HAL open science

Health Monitoring and Prognosis of Hybrid Systems

Saïd Zabi, Pauline Ribot, Elodie Chanthery

To cite this version:

Saïd Zabi, Pauline Ribot, Elodie Chanthery. Health Monitoring and Prognosis of Hybrid Systems. Annual Conference of the Prognostics and Health Management Society (PHM), Oct 2013, Nouvelle Orléans, United States. 12p. hal-01027538

HAL Id: hal-01027538

https://hal.science/hal-01027538v1

Submitted on 22 Jul 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Health Monitoring and Prognosis of Hybrid Systems

Said Zabi[1], Pauline Ribot[2], and Elodie Chanthery[3]

[1,2,3] *CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France*

[1,2] *Univ de Toulouse, UPS, LAAS, F-31400 Toulouse, France*
said.zabi@laas.fr
pauline.ribot@laas.fr
[3] *Univ de Toulouse, INSA, LAAS, F-31400 Toulouse, France*
elodie.chanthery@laas.fr

ABSTRACT

Maintenance and repair of complex systems are an increasing part of the total cost of final product. Efficient diagnosis and prognosis techniques have to be adopted to detect, isolate and anticipate faults. Moreover the recent industrial systems are naturally hybrid: their dynamic behavior is both continuous and discrete. This paper presents an architecture of health monitoring and prognosis for hybrid systems. By using model and experience-based approach we propose an implementation of an integrated diagnosis/prognosis process based on Weibull probabilistic model. This article focuses particularly on the prognosis algorithm description. The process has been implemented and tested on Matlab. Simulation results on a water tank system show how prognosis and diagnosis interact into the architecture.

1. INTRODUCTION

Due to new technologies, the development of industrial systems is increasingly complex. It becomes difficult to manage and anticipate the behavior of these systems, especially when they are exposed to failures. In time of economic crisis, it is particularly essential to improve the system availability by reducing maintenance and repair costs. Efficient diagnosis and prognosis techniques have to be adopted to detect, isolate and anticipate faults leading to failures. Health monitoring of industrial systems aims at determining the health state of systems at any time in order to optimize their functioning and act in case of malfunctions. Diagnosis helps to determine the current health state of a system. Malfunctions or failures may be anticipated by a prognostic reasoning on the system. No definition of prognosis is really stated in the scientific community. It is more often related to the health state prediction in the future or to the calculation of the remaining useful life (RUL). This temporal prediction gives the date at which the system is not operational anymore and must to be repaired. When the system is in such a state, it is said to be in a *failure mode*. Before this state, it can be either in a *nominal mode*, or after the occurrence of one or several faults in a *faulty mode* or degraded mode. Prognosis requires the knowledge of the current health state of the system through a diagnosis, hence the need of a health monitoring method integrating diagnosis and prognosis.

Recent industrial systems exhibit an increasing complexity of behaviors that are both continuous and discrete. It has become difficult to ignore the fact that most systems are hybrid. Therefore this paper proposes to use the techniques of model-based diagnosis and prognosis in the framework of hybrid systems. We propose to enrich the commonly used modeling framework for hybrid systems with available knowledge about aging or degradation of the system. Systems are continuously degrading according to operational conditions. According to information available on the system, it is possible to establish physical aging laws or time-dependent fault probabilities based on the feedback. This temporal and/or stochastic information should be taken into account in the model of the hybrid system.

The main ideas on the integrated architecture for diagnosis and prognosis of hybrid systems we proposed have been published in (Chanthery & Ribot, 2013). This paper focuses particularly on the prognosis algorithm description. It begins with a brief overview of the related work in diagnosis and prognosis on hybrid systems. Section 3 gives an overview of our architecture for interleaving diagnosis and prognosis processes. The diagnosis process is briefly described. Then section 4 presents the hybrid system modeling for prognosis. Section 5 is the core of the article. It details the prognosis algorithm. The algorithm has been implemented. Simulation results are given on Section 6, illustrated by a water tank

Saïd Zabi et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

system. Finally, Section 7 concludes the paper and proposes some future work.

2. Related Work

There has been considerable work on diagnosis of hybrid systems on one hand, and on prognosis on the other hand. However, to the best of our knowledge, very few studies succeed in coupling diagnosis and prognosis and the authors could not find related work dealing with prognosis on hybrid systems.

A formal generic modeling framework for a complex system is presented in (Ribot, Pencolé, & Combacau, 2009) that encapsulates the knowledge used by diagnosis and prognosis. In this work, the authors establish a coupling of diagnosis and prognosis based on a characterization of complex system modes but no algorithm and implementation have been proposed. Another approach has been proposed in (Roychoudhury & Daigle, 2011). The authors propose a common framework for diagnosis and prognosis thanks to a state representation that describes the nominal behavior of the system and fault progression. However, there is absolutely no hybrid or discrete aspect in this work. The model used is a state model that specifies the system behavior in nominal modes and in faulty modes. A parameter vector and an associated evolution equation are used to represent fault progression over time. The method consists in building an observer from the nominal behavior to perform fault detection. The identification is made from a set of observers that are built for each fault. Prognosis consists in predicting the remaining useful life (RUL) for each fault using an estimator based on a fault progression model.

Most of the works on discrete event systems consider prognosis as a prediction of an event trajectory (Cao, 1989) or fault event occurrences (Genc & Lafortune, 2006). The term "predictability" of a fault event introduced by (Cao, 1989) is based on the system observability property. It is clearly related to the diagnosability notion in discrete event systems: "it is certain that a critical event will take place". (Genc & Lafortune, 2009) demonstrates that any predictable event is diagnosable. An extension for the prediction of event patterns is proposed in (Jéron, Marchand, Genc, & Lafortune, 2008). In these studies, the system model is a classical automaton in which only ordered, undated and without delay event sequences are considered.

To perform prognosis, it is required to take the temporal aspect into account to compute the RUL of the system. Only (Khoumsi, 2009) uses a timed automata (TA) in order to prognose a fault event on the system. Clock ticks are added to transitions of the TA to determine the dated trajectories leading to fault events. No notion of uncertainty (neither by mean of probabilities nor intervals) is taken into account in these timed automata. However, uncertainty is intrinsically linked to prognosis.

In (Zemouri & Faure, 2006), the evolution of the system operating state is modeled by a stochastic timed automaton (STA). A stochastic distribution $f(t)$ is associated to each transition of the automaton. The distribution $f(t)$ gives the probability of occurrence for x_{j+1} at time t_{j+1} after the occurrence of x_j at time t_j. In this study, events occurring in the system are represented in the states of the stochastic timed automaton that does not take the hybrid dynamics of the system into account.

(Castaneda, Aubry, & Brinzei, 2010) proposes a stochastic hybrid automaton to evaluate the system dynamic reliability. The stochastic hybrid automaton represents the possible behavioral modes of the system. The stochastic part helps to take faults and uncertainties about system knowledge into account. The system switches from one mode to another with events that may be deterministic or stochastic. Stochastic events occur when a threshold on their probability law has been reached. In this study, stochastic transitions have a constant rate. The model is simulated to obtain availability and reliability defined as the probabilistic evaluation of the hybrid system failure.

In order to complete previous works, the first issue investigated in this paper is the representation of the behavior of the hybrid system and the uncertainty of its degradation into a single model. The second issue concerns the implementation and the test of a health monitoring method coupling diagnosis and prognosis.

3. Architecture for Interleaving diagnosis and prognosis processes

In this study, a system is supposed to start operating in nominal behavior. A set of critical faults has been anticipated for the system and each one of them may occur at any time from each nominal state. Faults are supposed to be permanent: it means that once a fault has occurred, the system evolves in what is called a *faulty mode*. This degradation can evolve into a worst degraded mode. Finally, when the system is not operational anymore, it is said to be in a *failure mode*. Without maintenance or repair action, the evolution of a system is then supposed to be unidirectional. This evolution of the system from a nominal mode to a failure mode is illustrated in Figure 1.

The combination of faults leading to a failure can be established from a fault tree analysis (Rausand & Hoyland, 2004). With this analysis and the sequence of fault dates predicted by prognosis, it is simple to obtain the system RUL that corresponds to the remaining time until the system failure. This fault analysis allows to link our prognosis definition to the one commonly used in the PHM community (Prognostics and Health Management).

This section begins by describing the architecture interleav-

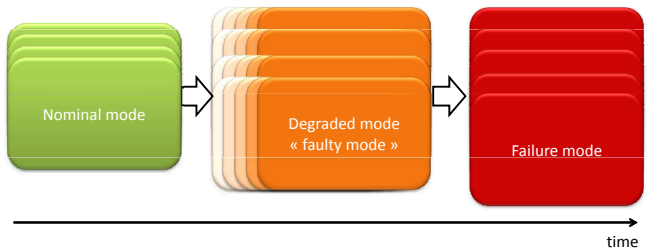


Figure 1. Unidirectional system evolution without maintenance or repair action.

ing diagnosis and prognosis. It will then go on the description of the diagnosis process.

3.1. Overview of the architecture

The originality of our work is to interleave diagnosis and prognosis processes that are too often considered separately. This section presents an architecture coupling diagnosis and prognosis into a single new process called *InterDP*. This architecture, illustrated in Figure 2, has already been described in (Chanthery & Ribot, 2013).

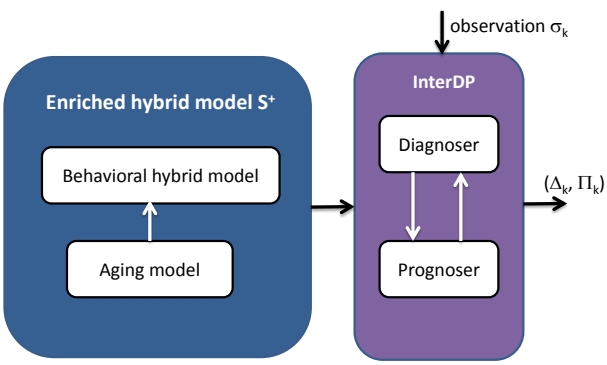


Figure 2. *InterDP* process interactions

Inputs of the *InterDP* process are an enriched hybrid model S^+ defined for our methodology and observations that will be formally defined in the next section. To put it in simple terms, the enriched hybrid model contains all the knowledge about the system behavior and degradation that is required to perform diagnosis and prognosis. Observations are a set of observable events. These events are built from information issued for example from the sensors that are embedded in the system.

The output of InterDP at each clock tick t_k is a couple (Δ_k, Π_k) of diagnosis and prognosis results. Diagnosis Δ_k is performed by a hybrid diagnoser (Bayoudh, Travé-Massuyes, Olive, & Space, 2008). It is built off-line from the hybrid automaton part of the enriched hybrid model. On-line, it takes as input the observations and provides a vector containing all possible modes (nominal, faulty or failure modes) for the system that are consistent with observations:

$$\Delta_k = \begin{bmatrix} \Delta_k^1 \\ \Delta_k^2 \\ \vdots \\ \Delta_k^D \end{bmatrix}$$

where D is the number of diagnosis hypotheses. Δ_k can be seen as a belief state of the hybrid system.

Prognosis Π_k is performed by a prognoser. On-line, at each clock tick t_k, the prognoser takes as input the enriched hybrid model and the result of the diagnosis process and provides a vector:

$$\Pi_k = \begin{bmatrix} \Pi_k^1 \\ \Pi_k^2 \\ \vdots \\ \Pi_k^D \end{bmatrix}$$

where Π_k^i is associated to a diagnosis Δ_k^i and represents the most likely sequence of dated faulty modes leading to the system failure:

$$\Pi_k^i = (\{f_1, d_1\}, \ldots, \{f_j, d_j\}, \ldots, \{f_N, d_N\}).$$

where d_i is the date occurrence of fault f_j and N represents the number of degraded modes before the failure mode.

One hypothesis in our work is that the system is assumed to be diagnosed after each new observation, that is to say when a change in measurements is detected. As diagnosis consists in monitoring the diagnoser, the diagnosis computation duration can be considered as instantaneous. It is also supposed that between two different observations, both diagnosis and prognosis can be performed. Let t_k be an occurrence of an observation, t_{k+1} be the occurrence of the next observation and CT_p be the computation time for prognosis.

Hypothesis 1. *The computation time of the prognosis process is smaller than the interval between two different observations.*

$$CT_p \leq (t_{k+1} - t_k) \tag{1}$$

3.2. Diagnosis process

The diagnosis process in *InterDP* has been described in detail in (Chanthery & Ribot, 2013) and is not the focus point of this paper. We just recall here the main steps of the process. The hybrid diagnoser is built from the hybrid behavioral automaton of S^+ that is formally defined is the next section. On-line it takes as input the set of observations on the system. The diagnosis method for hybrid systems that is adopted for our approach is the one developed in (Bayoudh, 2009), (Bayoudh et al., 2008). It interlinks a standard diagnosis method for continuous systems, namely the parity space method, and a standard diagnosis method for DES, namely the diagnoser method (Sampath, Sengputa, Lafortune, Sinnamohideen, &

Teneketsis, 1995). The diagnosis part of the methodology may be decomposed into three parts:

- diagnose the continuous part of the system,
- abstract the continuous part in terms of discrete events and enrich the discrete part of the system with discrete events that come from the abstraction of the continuous part,
- then apply the diagnoser method on the resulting discrete event system in order to build a diagnoser able to follow on-line the behavior of the system according to the observable events.

4. HYBRID SYSTEM MODELING FOR PROGNOSIS

4.1. Hybrid formalism

The modeling framework that is adopted for hybrid systems is based on a hybrid automaton (Henzinger, 1996). The hybrid automaton is defined as a quintuple $S = (\zeta, Q, \Sigma, T, C, (q_0, \zeta_0))$ where:

- ζ is the set of continuous variables that comprises input variables $u(t) \in R^{n_u}$, state variables $x(t) \in R^{n_x}$, and output variables $y(t) \in R^{n_y}$. The set of directly measured variables is denoted by ζ_{OBS}.
- Q is the set of discrete system states. Each state $q_i \in Q$ represents a behavioral mode of the system. It includes nominal and anticipated faulty modes, including failure modes. The anticipated faulty modes are faulty modes that are known to be possible on the system. The unknown mode can be added to model all the non-anticipated faulty situations.
- Σ is the set of events that correspond to discrete control inputs, autonomous mode changes and fault occurrences. $\Sigma = \Sigma_{uo} \cup \Sigma_o$, where $\Sigma_o \subseteq \Sigma$ is the set of observable events and $\Sigma_{uo} \subseteq \Sigma$ is the set of unobservable events.
- $T \subseteq Q \times \Sigma \rightarrow Q$ is the partial transition function. The transition from mode q_i to mode q_j with associated event σ_{ij} is noted $t(q_i, \sigma_{ij}, q_j)$ and we have $T(q_i, \sigma_{ij}) = q_j$. T also denotes the set of transitions.
- $C = \bigcup_i C_i$ is the set of system constraints linking continuous variables. C_i denotes the set of constraints associated to the mode q_i. C represents the set of differential and algebraic equations modeling the continuous behavior of the system. The continuous behavior in each mode is assumed to be linear.
- $(\zeta_0, q_0) \in \zeta \times Q$, is the initial condition.

The occurrence of a fault is modeled by a discrete event $f_i \in \Sigma_F$. Σ_F is the set of fault events associated to the anticipated faults of F. Without loss of generality it is assumed that $\Sigma_F \subseteq \Sigma_{uo}$. The discrete part of the hybrid automaton is given by $M = (Q, \Sigma, T, q_0)$, which is called the *underlying discrete event system (DES)* and the continuous behavior of the hybrid system is modeled by the so-called *underlying multi-mode system* $\Xi = (\zeta, Q, C, \zeta_0)$. An example of a hybrid system is given in Figure 3.

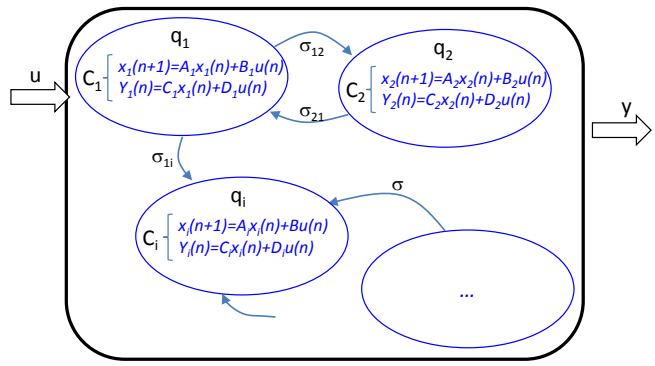


Figure 3. Example of an hybrid system

This hybrid automaton describes the set of knowledge useful to achieve model-based diagnosis. In order to perform prognosis, it is necessary to enrich the hybrid model by adding the available knowledge about the aging or the degradation of the system. A way to take the uncertainty on the degradation function into account is to introduce probability measures for each state that represents a mode of the system.

4.2. Aging modeling

The modeling framework that is adopted for the system degradation is based on the Weibull probabilistic model (Ribot & Bensana, 2011). A particular way for representing the remaining useful life of systems is to establish a fault probability from reliability analyses at different stress levels (operating conditions) (Hall & Strutt, 2003; Vachtsevanos, Lewis, Roemer, A.Hess, & Wu, 2006). Stress is defined as the set of internal and external conditions/factors that may have an impact on the system behavior. The parametrized Weibull model is often used in reliability for life data analyses due to its flexibility (Ferreiro & Arnaiz, 2008):

$$W(t, \beta, \eta, \gamma) = \frac{\beta}{\eta}\left(\frac{t-\gamma}{\eta}\right)^{(\beta-1)} e^{-(\frac{t-\gamma}{\eta})^{\beta}} \qquad (2)$$

where $t \geq 0$, $\beta \geq 0$, $\eta \geq 0$ and $\gamma \in [-\infty; \infty]$. The scale characteristic η defines the characteristic life of the system and corresponds to the mean life expectancy for a studied population sample. The shape characteristic β modifies the probability density function (pdf) nature and allows to model the different life phases of a system defined by the idealized bathtub curve of reliability. The location characteristic γ shifts the curve from the origin. It defines the system minimal life. The case $\gamma > 0$ means that the fault probability is zero until a date γ. In most cases, we assume $\gamma = 0$. The characteristic η is stress-dependent while β is assumed to remain constant across different stress levels.

Weibull characteristics $\beta_j^{q_i}, \eta_j^{q_i}, \gamma_j^{q_i}$ model the aging evolution of a system that leads to a fault f_j in a behavioral mode q_i and totally define the fault probability distribution $f_j^{q_i}$:

$$f_j^{q_i}(t) = \int_0^t W(t, \beta_j^{q_i}, \eta_j^{q_i}, \gamma_j^{q_i})dt. \quad (3)$$

The fault probability density function $W(t, \beta_j^{q_i}, \eta_j^{q_i}, \gamma_j^{q_i})$ has to give at any time the probability that the fault f_j occurs in the system from a mode q_i. Weibull characteristics $\beta_j^{q_i}$ and $\eta_j^{q_i}$ are fixed by the mode q_i of the system. The location characteristic $\gamma_j^{q_i}$ can be used to memorize the degradation evolution of the system in the past modes from the operation start of the system (Ribot & Bensana, 2011). At first, the system is in a mode q_0. If the system has never been used, q_0 obviously represents the nominal mode and we suppose that $\forall f_j, \quad \gamma_j^{q_0} = 0$ as previously explained. This characteristic $\gamma_j^{q_i}$ will be modified to take degradation in each behavioral mode into account during the system operation.

The occurrence date d_{f_j} of a fault event f_j for the system in mode q_i can be determined from a decision criterion $P_{max_{f_j}}$ that corresponds to a probability threshold beyond which the risk becomes unacceptable:

$$d_{f_j} \text{ such that } \int_0^{d_{f_j}} W(t, \beta_j^{q_i}, \eta_j^{q_i}, \gamma_j^{q_i})dt = P_{max_{f_j}}. \quad (4)$$

4.3. Enriched hybrid model

In each mode q_i, the system is subject to different aging laws. The set of aging laws is supposed to be accurately known.

Hypothesis 2. *An aging law of a system is supposed to be continuous over time.*

The consequence of this hypothesis is that the initial condition for an aging law at time t_{k+1}^+ is the value at t_{k+1}^-, when the system has not yet commuted between two modes.

To take into account the different aging laws, the hybrid system is then described as an enriched hybrid automaton $S^+ = (\zeta, Q, \Sigma, T, C, \mathcal{F}, (q_0, \zeta_0))$, where $\mathcal{F} = \{F^{q_i}, i \in \{1, \ldots, card(Q)\}\}$ is the set of aging laws associated to behavioral modes or the system. F^{q_i} is a vector of aging laws for each anticipated fault in the mode q_i. For example, in a system where N_F faults are considered:

$$F^{q_i}(t) = \begin{bmatrix} f_1^{q_i}(t) \\ f_2^{q_i}(t) \\ \ldots \\ f_{N_F}^{q_i}(t) \end{bmatrix} \quad (5)$$

where $f_j^{q_i}$ represents the probability distribution of the fault f_j at any time in mode q_i.

It can be noticed that as opposed to (Ribot & Bensana, 2011), the hybrid automaton we propose represents behavioral modes and not operational modes based on function availability.

For example, for a system with two nominal modes q_{01}, q_{02}, two possible actions a_1, a_2 that are observable events, and two faults f_1 and f_2, a possible model is given in Figure 4. This system is in a failure mode when f_1 and f_2 have occurred. If only one fault occurred, then the system is in a faulty mode.

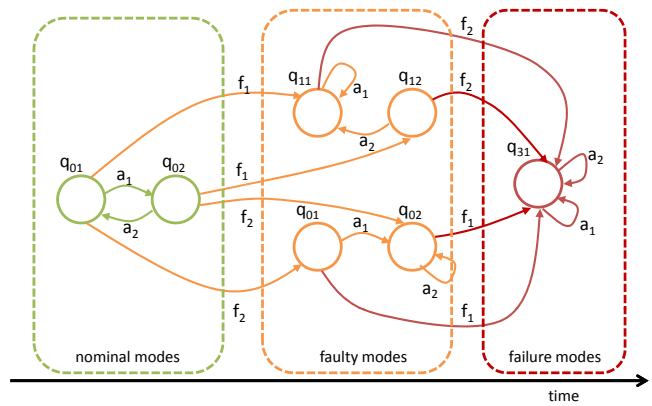


Figure 4. Example for a system with 2 nominal modes

5. Prognosis

The focus of this article is on the prognosis process. It consists, at each clock tick t_k, in computing the most likely fault sequence Π_k until the system failure. Algorithm 1 describes how the prognosis process is structured and introduces the three main functions of the process.

Algorithm 1: Prognosis of a hybrid system

Inputs: enriched model S^+, on-line diagnosis Δ_k
Outputs: RUL, fault sequences Π_k

```
k = 0
q− ← q0 {Mode initialization}
for each anticipated fault f_j ∈ Σ_f do
    (f_j^{q−}, d_{f_j}) ← InitializeAgingParameters(q−)
end for
(Π_0, RUL) ← PredictFaultSequence(S+, q−, {d_{f_j}})
repeat
    k++
    if Δ_k ≠ q− then
        q+ ← Δ_k
        {(f_j^{q+}, d_{f_j})} ←
        UpdateAgingParameters(S+, q−, q+)
        q− ← q+
        (Π_k, RUL) ← PredictFaultSequence(S+, q−,
        {d_{f_j}})
    end if
until RUL = 0
```

Prognosis takes as input information on aging laws in S^+ associated to the set of anticipated faults $\mathcal{F}$ and the set of behavioral models Q. It takes also as input the current result of the diagnosis Δ_k to update on-line the system aging laws according to the operation time in each behavioral mode. After each observable event, the appropriate aging laws are selected (l.11) according to the mode that is estimated by the diagnosis and the fault probability value reached in previous modes. Then the prognosis process predicts the most probable fault sequence (l.6,14) supposing that the system remains in the current mode.

Thus three main functions may be distinguished: the aging parameters initialization (l.3), the aging parameters update (l.11) and the fault sequence prediction (l.6,14). The following subsections describe precisely how the prognosis process is built and interleaved with diagnosis.

5.1. Aging function parameters initialization

The system is degrading in different ways leading to fault occurrences that may provoke a failure. This degradation depends on the mode of the hybrid system. For each mode, the degradation embodies the impact of stress factors. We recall that the aging dynamic of the system exposed to a fault f_j in mode q_i is modeled by a set of Weibull parameters $\beta_j^{q_i}$,$\eta_j^{q_i}$ and $\gamma_j^{q_i}$ in the enriched model S^+.

For the simplicity of the presentation, we first assume that there is no problem of diagnosability in the system that is studied. It means that at each clock tick, the system mode is totally known, i-e. non ambiguous and is given by the diagnosis process. This is a high hypothesis, and the case of ambiguity in the system state has to be studied in the future. Nevertheless, an easy solution to transform an ambiguous case into a non ambiguous one is to consider that the system is in its most probable state.

When the prognosis process is started, the system is in initial state (q_0, ζ_0). The aging law $f_j^{q_0}(t)$ associated to each anticipated fault f_j is initialized. Parameters $\beta_j^{q_0}$ and $\eta_j^{q_0}$ are assumed to be fixed and derived from reliability analyses. If the studied system has never been used before, q_0 obviously represents the nominal mode and we suppose that $\forall f_j,\ \gamma_j^{q_0} = 0$ as previously explained. If q_0 is not nominal, the enriched model has to give information about the initial values of $\gamma_j^{q_0}$.

The occurrence date d_{f_j} of each anticipated fault f_j for the system in mode q_0 is then determined from a decision criterion $P_{max_{f_j}}$:

$$\int_0^{d_{f_j}} W(t, \beta_j^{q_0}, \eta_j^{q_0}, \gamma_j^{q_0})dt = P_{max_{f_j}}. \tag{6}$$

All along its operation, the aging probability of the system exposed to a fault f_j is denoted by P_{f_j}. It is evaluated on-line with diagnosis and predicted to determine the RUL. Knowing the aging dynamic of the system in the initial mode and the different aging threshold, the prognosis process predicts the most probable fault sequence until the failure mode. This prediction process is done each time the diagnosis process updates the current mode of the system. The next two sections describe how the aging parameters need to be updated after a new diagnosis result and how fault sequences are then predicted.

5.2. Aging parameters update

Algorithm 2 describes how the aging functions are updated after each new diagnosis result.

Algorithm 2: UpdateAgingParameters

Inputs: Enriched model S^+, previous behavioral mode $q-$, new behavioral mode $q+$ provided by diagnosis

Outputs: New aging laws f_j^{q+} for each anticipated fault f_j, new dates of fault occurrences d_{f_j}

1: **for** each anticipated fault $f_j \in \Sigma_f$ **do**
2: $P_{f_j} \leftarrow$ ComputeAgingProbability(f_j^{q-}) {with Equation (7)}
3: $\gamma_j^{q+} \leftarrow$ ComputeLocationParameter(P_{f_j},$q+$) {with Equations (8) (9)}
4: **end for**

On receipt of a new mode estimation $q+$ at time t^+, the Weibull aging functions associated to faults in the mode $q+$ are updated according to the time spent by the system in previous mode $q-$. The aging probability associated to a fault f_j that the system has reached in past mode(s) at t^+ is computed with

$$P_{f_j} = \int_0^{t^+} W(t, \beta_j^{q-}, \eta_j^{q-}, \gamma_j^{q-})\mathrm{d}t. \tag{7}$$

To memorize this aging probability P_{f_j}, a new value for characteristic γ_j^{q+} of aging model associated to the fault f_j in the new mode $q+$ is computed:

$$\gamma_j^{q+} = (t^+ - \delta) \text{ such that } \int_0^{\delta} W(t, \beta_j^{q+}, \eta_j^{q+}, 0)\mathrm{d}t = P_{f_j}. \tag{8}$$

With the above equation, we introduce a mathematical manipulation to memorize the aging probability P_{f_j} reached in past modes into new aging models for mode $q+$ from t^+. By this calculation, the continuity of any aging function is guaranteed in all mode change points and

$$\int_0^{t^+} W(t, \beta_j^{q+}, \eta_j^{q+}, \gamma_j^{q+})\mathrm{d}t = P_{f_j} \tag{9}$$

Figure 5 illustrates how Weibull pdf associated to a fault f_1

evolves in order to describe the fault aging in two behavioral modes q_{01} and q_{02}. The two curves on the top illustrate the Weibull pdf associated to fault f_1 in mode q_{01} (on the left) and in mode q_{02} (on the right). Let suppose that the system is in mode q_{01} at $t = 0$, then P_{f_1} corresponds to the area under the curve. When the system switches into mode q_{02}, the continuity condition on the aging law (Hypothesis 2) implies that the area under the first curve is equal to the area under the second one. So the second curve is translated on the left.

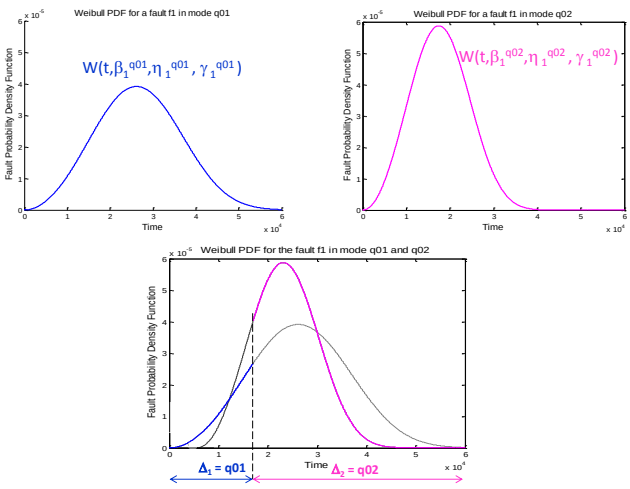


Figure 5. Weibull pdf associated to a fault f_1 for a system in two behavioral modes

5.3. Fault sequence prediction

To determine the sequence of fault events that leads to the system failure, we need to define a discrete fault event automaton (l.4) to extract the system faulty evolutions. This automaton describes all the possible fault transitions between operating modes until the system failure. The Discrete Fault Event System (DFES) is obtained by the projection of the underlying DES on the fault events (see Section 4.1). It corresponds to an abstraction of the hybrid system which contains fault events only. The DFES is a finite state machine $M_f = (Q_f, \Sigma_f, T_f, q_{f0})$ formally defined as follows:

- Q_f is the set of discrete states of the system,
- Σ_f is set of the fault events,
- $T_f \subseteq (Q_f \times \Sigma_f \to Q_f)$ is the partial transition function,
- $q_{f0} = q_0$ is the initial state corresponding to the nominal mode of the system.

The DFES of the example described in Figure 4 is illustrated by Figure 6.

Once a mode change is detected by diagnosis, the idea is to run through the DFES and predict the fault sequences until the system failure. A state q_c in a DFES is a deadlock if

$$\forall f \in \Sigma_f,\ T_f(q_c, f) = \emptyset \tag{10}$$

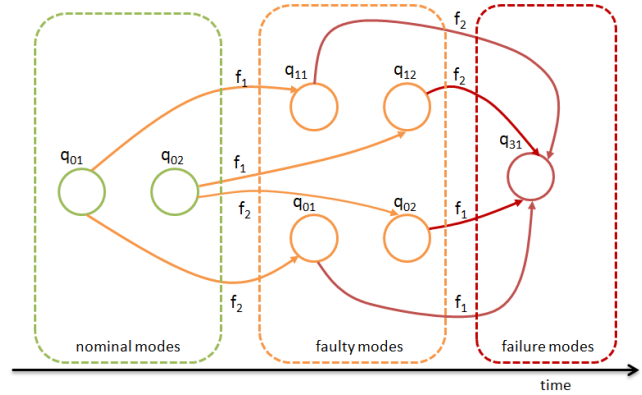


Figure 6. The DFES

The stop criterion for the prediction function is a dead-lock (for example mode q_{31} in Figure 6).

For each fault f_j that has not occurred yet, the algorithm computes the date of occurrence of f_j. The minimum value in this set of dates is denoted d_{min_1}: it corresponds to the date of occurrence of the next fault, denoted f_{min_1} (l.6). The set of faults whose occurrence has not been predicted yet is denoted by Σ_{ff}, then Σ_{ff} is updated and $\Sigma_{ff} = \Sigma_f \backslash f_{min_1}$ (l.7). It then contains faults whose occurrence date is superior to d_{min_1}. At d_{min_1}, the system is predicted to switch into fault mode $q_{f_{min_1}}$. New aging models in fault mode $q_{f_{min_1}}$ (described by the Weibull pdf $W(t, \beta_j^{q_{f_{min_1}}}, \eta_j^{q_{f_{min_1}}}, \gamma_j^{q_{f_{min_1}}})$) have to be updated for each fault f_j in Σ_{ff} (l.12). The mode change predicted at d_{min_1} may result in a modification of fault dates $\{d_{f_j}\}$ that have been previously computed.

As for an aging parameter update resulting from a change in diagnosis, characteristic $\gamma_j^{q_{f_{min_1}}}$ of aging models in future mode $q_{f_{min_1}}$ has to be computed from the fault probability $P_{f_j}^1$ the system could have reached at predicted commutation time d_{min_1}. Let q_c denotes the current system mode, for each fault f_j in Σ_{ff} :

$$P_{f_j} = \int_0^{d_{min_1}} W(t, \beta_j^{q_c}, \eta_j^{q_c}, \gamma_j^{q_c})\mathrm{d}t, \tag{11}$$

and $\gamma_j^{q_{f_{min_1}}} = (d_{min_1} - \delta)$ such that

$$\int_0^{\delta} W(t, \beta_j^{q_{f_{min_1}}}, \eta_j^{q_{f_{min_1}}}, 0)\mathrm{d}t = P_{f_j}. \tag{12}$$

Characteristic $\gamma_j^{q_{f_{min_1}}}$ allows to memorize the system aging in all past modes from q_0 and guarantee the continuity of aging laws. The date d_{f_j} of fault occurrences in Σ_{ff} are modified as follows:

$$\int_0^{d_{f_j}} W(t, \beta_j^{q_{f_{min_1}}}, \eta_j^{q_{f_{min_1}}}, \gamma_j^{q_{f_{min_1}}})dt = P_{max_j}. \tag{13}$$

The next possible fault f_{min_2} after f_{min_1} is determined from the minimal predicted fault date d_{min_2} for faults in Σ_{ff}. Then $\Sigma_{ff} = \Sigma_{ff} \backslash f_{min_2}$. Fault propagation is studied as explained above to compute $\gamma_j^{q_{f_{min_2}}}$ for faults that have not reached their probability threshold at d_{min_2} using new aging models for mode $q_{f_{min_2}}$ and the process reiterates.

The prognosis process computes the most likely future sequence Π_k of dated fault events according to a diagnosis Δ_k:

$$\Pi_k = (\{f_{min_1}, d_{min_1}\}, \{f_{min_2}, d_{min_2}\}, \ldots, \{f_{min_N}, d_{min_N}\}). \quad (14)$$

Algorithm 3 sums up the procedure of fault sequence prediction.

Algorithm 3: PredictFaultSequence

Inputs: Enriched model S^+, Current mode q_c, Dates of fault occurrence $\{d_{f_j}\}$
Outputs: Fault sequence Π_k, RUL

$\Pi_k = \emptyset$
$\Sigma_{ff} \leftarrow \Sigma_f$ {Σ_{ff} is the set of faults to be predicted}
$i \leftarrow 1$
ConstructDFES(S^+)
while $\exists f \in \Sigma_{ff} \mid T_f(q_c, f) \neq \emptyset$ {q_c is not a dead-lock} **do**
  $(f_{min}(i), d_{min}(i)) \leftarrow$ PredictNextFault(Σ_{ff}, $\{d_{f_j}\}$)
  $\Sigma_{ff} \leftarrow \Sigma_{ff} \backslash f_{min}$
  $\Pi_k = \Pi_k \cup \{(f_{min}(i), d_{min}(i)\}$
  RUL $\leftarrow d_{min}(i)$
  $q_f \leftarrow T_f(q_c, f_{min}(i))$ {System is predicted to switch in mode q_f at d_{min}}
  for each anticipated fault $f_j \in \Sigma_{ff}$ **do**
    $(f_j^{q_f}, d_{f_j}) \leftarrow$ UpdateAgingParameters(q_c,q_f) {with Equations (11) (12) (13)}
  end for
  $i \leftarrow i + 1$
  $q_c \leftarrow q_f$
end while

6. Experimental results

HYDIAG is a software program on MATLAB developed by the DISCO team. It performs diagnosis of hybrid systems (Bayoudh et al., 2008). The idea was to enrich it with Weibull aging models to performs prognosis, to implement the prognosis algorithm and interleave diagnosis and prognosis processes into a single one module named *InterDP*. This has been implemented and tested on a water tank system.

6.1. Modeling of a Water tank system

InterDP has been applied to the case study of a water tank system (Figure 7). This system is composed of one tank with two hydraulic pumps (P_1, P_2) providing water with different rates. Water flows through a valve at the bottom of the tank depending on the system control. Three sensors located

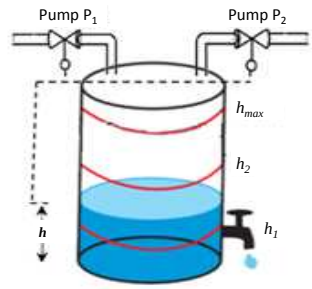
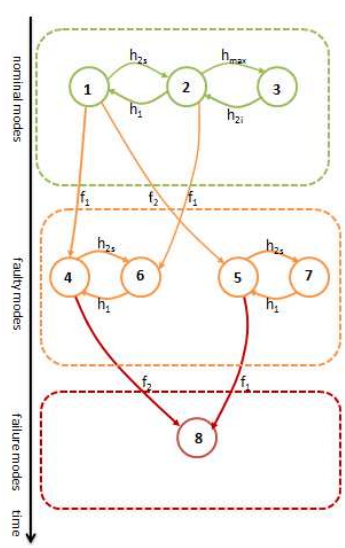


Figure 7. Water tank system

at different tank levels (h_1, h_2, h_{max}) detect the water level and allows to set the control of the pumps (on/off). If the water level h is between h_1 and h_2, both pumps P_1 and P_2 are turned on. If $h_2 < h < h_{max}$, only P_1 is on and when $h \leq h_{max}$, the pumps are turned off. It is assumed that the pumps may fail only if they are on.

The discrete model of water tank and the controls of pumps are given in Figure 8. Eight discrete behavioral modes has been identified $Q = \{q_1, q_2, \ldots, q_8\}$. Discrete events in $\Sigma = \{h_1, h_{2s}, h_{2i}, h_{max}, f_1, f_2\}$ allow the system to switch into different modes. Observable events are $\Sigma_o = \{h_1, h_{2s}, h_{2i}, h_{max}\}$. They result from sensor measurements of water level in the tank. Two faults are anticipated $\Sigma_f = \{f_1, f_2\}$ and are not observable: f_1 corresponds to the failure of P_1 and f_2 is the failure of P_2. The parameter values of aging models $\mathcal{F} = \{F^{q_i}\}$ associated to every behavioral mode q_i are reported in Table 1. Let us recall that the Weibull characteristics γ is assumed to be equal to zero at the system start.



Mode \ Pump	Pump1	Pump2
1	ON	ON
2	ON	OFF
3	OFF	OFF
4	Fail	ON
5	Fail	OFF
6	ON	Fail
7	OFF	Fail
8	Fail	Fail

Figure 8. Water tank DES model

Table 1. Weibull parameters of aging models

Aging laws		β	η
F^{q_1}	$f_1^{q_1}$	1.5	3000
	$f_2^{q_1}$	1.5	4000
F^{q_2}	$f_1^{q_2}$	2	3000
	$f_2^{q_2}$	1	7000
F^{q_3}	$f_1^{q_3}$	1	8000
	$f_2^{q_3}$	1	7000
F^{q_4}	$f_1^{q_4}$	NaN	NaN
	$f_2^{q_4}$	2	4000
F^{q_5}	$f_1^{q_5}$	2	3000
	$f_2^{q_5}$	NaN	NaN
F^{q_6}	$f_1^{q_6}$	NaN	NaN
	$f_2^{q_6}$	1	7000
F^{q_7}	$f_1^{q_7}$	1	8000
	$f_2^{q_7}$	NaN	NaN
F^{q_8}	$f_1^{q_8}$	NaN	NaN
	$f_2^{q_8}$	NaN	NaN

The underlying continuous behavior of every discrete mode q_i for $i \in \{1..8\}$ is represented by the same state space:

$$\begin{cases} X(k+1) &= AX(k) + BU(k) \\ Y(k) &= CX(k) + DU(k) \end{cases} \tag{15}$$

where the state variable X is the water level in the tank, continuous inputs U are the flows delivered by the pumps P_1, P_2 and the flow going through the valve, $A = \begin{pmatrix} 1 \end{pmatrix}$, $B = \begin{pmatrix} eTe/S \\ eTe/S \\ eTe/S \end{pmatrix}$ with Te the sample time, S the tank base area and $e_i = 1$ (resp. 0) if the pump is turned on (resp. turned off), $C = \begin{pmatrix} 1 \end{pmatrix}$ and $D = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$.

The continuous behavior is abstracted to build new observable discrete events Rox_y using the parity space approach. The enriched discrete event model of the hybrid system is used to build the diagnoser that will allow to track the system mode after each new observation.

The process *InterDP* was tested on this water tank hybrid model. Both diagnosis and prognosis are performed.

6.2. Simulation results

6.2.1. Simulation parameters

The time horizon is fixed at $T_{sim} = 4000$h, the sampling period is $T_s = 36$s and the filter sensitivity for the diagnosis is set as $T_{filter} = 3$min. The residual threshold is 10^{-12} as in (Bayoudh, 2009).

The scenarios involve a variant use of water (max flow rate = 1200L/h) depending on user needs during 4000h. Pumps are automatically controlled to satisfy the specifications indicated above. Flow rate of P_1 and P_2 are respectively 750L/h and 500L/h.

The diagnoser issued from the diagnosis process is given in Figure 9. Its computation is done off-line. Each state of the diagnoser indicates the belief state in the model enriched by the abstraction of the continuous part of the system, with a tag that gives the set of faults that have occurred on the system. This set is empty in case of nominal mode. This diagnoser shows that the tested system is diagnosable.

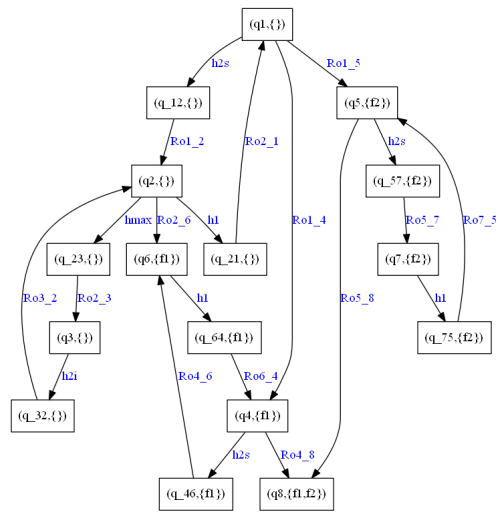


Figure 9. Diagnoser state tracker

Two fault scenarios have been simulated. In the Scenario 1, fault f_1 on the pump P_1 was injected after 3500h, fault f_2 is not injected. In the Scenario 2, fault f_2 is injected after 2000h, fault f_1 is not injected.

6.2.2. Scenario 1

Figure 10 shows the diagnoser belief state for Scenario 1 just before and after the fault f_1 occurrence. Results are consistent with the scenario: before 3500h, the belief states of the diagnoser are always tagged with a nominal diagnosis. After 3500h, all the states are tagged with f_1.

Figure 11 illustrates the predicted date of fault occurrence (d_{f_1} and d_{f_2}). At the beginning of the process, the prognosis result is: $\Pi_0 = (\{f_1, 4120\}, \{f_2, 5105\})$. It can be noted that the predicted dates d_{f_1} and d_{f_2} of f_1 and f_2 globally increase. Indeed, the system oscillates between stressful modes and less stressful modes. To make it simple, we can consider that in some modes, the system does not degrade, so the predicted dates of f_1 and f_2 are postponed.

Before 3500h, the predicted date of f_1 is lower than the one of f_2. After 3500h, the predicted date of f_2 is updated, knowing that the system is in a degraded mode. The prognosis result is $\Pi_{3501} = (\{f_2, 5541\})$.

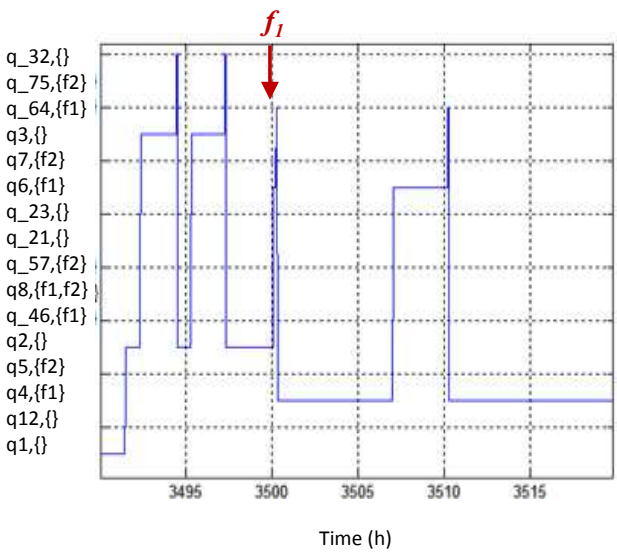


Figure 10. Diagnoser belief state for scenario 1

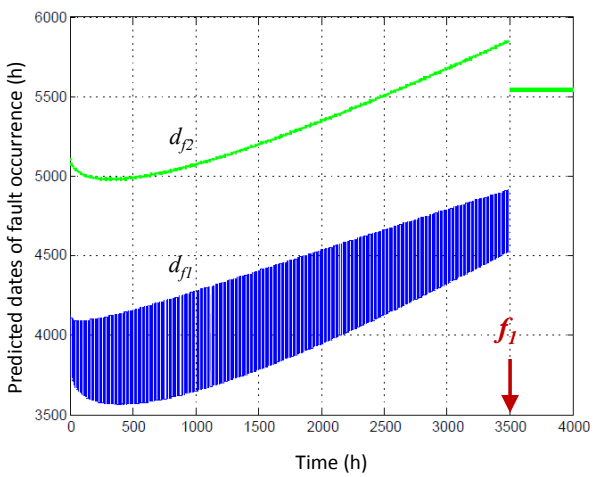


Figure 11. Prognosis results for scenario 1: d_{f_1}, d_{f_2}

Figure 12 shows the RUL of the system. Because of the oscillation of the system between stressful modes and less stressful modes, the RUL value oscillates between two values that globally decrease. The maximum bound corresponds to the less stressful mode whereas the minimum bound corresponds to the more stressful one. After the occurrence of the fault, the RUL is updated.

6.2.3. Scenario 2

Figure 13 shows the diagnoser belief state for Scenario 2 just before and after the fault f_2 occurrence. Results are coherent with the Scenario: before 2000h, the belief states of the diagnoser are always tagged with a nominal diagnosis. After 2000h, all the states are tagged with f_2.

Figure 14 illustrates the predicted date of fault occurrence

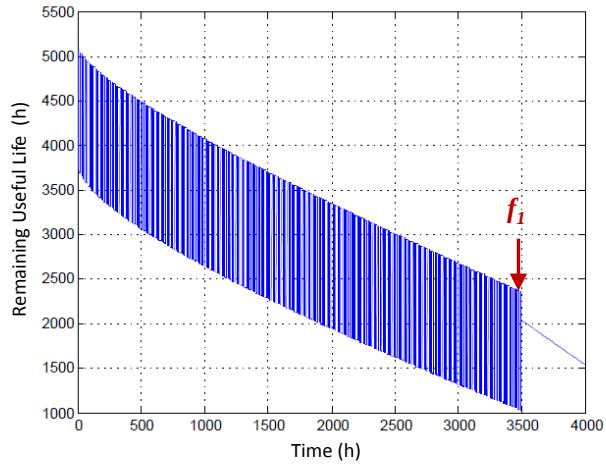


Figure 12. Prognosis results for scenario 1: RUL

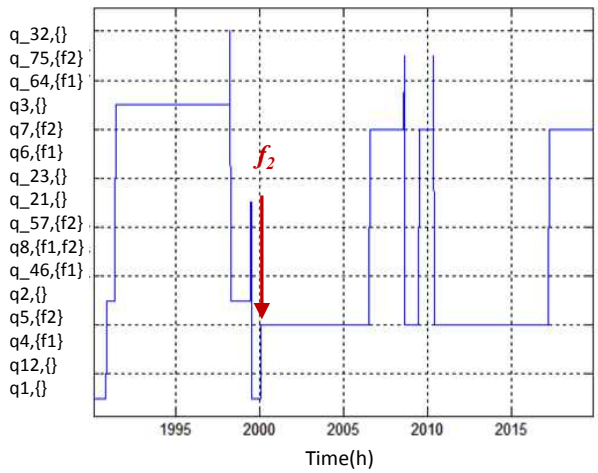


Figure 13. Diagnoser belief state for scenario 2

(d_{f_1} and d_{f_2}). At the beginning of the process, the prognosis result is: $\Pi_0 = (\{f_1, 4120\}, \{f_2, 5105\})$.

Before 2000h, the predicted date of f_1 is lower than the one of f_2. The prognosis result is After 2000h, the predicted date of f_1 is updated, knowing that the system is in a degraded mode. The prognosis result is $\Pi_{2010} = (\{f_1, 3947\})$.

Figure 15 shows the RUL of the system.

7. Conclusion and Future Work

This paper proposed an architecture of health monitoring and prognosis for hybrid systems. Diagnosis and prognosis are interleaved into a single process called *InterDP*. An original enriched hybrid model is proposed. It is used both for describing the hybrid behavior dynamic and for modeling the degradation of the system. This paper is focused on the prognosis process whose algorithm is given in detail. An implementation has been done in Matlab and *InterDP* has been

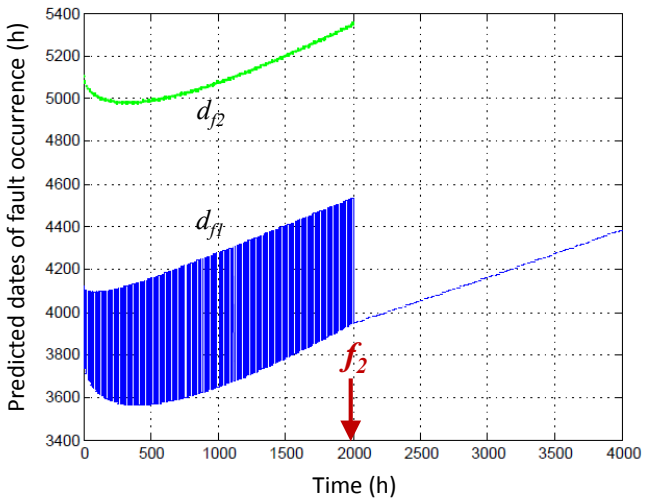


Figure 14. Prognosis results for scenario 2: d_{f_1}, d_{f_2}

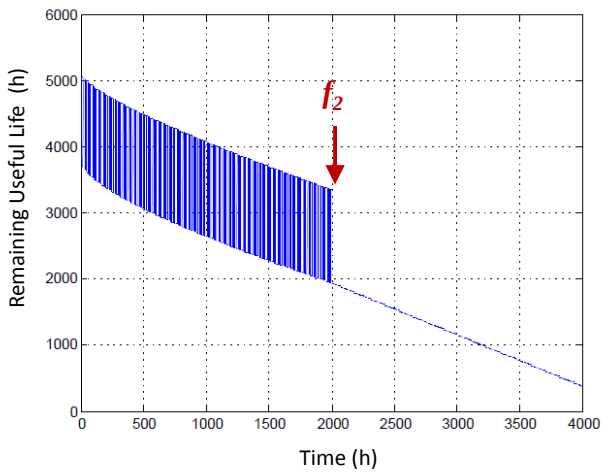


Figure 15. Prognosis results for scenario 2: RUL

tested on a water tank system. Simulation results show how the prognosis is updated thanks to the diagnosis.

In future work, we will study the case of a non diagnosable system, where the diagnoser provides a set of belief states. This will lead to a complexity problem both for diagnosis and prognosis processes. If a fault sequence has to be predicted for each belief state, this could combinatorial explode. Another complexity problem concerns the number of anticipated faults in each behavioral state. To reduce the model size of the hybrid system, only the most critical fault in each behavioral mode could be taken into account.

References

Bayoudh, M. (2009). *Active diagnosis of hybrid systems guided by diagnosability properties - application to autonomous satellites*. Unpublished doctoral dissertation, Université de Toulouse.

Bayoudh, M., Travé-Massuyes, L., Olive, X., & Space, T. A. (2008). Hybrid systems diagnosis by coupling continuous and discrete event techniques. *IFAC World Congress*, 7265–7270. doi: 10.3182/20080706-5-KR-1001.01229

Cao, X. (1989). The predictability of discrete event systems. *IEEE Transactions Automatic Control*, *34*(11), 1168–1171. doi: 10.1109/9.40745

Castaneda, G.-A. P., Aubry, J.-F., & Brinzei, N. (2010). DyReIA (Dynamic Reliability and Assessment). In *1st workshop on dynamic aspects in dependability models for fault-tolerant systems, DYADEM-FTS 2010.* Valence, Spain. doi: 10.1145/1772630.1772642

Chanthery, E., & Ribot, P. (2013, March). An integrated framework for diagnosis and prognosis of hybrid systems. In *the 3rd workshop on hybrid autonomous system (HAS).* Roma, Italy.

Ferreiro, S., & Arnaiz, A. (2008). Prognosis based on probabilistic models and reliability analysis to improve aircraft maintenance. In *International Conference on Prognostics and Health Management.* Denver, USA.

Genc, S., & Lafortune, L. (2006). Predictability in discrete-event systems under partial observations. In *6th ifac symposium on fault detection, supervision and safety of technical process (SAFEPROCESS)* (pp. 1531–1536). Beijing, China P.R.. doi: 10.3182/20060829-4-CN-2909.00243

Genc, S., & Lafortune, S. (2009). Predictability of event occurrences in partially-observed discrete-event systems. *Automatica*, *45*(2), 301-311. doi: 10.1016/j.automatica.2008.06.022

Hall, P. L., & Strutt, J. E. (2003). Probabilistic physics-of-failure models for component reliabilities using monte carlo simulation and weibull analysis: a parametric study. *Reliability Engineering and System Safety*, *80*, 233–242. doi: 10.1016/S0951-8320(03)00032-2

Henzinger, T. (1996). The theory of hybrid automata. In *Proceedings of the 11th annual IEEE symposium on logic in computer science* (pp. 278–292). doi: 10.1109/LICS.1996.561342

Jéron, T., Marchand, H., Genc, S., & Lafortune, S. (2008, July). Predictability of sequence patterns in discrete event systems. In *the 17th IFAC World congress* (p. 537). doi: 10.3182/20080706-5-KR-1001.00091

Khoumsi, A. (2009, June). Fault prognosis in real-time discrete event systems. In *DX'09* (p. 259).

Rausand, M., & Hoyland, A. (2004). *System reliability theory: models, statistical methods and applications*. Wiley. doi: 10.1002/9780470316900

Ribot, P., & Bensana, E. (2011, September 18-22). A generic adaptative prognostic function for heterogeneous multi-cmoponent systems: application to helicopters. In *ESREL 2011, European Safety & Reliability*

Conference. Troyes, France. doi: 10.1201/b11433-53

Ribot, P., Pencolé, Y., & Combacau, M. (2009, October 11-14). Diagnosis and prognosis for the maintenance of complex systems. In *IEEE International Conference on Systems, Man, and Cybernetics.* USA. doi: 10.1109/ICSMC.2009.5346718

Roychoudhury, I., & Daigle, M. (2011, October 4-7). An integrated model-based diagnostic and prognostic framework. In *22nd International Workshop on Principle of Diagnosis.* Murnau, Germany.

Sampath, M., Sengputa, R., Lafortune, S., Sinnamohideen, K., & Teneketsis, D. (1995). Diagnosability of discrete-event systems. *IEEE Trans. on Automatic Control*, *40*, 1555-1575. doi: 10.1109/9.412626

Vachtsevanos, G., Lewis, . L., Roemer, M., A.Hess, & Wu, B. (2006). *Intelligent fault diagnosis and prognosis for engineering systems*. Wiley. doi: 10.1002/9780470117842.index

Zemouri, R., & Faure, J.-M. (2006, October 4-6). Diagnosis of discrete event system by stochastic timed automata. In *IEEE International Conference on Control Applications* (pp. 1861–1866). Munich, Germany. doi: 10.1109/CACSD-CCA-ISIC.2006.4776924