2017-10

Prognostic systems representation in a function-based bayesian model during engineering design

LHer, Guillaume; Van Bossuyt, Douglas L.; OHalloran, Bryan M.

International Journal of Prognostics and Health Management

LHer, Guillaume, Douglas L. Van Bossuyt, and Bryan M. OHalloran. "Prognostic systems representation in a function-based bayesian model during engineering design." International Journal of Prognostics and Health Management 8.2 (2017): 23.
http://hdl.handle.net/10945/65183

Prognostic systems representation in a function-based Bayesian model during engineering design

Guillaume L'Her[1,*], Douglas L. Van Bossuyt[2], Bryan M. O'Halloran[3]

[1] *Colorado School of Mines, Golden, CO, 80401, USA*
glher@mines.edu
[2] *KTM Research, LLC, Tualatin, OR, 97062, USA*
douglas@ktmresearch.com
[3] *Naval Postgraduate School, Monterey, CA, 93943, USA*
bmohallo@nps.edu

ABSTRACT

Prognostics and Health Management (PHM) systems are usually only considered and set up in the late stage of design or even during the system's lifetime, after the major design decision have been made. However, considering the PHM system's impact on the system failure probabilities can benefit the system design early on and subsequently reduce costs. The identification of failure paths in the early phases of engineering design can guide the designer toward a safer, more reliable and cost-efficient design. Several functional failure modeling methods have been developed recently. One of their advantages is to allow for risk assessment in the early stages of the design. Risk and reliability functional failure analysis methods currently developed do not explicitly model the PHM equipment used to identify and prevent potential system failures. This paper proposes a framework to optimize prognostic systems selection and positioning during the early stages of a complex system design. A Bayesian network, incorporating the PHM systems, is used to analyze the functional model and failure propagation. The algorithm developed within the proposed framework returns the optimized placement of PHM hardware in the complex system, allowing the designer to evaluate the need for system improvement. A design tool was developed to automatically apply the proposed method. A generic pressurized water nuclear reactor primary coolant loop system is used to present a case study illustrating the proposed framework. The results obtained for this particular case study demonstrate the promise of the method introduced in this paper. The case study notably exhibits how the proposed framework can be used to support engineering design teams in making better informed decisions early in the design phase.

1. INTRODUCTION

An increasing number of systems use Prognostics and Health Management (PHM) hardware to detect future failures and allow for preventive maintenance and recovery actions, automated or manual. However, the hardware is often added after the system has been built or during the late stages of the design. In the early stages of the design, PHM is currently not seriously considered, despite the consequent impact it can have on the design choices made for the system. The goal of PHM is to allow systems operators to catch incipient failures early enough to be able to prevent or correct them. The consideration of PHM hardware in the early phase of engineering design can optimize the system design toward this goal. PHM systems can effectively be used to reduce the likelihood of failure of a component. Hence, a system can be designed with PHM hardware instead of expensive redundancies while maintaining a similar system reliability. The earlier in the design phase a potential system fault is discovered, the less costly the design required modifications can be (Chang, 2002). Being able to consider prognosis in the early phases by modeling the impact of PHM hardware allows the designer to limit the costly system changes while increasing the system reliability.

Existing risk and reliability analysis methods are either too rigid and require an advanced design, or cannot model a PHM system. For example, the widely used Probabilistic Risk Assessment (PRA) method is able to model failure detection and recovery actions, but is limited by its rigidity, time-consuming changes, and by its use in the late phases of design. Functional failure methods can be used in the early phases of design by considering only the functionality of a system, with no specific component requirements. However, these methods presents inherent difficulties to model PHM systems.

*Corresponding Author

The Prognostics and Health Analysis to Support Engineering Design (PHASED) method is proposed in this paper. The PHASED method creates a framework that enables the use of a functional failure method in various stages of design, notably early on, coupled with PHM hardware considerations. A Bayesian network representing the interaction between the components in the system is used to compute the functional failure propagation probabilities. An optimized configuration for PHM equipment positioning in the system is automatically given to the system designer, who can then decide to move forward with it or modify the system, according to the system failure probability returned by the algorithm. A design tool was developed to automatically apply the proposed method.

Section 2 presents the context of this paper and introduces various methods used throughout this paper. It additionally summarizes the state of the art for accounting for the presence of PHM systems during different design phases. In section 3, the proposed methodology is presented. A case study representing a simplified pressurized water reactor plant is defined within the scope of this paper in section 4. It is used to demonstrate the proposed methodology. The method results and future work are discussed in section 5. Finally, the conclusion is given in section 6.

2. BACKGROUND

2.1. Prognostics and Health Management

Prognostics and Health Management (PHM) as a field was introduced by NASA in 1990 (Elattar, Elminir, & Riad, 2016). PHM analyzes past failure data to devise ways to assess the system health based on current monitoring data. It can consequently allow for informed condition-based maintenance and extend the system lifetime or prevent failure, thus limiting cost of maintenance and allowing for a safer, more predictable system (Sun, Zeng, Kang, & Pecht, 2012), (Agarwal, Lybeck, Pham, Rusaw, & Bickford, 2015). More and more complex systems already make extensive use of PHM systems, across various industries such as automotive, aeronautics or nuclear (Coble, Ramuhalli, Bond, Hines, & Upadhyaya, 2015), but widespread industry application is still lagging behind (López, Márquez, Fernández, & Bolaños, 2014). Those systems are mainly used to reduce maintenance costs by moving toward a more condition-based maintenance schedule. PHM is often added to a system as an afterthought, in order to solve reliability and risks issues when they start arising. PHM system modeling can, however, also be used in the design phase to make important decisions, drive the probability of failure of components down, and avoid unnecessary design costs.

Most of the developments in the PHM field aim at improving the diagnosis and prognosis capability in various systems. This is seen through the development of sensors and measurement techniques (Lin, Zakwan, & Jennions, 2017; Xiao, 2016), more adequate data analysis methods (Sankavaram et al., 2016) and the introduction of decision algorithms for smart manufacturing processes (Choo, Adams, Weiss, Marvel, & Beling, 2016). The potential impact on the design of the application of PHM techniques during the early design phase is rarely considered.

2.2. Bayesian network

A Bayesian network is a directed acyclic graphical probabilistic model that represents a set of variables and their conditional dependencies (Pearl, 1985). It is composed of a set of nodes X, a directed and acyclic graph to link them, and a conditional distribution for each node given its parents, $P(X_i|\ Parents(X_i))$. Within the scope of the work presented in this paper, the conditional distribution are represented by conditional probability tables, giving the distribution over the states of X_i for each combinations of parent values. A Bayesian network being acyclic by definition, several models representing the various components' interaction and feedback loops in the system have to be considered.

Bayesian networks have been the subject of a growing popularity to model systems and conduct reliability analysis (Doguc & Ramirez-Marquez, 2009; Torres-Toledano & Sucar, 1998), and have shown significant advantages when compared to widespread methods such as Reliability Block Diagram (RBD) or Fault Tree Analysis (FTA) (Langseth & Portinale, 2007). Weber, Medina-Oliva, and Simon (2012) gives a useful overview of the use of Bayesian networks in the risk and reliability field.

2.3. Functional model

In order to model a system in the early conception phase, functional models were developed, and with them various functional failure analysis approaches were devised. These functional models are gaining traction within the industry due to their ability to discover faults and propagation paths early on in the design process, cutting costs to make the product evolve toward a safe and reliable prototype.

A functional model is a graphical representation of a system functionalities (Eisenbart, Blessing, & Gericke, 2012). It comprises a set of functions performed within the system and the flows connecting them together. The Functional Basis for Engineering Design (FBED), developed by Stone and Wood (2000), defines a specific taxonomy allowing for widespread and unified use of this type of model. In this taxonomy, for example, a tank of water would be characterized by "Provision - Store - Contain". One of the main advantages of this system definition is its applicability throughout all design stages, notably in the very early conception stages, when the specific components and requirements are yet to be determined, and when erroneous costly decisions can be taken

by engineering teams. The present paper uses the taxonomy developed within the FBED method and generates functional models based on the Functional Flows Block Diagram (FFBD) method. FFBD was developed in the late 50s for US defense applications. It introduces logical gates in a block diagram.

2.4. PHM in risk and reliability analyses

Prognostic and Health Management systems are not commonly considered in the early stages of design due to the lack of adequate analysis methods. With relation to PHM systems modeling in a system, two categories of risk and reliability analyses appear: *PHM-potential* methods and *non-PHM* methods.

2.4.1. PHM-potential methods

The PHM-potential methods can be used to account for PHM hardware within a system. However, those methods are either limited by the need for an advanced existing design or by the lack of flexibility when implementing PHM hardware modeling.

Probabilistic Risk Assessment (PRA) methods (Smith et al., 2005) identify and analyze the consequences of initiating events in a system, by playing out the accident sequence and computing the probability of the system being safe. The use of this method is, for example, required in the nuclear industry to justify the plant safety in a variety of initiating events (U.S. NRC, 2016). PRA can account for fault detection and corrective action success in each specific accident sequence. It cannot be effortlessly modified to compare the outcome of different selection and position of PHM hardware within the system and can be cumbersome to modify.

The Functional Failure Identification and Propagation (FFIP) propagates failures through a functional model using Flow State Logic (FSL) (Jensen, Tumer, & Kurtoglu, 2009; Kurtoglu & Tumer, 2008). Representing the impact of PHM hardware in the system requires the modification of the functional model and the FSL associated, an expensive (both in time and resources) undertaking.

In order to circumvent the limitations from FFIP, a method to integrate PHM system in a functional model and optimize the selection of the hardware was developed by Stack and Van Bossuyt (2015), the Prognostic System Variable Configuration Comparison (PSVCC). It introduced an algorithm allowing a designer to define potential PHM hardware to set up in the system and essentially performed a modified FFIP analysis on the new system created. This method did not consider a number of parameters such as the management and maintenance team decisions, or the use of generic databases. It based its failure propagation on the FFIP method, rendering the method challenging to scale up. PSVCC was considered an inspiration for the proposed method in the present paper, even though the two methods have little in common.

Continuous Time Bayesian Network (CTBN) can be used to account for loops in a Bayesian network by considering the time component (Gopalratnam, Kautz, & Weld, 2005; Nodelman, Shelton, & Koller, 2002). A reliability analysis based on CTBN was developed by Boudali and Dugan (2006). More recently, a prognostics method based on CTBN was introduced to account for PHM sensors in a system (Perreault, Thornton, Strasser, & Sheppard, 2015). This method was applied to a system in order to predict faults and act on them to prevent system failure. While adequate — though computionally very intensive even for small complex systems — for use during the system operational lifetime, it is not applicable in the early stages of a design, when the discrete time component for the functions states in the system is not known. It is also not made to select an optimized sensor configuration through a system. Finding a way to reconcile the Prognostics CTBN with the method proposed in this paper represent an interesting future direction for this field.

A methodology for probabilistic prognosis of a system using a dynamic Bayesian Network was recently proposed (Bartram & Mahadevan, 2015). A Hybrid Bayesian Network (HBN) framework was introduced (Neil & Marquez, 2012) to account for repair time and derive system availability. This method is again applicable on finished and operating designs only, consequently limiting its use in the design stages.

2.4.2. Non-PHM methods

Fault Tree Analysis (FTA) is often included in PRA. It can also be used independently, which allow its use in earlier stages of the design, although still pretty advanced (Ericson, 1999). Indeed, the components must be known in order to create the fault tree. Besides the need for an advanced design, this analysis method cannot be used to model corrective actions after a fault detection.

Failure Modes and Effect Analysis (FMEA) and its variant Failure Modes, Effects and Criticality Analysis (FMECA) (U.S. Department of Defense, 1949), are widely used risk and reliability analysis methods (Liu, Liu, & Liu, 2013). They are based on the computation of a risk probability number computed from several parameters, the probability of a failure, its detectability, its severity, and its criticality for a FMECA. PHM systems can be considered by the engineers while deriving the different parameters but no framework is provided naturally. A few frameworks which could account for health management sensors within FMECA were developed (Conroy, Stecki, & Thorn, 2016; Kacprzynski, Roemer, & Hess, 2002). However, these methods exhibit the weakness of a FMECA analysis, namely the needs for an advanced design and for a variety of experts, subject to bias.

The Functional Failure Design Method (FFDM) (Stone, Tumer, & Wie, 2005) and the Risk in Early Design (RED) (Lough, Stone, & Tumer, 2009) are among the main methods of functional failure analyses, based on functional models. These methods are based on historical functional failure data to identify the weak points of a system. The PHM systems cannot be modeled using these methods.

2.5. Human Reliability Analysis

Human Reliability Analysis (HRA) is used in the proposed framework to compute database information about the probability of success of corrective actions undertaken following PHM data analysis. It estimates the contribution of human failure to the system risk and reliability. Several noteworthy methods of HRA have been developed over the years, specifically for the nuclear industry, such as the THERP method (Swain & Guttmann, 1983), the SPAR-H method (Gertman, Blackman, Marble, Byers, & Smith, 2005) or ATHEANA (Cooper et al., 1996).

When considering maintenance and recovery actions, HRA is an important analysis to perform in order to account for human mistakes. Numerous HRA methods are criticized for not being plant-specific enough and relying on potentially outdated data (Spurgin & Lydell, 2002). SPAR-H, a more recent and now widespread method in the industry, is used in this paper.

In the SPAR-H method, the Human Error Probability (HEP) is defined from the combination of the Performance Shaping Factors (PSF) corresponding to different essential parts of the maintenance success such as stress factor or task complexity (Boring & Blackman, 2007). It follows Eq. (1). $NHEP$ is defined in HRA as being equal to 0.001 for action-based maintenance. The PSF includes the available time to perform a task, the associated stress, the team's experience and training, the complexity of the task, the ergonomics of the systems, the quality of available procedures, the team's fitness for duty and the work processes. The PSF values used in this paper are taken from the SPAR-H method.

$$HEP = \frac{NHEP * PSF}{NHEP * (PSF - 1) + 1} \tag{1}$$

2.6. Review

Functional models form a category of systems modeling that can allow for risk and reliability analysis in the early stages of a design. Bayesian networks provide a mathematical framework that can be used to represent such system models and to propagate failure probabilities through the models. PHM equipment's goal is to catch incipient failures early enough to attempt to correct them. HRA methods evaluate the likelihood of success of a correction.

3. METHODOLOGY

The PHASED methodology presented in this paper aims at incorporating PHM hardware in a system during the early phase of engineering design. The probabilities of failure of a critical point in the system are obtained for various PHM hardware configuration through the system and the optimized configuration is computed. The method can be divided into five main parts, as seen in Figure 1. These five parts include a logical functional model, databases, trees finding, PHM sensor selection and Bayesian network solver.

The PHASED method is based upon a functional representation of a system as developed by Stone and Wood (2000), augmented with Success Tree Analysis (STA). STA is the inverse of the FTA (Andrews & Dunnett, 2000). It describes the various steps needed to lead to a healthy system. Logical gates are introduced to define the steps connectivity. Five distinct databases are necessary to represent various system information. Recommended approaches to be used in order to populate these databases are presented. The databases shown in this paper fall into two categories, system-specific and generic. The generic databases comprise information about PHM hardware efficiency, emergent function weaknesses and impacts of function failure on subsequent linked flows quality. The system-specific databases tie the probability of corrective action success to each particular system function and flow, as well as code the management decision making towards maintenance tasks, the reliance on scheduled maintenance and PHM sensors indications. The different trees representing the system are obtained using a combined risk-critical and reliability-critical approach. For each computed tree, the selection and positioning of PHM hardware through the designed system is generated. The Bayesian network representing the given tree through the complex system is consequently built and a risk and reliability analysis is performed. The positioning of PHM equipment is then iterated to compute the best possible system configuration. The best configuration of the configurations obtained for all paths is then selected and can be used to support the decision making.

This framework has been automated to facilitate the designers' task (L'Her, 2016). It can be noted that the use of a functional representation of the system allows for the applicability of the proposed method in various stages of design, including the early phases.

3.1. Logical Functional model

The functional model is based on the Functional Basis for Engineering Design (FBED) method, and can be constructed from an existing Pipe and Instrumentation Diagram (P&ID) or from a conceptual design. A functional model can be effectively built in the very early stages of design.

In this paper, the concept of a logical functional model is

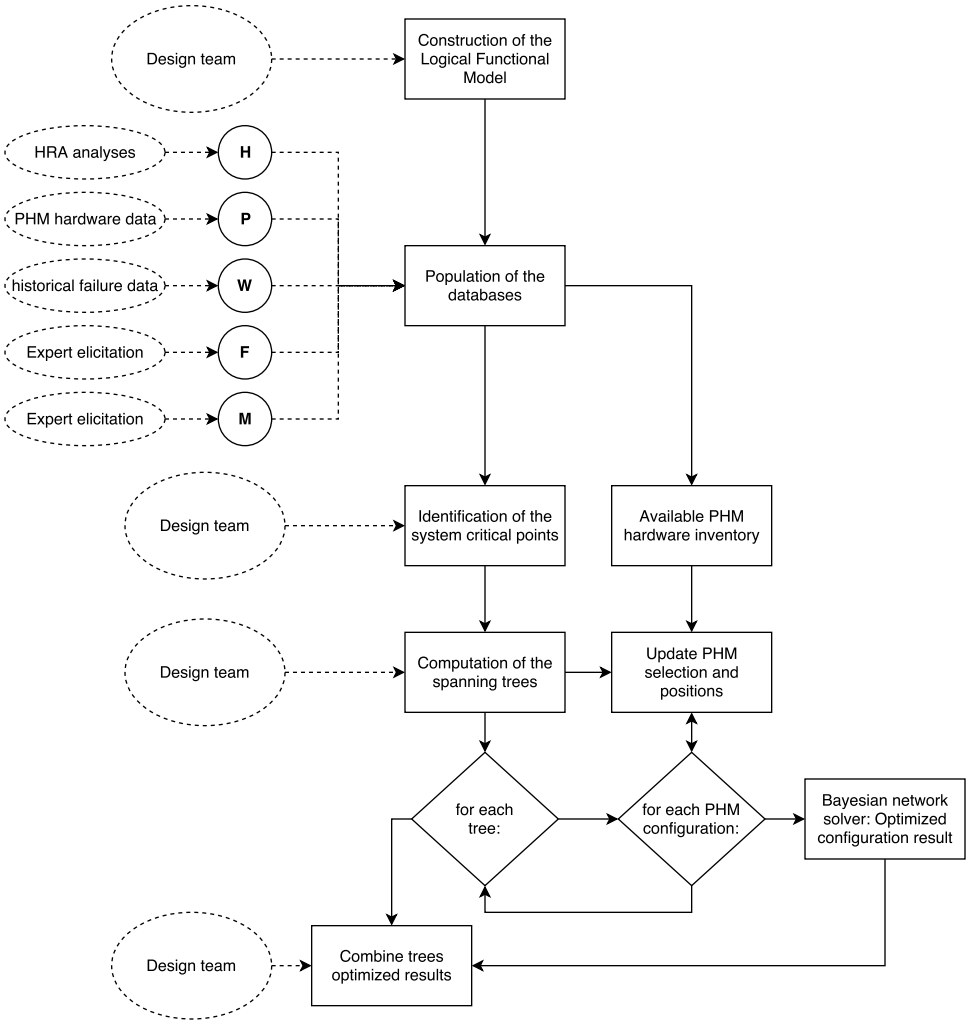

Figure 1. Methodology (dashed lines represent inputs)

introduced based upon FBED taxonomy and evolved FFBD method. FBED does not consider the use of logical gates linking several flows to functions. The method developed in this paper adds logical gates within the functional model representing the system. By default, if no gate is represented, an AND-gate is assumed, i.e. the receiving function needs all input flows to operate. A logical functional model allows for the input of more detailed information from the desired system. Figure 2 exhibits the difference between a logical functional model and its classical equivalent.

In the logical functional model shown in Figure 2, flows 14, 24 and 34 respectively connect the functions 1, 2 and 3 to the function 4 using a *Voting-Or* (k-of-N) logical gate. If the function 4 nominal operation depends on all three incoming flows to be in a nominal state, this gate becomes an AND-gate. If only one of the three flows is necessary to the nominal function 4 operation, this gate becomes an OR-gate.

The use of a logical functional model permits the encoding of information such as redundancies and fail-safe functions, prevalent in complex systems, to a functional model.

3.2. Databases

The objective of this paper is to propose a method to model and optimize positioning of PHM hardware throughout a complex system and obtain failure propagation paths using a Bayesian network. It does not aim at developing a set of values and rules to populate the aforementioned databases. Consequently, in this paper the databases are populated using values derived from expert opinion and industry resources for the purpose of illustrating the method; these specific values should not be used as-is for safety-critical analysis.

Five independent databases are to be used in this method. It is interesting to note that three of these databases are generic and not system-dependent, meaning that they can be reused across various systems. This ensues from the use of the unified taxonomy developed within FBED.

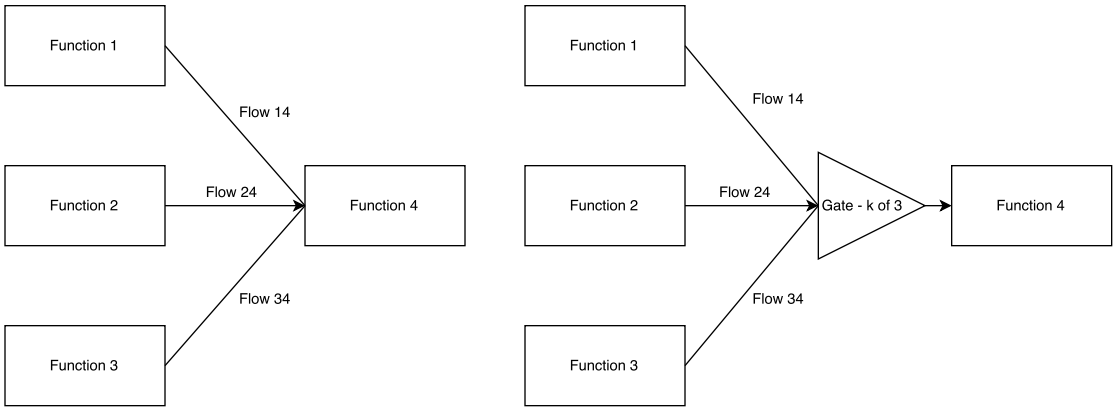

Figure 2. Classical Functional model (left) and Logical Functional model (right).

Name	Function or Flow	Efficiency			False alarm
		Failure	Concern		
			low	high	
1	Function A	$\varepsilon_{1,A}$	-	-	$e_{1,A}$
...					
n	Function A	$\varepsilon_{n,A}$	-	-	$e_{n,A}$
	Flow Z	$\varepsilon_{n,f,Z}$	$\varepsilon_{n,l,Z}$	$\varepsilon_{n,h,Z}$	$e_{n,Z}$

Table 1. PHM hardware database architecture - **P**

System Function or Flow [ID]	Correction success			Mishandling
	Failure	Concern		
		low	high	
Function A category [ID A]	ρ_A	-	-	β_A
...				
Flow Z category [ID Z]	$\rho_{f,Z}$	$\rho_{l,Z}$	$\rho_{h,Z}$	β_Z

Table 2. Correction success database architecture - **H**

The p-database, noted **P**, contains information relative to the type of PHM hardware available to the system. This includes the efficiency ε of the hardware to identify different flow and function weaknesses and the false alarm rate e. The efficiency represents the probability that a PHM hardware will correctly detect a flow or function weakness. The efficiency of a particular PHM hardware depends on the function or flow it is surveilling. The architecture of **P** can be found in Table 1. This database represents the specific hardware manufacturing specification data. However, it often happens that this data is not available to the designer. In such cases, values based on historical performance of functionally similar equipment can prove sufficient.

The m-database, noted **M**, contains information relative to the maintenance team management and decisions. This database accounts for potential team shortages or a managerial decision to ignore PHM data (γ). It also allows to account for scheduled maintenance not condition-based (μ).

This database permits a better refinement of the simulation. In the case **M** is not given, the algorithm considers the management to be in total support of the PHM hardware warnings. A maintenance team would thus be sent to repair a function or flow every time a weakness is detected by the PHM equipment. This database represents a challenge to populate efficiently. Indeed, the μ values should be considered to account for the maintenance of directly dependent function or flows, and the distinction between direct and indirect dependance of functions and flows can be subject to interpretation by the designers. In this paper, only the immediately connected flows and functions were considered impactful and as such, integrated in the μ values computation.

The h-database, noted **H**, contains information relative to the corrective actions. For each function and flow in the constructed functional model, the designer computes a likelihood of timely repair ρ in the case of a successful detection by the PHM hardware. The designer also generates a likelihood of mishandling β if the PHM signal originated from a false alarm. Maintenance or repair actions are dependent on the system itself. Hence, this database is considered system-specific and often cannot be reused.

To populate **H**, two approaches are possible: the automatic pre-planned actions and the human (maintenances, repairs, manual switches to redundant systems, etc.) actions. The HRA methodology can be applied to the studied system to account for the human side of corrective actions. The main question to answer when computing HRA probability is: Can the risk-critical and the reliability-critical functions defined for the system be protected? When a weakness is detected by a PHM sensor, different parameters (time to repair, maintenance team experience and training, work processes, procedures, etc.) are considered to compute a probability of successful action. In the case of automatic actions, their relevance and time efficiency can be obtained using various methods such as a simplistic PRA model. Table 2 presents the database structure.

Function	Flow	Flow quality			
		Failure	Concern		Nominal
			low	high	
Function A	Flow 1	$\lambda_{f,A\to1}$	$\lambda_{l,A\to1}$	$\lambda_{h,A\to1}$	$\lambda_{n,A\to1}$
	...				
	Flow N	$\lambda_{f,A\to N}$	$\lambda_{l,A\to N}$	$\lambda_{h,A\to N}$	$\lambda_{n,A\to N}$
...					

Table 3. Function failure link database architecture - **F**

Function or external Flow (Deepest level)	Emergent weakness probability (per year or per use)		
	Failure	Concern	
		low	high
Function A	ω_A	-	-
...			
Flow Z	$\omega_{f,A}$	$\omega_{l,A}$	$\omega_{h,A}$

Table 4. Emergent weakness database architecture - **W**

The f-database, noted **F**, links a function failure to an outgoing flow weakness. Its goal is to indicate to the simulation the likelihood λ of failure type propagation through the system. Within this study, a function state is considered binary. It is either in a failed state or in a nominal state. However, a function failure exhibits a non-binary interaction with the outgoing flows, due to various potential physical causes, giving way to the notion of *flow quality*. In this paper, the outgoing flows' quality from a failed function can be categorized in the following subset s of probability λ: failed (probability λ_f), of low concern (probability λ_l), of high concern (probability λ_h), or nominal (probability λ_n). The database architecture is explicited in Table 3.

The flow quality can then be detected with varying degrees of efficiency by PHM equipment, according to data in **P**. This modifies the propagation probabilities of the failure through the system. For each function in the system, the designers compute the probability of the output flows quality being in each of the states of s. Populating **F** can prove challenging to the designer and mostly rest on expert judgment. Consequently, the designer may skip this step for the unresolved function failure's impact. The algorithm will then automatically modify s by rendering it binary: failed or nominal flow.

The w-database, noted **W**, contains information relative to the independent function failure probability. It is similar to the database used in methods such as FFIP or FFDM, and the populating algorithms are identical. In this paper, it is assumed that there is no independent internal flow failure probability, but there is an independent external flow failure probability λ. The external flow failure probability links the functional model to the system boundaries. In other words, in the proposed method, a flow within the system can only fail if its parents function fail. However, the functions beyond the system boundaries are not simulated explicitly. The potential failure of such functions is thus carried into the system by independent failure probabilities of external flows.

W can be populated using component-level historical failure data and mapping each component failure to a function or flow failure. This mapping function is not straightforward, as physical effects from component specifications can affect the function or flow failure.

In this study, illustrative rates of occurence have been selected, based on expert elicitation. The correctness of the data considered does not impact the methodology algorithm.

3.3. Trees computing

A tree represent a path through a functional system. In order to compute all the possible paths through a system, several trees might be needed.

The paths are computed by going through the given model using the following algorithm A.1:

(A.1) Step 1 The entry points are identified. An entry point is a function or a flow within the logical functional model for which the parent functions or flows are either the boundary or non-defined. An entry point represent a point of entry for a complex system.

(A.1) Step 2 The *risk-critical point* and the *reliability-critical point* are defined by the design team. The risk-critical point is the function or flow which represent a failure of the system leading to a safety issue. The reliability-critical point is the function or flow which represent the failure of the system to operate as designed.

(A.1) Step 3 Starting from each entry point in turn, an algorithm computes the tree leading to the risk-critical point, as well as the tree leading to the reliability-critical point.

A simplified mockup of a functional model can be seen in Figure 3. This mockup is used to illustrate the algorithm used to compute the various paths through a system. The letters represent functions, while the connections between the letters represent flows. In the simple logical functional model of interest, two gates are considered, OR and AND. The system is considered isolated, not receiving input from outside its boundary. In this simplified example, the function set S is populated with $S = \{A, F\}$ (step 1). The risk-critical point is selected to be the function A, and the reliability-critical point is given to be the function E (step 2).

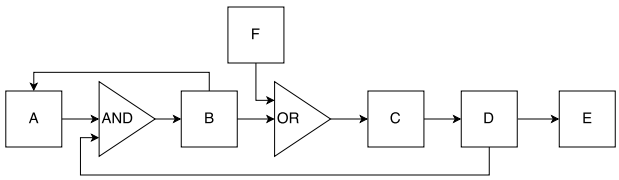

Figure 3. Simplified mockup example

Starting from function A, the algorithm computes the possible next function in the tree. Only B is possible. From B, two distinct paths can be followed. The first one goes back to A. This is a loop, and the tree is thus discarded. The second possibility is to go to C, and then D. From D, two more paths can be followed. The first one goes back to B, again generating a closed loop. Consequently, the longest possible tree $P_{A,rel}$ to attain the reliability-critical point E is shown in figure 4. The longest tree $P_{A,ris}$ to attain the risk-critical point A is obviously the tree containing only the node A.

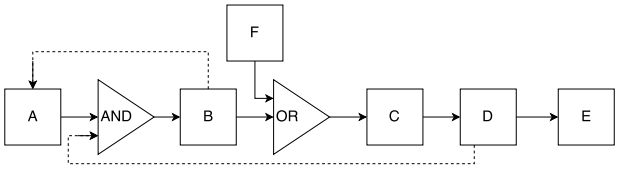

Figure 4. $P_{A,rel}$ tree through the system

Starting from function F and following the same algorithm, the longest non-looping tree $P_{F,ris}$ to get to risk-critical point A is obtained. The longest tree $P_{F,rel}$ to the reliability-critical point E is also computed. Both paths are presented in figure 5.

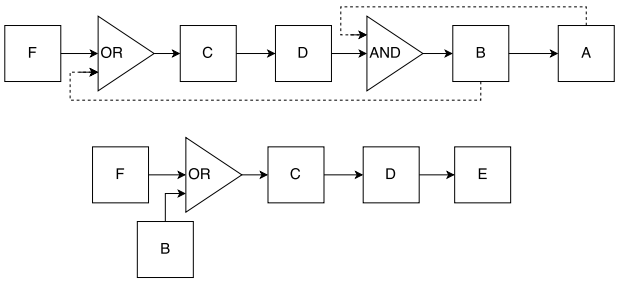

Figure 5. $P_{F,ris}$ (top) and $P_{F,rel}$ (bottom) trees through the system

In order to avoid redundancies, the algorithm then combines the obtained trees to eliminate the subtrees. A subtree is a computed tree that appears fully in another computed tree. In the example considered, it is easy to see that $P_{A,ris}$ is a subtree of $P_{F,ris}$, and that $P_{F,rel}$ is a subtree of $P_{A,rel}$. Consequently, in the simplified illustrative example considered, the combination of the two trees $P_{A,rel}$ and $P_{F,ris}$ represents the whole system, from a risk point of view and from a reliability point of view.

3.4. PHM hardware positions

Theoretically, in order to select the best possible combination of PHM hardware and their position in the system, each combination has to be considered, analyzed, and compared with the others. This is, however, not practical, due to computational time issues. Consequently, an algorithm is devised to select the best possible combinations of the PHM hardware positions throughout the system from a reduced list.

The combinations list reduction is rendered possible by using several assumptions. The first assumption is that each particular PHM equipment can only monitor specific categories of flows and functions with varying efficiency. This allows the algorithm to not link incompatible functions or flows and PHM hardware. The second assumption, which is optional, is that the inventory available to the designer is limited, for example, by incompatible hardware size or cost. Hence, the designer can inform the algorithm that only n sensors of type X are at its disposal. The third assumption is that the best results will be obtained with the maximum possible number of sensors in the system. Finally, a fourth assumption accounts for potential constraints, forcing a function or flow to be monitored by a specific hardware or to not be monitored.

Assumptions 1 and 2 are easily justified. Assumption 3 can be argued with, based on the hardware efficiency and especially its false alarm rate for a given function or flow being monitored. An inadequate PHM sensor could theoretically increase failure probability of a function, if the false alarm rate is high enough and the correction success rate low enough. That said, the approximation holds sufficiently well most of the time, allowing the designer to converge on a reasonable functional model. At this point, the designer can lift the third assumption and compute a final optimization for PHM positions in the system, the first two assumptions limiting the number of possible permutations to keep the computation time within reason.

3.5. Bayesian network nodes

We can recall that a Bayesian network is composed of nodes, linked together by relationships. Each node probabilistic outcome impacts its daughter nodes. Consequently, in a Bayesian network, knowing the state of the parents (e.g. $P(W)$ and $P(C)$) automatically gives the state of the children (e.g. $P(F|W,C)$). The model does not need to know anything else other than the parent nodes' states about the system. This presents a certain advantage for a complex system by not requiring extensive computer memory use.

In this paper, three categories of Bayesian network nodes are considered from within the logical functional model: the gates, the functions, and the flows. The functions and flows categories are each divided into four nodes to account for the potential presence of PHM hardware. The four different nodes associated to a function or a flow are (1) weakness, (2) detection, (3) correction and (4) failure. Logical gate nodes can be used to model system redundancies or requirements. Each specific node can be attributed a database. The weakness nodes use **W**, the detection nodes use **P**, and the correction nodes use **H**. The failure nodes use **F** to link with their child weakness nodes.

Figure 6 presents an excerpt of a system based on the functional prognostics Bayesian network model. In this subsys-

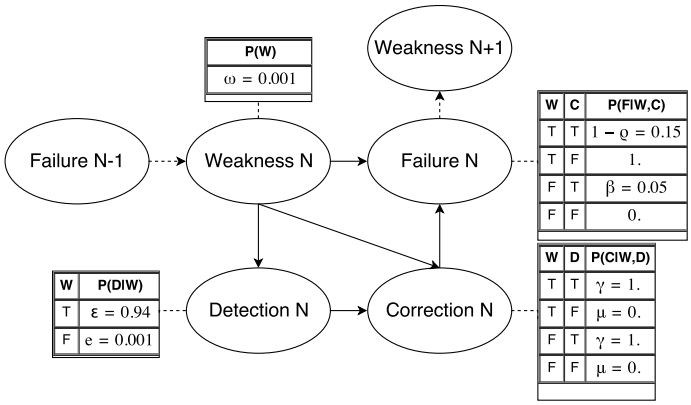

Figure 6. Example of conditional probability tables in the proposed prognostic bayes net.

	F14	Y				N			
	F24	Y		N		Y		N	
	F34	Y	N	Y	N	Y	N	Y	N
Gate	Y	1	0	0	0	0	0	0	0
	N	0	1	1	1	1	1	1	1

Table 5. Conditional Probability Table for an AND-gate of size 3.

tem, an emergent weakness of a monitored component is considered with a rate of 0.001 per year ($\omega = 0.001$). The PHM hardware used has a 94% efficiency to detect a weakness of the particular function or flow, and a 0.001 chance of signaling a false alarm ($\varepsilon = 0.94$ and $e = 0.001$). Every time the sensor detection model gives a positive signal, the maintenance team is sent to repair the function or flow, or the automatic corrective actions devised are activated. The corrective action success rate is set at 85%, and the maintenance team failing the function or flow even though the signal was only a false alarm is considered to happen 5% of the time ($\rho = 0.85$ and $\beta = 0.05$).

The gates category is simply used to model a more complex flows-to-function relationship in the system design. In that sense, a gate node can only take two probabilistic outcomes: true or false. Tables 5, 6 and 7 show respectively, for an AND-gate, an OR-gate and a Voting-Or gate, how a gate is modeled in the functional Bayesian network representation of the logical functional model from Figure 2. In those tables, Y represents a nominal state and N represents a failure state.

The nodes associated with the functions category are considered *binary events* in this paper. A binary event is defined as a node being in one of two states. Consequently, a function node (weakness, detection, correction, and failure nodes) can only be in one of two states, true or false. Each state carries a specific probability, dependent on the states of the parents' nodes.

	F14	Y				N			
	F24	Y		N		Y		N	
	F34	Y	N	Y	N	Y	N	Y	N
Gate	Y	1	1	1	1	1	1	1	0
	N	0	0	0	0	0	0	0	1

Table 6. Conditional Probability Table for an OR-gate of size 3.

	F14	Y				N			
	F24	Y		N		Y		N	
	F34	Y	N	Y	N	Y	N	Y	N
Gate	Y	1	1	1	0	1	0	0	0
	N	0	0	0	1	0	1	1	1

Table 7. Conditional Probability Table for a 2-of-3 gate.

For the flows weakness nodes, four flow quality states are considered, from the ensemble s. We recall that the ensemble s represents the following flow quality states: failed, of low concern, of high concern, or nominal. The nodes associated with flow quality from s are named *s-events*. The PHM hardware efficiency and error rate is impacted by the flow quality, following the data given in **P**. The four-states weakness node eventually translates to a binary event representing the flow failure node, using inputs from the binary events modeling the detection node and the correction node. The four-states weakness is thus used to refine the flow failure probability.

In order to illustrate the algorithm presented, a small example is given. Figure 7 presents a very simple functional model, and Figure 8 represents its translation into the proposed method model, provided each function and flow are linked to a PHM device. The interaction with the various databases, **F**, **H**, **M**, **P** and **W** is also shown in Figure 8. If the designer were to force the flow f_{12} not to be equipped with a PHM hardware, the nodes *Detection* f_{12} and *Correction* f_{12} would disappear from the model, along with the connections.

3.6. Bayesian network algorithm

The Bayesian network used to describe the whole system is based upon the following algorithm A.2 steps. Steps 1 through 4 are applied to a function. Step 5 links a function with its outgoing flows. Steps 6 through 8 are applied to a flow. Step 9-a links a flow with its receiving function. Step 9-b is applied to a gate.

(A.2) Step 0 This step is optional. In a Bayesian network, the designer can set the states of several functions and flows. While not particularly useful in generating the risk and reliability analysis on the system during the early design phase, it can be noted that this feature can be used simultaneously as a Prognostics and Diagnostics tool during the operational lifetime of the system.

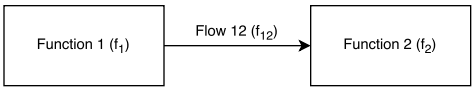

Figure 7. Simple functional model.

(A.2) Step 1 The algorithm computes the probability of a function weakness, given the observed evidence (step 0) or **W**. The function weakness node is a binary event, meaning that it can only take one of two potential states. Consequently, either there is a weakness (state $S_W = Y$) or the function is in a nominal state ($S_W = N$), as represented by W in Eq. (2) for the function f_1.

$$P(W_{f_1}) = \begin{cases} \omega_{f_1} & \text{if } S_W = Y \\ \bar{\omega_{f_1}} = 1 - \omega_{f_1} & \text{if } S_W = N \end{cases} \tag{2}$$

(A.2) Step 2 The weakness probability associated with f_1 has been computed in step 1. The probability of being in a state of detection by a PHM hardware can now be calculated. Several potential states can be described to link a weakness of a function to its detection. There could effectively be a weakness, and this weakness could be detected according to the attached hardware efficiency ε_{phm,f_1}. The probability of this event will be noted d_{ε,f_1}. Alternatively, there might be no function weakness, but a false alarm is raised by the hardware according to its false alarm rate e_{phm,f_1}. The probability of this event will be noted d_{e,f_1}. The combination of these two events forms the probability of a detection.

The probability of being in the state of non detection, is obviously the complement of the probability of detection. Either the weakness is present and not detected, or there is no weakness, and no false alarm is raised. The different probability paths leading to the probability of detection are represented in Table 8.

Given the function weakness probability, the detection conditional probability matrix $P(D_{f_1}|W_{f_1})$ obtained is displayed in Eq. (3).

$$P(D_{f_1}|W_{f_1}) = \begin{cases} d_{\varepsilon,f_1} + d_{e,f_1} & \text{if } S_D = Y \\ d'_{\varepsilon,f_1} + d'_{e,f_1} & \text{if } S_D = N \end{cases} \tag{3}$$

Where:

$$\begin{aligned} d_{\varepsilon,f_1} &= \omega_{f_1}\varepsilon_{phm,f_1} \\ d_{e,f_1} &= (1-\omega_{f_1})e_{phm,f_1} \\ d'_{\varepsilon,f_1} &= \omega_{f_1}(1-\varepsilon_{phm,f_1}) \\ d'_{e,f_1} &= (1-\omega_{f_1})(1-e_{phm,f_1}) \end{aligned}$$

This step is performed if and only if the function of interest is equipped with a PHM hardware. Indeed, if a PHM hardware is not attached to the function, the detection is obviously non-existent, implying $\varepsilon_{phm,f_1} = 0$ and $e_{phm,f_1} = 0$. Entering these numbers in Eq. (3), we obtain Eq. (4).

$$P(D_{f_1}|W_{f_1}) = \begin{cases} 0 & \text{if } S_D = Y \\ 1 & \text{if } S_D = N \end{cases} \tag{4}$$

(A.2) Step 3 The weakness probability and the detection probability have been computed respectively in step 1 and step 2. This third step estimates the probability of a corrective action being attempted. The potential scenarios leading to the corrective action are treated by the conditional probability table presented in Table 8. In the case of an actual weakness, the detector could detect the weakness (d_{ε,f_1}). Then, the maintenance team is sent to repair according to a decision probability γ_{f_1} given by **M**. Alternatively, the weakness is not detected (d'_{ε,f_1}) but a scheduled non required maintenance is done on the function, according to a probability μ_{f_1} also given by **M**. The combination of these two events translate to a probability noted c_{ω,f_1}. Maintenance can also be carried out on the function if no weakness actually happened. This event is true if a false alarm (d_{e,f_1}) caused the maintenance team to mobilize or if a scheduled non required maintence is performed. These two events can be combined to obtain a probability noted $c_{\bar{\omega},f_1}$. Finally, the corrective action probability can be calculated by combining c_{ω,f_1} with $c_{\bar{\omega},f_1}$.

Considering the fact that the corrective action is a binary event, the probability of no corrective action being carried out is obviously the complement of the probability that a corrective action is performed.

Given the function weakness probability $P(W_{f_1})$ and the detection probability matrice $P(D_{f_1}|W_{f_1})$, the conditional corrective action matrice obtained is displayed in Eq. (5).

$$P(C_{f_1}|W_{f_1}, D_{f_1}) = \begin{cases} c_{\omega,f_1} + c_{\bar{\omega},f_1} & \text{if } S_C = Y \\ c'_{\omega,f_1} + c'_{\bar{\omega},f_1} & \text{if } S_C = N \end{cases} \tag{5}$$

Where:

$$\begin{aligned} c_{\omega,f_1} &= d_{\varepsilon,f_1}\gamma_{f_1} + d'_{\varepsilon,f_1}\mu_{f_1} \\ c_{\bar{\omega},f_1} &= d_{e,f_1}\gamma_{f_1} + d'_{e,f_1}\mu_{f_1} \\ c'_{\omega,f_1} &= d_{\varepsilon,f_1}(1-\gamma_{f_1}) + d'_{\varepsilon,f_1}(1-\mu_{f_1}) \\ c'_{\bar{\omega},f_1} &= d_{e,f_1}(1-\gamma_{f_1}) + d'_{e,f_1}(1-\mu_{f_1}) \end{aligned}$$

(A.2) Step 4 Based on the weakness probability and the corrective action probability calculated respectively in step 1 and step 3, the algorithm computes the probability of the function failure using the conditional probability table displayed in Table 8. In the case of an actual weakness, the path leading

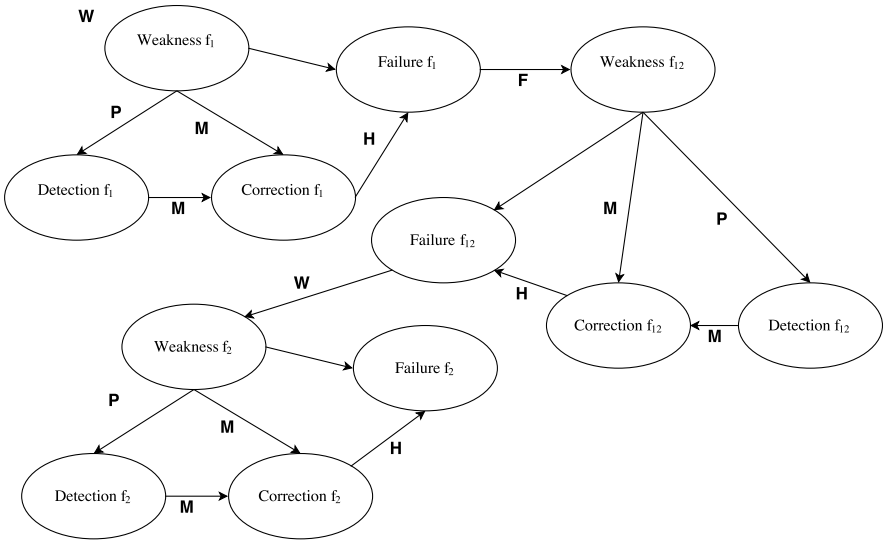

Figure 8. Translation of a simple function model (Figure 7) to its functional prognostic Bayesian network form.

$P(D_{f_1}\|W_{f_1})$	Weakness f_1	Y		N	
Detection f_1	Y	ε_{phm,f_1}		e_{phm,f_1}	
	N	$1-\varepsilon_{phm,f_1}$		$1-e_{phm,f_1}$	
$P(C_{f_1}\|W_{f_1},D_{f_1})$	Weakness f_1	Y		N	
	Detection f_1	Y	N	Y	N
Correction f_1	Y	γ_{f_1}	μ_{f_1}	γ_{f_1}	μ_{f_1}
	N	$1-\gamma_{f_1}$	$1-\mu_{f_1}$	$1-\gamma_{f_1}$	$1-\mu_{f_1}$
$P(F_{f_1}\|W_{f_1},C_{f_1})$	Weakness f_1	Y		N	
	Correction f_1	Y	N	Y	N
Failure f_1	Y	$1-\rho_{f_1}$	1	β_{f_1}	0
	N	ρ_{f_1}	0	$1-\beta_{f_1}$	1

Table 8. Conditional probability tables for the weakness detection, correction and failure of a function f_1

to a failure can be that no corrective action was performed (c'_{ω,f_1}), or that a corrective action was performed (c_{ω,f_1}) but was unsuccessful, according to the ρ_{f_1} value given in **H**. Alternatively, a function can fail if there was no weakness but a corrective action was still performed ($c_{\bar{\omega},f_1}$) and generated a function failure according to the mishandling probability β_{f_1} given by **H**. The failure probability of f_1 is obtained by combining these different scenarios.

The probability that the function does not fail is obtained by combining the following three possibilities. A weakness was present but was corrected following the ρ_{f_1} value. No weakness was present and no action was performed. No weakness was present and the performed action did not fail the function, according to the mishandling probability β_{f_1}. This corresponds to the complement of the probability of failure.

Given the function weakness probability $P(W_{f_1})$ and the correction probability matrice $P(C_{f_1}|W_{f_1},D_{f_1})$, the conditional failure matrice obtained is displayed in Eq. (6).

$$P(F_{f_1}|W_{f_1},C_{f_1}) = \begin{cases} f_{f_1} & \text{if } S_F = Y \\ f'_{f_1} & \text{if } S_F = N \end{cases} \tag{6}$$

Where:

$$f_{f_1} = c_{\omega,f_1}(1-\rho_{f_1}) + c'_{\omega,f_1} + c_{\bar{\omega},f_1}\beta_{f_1}$$
$$f'_{f_1} = c_{\omega,f_1}\rho_{f_1} + c'_{\bar{\omega},f_1} + c_{\bar{\omega},f_1}(1-\beta_{f_1})$$

(A.2) Step 5 A function failure can be linked to different outgoing flow qualities. The flow quality represents a state of weakness and is modeled by an s-event. An s-event is an event that can take four distinct states of flow quality. The quality of a flow can be categorized as failed (λ_f), of low concern (λ_l), of high concern (λ_h), or nominal (λ_n). The algorithm considers that the quality of a flow cannot spontaneously change. The quality of a flow can only change when

the function it originates from is in a failed state. This represents a limitation of the simulation, as it does not allow for the treatment of failure flows going through a function without failing it.

F contains the detailed data for each specific function-flow connection. The probability of weakness $P(W_{f_{12}}|F_{f_1})$ is displayed in Eq. (7).

$$P(W_{f_{12}}|F_{f_1}) = \begin{cases} w_{f,f_{12}} & \text{if } S_W = f \\ w_{l,f_{12}} & \text{if } S_W = l \\ w_{h,f_{12}} & \text{if } S_W = h \\ w_{n,f_{12}} & \text{if } S_W = n \end{cases} \tag{7}$$

Where:

$$\begin{aligned} w_{f,f_{12}} &= \lambda_{f,f_{12}} f_{f_1} \\ w_{l,f_{12}} &= \lambda_{l,f_{12}} f_{f_1} \\ w_{h,f_{12}} &= \lambda_{h,f_{12}} f_{f_1} \\ w_{n,f_{12}} &= f_{f_1} + f'_{f_1} - \textstyle\sum_{i\in[f,h,l]} f_{f_1}\lambda_i \end{aligned}$$

(A.2) Step 6 The weakness probability of f_{12} has been computed in step 5. The probability of a detection can now be calculated. Similar to step 2, several potential states can be described to link a weakness of a flow to its detection. There could effectively be a flow quality weakness, which is detected according to the attached hardware efficiencies. The hardware efficiencies for a flow are given for the three states of degraded operation, low concern ($\varepsilon_{phm,l,f_{12}}$), high concern ($\varepsilon_{phm,h,f_{12}}$) and failed ($\varepsilon_{phm,f,f_{12}}$). The probability of a scenario in which a weakness is present and detected will be noted $d_{\varepsilon_i,f_{12}}$, for $i \in [f,l,h]$. Alternatively, a detection might occur if there is no flow weakness, but a false alarm is raised by the hardware according to its false alarm rate $e_{phm,f_{12}}$. The probability of this event will be noted $d_{e,f_{12}}$. The combination of these two events forms the probability of a detection. This can be seen in Table 9.

The probability of not having a detection is the complement of the probability of having a detection. Indeed, if the flow quality is not nominal, the detector might fail to detect it, with a probability depending on its efficiency. If the flow quality is nominal, the detector can also not signal any issue, based on its false alarm rate.

Given the flow weakness probability, the detection conditional probability matrice $P(D_{f_{12}}|W_{f_{12}})$ obtained is displayed in Eq. (8).

$$P(D_{f_{12}}|W_{f_{12}}) = \begin{cases} \sum_{i\in[f,h,l]} d_{\varepsilon_i,f_{12}} + d_{e,f_{12}} & \text{if } S_D = Y \\ \sum_{i\in[f,h,l]} d'_{\varepsilon_i,f_{12}} + d'_{e,f_{12}} & \text{if } S_D = N \end{cases} \tag{8}$$

Where:

$$\begin{aligned} d_{\varepsilon_i,f_{12}} &= \omega_{i,f_{12}}\varepsilon_{phm,i,f_{12}} \\ d_{e,f_{12}} &= \omega_{n,f_{12}} e_{phm,f_{12}} \\ d'_{\varepsilon_i,f_{12}} &= \omega_{i,f_{12}}(1-\varepsilon_{phm,i,f_{12}}) \\ d'_{e,f_{12}} &= \omega_{n,f_{12}}(1-e_{phm,f_{12}}) \end{aligned}$$

This step is performed if and only if the function of interest is equipped with a PHM hardware. Indeed, if a PHM hardware is not attached to the function, the detection is obviously in a false state.

(A.2) Step 7 The flow weakness probability and the detection probability have been computed respectively in step 5 and step 6. This next step estimates the probability of a corrective action being attempted. The potential scenario leading to the corrective action is treated by the conditional probability table presented in Table 9. In the case of an actual weakness, the detector could detect the weakness according to the corresponding hardware efficiency for each flow quality ($d_{\varepsilon_i,f_{12}}$ for $i \in [f,l,h]$). If the event is detected, the maintenance team is sent to repair according to a decision probability $\gamma_{i,f_{12}}$ for $i \in [f,l,h]$, based on the team management and the detected flow quality weakness. The decision probability is given by **M**. Alternatively, the weakness could be undetected ($d'_{\varepsilon_i,f_{12}}$ for $i \in [f,l,h]$) but a scheduled non required maintenance could be performed on the system which would impact the flow, according to a probability $\mu_{f_{12}}$ also given by **M**. The combination of these two events translates to a probability noted $c_{\omega,i,f_{12}}$ for $i \in [f,l,h]$. Maintenance can also be carried out on the system, with a direct impact on the flow f_{12} if no weakness actually happened. This event is true if a false alarm ($d_{e,f_{12}}$) caused the maintenance team to mobilize or if a scheduled non required maintence is performed. These two events can be combined to obtain a probability noted $c_{\bar{\omega},f_{12}}$. Finally, the corrective action probability can be calculated by combining $c_{\omega,i,f_{12}}$ with $c_{\bar{\omega},f_{12}}$ for $i \in [f,l,h]$.

The probability that no corrective action is attempted is the complement of the probability that a corrective action is carried out.

Given the flow weakness probability $P(W_{f_{12}}|F_{f_1})$ and the detection probability matrice $P(D_{f_{12}}|W_{f_{12}})$, the conditional corrective action matrice obtained is displayed in Eq. (9).

$$P(C_{f_{12}}|W_{f_{12}},D_{f_{12}}) = \begin{cases} \sum_{i\in[f,l,h,n]} c_{\omega_i,f_{12}} & \text{if } S_C = Y \\ \sum_{i\in[f,l,h,n]} c'_{\omega_i,f_{12}} & \text{if } S_C = N \end{cases} \tag{9}$$

Where, for $i \in [f,l,h]$:

$P(D_{f_{12}}\|W_{f_{12}})$	Weakness f_{12}	Failed	Low concern	High concern	Nominal
Detection f_{12}	Y	$\varepsilon_{phm,f,f_{12}}$	$\varepsilon_{phm,l,f_{12}}$	$\varepsilon_{phm,h,f_{12}}$	$e_{phm,f_{12}}$
	N	$1-\varepsilon_{phm,f,f_{12}}$	$1-\varepsilon_{phm,l,f_{12}}$	$1-\varepsilon_{phm,h,f_{12}}$	$1-e_{phm,f_{12}}$

$P(C_{f_{12}}\|W_{f_{12}},D_{f_{12}})$	Weakness f_{12}	Failed		Low concern		High concern		Nominal	
	Detection f_{12}	Y	N	Y	N	Y	N	Y	N
Correction f_{12}	Y	$\gamma_{f,f_{12}}$	$\mu_{f,f_{12}}$	$\gamma_{l,f_{12}}$	$\mu_{l,f_{12}}$	$\gamma_{h,f_{12}}$	$\mu_{h,f_{12}}$	$\gamma_{n,f_{12}}$	$\mu_{n,f_{12}}$
	N	$1-\gamma_{f,f_{12}}$	$1-\mu_{f,f_{12}}$	$1-\gamma_{l,f_{12}}$	$1-\mu_{l,f_{12}}$	$1-\gamma_{h,f_{12}}$	$1-\mu_{h,f_{12}}$	$1-\gamma_{n,f_{12}}$	$1-\mu_{n,f_{12}}$

$P(F_{f_{12}}\|W_{f_{12}},C_{f_{12}})$	Weakness f_{12}	Failed		Low concern		High concern		Nominal	
	Correction f_{12}	Y	N	Y	N	Y	N	Y	N
Failure f_{12}	Y	$1-\rho_{f,f_{12}}$	1	$1-\rho_{l,f_{12}}$	1	$1-\rho_{h,f_{12}}$	1	$\beta_{f_{12}}$	0
	N	$\rho_{f,f_{12}}$	0	$\rho_{l,f_{12}}$	0	$\rho_{h,f_{12}}$	0	$1-\beta_{f_{12}}$	1

Table 9. Conditional probability tables for the weakness detection, correction and failure of a flow f_{12}

$$
\begin{aligned}
c_{\omega_i,f_{12}} &= d_{\varepsilon_i,f_{12}}\gamma_{i,f_{12}} + d'_{\varepsilon_i,f_{12}}\mu_{f_{12}}\\
c_{\omega_n,f_{12}} &= d_{e,f_{12}}\gamma_{n,f_{12}} + d'_{e,f_{12}}\mu_{f_{12}}\\
c'_{\omega_i,f_{12}} &= d_{\varepsilon_i,f_{12}}(1-\gamma_{i,f_{12}}) + d'_{\varepsilon_i,f_{12}}(1-\mu_{f_{12}})\\
c'_{\omega_n,f_{12}} &= d_{e,f_{12}}(1-\gamma_{n,f_{12}}) + d'_{e,f_{12}}(1-\mu_{f_{12}})
\end{aligned}
$$

(A.2) Step 8 Based on the flow weakness probability and the corrective action probability calculated respectively in steps 5 and 7, the algorithm computes the probability of the flow failure using the conditional probability table displayed in Table 9.

In the case of an actual weakness, either of low concern, of high concern, or failed, the path leading to a failure can be that no corrective action was performed $\left(\sum_{i\in[f,l,h]} c'_{\omega_i,f_{12}}\right)$, or that a corrective action was performed $\left(\sum_{i\in[f,l,h]} c_{\omega_i,f_{12}}\right)$ but was unsuccessful, according to the $\rho_{i,f_{12}}$ values for $i \in [f,l,h]$ given in **H**. Alternatively, the flow can fail if there was no weakness but a corrective action was still performed on the system ($c_{\omega_n,f_{12}}$) and generated a function failure according to the mishandling probability $\beta_{f_{12}}$ given by **H**.

Given the function weakness probability $P(W_{f_1})$ and the correction probability matrice $P(C_{f_1}|W_{f_1},D_{f_1})$, the conditional failure matrice obtained is displayed in Eq. (10).

$$
P(F_{f_{12}}|W_{f_{12}},C_{f_{12}}) = \begin{cases} f_{f_{12}} & \text{if } S_F = Y \\ f'_{f_{12}} & \text{if } S_F = N \end{cases} \tag{10}
$$

Where:

$$
\begin{aligned}
f_{f_{12}} &= c_{\omega_n,f_{12}}\beta_{f_{12}} + \textstyle\sum_{i\in[f,l,h]} c_{\omega_i,f_{12}}(1-\rho_{i,f_{12}}) + c'_{\omega_i,f_{12}}\\
f'_{f_{12}} &= c_{\omega_n,f_{12}}(1-\beta_{f_{12}}) + c'_{\omega_n,f_{12}} + \textstyle\sum_{i\in[f,l,h]} c_{\omega_i,f_{12}}\rho_{i,f_{12}}
\end{aligned}
$$

If the next Bayesian node in the model is a logical gate, the algorithm goes to step 9-b, otherwise, it goes to step 9-a.

(A.2) Step 9-a A failed flow is considered to fail a receiving function since the function will not be able to perform its task without a necessary flow. The failure probability obtained in step 8 for the flow is thus passed fully to the next function in the model. The emergent weakness of the next function in the model is also considered. Consequently, given $P(F_{f_{12}}|W_{f_{12}},C_{f_{12}})$, the weakness probability seen by the next function is displayed in the conditional failure matrice in Eq. (11).

$$
P(W_{f_2}|F_{f_{12}}) = \begin{cases} f_{f_{12}} + \omega_{f_2} & \text{if } S_W = Y \\ f'_{f_{12}} - \omega_{f_2} & \text{if } S_W = N \end{cases} \tag{11}
$$

The algorithm returns to step 1.

(A.2) Step 9-b Logical gates can combine several flows and compute the next function weakness associated. Consider another flow, f_{32}, supplying a redundant flow to function f_2. An OR-gate is placed in the model, so that only one flow, f_{12} or f_{32} is needed for function f_2 to operate nominally.

The probability that the flow failures propagate through the gate $g_{12,32}$ to the next function weakness, $P(F_{f_2}|F_{f_{12}},F_{f_{32}})$, is shown in Eq. (12).

$$
P(F_{g_{12,32}}|F_{f_{12}},F_{f_{23}}) = \begin{cases} f_{f_{12}}f_{f_{32}} & \text{if } S_F = Y \\ f'_{f_{12}}f_{f_{32}} + f_{f_{12}}f'_{f_{32}} + f'_{f_{12}}f'_{f_{32}} & \text{if } S_F = N \end{cases} \tag{12}
$$

The gate failure probability becomes the new flow failure probability. The algorithm returns to step 9-a.

3.7. Engineering decision framework

For each specific permutations retained, and for each tree computed through the system, the prognostic functional Bayesian network is automatically updated by the algorithm. In order to compare the various possibilities, a score must be computed for each possibility. This score is defined as the failure probability of the *critical failure point*. The critical failure point is the reliability-critical point or the risk-critical

Trees	Configuration A	Conf. B	Conf. C
A	$P(f_A\|C_A)$	$P(f_A\|C_B)$	$P(f_A\|C_C)$
B	$P(f_B\|C_A)$	$P(f_B\|C_B)$	$P(f_B\|C_C)$
C	$P(f_C\|C_A)$	$P(f_C\|C_B)$	$P(f_C\|C_C)$

Table 10. Combination of the optimized configurations

point, depending on the tree considered, defined by the design team in the algorithm A.1. It can be a function or a flow. The failure probability of the critical failure point englobes all of its ancestors' failure probabilities. It can thus be equated to the system's failure probability, according to the critical failure point identified by the design team.

The goal of the engineering decision framework is to optimize the PHM sensor selection and locations. The critical failure point is consequently used as an objective function for the optimization algorithm. The PHM sensors' availability is considered as the constraint.

For each tree computed, an optimized PHM sensor selections and locations map is obtained. These configurations may differ depending on the considered tree. In order to reconcile the configurations and compute the optimized configuration and final failure probabilities for the system as a whole, the following algorithm A.3 is adopted:

(A.3) Step 1 The engineering team decides on some thresholds probabilities for the risk-critical failure probability and for the reliability-critical failure probability.

(A.3) Step 2-a If one of these thresholds is not met, the system design is to be modified. The available sensors cannot be used to lower the probabilities under the thresholds, the system is consequently considered insufficient.

(A.3) Step 2-b If the thresholds are met, the combination of the configurations can be attempted. For each individual tree computed, the optimized configurations obtained for all the other trees are applied, as shown in Table 10.

(A.3) Step 3 A configuration is eliminated if the resulting probability of critical node failure passes above the defined threshold. The engineering team can then select the most appropriate configuration for their system. It is possible that no configurations can satisfy the threshold for each and every tree. In such a case, the system is considered insufficient. The specific failing configurations can be analyzed to determine the best course of action for the design team.

3.8. Automatic framework

A framework was developed to facilitate the application of the PHASED method (L'Her, 2016). This framework was developed mostly using Python and the pgmpy package (Ankan & Panda, 2015) and is not computationally optimized. The software allows users to easily input the various identified trees for a logical functional model and populate the databases using an open source human readable data serialization language, YAML. The software then computes the user-defined critical points failure probabilities and outputs the results.

3.9. Review

The PHASED methodology has been divided into six main steps. A logical functional model is built for the system and databases are created. Trees parsing the logical functional model are computed. Various PHM hardware configuration are obtained for each tree, and the resulting model is translated into Bayesian networks. The Bayesian network are solved and output the probability of failures of user-defined critical points in the model. The engineering team combines the different results obtained for each tree to generate the optimum PHM hardware configuration. Informed design decisions can then be made to improve upon it. Table 11 reviews the main points of the PHASED methodology presented in this paper.

4. CASE STUDY

To illustrate the PHASED method introduced in this paper, a simplified pressurized water reactor case study is discussed. The Piping and Instrumentation Diagram (P&ID) is drawn in Figure 9. Only the top-level components are considered, to represent an early design phase. The system studied contains the nuclear reactor core. One primary pump is designed, supplied in electricity by either a derivation of the main generator output or by one of two backup diesel generators. The water in the primary vessel is kept liquid by a pressurizer. The steam generated by a steam generator activates the turbine, which feeds into the electricity generator. The vapor is then condensed back to liquid using a condenser, and pumped back to the steam generator using a pump only fed by the electricity generator.

This system intentionally does not correspond to an existing PWR design. This section demonstrates how to use the proposed method to assess the power plant early design considering prognostic and health management conducted throughout the system lifetime. Various design improvement are consequently analyzed, such as removing or adding redundancies into the system and observing their impact.

This case study illustrates how the proposed tool can be used by a designer from the project onset. The following steps will be demonstrated for the study:

1. Construction of the logical functional model,
2. Definition of the system risk-critical and reliability-critical nodes,
3. Computation of the spanning trees,
4. Description of the available PHM hardware inventory,
5. Population of the databases,

Logical functional model	A logical functional model is built to represent the system.
Databases	Five databases are created, to encode information about: • the PHM hardware (**P**), • the function failure rates (**W**), • the corrective actions likelihood of success (**H**), • the management style (**M**), • the link between a function failure and its outgoing flow qualities (**F**).
Trees	Trees representing various paths through the system are computed in order to parse the whole system, to compensate for the fact that a Bayesian network cannot model feedback loops and to account for both a system risk and a system reliability point of view.
PHM hardware selection	The possible configurations of PHM harware selection and positioning in the system are computed. The optimized configuration will be obtained from the set of the possible configurations, reduced according to several assumptions.
Bayesian network nodes	The logical functional model is enhanced with the PHM hardware selection and position through the system. The resulting system is translated to a Bayesian network.
Bayesian network algorithm	The properties of Bayesian networks are used to compute the failure probabilities for every node (function or flow) in the model, using data from the given databases.
Engineering decision	For each identified tree, an optimized PHM hardware configuration is obtained, along with a failure probability of the user-defined critical point. Engineers combine these configurations to select the best PHM-enhanced system for their design or to modify the system if no acceptable configuration is obtained.

Table 11. Review of the PHASED methodology

6. Desired results and analysis

4.1. Logical functional model

A formalism, based on open source human readable data serialization language, namely YAML, has been adopted to facilitate the designers' task. An integrated drawing tool will be important to ensure comfort and improved quality assurance.

Figure 10 (appendix) translates the P&ID from Figure 9 into the logical functional model introduced in section 3.1. Due to the very nature of a Bayesian network, feedback loops cannot be taken into account. This is shown using the *disconnected* links (dotted lines). To simulate those feedbacks, the flows are considered to go out of the system boundaries before coming back. Effectively, this limits the case study to once-through cycles for each identified trees.

Several flows are indicated in the system using *unused* links (dashed lines). These flows are not relevant to any failure propagation. However, they can be important in regard to the PHM modeling of the system by giving precious information on the system health by monitoring a priori uninteresting flows. One example of such flows would be, in our study, the acoustic energy.

Some flows can also be considered to have multiple directions. This is the case of the equilibrium reaction resulting in thermal exchange. Heat is added to one function but subtracted from another (cooling), displaying, in effect, a bidirectional flow. A thermal flow is considered unidirectional if the effect of one direction can be neglected. For example, the heat sink is considered large enough that the heat gained from cooling the secondary circuit does not impact the temperature of the heat sink significantly.

4.2. Critical nodes

The critical nodes, used to compare the various PHM hardware selection and position combinations, can be identified as the main function or flow of interest of our system. In the considered case study (Figure 10), this corresponds to two different functions or flows, the risk-critical one, and the reliability-critical one, which are different. The reliability-critical point selected is the electricity generation function, exhibiting a little gray square. The risk-critical point is chosen to be the vessel function, marked with a little red square.

4.3. Spanning trees

The trees representing the system as a whole, from a risk standpoint as well as from a reliability standpoint, are computed. The entry nodes set is obtained, composed of *Signal - Control* (flow to the Core); *Convert - Convert* (Pressurizer); *Provision - Supply* (Heat sink); and *Material - Liquid* (flow to the backup generators).

From each of these functions, the trees needed to reach the

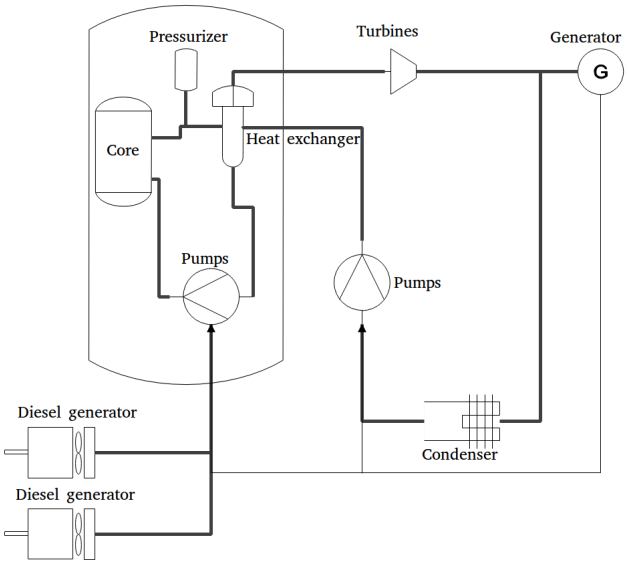

Figure 9. Case study - Simplified P&ID of a Nuclear Power Plant.

risk-critical point and the reliability-critical point are computed. The different trees computed for this example are given in the appendix, in Figures 11 and 12. Two trees can represent the whole system. Indeed, the trees emanating from the *Convert - Convert* (Pressurizer), from the *Material - Liquid* (flow to the backup generators), the risk-critical tree starting from *Signal - Control* (flow to the Core) and the reliability-critical tree starting from *Provision - Supply* (Heat sink) are subtrees of the ones displayed.

4.4. PHM hardware inventory

Several categories of PHM equipment have been considered in this study. Table 12 presents an excerpt of **P** used in this particular study. Note again that the PHM harware and their displayed values are fictional. They are only used for illustrative purposes. As explained in section 3.6, function weakness nodes are a binary event, while flow weakness nodes are s-events, pointing to the fact that a PHM hardware associated to a function is defined to only have a relevant failure detection efficiency. The function and flow name set can be of various degrees of specification, according to the taxonomy introduced by Stone and Wood (2000). If a PHM hardware is entered in the database with relation to Gas (Material-Gas flow) only, a flow defined as Material in the logical functional model cannot use that sensor. If, on the contrary, a PHM hardware is linked to Material, it can be used with any Material flow defined in the system (e.g liquid, gas, solid, ...) with the same efficiency and false alarm parameters. It is important to note that this is a factor in computation time reduction. The more precise the database **P**, the more adequate and succinct the PHM selection and position combinations analyzed will be.

	Function or Flow	Efficiency			False alarm
		Failure	Concern		
			low	high	
PHM_1	Convert	0.95	-	-	0.01
PHM_2	Energy - Thermal	0.999	0.90	0.98	0.1
PHM_3	Control Magnitude	0.995	-	-	0.02
	Branch	0.995	-	-	0.02
	Channel	0.995	-	-	0.02
	Material	0.5	0.4	0.45	0.2

Table 12. Inventory of the PHM hardware

Function or external Flow (Deepest level)	Emergent weakness probability (per year or per use)		
	Failure	Concern	
		low	high
Divide	1×10^{-5}	-	-
Extract	1×10^{-5}	-	-
Remove	1×10^{-5}	-	-
Separate	3×10^{-5}	-	-
Distribute	1×10^{-5}	-	-
...			
Thermal	1×10^{-5}	1×10^{-3}	1×10^{-4}

Table 13. Emergent weakness - **W**

4.5. Population of the databases

Tables 12, 13, 14 and 15 display the data considered to analyze the presented case study. For illustrative purposes only, this subsection goes over an example of a potential algorithm to populate the database in each case.

For **P** (Table 12), three pieces of hardware are considered available. The hardware PHM_2 is shown. It represents a resistance temperature detector. The manufacturer data, in correlation with the system desired nominal flow, can be used to compute its efficiency at detecting flows of various "qualities". The likelihood of detecting a low concern flow quality is taken as 90%. An efficiency of 98% is considered for high concern flow quality. Finally, an efficiency of 99.9% is obtained for the hardware to detect a failed flow. The false alarm rate is taken as being 0.1%.

W can be populated using several sources of information. An excerpt of the database used within the scope of the case study is presented in Table 13.

Once a weakness is detected for a specific flow or function, a corrective action can be undertaken to restore the system health. This corrective action is considered successful if the flow or function weakness is restored, either by fixing it directly or by acting on neighboring functions or flows. HRA can, for example, be used to estimate the probability of a successful correction following a detected weakness. Take, for example, the case of a weakness detection in one of the diesel backup generators (Convert - Convert function). For this component, one could derive that the maintenance team

System Function or Flow [ID]	Correction success			Mishandling
	Failure	Concern		
		low	high	
Provision - Store - Contain [Vessel]	0.75	-	-	1×10^{-2}
Provision - Store [Core]	0.75	-	-	1×10^{-2}
Provision - Store - Contain [Primary]	0.75	-	-	1×10^{-2}
Convert - Convert [Pressurizer]	0.75	-	-	1×10^{-2}
Convert - Convert [SG]	0.75	-	-	1×10^{-2}
...				
Energy - Thermal [Core-Primary]	0.75	0.95	0.85	1×10^{-2}
Energy - Hydraulic [Pressurizer-Primary]	0.75	0.95	0.85	1×10^{-2}

Table 14. Correction success - **H**

Function	Flow	Failure	Concern		Nominal
			low	high	
Provision - Store - Contain	Energy - Thermal	0.6	0.1	0.3	0.
	Material - Liquid	0.7	0.1	0.2	0.
	Material - Gas	0.6	0.1	0.3	0.

Table 15. Function failure link database architecture - **F**

has little spare time to fix the problem (SPAR-H PSF multiplier 10) and works under high stress (SPAR-H PSF multiplier 2). The task is not difficult (SPAR-H PSF multiplier 1) and the team is highly trained (SPAR-H PSF multiplier 0.5), but the procedure is lacking (SPAR-H PSF multiplier 5) and the system ergonomy is not adequate (SPAR-H PSF multiplier 10). Using Eq. (1), a probability of successful repair of 66.6% is derived for this particular weakness.

Some questions, such as the system ergonomy or the team training, are unknown to the designer during the early phase of design. This is covered by HRA methodology using a *not enough information* category. Similar existing systems could be used as reference by the design team to obtain meaningful probabilities. The data can be refined when more precise information is obtained.

Linking a function failure to its impact on outgoing flows can also be challenging. A potential method is exhibited on the impact of the failure of a *Provision - Store - Contain* function on outgoing flows. In this case, the outgoing flow of interest is considered to be thermal energy. The failure of the function could be translated to a small, intermediate or large leak, which would in turn impact the outgoing flow quality in different ways. Probabilities of each event can be attributed based on historical data and engineering deduction. Efficiently and automatically populating **F** could warrant additional research efforts.

The case study presented does not account for the presence of the m-database **M**, considering instead a management trusting blindly in the PHM hardware data analysis and moving away from any form of non-condition based maintenance on PHM-equipped functions and flows.

4.6. Results

A reference case is computed by considering no PHM hardware anywhere in the system, which is the nominal case in early design. This reference will be used to estimate the gain from the possible use of PHM systems throughout the case study. The trees representing the whole system interactions, reliability-centered and risk-centered, are computed. For each tree, the optimized PHM sensor configuration is obtained. The combined optimized configuration can then be defined. For the present case study, one considers the risk-critical tree optimized PHM configuration to be dominant. The reliability-critical point failure probability will thus be dependent on the PHM configuration obtained for the risk analysis. In most systems with limited redundancies, such as the one considered in the case study, non-representative of a real nuclear power station design, PHM can often only diminish the immediate system reliability due to required offline maintenance operations.

In order to estimate the gain from the possible use of PHM systems, failure propagation probabilities within the "bare" functional model design are computed. The failure probability for both the risk-critical tree and the reliability-critical tree are calculated. Given the defined database, the Bayesian network identifies the probability of the risk-critical point failure to be $1.37 \times 10^{-2}\ y^{-1}$. The probability of the reliability-critical point failure is $3.65 \times 10^{-2}\ y^{-1}$.

The optimized PHM hardware selection and positioning is obtained for the risk-critical tree. The configuration is given in Figure 10. This configuration is also considered for the reliability study, as in our case, the most information can be obtained from the risk analysis.

Using PHM sensors through the systems and the given corrective actions from the database, the probability of failure of the risk-critical point is reduced by 75%, from $1.37 \times 10^{-2}\ y^{-1}$ to $3.35 \times 10^{-3}\ y^{-1}$. Simultaneously, the probability of failure of the reliability-critical point is increased by 70%, from $3.65 \times 10^{-2}\ y^{-1}$ to $6.24 \times 10^{-2}\ y^{-1}$.

5. DISCUSSIONS AND FUTURE WORK

The PHASED methodology proposed in this paper, applied to the considered case study, shows the potential benefit from considering PHM in the early phase of design. It notably gives a more realistic analysis of the failure probabilities in the designed system and helps designers select adequate sensors and associated function or flow to monitor. In the present

case study, it can be surmised that the reliability can be improved by increasing the number of redundancies. Without such redundancies, the system can be made safer using PHM equipments at the expense of the reliability.

The PHASED methodology could be improved by more efficiently identifying the system weaknesses and guiding the design team toward potential solutions and to resolve weak points. In the current state of the proposed methodology, the design team must identify manually the function or flows to better monitor them or make them redundant in order to lower the overall probability of failure. It is also dependent on data that might prove difficult to obtain with high confidence, such as a detector efficiency when monitoring a generic function or flow, covering a varying range of parameters (flow velocity, component size, etc.). However, this is a common and widely known issue in risk engineering.

Bayesian networks may also require the use of a significant amount of data to derive conditional probability tables, especially when the number of parent nodes increases. This directly impacts the computer resources needed for the calculations and can be a limiting factor in the immediate industrial application of PHASED.

It is interesting to note that the probability of failure of the reliability-critical point can increase with the use of PHM hardware to monitor functions or flows, if those are essential to the reliability-critical point. This is especially true for non-redundant functions or flows that cannot be repaired online. Indeed, for such functions or flows, no repair actions can be attempted without shutting down the system. The rate of reliability failure would be the sum of the failure probability seen with no sensor and of the probability of a false alarm, both cases causing a reliability issue. However, if the false alarm rate is not too high, by itself this does not indicate a detrimental aspect of the PHM sensors use to the reliability in the long term. The outage time and cost to repair a detected weakness would be beneficial in the long term, extending the component's lifetime when compared to the outage time and cost incurred by a sudden unexpected failure. An analysis of this assumption has not been performed within the frame of this paper.

An interesting concept from Bayesian networks can be discussed. The PHASED method proposed in this paper could eventually be able to replace the real world by simulating the system. *Observed evidence* can be used to compute the failure propagation within the functional Bayesian network. The observed evidence encodes some simulated knowledge of the system into the model and observes the consequences on the final probabilistic states of each node, including parents' nodes. In this regard, the algorithm developed for the PHASED methodology could eventually also be used as an online diagnostic tool, depending on the propagation direction contemplated, following the flows or not. Consider that a functional model is constructed representing the real operating final design. If a failure is observed in the real world, the information can be coded into the simulation method proposed in this paper. Given this observed evidence, the probabilities of every node throughout the model update to account for it. The likely cause of the failure could then be identified more easily. It can provide the engineering team with useful information about the likely underlying cause. This aspect of the methodology will be the subject of future work.

Future work could include the use of a continuous time Bayesian network instead of the static Bayesian network presented. This would allow natural feedback loops within the designed system, eliminating the need for several spanning trees.

A limitation of one PHM hardware monitoring a function or flow has been applied in this framework. Sensor fusion can be integrated to the algorithm to remove this limitation and allow for lowering the false alarm rate while improving the detection efficiency.

More efficient PHM hardware selection and position algorithm could be devised in order to improve computational time performances, and hardware costs could also be considered when building the available inventory.

Adding PHM hardware to a system introduces a new source of failure and uncoupled flows through the system. Uncoupled Failure Flow State Reasoning (UFFSR) defines a methodology to account for the uncoupled flows within the scope of a functional model (O'Halloran, Papakonstantinou, & Van Bossuyt, 2015; Ramp & Van Bossuyt, 2014). The consideration of such non-nominal failure propagation is of importance in complex systems. The merging of the proposed method with UFFSR would be beneficial to the risk and reliability analysis of a complex system. This represents a potential future endeavour, though computational RAM might be a limiting factor for widespread use.

Finally, the development of official, complete functional databases could be undertaken to facilitate the use of the proposed framework. Uncertainties on the data given in the various databases, and their impact on the system risk and reliability, might also be considered.

6. CONCLUSION

The PHASED methodology and the associated automatic framework presented in this paper have been shown to perform adequately in the given objectives. An example of the potential use of the framework has been introduced using the case study of a simplified nuclear power plant, demonstrating its capabilities.

It was shown that the modeling of PHM hardware during the early design phase can give a more realistic view of the risk and reliability failure probabilities of the system. The

PHASED methodology provides the engineering design team with adaptable risk and reliability analysis, allowing them to make better informed decisions about the design in the early phase. The methodology can be used to reduce the cost of a system by replacing expensive redundancies (upfront cost, preventive maintenance) with PHM monitoring while upholding the system's failure probability.

Existing risk and reliability methods all encounter limitations when trying to consider PHM systems during the early phase of design. The work presented in this paper offers a viable solution to this problem within an automated framework. It also offers the potential to be the basis for a complete integrated framework for prognostics and health management oriented design and for online diagnostics simulation.

ACKNOWLEDGMENT

This work was partially supported by United States Nuclear Regulatory Commission Grant NRC-HQ-84-15-G-0016. Any opinions or findings of this work are the responsibility of the authors, and do not necessarily reflect the views of the sponsors or collaborators.

NOMENCLATURE

P	PHM hardware database
ε	PHM hardware efficiency
e	PHM hardware false alarm rate
M	Management decision database
γ	Management decision to ignore PHM data
μ	Non-condition-based scheduled maintenance
H	Corrective action database
ρ	Maintenance success rate
β	Likelihood of mishandling during unnecessary maintenance
F	Link between function and flow failure database
λ	Function to flow failure propagation probability
W	Weakness database
ω	Emergent weakness probability

ACRONYMS

$ATHEANA$	A Technique for Human Event Analysis
$CTBN$	Continuous Time Bayesian Network
$FBED$	Functional Basis for Engineering Design
$FFBD$	Functional Flows Block Diagram
$FFDM$	Function Failure Design Method
$FFIP$	Function Failure Identification and Propagation
$FMEA$	Failure Modes and Effects Analysis
$FMECA$	Failure Modes, Effects, and Criticality Analysis
FTA	Fault Tree Analysis
FSL	Function State Logic
HBN	Hybrid Bayesian Network
HEP	Human Error Probability
HRA	Human Reliability Analysis
$PHASED$	Prognostics and Health Analysis to Support Engineering Design
PHM	Prognostics and Health Management
$P\&ID$	Pipe and Instrumentation Diagram
PRA	Probabilistic Risk Assessment
PSF	Performance Shaping Factors
$PSVCC$	Prognostic System Variable Configuration Comparison
PWR	Pressurized Water Reactor
RBD	Reliability Block Diagram
RED	Risk in Early Design
$SPAR\text{-}H$	Standardized Plant Analysis Risk-Human
STA	Success Tree Analysis
$THERP$	Technique for Human Error-Rate Prediction
$UFFSR$	Uncoupled Flow Failure State Reasoning
$YAML$	YAML Ain't Markup Language

REFERENCES

Agarwal, V., Lybeck, N., Pham, B. T., Rusaw, R., & Bickford, R. (2015). Prognostic and health management of active assets in nuclear power plants. *International Journal of Prognostics and Health Management*.

Andrews, J. D., & Dunnett, S. J. (2000). Event-tree analysis using binary decision diagrams. *IEEE Transactions on Reliability*, *49*(2), 230–238.

Ankan, A., & Panda, A. (2015). *Mastering probabilistic graphical models using python*. Packt Publishing Ltd.

Bartram, G., & Mahadevan, S. (2015). Probabilistic prognosis with dynamic bayesian networks. *International Journal of Prognostics and Health Management*.

Boring, R. L., & Blackman, H. S. (2007). The origins of the SPAR-H methods performance shaping factor multipliers. In *2007 IEEE 8th human factors and power plants and HPRCT 13th annual meeting* (pp. 177–184). Monterey, CA.

Boudali, H., & Dugan, J. (2006, mar). A Continuous-Time Bayesian Network Reliability Modeling, and Analysis Framework. *IEEE Transactions on Reliability*, *55*(1), 86–97. doi: 10.1109/TR.2005.859228

Chang, A. S. (2002). Reasons for Cost and Schedule Increase for Engineering Design Projects. *Journal of Management in Engineering*, *18*(1). doi: 10.1061/(ASCE)0742-597X(2002)18:1(29)

Choo, B. Y., Adams, S. C., Weiss, B. A., Marvel, J. A., & Beling, P. A. (2016). Adaptive multi-scale prognostics and health management for smart manufacturing systems. *International Journal of Prognostics and Health Management*.

Coble, J., Ramuhalli, P., Bond, L., Hines, J. W., & Upadhyaya, B. (2015). A review of prognostics and health management applications in nuclear power plants. *International Journal of Prognostics and Health Management*.

Conroy, P., Stecki, J., & Thorn, A. (2016). Influence of Environmental Loading Factors on System Design. In *European Conference of the Prognostics and Health Management Society 2016.* Bilbao, Spain.

Cooper, S. E., Ramey-Smith, A. M., Wreathall, J., Parry, G. W., Bley, D. C., Luckas, W. J., ... Barriere, M. T. (1996). *A technique for human error analysis (ATHEANA)* (Tech. Rep.). U.S Nuclear Regulatory Commission, NUREG/CR-6350.

Doguc, O., & Ramirez-Marquez, J. E. (2009). A generic method for estimating system reliability using bayesian networks. *Reliability Engineering & System Safety*, *94*(2), 542–550.

Eisenbart, B., Blessing, L., & Gericke, K. (2012). Functional Modelling Perspectives Across Disciplines : a Literature Review. In *International Design Conference - Design 2012.* Dubrovnik, Croatia.

Elattar, H., Elminir, H., & Riad, A. (2016). Prognostics: A literature review. *Complex Intell. Syst.*, *2*, 125–154. doi: 10.1007/s40747-016-0019-3

Ericson, C. (1999). Fault tree analysis - a history. In *Proceedings of the 17th International Systems Safety Conference.* Orlando, FL.

Gertman, D., Blackman, H., Marble, J., Byers, J., & Smith, C. (2005). The SPAR-H human reliability analysis method. *US Nuclear Regulatory Commission, NUREG/CR-6883*.

Gopalratnam, K., Kautz, H., & Weld, D. S. (2005). Extending continuous time bayesian networks. In *Proceedings of the 20th national conference on Artificial intelligence* (Vol. 2, p. 981). Pittsburgh, PA.

Jensen, D., Tumer, I. Y., & Kurtoglu, T. (2009). Flow state logic (FSL) for analysis of failure propagation in early design. In *ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (Vol. 8). San Diego, CA.

Kacprzynski, G. J., Roemer, M. J., & Hess, A. J. (2002). Health Management System Design: Development Simulation and Cost/Benefit Optimization. In *2002 Aerospace Conference Proceedings* (Vol. 6, pp. 3065–3072). Big Sky, MT.

Kurtoglu, T., & Tumer, I. Y. (2008). A Graph-Based Fault Identification and Propagation Framework for Functional Design of Complex Systems. *Journal of Mechanical Design*, *130*. doi: 10.1115/1.2885181

Langseth, H., & Portinale, L. (2007). Bayesian networks in reliability. *Reliability Engineering and System Safety*, *92*, 92–108.

L'Her, G. (2016). *PHASED.* `https://github.com/glher/PHASED`. GitHub.

Lin, Y., Zakwan, S., & Jennions, I. (2017). A bayesian approach to fault identification in the presence of multi-component degradation. *International Journal of Prognostics and Health Management*.

Liu, H.-C., Liu, L., & Liu, N. (2013). Risk evaluation approaches in failure mode and effects analysis: A literature review. *Expert Systems With Applications*, *40*, 828–838.

López, A. J. G., Márquez, A. C., Fernández, J. F. G., & Bolaños, A. G. (2014). Towards the Industrial Application of PHM: Challenges and Methodological Approach. In *European Conference of the Prognostics and Health Management Society 2014.* Nantes, France.

Lough, K. G., Stone, R., & Tumer, I. Y. (2009). The risk in early design method. *Journal of Engineering Design*, *20*(2), 155–173. doi: 10.1080/09544820701684271

Neil, M., & Marquez, D. (2012). Availability modelling of repairable systems using bayesian networks. *Engineering Applications of Artificial Intelligence*, *25*(4), 698–704.

Nodelman, U., Shelton, C. R., & Koller, D. (2002). Continuous time bayesian networks. In *Proceedings of the Eighteenth conference on Uncertainty in Artificial Intelligence* (pp. 378–387). Alberta, Canada.

O'Halloran, B. M., Papakonstantinou, N., & Van Bossuyt, D. L. (2015). Modeling of function failure propagation across uncoupled systems. *2015 Annual Reliability and Maintainability Symposium*. doi: 10.1109/RAMS.2015.7105107

Pearl, J. (1985). Bayesian networks: A model of self-activated memory for evidential reasoning. In *Seventh Annual Conference of the Cognitive Science Society* (pp. 329–334).

Perreault, L., Thornton, M., Strasser, S., & Sheppard, J. W. (2015). Deriving prognostic continuous time Bayesian networks from D-matrices. In *Proceedings of IEEE AUTOTESTCON, 2015* (pp. 152–161). National Harbor, MD.

Ramp, I. J., & Van Bossuyt, D. L. (2014). Toward an automated model-based geometric method of representing function failure propagation across uncoupled systems. In *ASME 2014 International Mechanical Engineering Congress and Exposition.* Montreal, Canada.

Sankavaram, C., Kodali, A., Pattipati, K., Singh, S., Zhang, Y., & Salman, M. (2016). An inference-based prognostic framework for health management of automotive systems. *International Journal of Prognostics and Health Management*.

Smith, C., Knudsen, J., Calley, M., Beck, S., Kvarfordt, K., & Wood, T. (2005). SAPHIRE basics - An Introduction to Probabilistic Risk Assessment via the Systems Analysis Program for Hands-On Integrated Reliability Evaluations (SAPHIRE) Software. *Idaho National Laboratory, Idaho Falls, ID*.

Spurgin, A., & Lydell, B. (2002). Critique of current human reliability analysis methods. In *Proceedings of the 2002 IEEE 7th Conference on Human Factors and Power Plants* (pp. 3–12). Scottsdale, AZ.

Stack, C., & Van Bossuyt, D. L. (2015). Toward a Functional Failure Modeling Method of Representing Prognostic Systems During the Early Phases of Design. *Proceedings of the ASME 2015 International Design Engineering Technical Conference & Computers and Information in Engineering Conference*(August 2015), DETC2015–46400.

Stone, R. B., Tumer, I. Y., & Wie, M. V. (2005). FFDM: The function failure design method . *Journal of Mechanical Design*. doi: 10.1115/1.1862678

Stone, R. B., & Wood, K. L. (2000). Development of a Functional Basis for Design. *Journal of Mechanical Design*, *122*, 359–370.

Sun, B., Zeng, S., Kang, R., & Pecht, M. (2012). Benefits and Challenges of System Prognostics. *IEEE Transactions on Reliability*, *61*(2).

Swain, A. D., & Guttmann, H. E. (1983). *Handbook of human-reliability analysis with emphasis on nuclear power plant applications. final report* (Tech. Rep.). Sandia National Labs., Albuquerque, NM (USA).

Torres-Toledano, J. G., & Sucar, L. E. (1998). Bayesian networks for reliability analysis of complex systems. In *Proceedings of the 6th Ibero-American Conference on Artificial Intelligence* (pp. 195–206). Lisbon, Portugal.

U.S. Department of Defense. (1949). *Procedures for performing a failure mode effect and critical analysis* (Tech. Rep. No. MIL-P-1629).

U.S. NRC. (2016). *Probabilistic risk assessment* (Tech. Rep. No. ML032200337). U.S. Nuclear Regulatory Commission.

Weber, P., Medina-Oliva, G., & Simon, C. (2012). Overview on Bayesian networks applications for dependability, risk analysis and maintenance areas. *Engineering Applications of Artificial Intelligence*, *25*, 671–682. doi: 10.1016/j.engappai.2010.06.002

Xiao, W. (2016). A probabilistic machine learning approach to detect industrial plant faults. *International Journal of Prognostics and Health Management*.

BIOGRAPHIES

Guillaume L'Her was born in France in 1988. He earned a Bachelor of Science in Physics from the University of Paris XI at Orsay in 2010 and a Master of Science in Nuclear Engineering from the French National Institute of Nuclear Science and Technology in 2012. He then spent four years as a Nuclear Reactor Core Engineer with EDF. He is now pursuing a PhD at Colorado School of Mines in the USA. His research interests comprise the development of risk and reliability analysis for complex systems such as nuclear reactor design and the development of fast nuclear reactor designs.

Douglas L. Van Bossuyt, Ph.D. is a partner at KTM Research, LLC in Tualatin, Oregon. KTM Research specializes in machine vision for manufacturing systems and vision-guided robotics, and has machines deployed in manufacturing facilities throughout North America and Asia. His research focuses on the intersection of design, system modeling, and risk analysis. Dr. Van Bossuyt received his PhD from the Complex Engineered Systems Design Laboratory at Oregon State University in 2012.

Bryan O'Halloran, Ph.D. is currently an Assistant Professor in the Systems Engineering department at the Naval Postgraduate School. Previously he was a Senior Reliability and Systems Safety Engineer at Raytheon Missile Systems and the Lead Reliability and Safety Engineer for hypersonic missile programs. He holds a Bachelor of Science degree in Engineering Physics and a Master of Science and Doctorate of Philosophy in Mechanical Engineering from Oregon State University. His current research interests include risk, reliability, safety, and failure modeling in the early design of complex cyber physical systems.

APPENDIX

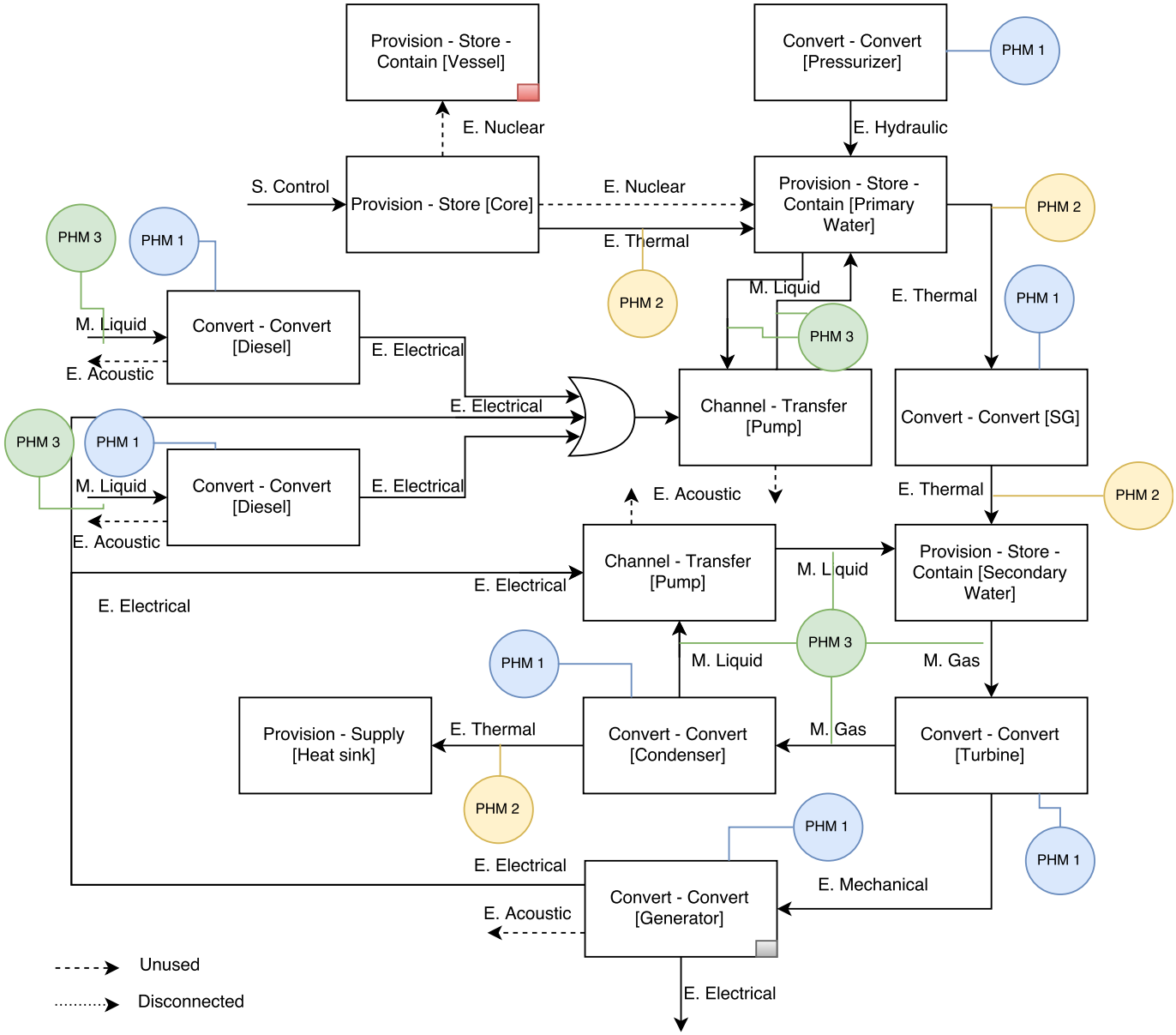

Figure 10. Case study - Simplified logical functional model of a Nuclear Power Plant - Optimized positions of PHM sensors

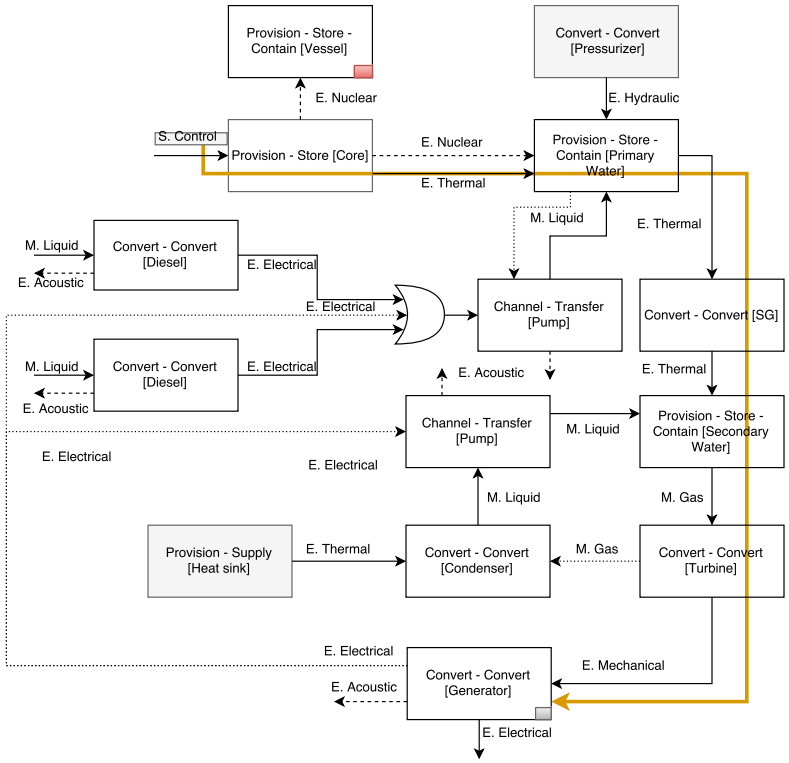

Figure 11. Case study - Spanning tree - reliability-critical node.

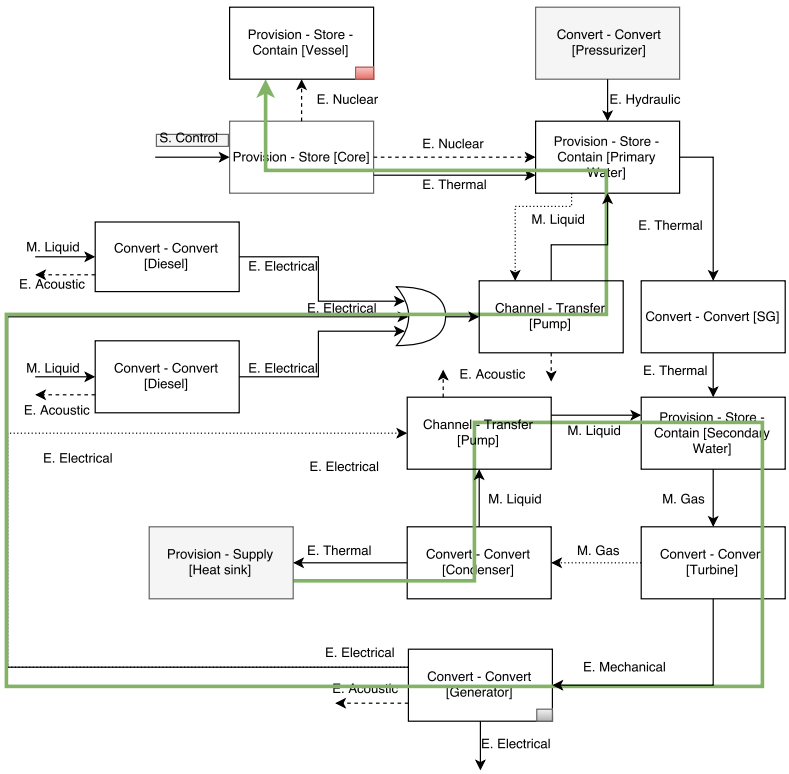

Figure 12. Case study - Spanning tree - risk-critical node.