Functional Prognostic Architecture for the Maintenance of Complex Systems

Pauline Ribot [*] **Yannick Pencolé** [**] **Michel Combacau** [*]

[*] *University of Toulouse; LAAS; Toulouse, France*
[**] *CNRS; LAAS; Toulouse, France*
{pribot,ypencole,combacau}@laas.fr

Abstract: This paper addresses the problem of the maintenance of a complex system of heterogeneous components. With the decreasing costs of sensors, it now becomes possible to really benefit of on-line adaptive prognostic methods in order to support the optimization of the maintenance actions and scheduling. In this paper, we propose a generic health monitoring architecture that encompasses the available prognostic methods and provides a common support for the maintenance decision on a complex system. The output of this architecture takes into account the composed nature of the system by not only providing component prognoses but also function prognosis. In order to provide a prognosis for the whole system, component prognoses are combined by taking into account functional dependencies in the system.

Keywords: Complex systems, dynamic degradation, maintenance engineering, probabilistic models, supervision.

1. INTRODUCTION

Nowadays, the maintenance efficiency of systems is an important economical and commercial issue. Maintaining a system basically consists in replacing components that are unable to perform their functions by new ones. In order to optimize the maintenance cost, it is necessary to perform preventive maintenance to prevent failure occurrence. Preventive maintenance actions consist in replacing components during a scheduled maintenance phase that are not faulty yet but that will probably become faulty before the date of the next scheduled maintenance phase. In the classical case, preventive maintenance is only based on reliability analyses that do not take into account the stress factors really influencing each component of the system during its own life. Among stress factors are included the system operational conditions, faults and their consequences that can occur on a component or on any components in its neighborhood. In order to take into account real stress factors that occur on a component it is required to perform dynamic preventive maintenance (also called predictive maintenance). This maintenance must be decided relying on an efficient and complete analysis of the health of the system when it is operating, which requires a diagnostic and prognostic analysis (Byington et al. (2004)). With the help of new technologies, it is now possible to embed within the system an on-line supervision architecture that is in charge of monitoring the system, detecting any problems, faults which require some maintenance actions and performing prognostics.

Prognostics is the capability to predict the *remaining useful life* (RUL for short) of systems in service: the RUL is defined as the remaining time until the system does not successfully fulfill its goal anymore and has to be repaired (Engel et al. (2000)). By performing a prognostic reasoning over the system, the maintenance architecture establishes whether a preventive action is pertinent at a given time. It is possible to provide an adaptive prognosis by taking into account real stress factors that occur on the system that will modify its RUL. Model-based prognostics is a technically comprehensive modelling approach often used for system failure prognostics (Valentin et al. (2003); Kirkland et al. (2004); Vachtsevanos et al. (2006); Muller et al. (2008)). Such a method relies on a continuous physics model of the system degradation and provides an accurate prognosis by identifying the possible stress causes. However, these methods consider the system as monolithic and do not handle the composed and heterogeneous nature of integrated systems.

In this paper, prognostics are performed in order to support preventive maintenance decisions for aeronautical systems [1] by developing two points:

- *Characterization of Function Prognoses*
  Determining prognosis for a complex system consists in determining the time until the next failure (function loss). Then system prognostics does not only rely on component prognoses but also on function prognoses that rely on functional dependencies in the system. Let us consider the example of a function in a complex system that is implemented by at least two redundant components. If one of the components is not able to perform its function, it may not be replaced if the function is still performed by the other component. From a system view, the function is not lost.
- *Presentation of a generic on-line health maintenance architecture for a complex system.*
  A generic on-line health maintenance architecture for

[1] This work is supported by the research project ARCHISTIC on the maintenance of aeronautical systems in collaboration with AIRBUS and the National Engineering School of Tarbes, FRANCE.

a complex system is presented that takes into account the fact that the system components are heterogeneous, thus several prognostic methods can be used depending on the available models and the available sensors. This architecture provides a common representation as prognosis output for the system functions in order to combine them to provide the prognosis of the complex system. Our generic architecture fits in the *Health Assessment* and the *Prognostics* modules of the OSA/CBM architecture introduced in Lebold and Thurston (2001).

This paper is organized as follows. Section 2 presents the modeling of a complex system and the formalism used in following sections. Section 3 introduces the health maintenance architecture for complex systems and its objective. In Section 4, the prognostic function is characterized. Then the prognosis output is defined for a component and a way to combine component prognoses is presented in order to get a prognosis for the whole system. Finally, Section 5 presents an example that illustrates the prognosis combination.

2. MODELING OF A COMPLEX SYSTEM

An aircraft is composed of several aeronautical systems (braking system, air-conditioning system, ...). Each aeronautical system is a complex system that can be split down into a set of subsystems, any subsystem implementing a function. A function also relies on subfunctions. The prognosis for a complex system relies on functional dependencies. This section introduces the formalism and presents a model of a complex system taking the functional dependencies into account.

2.1 Structural Model

A complex system S is composed of a set of components. A component is a line replaceable unit (LRU for short) which is the smallest device that can be replaced for maintenance purposes. Let $Comp$ denote the set of system components. A component $C_i \in Comp$ implements a set of functionalities noted $FuncT_i$. The set of functionalities implemented by the whole system is denoted $FuncT$, where $FuncT = \bigcup_i FuncT_i$. A functionality $Fu_j \in FuncT_i$ is an elementary function that is implemented by a component C_i and supported by a set of resources gathered in this component C_i. Let $\mathcal{R}_i$ denote the set of resources contained in the component C_i and $\mathcal{R}$ the set of system resources, where $\mathcal{R} = \bigcup_i \mathcal{R}_i$. In this paper, the problem is restricted to the case where a functionality $Fu_j \in FuncT_i$ is supported by only one resource in $\mathcal{R}_i$. A function results from an association of functionalities. The functions are combined to finally perform the system functions. Let $SFunc$ denote the set of system functions and $Func$ be the set of all functions and subfunctions performed by the system, then $SFunc \subseteq Func$ and $FuncT \subseteq Func$.

Formally, the functionalities in $FuncT$ supported by a resource in $\mathcal{R}$ are defined by the mapping F_r on the set of component resources $\mathcal{R}$:

$$\begin{cases} \mathcal{R} \xrightarrow{F_r} P(FuncT) \\ R_i \xrightarrow{F_r} F_r(R_i) = \{Fu_1 \ldots Fu_m\}, \end{cases}$$

where $P(FuncT)$ denotes the power set of $FuncT$. Now, we redefine the system S from a functional view by taking into account the functions it has to realize.

Definition 1. (functional system). A system S is composed of a set $Sys^{(m)}$ of m components $\{C_1, \ldots, C_m\}$ and performs a set of functions $SFunc$:

$$S =< Sys^{(m)},\ \{SFu_1, \ldots, SFu_p\} >,$$

where $Sys^{(m)}$ denotes a subsystem of m components in $Comp$ and $\{SFu_1, \ldots, SFu_p\}$ is the set of the functions provided by the system S, $SFunc = \{SFu_1, \ldots, SFu_p\}$.

A system S has functional requirements and provides high-level functions $\{SFu_1, \ldots, SFu_p\}$ to its environment. In this way, a system can be viewed as a super-component and can also be defined as a set of subsystems. Then the role of the system is to implement higher-level functions. The set of functions $SFunc$ provided by a system S can be defined as an extension F_s of the previously defined mapping F_r. Let $P(Comp)$ denote the power set of $Comp$:

$$\begin{cases} P(Comp) \xrightarrow{F_s} P(Func) \\ S \xrightarrow{F_s} F_s(S) = \{SFu_1, \ldots, SFu_p\} \end{cases}.$$

Figure 1 illustrates all the relationships that have been previously defined.

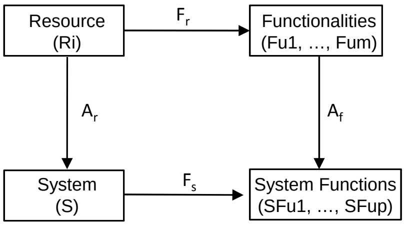

Fig. 1. System, resources and functionalities

The mapping A_f defines the aggregation of the functional dependencies existing between a system function SFu_i performed by a system S and the functionalities implemented by the components of this system:

$$\begin{cases} P(FuncT) \xrightarrow{A_f} SFunc \\ \{Fu_j, \ldots, Fu_k\} \xrightarrow{A_f} A_f(\{Fu_j, \ldots, Fu_k\}) = SFu_i. \end{cases}$$

The functionality set $\{Fu_j, \ldots, Fu_k\}$ is involved in the realization of the system function SFu_i. A mapping A_r between each resource $R_i \in \mathcal{R}$ and the system S can be defined by $S = F_s^{-1}[A_f[F_r(R_j)]] = A_r(R_j)$. The relations between functionalities are described in a functional model of the system by introducing two new mappings $Pred$ and n/m.

2.2 Functional dependencies

A functional model describes the interactions between functionalities in $FuncT$ and illustrates dependencies between functions from which the system functions in $SFunc$ can be performed. The mapping $Pred$ helps to represent these functional dependencies. The notation $Fu_i \in Pred(Fu_j)$ means that Fu_i is directly involved in the realization of Fu_j. Fu_i is called a predecessor of Fu_j.

$$\begin{cases} Func \xrightarrow{Pred} P(Func) \\ Fu_i \xrightarrow{Pred} Pred(Fu_i) = \{Fu_j, \ldots, Fu_k\} \end{cases}$$

This mapping $Pred$ can be used to redefine a functionality and a system function. A function Fu_i in $FuncT \subseteq Func$ is an elementary function, a functionality, if it has no predecessor in the functional representation of the system:

$$Pred(Fu_i) = \emptyset.$$

A functionality does not depend on the realization of another functionality or function. A function is noted SFu_i as a system function iff it is not a predecessor of another function in the system S_i:

$$\forall Fu_j \in Func, \quad SFu_i \notin Pred(Fu_j).$$

The mapping $Pred$ can be graphically represented as a tree structure, called a function tree (Rausand and Hoyland (2004)). The definition of the mapping $Pred$ is not accurate enough if a function has several predecessors, several functions in $Pred(Fu_i)$. The relation between these predecessors is not specified. For example, if $Pred(Fu_1) = \{Fu_2, Fu_3\}$, the relation between Fu_2 and Fu_3 is not expressed and it could be a conjunction (Fu_2 and Fu_3) or a disjunction (Fu_2 or Fu_3). For this purpose another mapping n/m is associated to $Pred$ to clarify relations between functions. The mapping n/m means that at least n functions among the m functions in $Pred(x)$ must work to make the function x work. The n/m mapping is defined by:

$$\begin{cases} P(Func) \xrightarrow{n/m} P\left[P(Func)\right] \\ n/m[(Pred(Fu_t)] = \{X \subseteq Pred(Fu_t) \;\; s.t. \;\; ||X|| = n\} \end{cases}$$

With this notation $n/m[Pred(Fu_t)] = \{X_1, X_2 ... X_k\}$, each X_i is a set of n Fu_j whose interaction is sufficient to make Fu_t work properly and only one X_i is required to ensure the proper work of Fu_t. It must be noticed that the n/m mapping is an extension of the classical AND and OR logical operators. Indeed, it is easy to show that:

- $m/m[Pred(Fu_t)] = \{\{Fu_1 ... Fu_m\}\}$: m Fu_j are required,
- $1/m[Pred(Fu_t)] = \{\{Fu_1\} ... \{Fu_m\}\}$: at least one Fu_j is required.

The sets $Comp$, $Func$ and the mappings F_r, F_s, $Pred$ derive from the knowledge available at the system design.

3. HEALTH MANAGEMENT ARCHITECTURE FOR COMPLEX SYSTEMS

3.1 Prognosis and diagnosis

Our prognosis problem consists in predicting the remaining useful life (RUL for short) of an avionic system. An avionic system is a complex system S composed of a set of heterogeneous components $Comp$ (pieces of hardware, software) that implement a set of functions $Func$ relying on a set of resources $\mathcal{R}$. Available knowledge about each resource $R_i \in \mathcal{R}$ may be very different then several methods for prognostics have to be used. This knowledge is described into ageing models for each resource in $\mathcal{R}$ (Wilkinson et al. (2004)). The ageing of resources is firstly initialized to the mean time to failure ($mttf$ for short) associated to components in which they are contained.

Another challenge facing prognostics for complex systems is to enhance the exploitation of current diagnostic results for a more precise prognosis. When a fault (also called anomaly in Brotherton et al. (2000)) occurs on one of its components, the functioning mode of the system is degraded and its RUL tends to drastically decrease and does not follow the nominal ageing law anymore: this is for instance the case when a crack occurs in a rotor blade of an helicopter (Vachtsevanos et al. (2006)). There is thus a direct link between the prognostic approach and the diagnostic results which is illustrated by Figure 2.

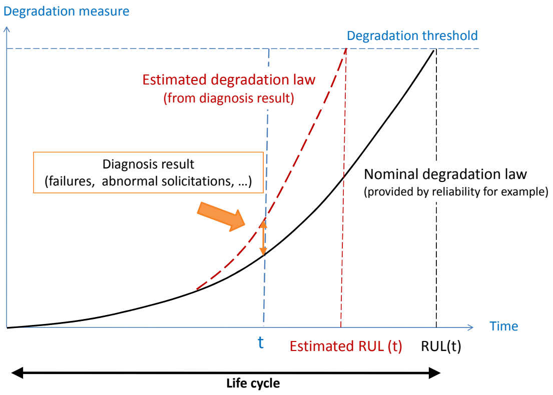

Fig. 2. Prognostics with diagnostic results

Diagnosis helps to estimate the deviations from the nominal ageing law by detecting the fault occurrences at a given time. The link between faults and the age of the component resource must be taken into account in order to provide a precise adaptive prognosis. More details about the diagnosis influence on the RUL can be found in Ribot et al. (2008).

3.2 Health Management and Maintenance Architecture

This section presents a generic maintenance architecture for a complex system that takes into account the heterogeneity of components by finally providing the same prognostic representation for every monitored component in $Comp$. Our generic architecture for the health management and the maintenance of the complex system is presented in Figure 3.

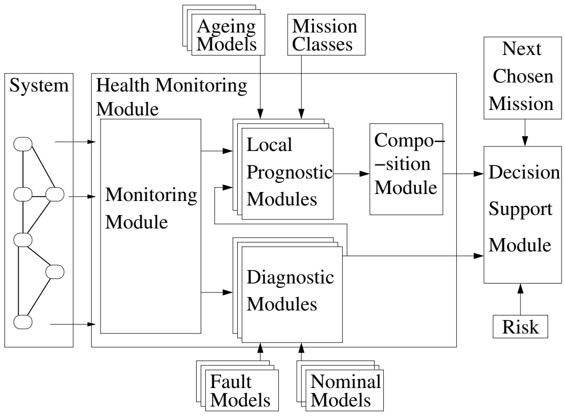

Fig. 3. Health Management and Maintenance Architecture.

This architecture is mainly divided into two modules. The first module is the health monitoring module. It is in charge of monitoring the system S when it is operating. The second module is a decision support module: it is in charge of deciding for maintenance actions relying on the outputs provided by the health monitoring module (Wilkinson et al. (2004)).

The health monitoring module is composed of four types of submodules. The monitoring module contains the sensors and all the communication protocols between sensors and the other modules in order to get the necessary observations for the diagnostic and the prognostic modules.

The diagnostic modules are in charge of performing fault diagnosis on the system S. These modules classically rely on a knowledge about the nominal behavior of the system components in $Comp$ (nominal models usually coming from the specification of the system) and some faulty behaviors (fault models usually coming from a failure mode and effect analysis, FMEA for short).

A local prognostic module is in charge of providing a current prognosis about a component C_i of the system S (see Section 4). It relies on a set of ageing models provided by the component designers that contain the knowledge about the resources in $\mathcal{R}_i$ and a specification of mission classes. To each resource in $\mathcal{R}_i$ corresponds an ageing model. An ageing model may rely on measured stress (physical or estimation models) or not (statistical models). Depending on the nature of the ageing models for resources in $\mathcal{R}_i$ and the type of the modeled component $C_i \in Comp$, the local prognostic module for a component C_i determines the ageing of $\mathcal{R}_i$ in terms of fault probability (in terms of number of cycles in Valentin et al. (2003)). A mission class describes a set of possible missions that the system S can realize: each class describes the expected stress conditions induced by any of these missions on the ageing of $\mathcal{R}_i$ if such a mission is effectively achieved. From these fault probabilities of resources, the local prognostic module provides failure probabilities for each functionalities in $FuncT_i$ implemented by the component C_i. Finally, a composition module provides prognoses of functions in $Func$ and the ageing probability of the whole system S (see Section 4.3).

The decision support module is in charge of providing maintenance recommendations. Recommendations may be to actually perform maintenance actions by replacing faulty components in $Comp$ or components suspected to fail in the future. Deciding whether a non-faulty component C_i has to be replaced or not depends on the failure probability of the system functions in $SFunc$ but also depends on the missions that the system S is supposed to perform before the next maintenance phase and the economical risk if the system fails before the next scheduled maintenance phase (see Ribot et al. (2008)).

4. CHARACTERIZATION OF THE PROGNOSTIC FUNCTION

To identify the output of the prognostic module, a generic prognostic function has to be characterized in order to provide component and function prognoses. The knowledge about each resource in $\mathcal{R}_i$ is contained in ageing models from which the failure probability of functionalities in $FuncT_i$ can be determined at any time.

4.1 Generic representation of prognosis output as Weibull probability density function

Due to its flexibility, the Weibull model can be used to represent the failure probability of functionalities and functions implemented by the system S that rely on resources in $\mathcal{R}$ (Humphrey et al. (2002) Kirkland et al. (2004)). The prognostic function provides Weibull probability density functions (pdf for short) for any functionality or function in $Func$ from which the decision support system determines the *estimated time to failure* (*ettf* for short) of system functions in $SFunc$ and then estimates the RUL of the system S (see Section 3.2). The *ettf* is defined as the remaining time until the next failure (function loss).

The Weibull model is often used in the field of reliability and life data analysis (Vachtsevanos et al. (2006)) to define a pdf. It is a parametrized probability distribution that is able to reproduce the behavior of other statistical distributions such as the exponential distribution and the normal distribution:

$$f(t;\beta,\eta,\gamma) = \frac{\beta}{\eta}\Big(\frac{t-\gamma}{\eta}\Big)^{(\beta-1)} e^{-(\frac{t-\gamma}{\eta})^{\beta}}$$

for $t \geq 0$, $\beta \geq 0$, $\eta \geq 0$ and $\gamma \geq 0$. In our case, $\gamma = 0$ that means failures are assumed to occur from $t = 0$. The parameter values β and η change the shape and the scale of the Weibull pdf. The shape parameter β is determined such that the Weibull model follows the idealized bathtub curve from reliability (Humphrey et al. (2002); Wilkinson et al. (2004)). In the sequel of this paper, the problem is restricted to the useful life period of any resource in $\mathcal{R}$. The Weibull pdf models the useful life period by fixing the value of the shape parameter $\beta = 1$, so that the Weibull model represents an exponential law.

The *ettf* evaluation for any functionality or function in $Func$ consists in determining the time t_p for which the failure probability of the considered function has reached a given threshold Pr:

$$ettf = t_p \quad \text{so that} \int_0^{t_p} f(t;\beta,\eta)dt = Pr,$$

where Pr is a probability threshold deriving from the risk level suitable with the next mission.

4.2 Local Prognostic Module

A local prognostic module for a component C_i aims to determine the *ettf* of each functionality in $FuncT_i$ by providing a Weibull pdf for each functionality implemented by C_i that represents its failure probability density. The *ettf* of the functionality is firstly initialized to the *mttf* of the associated resource. The failure probability of a functionality in $FuncT_i$ which relies on a resource $R_j \in \mathcal{R}_i$ evolves according to the ageing model of the resource R_j and the stress influencing the component C_i by which they are implemented. A stress on a component can be generated by a fault occurrence or an abnormal solicitation from other parts of the system. This stress affects the second parameter η of the Weibull pdf. The value of η modifies the scale of the Weibull pdf from which the *ettf* is evaluated. If the component C_i is highly and frequently used, the value η decreases, then the *ettf* of a functionality $Fu_i \in FuncT_i$ is shortened. If C_i is less stressed than expected, the value η increases and the *ettf* may also get longer.

The parameter η of the Weibull pdf can be determined by a function of the *stress factors* extracted from the set of available ageing models:

$$\eta = f(EF, DR, I, MC),$$

where EF are the environmental factors, DR is the diagnosis result, I represents the interactions between components in $Comp$, and MC is the class of future missions before the next maintenance phase.

Interactions between components in $Comp$ may induce some stress. If a component is abnormally used (or stressed), it can have an impact on other components in its neighborhood and then modify the ageing laws of resources in $\mathcal{R}$ and the failure probability of functionalities in $FuncT$. An extended FMEA could be used to quantify these interactions.

Some future stress can also be estimated by knowing the future missions (for instance energy required for the next missions). The future missions are sorted into mission classes according to the stress (low/medium/high frequency of use for instance) they induce. To each mission class corresponds a stress level, thus a value of η.

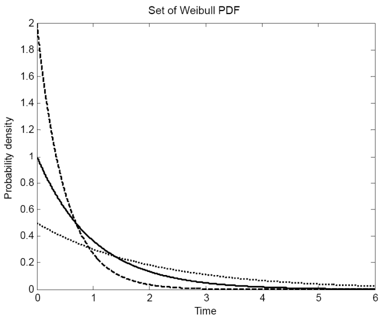

Fig. 4. Weibull probability densities for each mission class

The local prognostic module of a component C_i computes for each functionality in $FuncT_i$ a set of Weibull probability density functions with different values of η for the different mission classes, as shown in Figure 4.

4.3 Composition for System RUL

The concepts introduced by the previous section are about a component C_i and its functionalities in $FuncT_i$. Now, the challenge is to determine the global RUL for a specific system (Goebel and Eklund (2007)). The RUL of a system corresponds to the remaining time until the system could not perform one of its functions anymore. Then the RUL for a system S is evaluated from the *ettf* of its functions in $SFunc$ that are obtained by composing the Weibull pdf of functionalities in $FuncT$ provided by the local prognostic modules.

This composition relies on functional dependencies in the system S. According to the scheme described in Section 3.2, the computation of the RUL for a system must take the future missions into account (stress and risk). As the *ettf* for a functionality is a monotonic increasing and positive function of the parameter η, the reasoning about *ettf* can be directly done on the parameter η. The scheme described in the sequel is valid for a given mission class and must be repeated for each class of mission. For a function Fu_i which requires that all functions in $Pred(Fu_i)$ work properly, the η of the function Fu_i is given by the function in $Pred(Fu_i)$ with a minimal η:

$$\eta(Fu_i) = \underset{Fu_j \in Pred(Fu_i)}{Min} [\eta(Fu_j)].$$

On the contrary, for a function Fu_i which requires that only one function in $Pred(Fu_i)$ works properly, the η is given by the function with a maximal η:

$$\eta(Fu_i) = \underset{Fu_j \in Pred(Fu_i)}{Max} [\eta(Fu_j)].$$

More generally, if the requirements for Fu_i to be normal can be written using the n/m mapping, the η of the function Fu_i can be expressed as follows:

$$\eta(Fu_i) = \underset{X \in n/m[Pred(Fu_i)]}{Max} \left[\underset{Fu_j \in X}{Min} \eta(Fu_j) \right],$$

where $\|X\| = n$ and $\|Pred(Fu_i)\| = m$.

This definition points out that it is not necessary to calculate the *ettf* of each subset of n functions to get the result. It is sufficient to consider the set of the n functions whose η are the greatest and then to take the minimum of their η. This technique allows to compute the η parameter of each intermediate functions and to compose them later till having the *ettf* of the system functions in $SFunc$. The RUL of the system corresponds to the remaining time until the system does not perform successfully one of its functions then the RUL of the system is evaluated as the minimal value of the *ettf* of the system functions $\{SFu_1, \ldots, SFu_p\}$:

$$RUL(S) = min[ettf(SFu_1), \ldots, ettf(SFu_p)].$$

5. ILLUSTRATIVE EXAMPLE

Figure 5 depicts a simple but composed system with seven LRU whose goal is to enlighten. The system is composed of three power sources $PS1$, $PS2$ and $PS3$, two electrical switches $SW1$ and $SW2$, three wires $W1$, $W2$ and $W3$ and a lamp. This system is modelled by
$Comp = \{C_1, C_2, \ldots, C_7\}$,
$\mathcal{R} = \{PS_1, PS_2, PS_3, W_1, W_2, W_3, SW_1, SW_2, Lamp\}$,
$FuncT = \{Fu_1, Fu_2, \ldots, Fu_9\}$(respectively associated to the nine resources),
SFu_1 the system function (lighting).

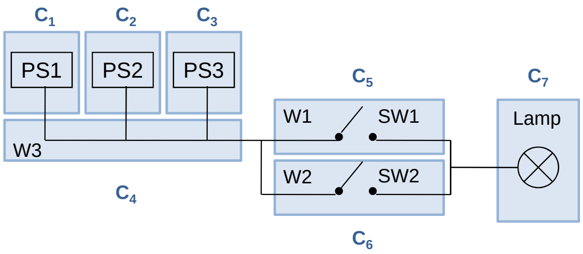

Fig. 5. Example of a lighting System

The system function SFu_1 is realized by the association of functionalities in $FuncT$ as it is represented in Figure 6. The root node is the system function SFu_1, the leaves are the functionalities in $FuncT$ that rely on the system resources in $\mathcal{R}$ and the internal nodes are intermediate functions (subfunctions). The mapping n/m specifies the relation between the functions at each node in the graphical representation.

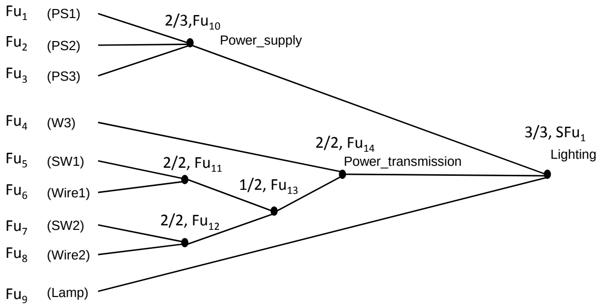

Fig. 6. Functional Dependency Model

This function tree is the graphical representation of the mapping $Pred$.

$Pred(SFu_1) = \{Fu_9, Fu_{10}, Fu_{14}\}$,
$Pred(Fu_{14}) = \{Fu_4, Fu_{13}\}$,
...
$Pred(Fu_1) = \emptyset$.

The functional relations must also be represented using the mapping n/m as follows:

$$SFu_1 = 3/3[Fu_9, 2/3(Fu_1, Fu_2, Fu_3)), \\ 2/2(Fu_4, 1/2(2/2(Fu_5, Fu_6), 2/2(Fu_7, Fu_8)))]$$

Assuming that $\eta(PS2) = \eta(PS3) > \eta(PS1)$ (PS_2 and PS_3 are identical and more robust than PS_1) $\eta(W_i) >> \eta(SW_j)$ (switches are more often faulty than wires) and $\eta(SW_i) > \eta(Lamp)$ (the lamp is weaker than the switches), we can determine that

$$\eta(Power\_supply) = \eta(PS2),$$

$$\eta(Power\_transmission) = Max[\eta(SW1), \eta(SW2)],$$

$$\eta(Lighting) = \\ Min[\eta(PS2), Max(\eta(SW1), \eta(SW2)), \eta(Lamp)].$$

Only two power sources are required to enlighten, then power sources are redundant. In the case where the *ettf* of Fu_2 is long, the replacement of $PS1$ is not necessary at the next maintenance phase, even if the *ettf* of Fu_1 is very short, because the *ettf* of the system function only depends on $PS2$ in this example. The value of $\eta(Lighting)$ characterizes the Weibull pdf for the function $Lighting$ from which the *ettf* of the system function and then the RUL of the system can be determined by the Decision Support System.

6. CONCLUSION

This paper presents a generic framework for the development of a health monitoring architecture that supports the preventive maintenance of complex systems with prognostics. There are several key points for implementing such an architecture.

Firstly it has to take into account the heterogeneity of the components of the system. This heterogeneity implies the use of different types of ageing models, prognostic methods. However, as the prognosis for the whole system does not only rely on component prognoses but also on functional dependencies in the complex system, it is necessary to get the same representation as Weibull pdf for the function prognoses in order to combine them to provide the system prognosis.

Secondly, as ageing models become more and more informative, it is important to use specialized sensors for prognostics in order to measure the stress factors and provide adaptive prognosis. Some stress factors can also be deducted from faulty behaviors which means that diagnostics is necessarily an input for adaptive prognostics and that is why our architecture is strongly related to the diagnosis module.

Finally, maintenance decisions do not only rely on failure probability of functions but also on the expected missions before the next scheduled maintenance phase. Depending on the expected missions, the set of required functionalities may change.

This paper describes a basic solution to compute prognostics and determine the RUL of a complex system. Future works will focus on two aspects. We will generalize our approach by extending it to the case in which functionality relies on several resources and we will take into account the acceptable probability threshold at the system level to compute the RUL of a system by combining the *ettf*.

REFERENCES

T. Brotherton, G. Jahns, J. Jacobs, and D. Wroblewski. Prognosis of faults in gas turbine engines. In *IEEE Aerospace Conference Proceedings*, volume 6, pages 163–171, 2000.

C.S. Byington, P.W. Kalgren, B.K. Dunkin, and B.P. Donovan. Advanced diagnostic/prognostic reasoning and evidence transformation techniques for improved avionics maintenance. In *IEEE Aerospace Conf. Proc.*, volume 5, pages 3424–3434, 2004.

S.J. Engel, B.J. Gilmartin, K. Bongort, and A. Hess. Prognostics, the real issues involved with predicting life remaining. In *IEEE Aerospace Conference*, volume 6, pages 457–469, USA, 2000.

K. Goebel and N. Eklund. Prognostic fusion for uncertainty reduction. In *Proc. of AIAA Infotech@ Aerospace Conf.*, 2007.

D. Humphrey, W. Schawlee, P. Sandborn, and D. Lorenson. Utilization life of electronic systems - aging avionics usable life and wear-out issues. In *presented at World Aviation Congress*, 2002.

L.V. Kirkland, T. Pombo, K. Nelson, and F. Berghout. Avionics health management: Searching for the prognostics grail. In *IEEE Aerospace Conference*, volume 5, pages 3448–3454, 2004.

M. Lebold and M. Thurston. Open standards for condition-based maintenance and prognostic systems. In *Maintenance and Reliability Conference*, 2001.

A. Muller, M-C. Suhner, and B. Iung. Formalisation of a new prognosis model for supporting proactive maintenance implementation on industrial system. *Reliability Engineering and System Safety*, 93:234–253, 2008.

M. Rausand and A. Hoyland. *System reliability theory: models, statistical methods and applications.* Wiley, 2004.

P. Ribot, Y. Pencolé, and M. Combacau. Prognostics for the maintenance of distributed systems. In *the International Conference on Prognostics and Health Management*, Denver, USA, 2008.

G. Vachtsevanos, F. L. Lewis, M. Roemer, A. Hess, and B. Wu. *Intelligent Fault Diagnosis and Prognosis for Engineering Systems.* Wiley, 2006.

R. Valentin, M. Osterman, and B. Newman. Remaining life assessment of aging electronics in avionic applications. In *Annual Reliability and Maintainability Symposium*, pages 313–318, 2003.

C. Wilkinson, D. Humphrey, B. Vermeire, and J. Houston. Prognostic and health management for avionics. *IEEE Aerospace Conf. Proc.*, 5:3435–3446, 2004.