: enhanced heuristic state space exploration; the issue of online reliability estimation; the integration of the framework with other reliability methodologies in order to take into account the effect of other subsystems, external conditions, operating profile, etc.; and consequence modeling in shipping; with the aim of pursing a holistic approach to reliability and risk modeling for the safe operation of ship systems.

References

[1] Guidelines for Formal Safety Assessment (FSA) for use in the IMO rulemaking process. IMO, 2002: MSC/Circ.1023-MEPC/Circ.392, IMO2002.

[2] Villemeur, A., Reliability, Availability, Maintainability and Safety Assessment: Methods and Techniques Vol. 1. 1992: John Wiley & Sons.

[3] Rausand, M. and A. Høyland, System Reliability Theory: Models, Statistical Methods and Applications. 2nd ed: John Wiley & Sons.

[4] Haasl, D.F., et al., Fault Tree Handbook, 1981, Nuclear Regulatory Commission, Washington, DC (USA). Office of Nuclear Regulatory Research.

[5] Recht, J.L., Failure Mode and Effect. National Safety Council, 1966.

[6] Lees, F.P., Loss Prevention in the Process Industries 2nd edition 1996, Elsevier.

[7] Dunjóa, J., et al., Hazard and operability (HAZOP) analysis. A literature review. Journal of Hazardous Materials, 2010. 173(1-3): p. 19-32.

[8] Lawley, H.G., Operability study and hazards analysis. Chem. Eng. Progress, 1974. 70: p. 45-56.

[9] Papadopoulos, Y., et al., Engineering failure analysis and design optimisation with HiP-HOPS. Engineering Failure Analysis, 2011. 18: p. 590–608.

[10] Papadopoulos, Y. and U. Petersen, Combining ship machinery system design and first principle safety analysis, in 8th Int. Marine Design Conference (IMDC03)2003: Athens, Greece.

[11] Soares C.G., et al., Risk-Based Ship Design: Methods, Tools and Applications. Papanikolaou A.D., Ed. Berlin, Germany: Springer-Verlag, 2009.

[12] Papadopoulos, Y. and C. Grante, Evolving car designs using model-based automated safety analysis and optimisation techniques. The Journal of Systems and Software, 2005. 76(1): p. 77-89.

[13] Joshi, A. et al., Automatic Generation of Static Fault Trees from AADL Models. In Proceedings of the IEEE/IFIP Conference on Dependable Systems and Networks' Workshop on Dependable Systems, DSN07-WADS, Edinburgh, Scotland-UK, June 2007.

[14] Prosvirnova, T. et al., The Altarica 3.0 project for Model-Based Safety Assessment. Safety, reliability and Risk Analysis: Beyond the Horizon 2013. Proceedings of ESREL conference, Amsterdam, The Netherlands, 2013.

[15] Pai, G.J. and Dugan, J.B., Automatic Synthesis of Dynamic Fault Trees from UML System Models. In Proceedings of the 13th International Symposium on Software Reliability Engineering, Annapolis, MD, pp. 243-256, 2002.

[16] Poucet, A., STARTS: Knowledge Based Tools for safety and reliability Analysis. Reliability Engineering and System Safety 30 (1990) 379-97.

[17] Xie, G., Xue, D., and Xi, S., TREE-EXPERT: A tree-based expert system for Fault Tree construction. Reliability Engineering and System Safety 40 (1993) 295-309.

[18] Bozzano, M., and Villafiorita, A., Improvong System reliability via Model Checking: The FSAP/NuSMV-SA Safety Analysis Platform. Lecture notes in Computer Science 2788 (2003) 49-62.

[19] Baier, C. and Katoen, J.P., Principles of Model Checking. MIT Press, 2008.

[20] Alur, R., et al., Hybrid Automata: An algorithmic approach to the specification and verification of hybrid systems. Lecture Notes in Computer Science 736 (1993) 209-29.

[21] Davis, M., Markov models and optimization. London: Chapman and Hall, 1993.

[22] Sahner, R., Trivedi, K.S., and Pulifiato, A., Performance and reliability analysis of computer systems. Kluwer Academic Publishers, Boston, MA, 1996.

[23] C.C. Pantelides, Z.E. Urban. Process Modelling Technology: A critical review of recent developments. 6th International Conference on Foundations of Computer-Aided Process Design (FOCAPD), Princeton, New Jersey, USA, 2004.

[24] K.-U. Klatt, W. Marquardt. Perspectives for process systems engineering—Personal views from academia and industry. Computers & Chemical Engineering. 33 (2009) 536-50.

[25] G. Stephanopoulos, G.V. Reklaitis. Process systems engineering: From Solvay to modern bio- and nanotechnology.: A history of development, successes and prospects for the future. Chemical engineering science. 66 (2011) 4272-306.

[26] Kartson, D., Balbo, G., Donatelli, S., Franceschinis, G., and Conte, G., Modelling with Generealized Stochastic Petri Nets. John Willey & Sons, NY, USA, 1994.

[27] Sanders W H. Construction and Solution of performability model based on stochastic activity networks. PhD Thesis, University of Michigan, 1988.

[28] Walker, M.D., and Papadopoulos, Y.I., Pandora: The Time of Priority AND gates. In INCOM 2006, France, pp. 237–242, 2006.

[29] Chiacchio, F., Compagno, L., D'Urso, D., Manno, G., and Trapani, N., Dynamic Fault Trees resolution: A conscious trade-off between analytical and simulative approaches. Reliability Engineering and System Safety 96 (11), 2011, 1515-26.
[30] Manno, G., Chiacchio, F., Compagno, L., D'Urso, D., and Trapani, N., Conception of Reparaible Dynamic Fault Trees and resolution by the use of RAATSS, a Matlab toolbox based on the ATS formalism. Reliability Engineering and System Safety 121 (2014) 250-262.
[31] Kakalis, N.M.P., G.G. Dimopoulos, and E. Ovrum, COSSMOS, model development and implementation I : Complex Ship Systems Modelling and Simulation, 2009, Det Norske Veritas.
[32] Kakalis, N.M.P., et al., COSSMOS, model development and implementation II. DNV, 2010.
[33] Kakalis, N.M.P., et al., COSSMOS, model development and implementation III. DNV, 2011.
[34] Dimopoulos, G.G. and N.M.P. Kakalis. An integrated modelling framework for the design, operation and control of marine energy systems in Proc. of 26th CIMAC, 2010, Bergen, Norway.
[35] Dimopoulos, G.G., C.A. Georgopoulou, and N.M.P. Kakalis. Modelling and optimisation of an integrated marine combined cycle system. in Proc. of 24th Int. Conf. on Energy, Cost, Optimization, Simulation and Environmental Impact of Energy Systems (ECOS). 2011. Novi-Sad, Serbia.
[36] Kakalis, N.M.P. and G. Dimopoulos. Managing the complexity of marine energy systems. Position Paper 11/2012, Det Norske Veritas, Research & Innovation, (available at: www.dnv.com).
[37] gPROMS. Available from: www.psenterprise.com/gproms.
[38] OREDA: offshore reliability data, 4th edition. SINTEF Industrial management, 2002.
[39] Siu, N., Risk assessment for dynamic systems: An overview. Reliability Eng. and System Safety, 1994. 43: p. 43-73.
[40] Henzinger, T.A., The theory of hybrid automata. In Proc. of the 11th Annual IEEE Symposium on Logic in Computer Science, LICS '96, pp. 278-292, New Brunswick, New jersey, 1996.
[41] Stochastic Hybrid Systems ed. C.G. Cassandras and J. Lygeros2007: Taylor & Francis.
[42] Labeau, P.E., Probabilistic Dynamics: Estimation of generalized unreliability through efficient Monte-Carlo simulation. Ann. Nucl. Energy, 1996. 23(No. 17): p. 1355-1369.
[43] Smidts, C., Probabilistic Dynamics: a comparison between continuous event trees and discrete event tree model. Reliability Eng. and System Safety, 1994. 44: p. 189-206.
[44] Devooght, J., Dynamic reliability. Adv. in Nuclear Science and Technology, 1997. 25.
[45] Marseguerra, M., et al., A concept paper on dynamic reliability via Monte Carlo simulation. Mathematics and Computers in Simulation, 1998. 47: p. 371-382.
[46] Kopustinskas V., Augutis J. and Rimkevičius S., Dynamic reliability and risk assessment of the accident localization system of Ingalina NPP RBMK-1500 reactor. Reliability Eng. and System Safety, 2005. 87: p.77-87.
[47] Labeau, P.E., A Monte-Carlo estimation of the marginal distributions in a problem of probabilistic dynamics. Reliability Eng. and System Safety, 1996. 52: p. 65-75.
[48] Lewis, E.E. and F. Böhm, Monte carlo simulation of Markov unreliability models. Nuclear Eng. And Design, 1984. 77(49-62).
[49] Podofillini, L., et al., Dynamic safety assessment: Scenario identification via a possibilistic clustering approach. Reliability Eng. And System Safety, 2010. 95: 534–549.
[50] Mercurio, D., et al., Identification and classification of dynamic event tree scenarios via possibilistic clustering: Application to a steam generator tube rupture event. Accident Analysis and Prevention, 2009. 41: 1180–1191.
[51] Horton, G., et al., Fluid Stochastic Petri Nets: Theory, Applications, and Solution, January 1996, NASA: Langley Research Center Hampton, Virginia 23681-0001.
[52] Trivedi, K.S. and V.G. Kulkarni. FSPNs: Fluid Stochastic Petri Nets. in 14th Int. Conference on Application and Theory of Petri Nets June 1993. Chicago, Illinois, USA: Springer-Verlag.
[53] Ciardo, G., D.M. Nicol, and K.S. Trivedi, Discrete-Event Simulation of Fluid Stochastic Petri Nets. Software Engineering, IEEE Transactions on, 1999. 25(2): p. 207-217.
[54] Manno, G., Zymaris, A., Kakalis, N.M.P., Chiacchio, F., Cipollone, F.E., Compagno, L., D'Urso, D., and Trapani, N., Dynamic reliability analysis of three nonlinear aging components with different failure modes characteristics. Safety, reliability and Risk Analysis: Beyond the Horizon 2013. Proceedings of ESREL conference, Amsterdam, The Netherlands, 2013.
[55] Yang, X. and M.S. Mannan, The development and application of dynamic operational risk assessment in oil/gas and chemical process industry. Reliability Eng. and System Safety, 2010. 95: p. 806-815.
[56] Domínguez-García, A.D., et al., An integrated methodology for the dynamic performance and reliability evaluation of fault-tolerant systems. Reliability Eng. and System Safety, 2008. 93: p.1628–1649.