



HAL
open science

Prognostics in switching systems: Evidential markovian classification of real-time neuro-fuzzy predictions.

Emmanuel Ramasso, Rafael Gouriveau

► To cite this version:

Emmanuel Ramasso, Rafael Gouriveau. Prognostics in switching systems: Evidential markovian classification of real-time neuro-fuzzy predictions.. IEEE Prognostics & Systems Health Management Conference, PHM'2010., Jan 2010, Macau, China. 9 p. hal-00459310

HAL Id: hal-00459310

<https://hal.science/hal-00459310>

Submitted on 23 Feb 2010

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.

Prognostics in Switching Systems: Evidential Markovian Classification of Real-Time Neuro-Fuzzy Predictions

RAMASSO Emmanuel, GOURIVEAU Rafael
FEMTO-ST Institute, UMR CNRS 6174 - UFC / ENSMM / UTBM
Automatic Control and Micro-Mechatronic Systems Department
emmanuel.ramasso@femto-st.fr, rgourive@ens2m.fr

Abstract – Condition-based maintenance is nowadays considered as a key-process in maintenance strategies and prognostics appears to be a very promising activity as it should permit to not engage inopportune spending. Various approaches have been developed and data-driven methods are increasingly applied. The training step of these methods generally requires huge datasets since a lot of methods rely on probability theory and/or on artificial neural networks. This step is thus time-consuming and generally made in batch mode which can be restrictive in practical application when few data are available. A method for prognostics is proposed to face up this problem of lack of information and missing prior knowledge. The approach is based on the integration of three complementary modules and aims at predicting the failure mode early while the system can switch between several functioning modes. The three modules are: 1) observation selection based on information theory and Choquet Integral, 2) prediction relying on an evolving real-time neuro-fuzzy system and 3) classification into one of the possible functioning modes using an evidential Markovian classifier based on Dempster-Shafer theory. Experiments concern the prediction of an engine health based on more than twenty observations.

I. INTRODUCTION

Prognostics reveals to be a very promising maintenance activity as it should permit to improve safety, plan successful missions, schedule maintenance, reduce maintenance cost and down time [1]. Also, industrials show a growing interest in this thematic which becomes a major research framework. However, considering the benefits that such technology may bring to the security, economics and resource management fields, real prognostics systems are still scarce in industry. From the research point of view, many developments exist to support the prognostics activity. They can be classified into three main categories, namely, model-based, data-driven, and experience-based approaches, and their applicability depends on the characteristics of the system under study [2, 3]. In practice, choosing an efficient technique depends on classical constraints that limit the applicability of the tools: available data, knowledge or experiences, dynamic and complexity of the system, implementation requirements (precision, computation time, etc.), available monitoring devices. Following that and assuming that it is often possible to instrument industrial plant with sensors, data-driven have been increasingly applied to prognostics (mainly techniques from artificial intelligence AI) and have shown improved performances over conventional approaches [4]. However, it is not easy to apply AI techniques

due to the lack of training data and of efficient procedures to extract specific knowledge. Thus, data-driven approaches are highly-dependent on the quantity and quality of operational data. This is the problem addressed in this paper: a method for prognostics is proposed to face up this problem of lack of information and missing prior knowledge in prognostics application. The approach is based on the integration of three modules and aims at predicting the failure mode early while the system can switch between several functioning modes. These modules are: 1) observation selection based on information theory and Choquet Integral, 2) prediction relying on an evolving real-time neuro-fuzzy system and 3) classification into one of the possible functioning modes using an evidential Markovian classifier based on Dempster-Shafer theory.

The paper is organized as follows. In section 2, the global prognostics architecture is presented in order to give an overview of the proposed prognostics procedure and of the complementary of the modules. Then, each step is more widely presented in sections 3, 4 and 5. Finally, the whole proposition is illustrated in a real-world prognostics problem concerning the prediction of an engine health.

II. PROGNOSTICS ARCHITECTURE

Prognostics within condition-based maintenance

Based on ISO 13381-1:2004 standard, "prognostics of future fault progressions requires foreknowledge of the probable failure modes, future duties to which the machine will/might be subjected, and a thorough understanding of the relationships between failure modes and operating conditions" [5]. Thereby, the whole aspects of failure analysis and prediction must be view as set of activities that all must be performed. This aspect is particularly pointed out within CBM concept (Condition-Based Maintenance). According to CBM practitioners, various activities, from data collection through the recommendation of specific maintenance actions, must be carried out to perform predictive maintenance. Usually, a CBM system is decomposed into seven layers, one of them being that of "prognostics" [6].

- Layer 1: The *Sensor Module* provides the system with digitized sensor or transducer data.
- Layer 2: The *Signal Processing Module* performs signal transformations and feature extractions.
- Layer 3: The *Condition Monitoring Module* compares on-line data with expected values; it should also be able to generate alerts.
- Layer 4: The *Health Assessment Module* determines if the system has degraded. It also generates a diagnostic record and suggests fault possibilities.

- Layer 5: The *Prognostics Module* predicts the future condition of the monitored system. It should be able to require data from all the previous modules.
 - Layer 6: The *Decision Support Module* provides recommended maintenance actions or alternatives on how to run the system until completed mission.
 - Layer 7: The *Presentation Module* can be built into a regular machine interface.
- In this paper, only layers from 2 to 5 are considered.

Prognostics architecture

The proposed prognostics architecture depicted in figure 1 aims at enabling to go from multidimensional data through the remaining useful life of a system. This prognostics procedure is composed of three complementary steps. Data are first processed (feature extraction, selection and cleaning). This step enables to feed an evolving neuro-fuzzy system which provides predictions of observations from t to $t+h$ (where h is the horizon of prediction). These predictions are finally analyzed by a classifier which provides the most probable state of the system at any time t . (Each step is more widely explained in its corresponding section.)

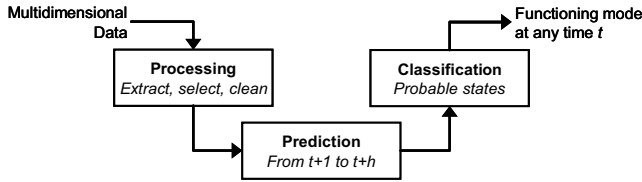


Figure 1: Prognostics architecture

The approach requires the data to be segmented into functioning modes. The granularity of the segmentation depends on the application. As an example, consider figure 2 where data concern the evolution of a health performance index for an engine. These data can be easily segmented into four functioning modes: "steady state" (the engine is working well), "degrading state" (the engine is being degraded), "transition state" (in between "steady" and "degrading"), and "critical state" (the engine will broke). The set comprising the data and the ground truth concerning the modes is called the training dataset.

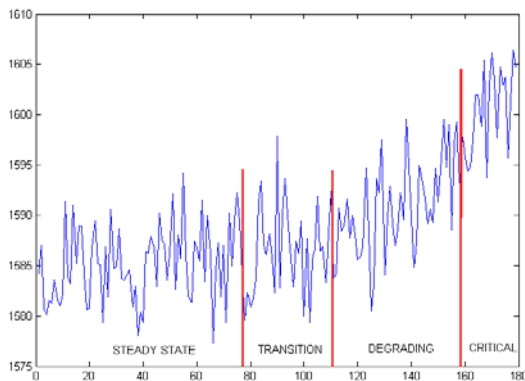


Figure 2: Segmentation into functioning modes

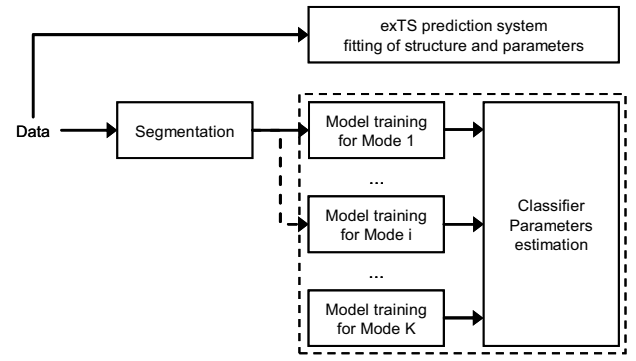


Figure 3: Training step

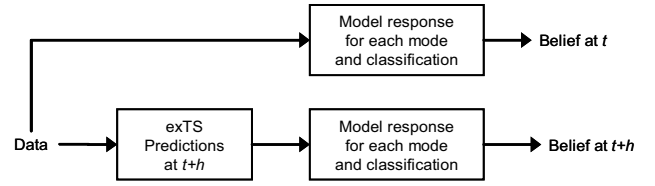


Figure 4: Testing step

Given the training dataset, an algorithm proposed in [7] is applied in order to select the most relevant set of features for each mode. This method relies on the Kullback-Leibler divergence and on the Choquet Integral. One advantage of this method is the computation of weights reflecting the importance of each feature and also of subsets of features. These weights are particularly interesting to quantify the redundancies and the complementarities of features (see [7] and cited references).

The training dataset is also used by both the neuro-fuzzy (NF) predictor and the classification module (figure 3). The NF predictor is based on the evolving extended Takagi-Sugeno system (exTS) introduced by [8]. The main advantage of this system is that it starts from scratch, i.e. any assumption on its structure and parameters initialization is necessary.

At classification step, "modeling algorithms" are used to provide a confidence value that reflects how likely is each functioning mode at each instant. These confidences values are then used in an Evidential Markovian Classifier (EMC) relying on the Transferable Belief Model framework [9]. This framework will be described later and is an extension of Dempster-Shafer theory of evidence [10, 11].

When the prediction and classification modules are correctly trained and after observing a new data at time t , the global prognostics architecture provides a belief concerning states at instant t and $t+h$ (figure 4).

III. FEATURE SELECTION

Objectives

The aim of this first step of the prognostics architecture is to identify the best set of features for each mode. The applied method was initially developed in [7].

Let adopt the following notations. A data (measured or computed at t) is denoted by $X_t = [x_{t,1} \ x_{t,2} \ \dots \ x_{t,F}]$, where F is

the dimension of the feature space. A dataset is the set of all data and is denoted by $X = \{X_1, X_2, \dots, X_T\}$ while the set of modes is denoted by $\Omega = \{M_1, M_2, \dots, M_K\}$. L_i is the set of data for which the ground truth is the functioning mode M_i . The other data are gathered in the dataset denoted R_i and the corresponding ground truth is denoted Z_i (it represents all modes except M_i). L_i and R_i are both of dimension F .

Given the mode M_i , the goal is to select the dimensions of L_i which brings the most important part of the information contained in the features. The resulting dataset will be denoted L_i' with dimension $F_i' \leq F$.

Theoretical background

The method works as follows. All possible combinations of features are considered. Given a particular combination $Y \subseteq X$, two probability distributions based on the data in L_i and Z_i can be estimated: these two probability distributions, denoted $P(Y|M_i)$ and $P(Y|Z_i)$, with $Y \subseteq X$, are actually defined conditionally to M_i and Z_i and are both expressed on the joint space of the considered set of features. The divergence between these distributions reflects how discriminative the current set of features for the considered mode is. The chosen divergence is the Kullback-Leibler one:

$$KL_i(Y) = \sum_Y P(Y|M_i) \log(P(Y|M_i)/P(Y|Z_i))$$

One set of 2^F divergence measures is thus computed for each mode. Given a mode M_i , the subset of features L_i' with the highest value of $KL_i(L_i')$ (and the lowest cardinality, i.e. F_i' as low as possible) is chosen. The subsets L_i' will then be used in classification (explained in the next section).

The method described in [7] also enables to build weights reflecting importance, redundancies and complementarities of each subset of feature. These weights are computed as follows. Let consider one subset $Y \subseteq 2^F$ in the set of divergence measures for a given mode M_i . The weight of each feature $f \in Y$ in the considered subset Y is given by:

$$v_f = \sum_{A \in Y-f} \frac{(n-|A|-1)!|A|!}{n!} \times (\mu(A \cup \{f\}) - \mu(A))$$

and the interaction coefficient between two features $f \in Y$ and $g \in Y$ is given by:

$$I_{f,g} = \sum_{A \in Y-\{f,g\}} \frac{(n-|A|-2)!|A|!}{(n-1)!} * (\mu(A \cup \{f,g\}) - \mu(A \cup \{f\}) - \mu(A \cup \{g\}) + \mu(A))$$

where the importance coefficient $\mu(S)$ of a subset S is given by the value of the divergence normalized by the divergence of the whole set of features:

$$\mu(S) = KL_i(S) / KL_i(X)$$

Given a mode M_i , the coefficients v and I represent the 2-additive Choquet Integral parameters while coefficient μ represents the generalized Choquet Integral parameters. The Choquet Integral is a powerful tool used for multicriteria decision making and information fusion (see references in [7]). In particular, this tool generalizes the weighted average. If the

interaction coefficient $I_{f,g} > 0$, the features f and g are said complementary while they are said redundant when the coefficient is negative. When all interactions are nil, coefficients v represent the weights of a weighted average [7].

This method has shown to be powerful in [7]. The main disadvantage is the necessity to compute a potentially high number of divergences that grows exponentially with the dimension F (until $F=15$, usual PC technologies and softwares are sufficient).

IV. TEMPORAL PREDICTIONS

Objectives

Assuming that data are defined in an multidimensional space (at any time t , $X_t = [x_{t,1} \ x_{t,2} \ \dots \ x_{t,F}]$, where F is the dimension of the feature space), the aims of the prediction module is to forecast in time the evolution of the data values:

$$X_t = [x_{t,1} \ x_{t,2} \ \dots \ x_{t,F}] \rightarrow X_{t+h} = [x_{t+p,1} \ x_{t+p,2} \ \dots \ x_{t+p,F}]$$

with $p=[1,h]$, h being the maximum horizon of prediction.

In practice, this global prediction can be performed by building a prediction system for each one of the sub-signals. For that purpose, many approaches exist in literature [12]. According to previous works [13], recent works focus on the interest of using hybrid systems for prediction purpose. More precisely, first order Takagi-Sugeno (TS) fuzzy models have shown improved performances over conventional approaches [4, 14]. In this paper, the evolving extended Takagi Sugeno system (exTS) introduced by [8] is used.

First order Takagi-Sugeno systems

A first order TS model can be seen as a multi-model structure consisting of linear models that are not necessarily independent [12]. It is based on the fuzzy decomposition of the input space. For each part of the state space, a fuzzy rule can be constructed to make a linear approximation of the input. The global output is a combination of the whole rules. Consider figure 5 to explain the first order TS model. This model has two inputs variables. Two membership functions (antecedent fuzzy sets) are assigned to each one of them. The TS model is finally composed of two fuzzy rules. (That can be generalized to the case of n inputs and N rules).

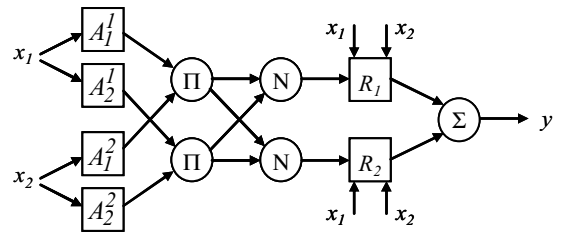


Figure 5: A First-order TS model with 2 inputs

The rules perform a linear approximation of inputs:

$$R_i: \text{if } x_1 \text{ is } A_i^1 \text{ and } \dots \text{ and } x_n \text{ is } A_i^n \\ \text{THEN } y_i = a_{i0} + a_{i1} \cdot x_1 + \dots + a_{in} \cdot x_n$$

where R_i is the i^{th} fuzzy rule, N is the number of rules, $X_i = [x_1, x_2, \dots, x_n]^T$ is the input vector, A_i^j denotes the antecedent fuzzy sets, $j=[1, n]$, y_i is the output of the i^{th} linear subsystem, and a_{iq} are its parameters, $q=[1, n]$. Let assume Gaussian antecedent fuzzy sets (this choice is justified by its generalization capabilities) to define the regions of fuzzy rules in which the local linear sub-models are valid:

$$\mu_j^i = \exp\left(-4 \frac{\|x - x^{i*}\|_j^2}{(\sigma_j^i)^2}\right)$$

where σ_j^i is the spread of the membership function, and x^{i*} is the focal point (center) of the i^{th} rule antecedent. The firing level (τ_i) and the normalized firing level (λ_i) of each rule are obtained as follows:

$$\tau_i = \mu_{i1}(x_1) \times \dots \times \mu_{in}(x_n), \quad \lambda_i = \tau_i / \sum_{j=1}^N \tau_j$$

The model output is the weighted averaging of individual rules' contributions:

$$y = \sum_{i=1}^N \lambda_i y_i = \sum_{i=1}^N \lambda_i x_e^T \pi_i$$

where $\pi_i = [a_{i0}, a_{i1}, \dots, a_{in}]$ is the vector parameter of the i^{th} sub-model, and $x_e = [1 \ X^T]^T$ is the expanded data vector.

A TS model has two types of parameters. The non-linear parameters are those of the membership functions (a Gaussian membership has two parameters: its center and its spread deviation). These kinds of parameter are referred to as premise or antecedent parameters. The second types of parameters are the linear ones that form the consequent part of each rule (a_{iq}). All this parameters must be tuned to fit to the studied problem. This is the aim of the learning procedure.

Learning procedure of exTS

The learning procedure of exTS is composed of two phases: (1) an unsupervised data clustering technique is used to adjust the antecedent parameters, (2) the supervised recursive least squares learning method is used to update the consequent parameters. These algorithms can not be fully detailed in this paper but are well described in [8, 15].

The exTS clustering phase processes on the global input-output data space: $z = [x^T; y^T]^T, z \in R^{n+m}$, where $n+m$ defines the dimensionality of the input/output data space. Each one of the sub-model of exTS operates in a sub-area of z . This clustering algorithm is based on the calculus of a potential which is the capability of a data to form a cluster (antecedent of a rule). The procedure starts from scratch and, as more data are available, the model evolves by replacement or upgrade of rules. This enables the adjustment of the antecedent parameters (the non-linear ones).

Note that the main advantages of this NF systems result from the clustering phase. Firstly, they an exTS is able to update the parameters without the intervention of an expert. Secondly, they it can be trained in online mode. Thirdly, it has a flexible structure that evolves as data are gathered: data are collected continuously which enables to form new rules or to modify an existing one. This last characteristic is very useful to take into account the non-stationary aspect of signal.

Using exTS for prediction

In forecasting applications, models are usually built by considering some past values of each input and output variables. Consider figure 6 as for an example. In this figure, the NF system is composed of 2 input variables (x_I and y), predictions are made at one step ahead ($t+1$), one regressor is used for variable y and two for variable x_I .

In the case of a mono-variable predictor (and assuming that future is for essence unknown), previous predictions can be used as for the inputs for next predictions. This type of architecture, named "cascade models" enables to perform multi-step ahead predictions (at $t+h$) without building various predictors (and thereby with a single learning phase).

Figure 7 shows the evolution of a performance index of an engine and the prediction that can be obtained thanks to an exTS. Note that in this figure, all predictions (from 51 to ###) where made at time $t = 50$.

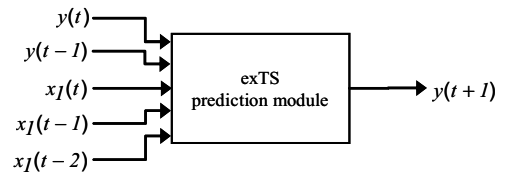


Figure 6: Forecasting model with regressors

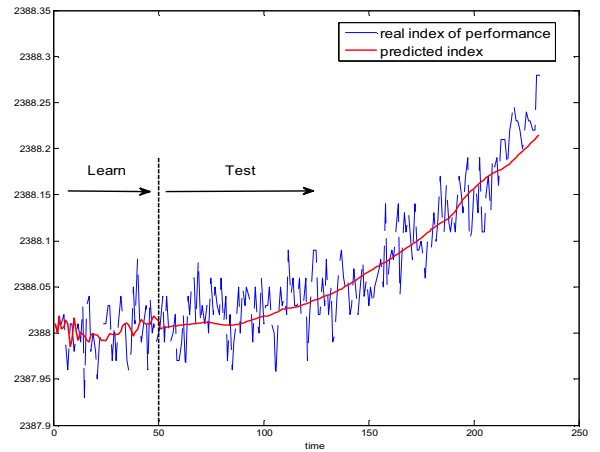


Figure 7: Example of multi-step ahead predictions

V. CLASSIFICATION OF TEMPORAL PREDICTIONS

Objectives

Given observations which can be measured at t or computed by exTS at $t+h$, the goal of this module is to provide a reliable classification into one of the possible functioning modes.

The proposed classification method relies on one model of data for each functioning mode. Given these models, a decision-making process is used to choose the best functioning mode given the observations. The decision-making process is temporal i.e. it embeds past and current knowledge on the functioning modes. The temporal aspects are taken into account using a Markovian state sequence recognition algorithm. The main idea of the proposed method is that the *sequence of modes*

leads to a more reliable conclusion concerning the real functioning mode than a decision based on modes only. This idea is not really new: in [16], authors proposed a similar approach based on Hidden Markov Models but their method requires prior information concerning the duration of the states. Most of methods implementing this idea were based on probability theory. We propose here to exploit a new framework called the ‘‘Transferable Belief Model’’ (TBM). This framework was proposed by Ph. Smets [9] and is based on belief functions (instead of probabilities) extending the work on Evidence Theory of Dempster and Shafer [10, 11]. The TBM has been shown to be powerful in particular for classification in many applications [17]. This is mainly due to the capacity of belief functions to model real situations without assuming prior, and also to the Generalized Bayesian Theorem (GBT) [18].

TBM for state sequence recognition

One of the first tools used for the analysis of temporal data in the TBM was proposed in 1999 by Rombaut et al. [19] but this tool is not robust to noise and no classification criterion was proposed. The generalized HMM [20] proposed in 2000 is narrowed down to possibility measures and thus is not able to manage belief functions. One advantage of their framework is the possibility to manage dependent observations by using fuzzy operators (but the authors used the product thereby assuming statistical independence). Other tools were developed by Pieczynski et al. [21, 22] who attempted to mix belief functions with Markov chains leading to promising results on several applications. This mixing alleviates the problem of prior modeling in Markov chains using Dempster’s rule of combination (which generalizes Bayesian inference [10]). However, either the prior or the belief state is evidential but not both (therefore underlying probability assumptions are present). In [23, 24] was also proposed a deterministic state machine where an original inference criterion relying on a conflict measure was proposed for classification of sequences. The main problem of this method is parameter setting for highly noisy data. Lastly, we proposed recently [25] an efficient method for state sequence recognition based on noisy observations in the TBM. It is a generalization of HMM mechanisms to belief functions and allows to use possibly dependent observations. This method is described in the sequel.

Belief functions in the Transferable Belief Model

We first recall the basics on belief functions necessary to understand the classification part.

The Basic Belief Assignment – Let $\Omega = \{M_1, M_2, \dots, M_K\}$ be the frame of discernment (FoD) gathering all possible and exclusive hypotheses. The number of hypotheses is called cardinal and denoted $|\Omega|$. The distributions of masses, also called Basic Belief Assignment (BBA), is defined on all possible subsets of the FoD which is $2^\Omega = \{\{M_1\}, \{M_2\}, \{M_1, M_2\}, \{M_3\}, \{M_1, M_2, M_3\}, \dots, \{\Omega\}\}$. A subset is denoted for example $S \subseteq \Omega$ or equivalently $S \in 2^\Omega$. The BBA is then defined as follows:

$$m : 2^\Omega \rightarrow [0,1]$$

$$S \subseteq \Omega \quad m^\Omega(S) \quad \text{with} \quad \sum_S m^\Omega(S) = 1$$

and a subset for which $m^\Omega(S) > 0$ is called a focal set. The BBA is said:

- *Normalized* when $m^\Omega(\emptyset) = 0$ (one can normalize a BBA easily by dividing each focal set by $1 - m^\Omega(\emptyset)$ and cancelling $m^\Omega(\emptyset)$).
- *Vacuous* when the whole belief mass is assigned to Ω . This BBA represents *total ignorance*.
- *Categorical* when the whole belief mass is assigned to a subset $B \subseteq \Omega$.

Given a BBA m^Ω , several functions can be computed which allow to interpret the BBA content and also to simplify combinations of BBA. The main functions are:

- The plausibility function: $pl^\Omega(S) = \sum_{C \cap S \neq \emptyset} m^\Omega(C)$ (which is strongly related to likelihood [11, 18])
- The credibility function: $bel^\Omega(S) = \sum_{\emptyset \neq C \subseteq S} m^\Omega(C)$, also called belief function.
- The commonality function: $q^\Omega(S) = \sum_{C \supseteq S} m^\Omega(C)$
- The weights of the canonical conjunctive decomposition: $w^\Omega(S) = \prod_{A \supseteq S} q(A)^{(-1)^{|A|-|S|+1}}$

The BBA can be recovered from any of these functions and the inverse formula are given in [11, 18].

The conjunctive rule of combination – Given two BBA provided by distinct and reliable sources of information, their fusion (or combination) can be done using the conjunctive rule of combination (CRC) defined by:

$$m_{1 \cap 2}(S) = \sum_{A \cap B = S} m(A) * m(B)$$

Using commonalities, the conjunctive rule reduces to: $q_{1 \cap 2}(S) = q_1(S) * q_2(S)$ i.e. to point-wise multiplication. Other combination rules were proposed in the literature. Of particular interest are the disjunctive rule of combination (the same formula as the CRC but replacing \cap by \cup) and the cautious CRC rule (called CCRC) [26] proposed when distinctness (i.e. independence) and reliability of the sources of information can be questioned. The CCRC is defined by:

$$w_{1 \otimes 2}(S) = w_1(S) \otimes w_2(S)$$

where the operator \otimes is the CCRC operator. The CCRC is based on pair of t-norm/conorm, for example, in [26], the author proposed to use Frank’s family:

$$w_{1 \otimes 2}(S) = \begin{cases} w_1(S) \wedge w_2(S) & \text{if } \tau = 0 \\ w_1(S) * w_2(S) & \text{if } \tau = 1 \\ \log_\tau \frac{(1 - \tau^{w_1(S)}) * (1 - \tau^{w_2(S)})}{\tau - 1} & \text{if } \tau \in]0, 1[\end{cases}$$

Conditioning – Given an imprecise and uncertain knowledge represented by a BBA m^Ω , one knows that the truth

is in one subset $B \subseteq \Omega$. The BBA m^Ω is updated by combining conjunctively both information using the CRC. The process is called conditioning:

$$m[B] = m \cap m_B$$

where m_B is a categorical BBA defined by $m_B(B)=1$ (conditioning in the TBM is denoted by brackets). Probabilistic conditioning is a special case recovered when m is Bayesian (and the BBA normalized).

Decision-making – Given a BBA m^Ω , one can be interested in making a decision based on this knowledge. For that, Smets [9] proposed the pignistic transformation:

$$BetP^\Omega \{m^\Omega\}(M_i) = \frac{1}{1 - m^\Omega(\emptyset)} \sum_{B \subseteq \Omega, M_i \in B} \frac{m^\Omega(B)}{|B|}$$

The Generalized Bayesian Theorem (GBT) – The Bayesian theorem allows computing a posterior probability distribution over a set of hypotheses given both a set of likelihoods (conditional to the hypotheses) and prior. In the TBM framework, this reasoning process has been extended by Smets [18] and called the GBT. The particularity of the GBT is that no prior is required for the posterior computation.

The GBT requires only a set of $|\Omega|$ conditional BBA $m^X[M_i]$ (where M_i plays the role of an hypothesis) while the prior is assumed vacuous. Each conditional BBA is defined on a set X which can be either discrete or continuous [27]. For classification problems, X represents (possibly multi-dimensional) observations and is generally continuous [17]. The GBT computes the posterior belief on Ω (i.e. the set of classes, here the modes) [18]:

$$pl^\Omega[x_t](S) = 1 - \prod_{M_i \in S} (1 - pl^X[M_i](x_t))$$

If a prior is available, it can be combined with the posterior belief using the CRC rule [].

Evidential Hidden Markov Models (EvHMM)

In [25], we proposed an extension of probabilistic Hidden Markov Models (HMM) to the TBM. To represent EvHMM, we use an extension of Directed Evidential Network [14] to the temporal domain that we call Temporal DEVN (TDEVN). A TDEVN is depicted in figure 8 (using the formalism proposed in [28]). The advantages of this framework are:

- Representation lack of knowledge on states, e.g. at the first instant the prior on states can be vacuous.
- Improvement of the estimation of the network parameters using a training set annotated by belief functions [25, 29].
- Improvement of the inference of the state sequence based on noisy observations using the evidential Viterbi-like decoder proposed in [25].
- Possibly non-distinct (not independent) observations can be considered using particular combinations rules [26].
- Combination of several types of formalism for uncertainty management in one common framework.

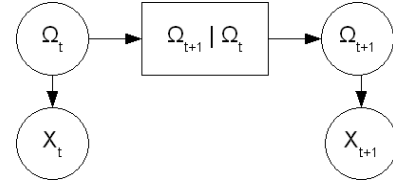


Figure 8: TDEVN representing EvHMM on two slices

Classification of sequence of observations in EvHMM – Given a set of T observations $X_{1:T}$ (with dimension F), it is possible to compute the likelihood of the sequence by computing recursively using the forward belief function. For a particular EvHMM λ , the forward variable is given by:

$$w_a^{\Omega_t}(S) = \left(\sum_{B \subseteq \Omega_{t-1}} m_a^{\Omega_{t-1}}(B) * w_a^{\Omega_t | \Omega_{t-1}}[B](S) \right) \otimes w_b^{\Omega_t | \mathfrak{R}}[X_t](S)$$

In order to represent missing prior at the first instant, one can use a vacuous BBA: $m_a^{\Omega_1}(\Omega_1) = 1$. If observations are distinct, one may use the commonalities instead of weights.

At each instant, the conflict value $m_a^{\Omega_t}(\emptyset)$ represents the amount of mass which is allocated to subsets not in 2^{Ω_t} and therefore, the total amount of conflict in the whole sequence quantifies how unlikely are the observations given the model [25]. The conflict value is linked to the plausibility by $m_a^{\Omega_t}(\emptyset) = 1 - pl_a^{\Omega_t}(\Omega_t)$. Thus, it is possible to compute the sequence plausibility for a particular EvHMM λ by:

$$L(\lambda) = \frac{1}{T} \sum_{t=1}^T \log pl_a^{\Omega_t}(\Omega_t)$$

and then the best model is given by maximizing $L(\lambda)$ over all models λ .

State sequence recognition in EvHMM – Given a sequence of noisy observations, one may be interested in knowing which state is the best one at a given instant. For that, two solutions are available. The first one relies on the *smoothing variable* and the second one relies on a *Viterbi-like decoding*.

a) The smoothing variable is computed by conjunctively combining both the forward and backward variables. The backward variable is given by [25]:

$$w_b^{\Omega_t}(S) = \sum_{B \subseteq \Omega_{t+1}} m_b^{\Omega_{t+1}}(B) * w_a^{\Omega_t+1 | \Omega_t}[B](S)$$

where $m_b^{\Omega_{t+1}}$ is the BBA obtained by conjunctively combining

$m_b^{\Omega_t+1}$ and $m_b^{\Omega_{t+1} | \mathfrak{R}}[X_{t+1}](S)$. The posterior weights

$w_a^{\Omega_t+1 | \Omega_t}$ are obtained by applying the GBT (by first transforming w into pl). The smoothing is thus given by:

$$w_\gamma^{\Omega_t}(S) = w_a^{\Omega_t}(S) \otimes w_b^{\Omega_t}(S)$$

The most probable state at t is then given by maximizing the pignistic probability over all states:

$$M_*^t = \arg \max_{M_i} \left[\text{BetP} \left\{ m_{\gamma}^{\Omega_t} \right\} (M_i) \right]$$

b) A Viterbi-like decoder allows propagating the decision at t over the states at subsequent instants. In [25], we proposed two algorithms and one of them is optimal in terms of maximum of plausibility. The second algorithm, called evidential Viterbi-like decoder is not optimal but allows to fully drawing benefit of belief functions framework. This algorithm is used in this paper since results are generally better than the previous one. The evidential Viterbi-like decoder is based on the propagation of the following metric $w_{\delta}^{\Omega_t}(S)$:

$$w_{\delta}^{\Omega_t}(S) = \left(\sum_{B \subseteq \mathfrak{S}_{t-1}} m_{\delta}^{\Omega_t-1}(B) * w_a^{\Omega_t|\Omega_t-1}[B](S) \right) \otimes w_b^{\Omega_t|\mathfrak{R}}[X_t](S)$$

The difference with the forward variable is the computation of the sum which is now done over subsets of \mathfrak{S}_{t-1} . This set is actually a subset of Ω_{t-1} and the hypotheses composing \mathfrak{S}_{t-1} are selected at each instant as follows. First, we compute the set of weights $w_{\delta,j}^{\Omega_t}$ defined by:

$$w_{\delta,j}^{\Omega_t}(S) = \left(\sum_{B \subseteq \Omega_{t-1}} m_{\delta}^{\Omega_t-1}(B) * w_a^{\Omega_t|\Omega_t-1}[B \cap M_j](S) \right) \otimes w_b^{\Omega_t|\mathfrak{R}}[X_t](S)$$

This equation is simply a forward propagation but conditioned on each mode $M_j \in \Omega_{t-1}$. One obtains a set of weights for each mode. These weights are then transformed into BBA $m_{\delta,j}^{\Omega_t}(S)$ and then into pignistic probability distributions. We thus obtain a set of $|\Omega_{t-1}|$ pignistic distributions defined on Ω_t conditionally to each mode. The best predecessor of a mode $M_i \in \Omega_t$ is thus found and stored as follows:

$$\psi_t(M_i) = \arg \max_{M_j \in \Omega_{t-1}} \text{BetP}^{\Omega_t|\Omega_t-1} \left\{ m_{\delta,j}^{\Omega_t} \right\} (M_i)$$

The set \mathfrak{S}_{t-1} is then composed by the union of predecessors:

$$\mathfrak{S}_{t-1} = \bigcup_i \psi_t(M_i)$$

At instant t , it is possible to compute the state sequence from t_0 . For that, we need first to compute the best mode M_*^t at t :

$$M_*^t = \arg \max_{M_j \in \Omega_{t-1}} \text{BetP}^{\Omega_t} \left\{ m_{\delta,j}^{\Omega_t} \right\} (M_i)$$

and then to use the backtracking process until t_0 :

$$M_*^{t-1} = \psi_t(M_*^t)$$

Parameter learning in EvHMM – In this paper, the training set is composed of observations for which the ground truth (real modes) is known. In this case, modes are not hidden but we use the Viterbi-like decoder developed in EvHMM in order to recognize the sequence of modes given observations. This approach has been used by several authors [16] based on probabilistic HMM. Each mode is thus modeled using an EM clustering assuming Gaussian mixtures. Given trained models, a new observation generates a set of likelihoods (one for each mode) which are then used in the GBT in order to compute the

posterior BBA on the set of modes given observations. These posterior BBA are then used to estimate automatically the evidential transition matrix. The method consists in computing the expected joint belief mass defined on the product space $\Omega_t \times \Omega_{t-1}$ over all time instants. In the case of possibly non-distinct observations, we have:

$$\hat{w}_a^{\Omega_t \times \Omega_{t-1}}(S) = \frac{1}{T-1} \sum_{t=2}^T w_b^{\Omega_t|\mathfrak{R} \uparrow \Omega_t \times \Omega_{t-1}|\mathfrak{R}}[X_t] \otimes w_b^{\Omega_{t-1}|\mathfrak{R} \uparrow \Omega_t \times \Omega_{t-1}|\mathfrak{R}}[X_{t-1}]$$

where $w_b^{\Omega_t|\mathfrak{R} \uparrow \Omega_t \times \Omega_{t-1}|\mathfrak{R}}[X_t]$ is called vacuous extension [] of $w_b^{\Omega_t|\mathfrak{R}}[X_t]$ onto the product space $\Omega_t \times \Omega_{t-1}$. For a BBA, the vacuous extension is defined by:

$$m_b^{\Omega_t|\mathfrak{R} \uparrow \Omega_t \times \Omega_{t-1}|\mathfrak{R}}[X_t](S) = \begin{cases} m_b^{\Omega_t|\mathfrak{R}}[X_t](B) & \text{if } S = B \times \Omega_{t-1} \\ 0 & \text{otherwise} \end{cases}$$

VI RESULTS

The evidential Viterbi-like decoder is illustrated on the challenge dataset of diagnostic and prognostics of machine faults from the first Int. Conf. on Prognostics and Health Management (2009) [30]. The dataset consisted of multiple multivariate time series (26 variables) with sensor noise (Fig. 2). Each time series was from a different engine of the same fleet and each engine started with different degrees of initial wear and manufacturing variation unknown to the user and considered normal. The engine was operating normally at the start and developed a fault at some point. The fault grew in magnitude until system failure.

Given a new observation sequence, the goal was to diagnose its current mode and to determine if the system was faulty. A fault occurred when a sequence of four modes was detected (steady \rightarrow transition \rightarrow degrading \rightarrow faulty).

From the 26 features, we first aimed at selecting only 8 of them. We first built all groups of 8 features and applied the feature selection method. For each mode, the group with the maximum value of the Kullback-Leibler divergence was selected. Then, we applied again the feature selection process considering all combinations of features among the group of eight. Note that the group of features for each mode was generally different.

One detector was built for each mode. For that, an EM was run on the training set using mixture of Gaussians with adaptive number of components.

For the prediction step, each feature was estimated with an exTS model for multi-step ahead prediction (a NF cascade model as explained before). Table 1 resumes the set of inputs variables used for that purpose. All predictions were made until time $t = 50$, so that, for each data test set, the prediction module provided the expected values of the considered performance index from time $t = 51$ to the end of the test series.

Table 1: Feature prediction with exTS

Feature	Inputs	Feature	Inputs
F 1	$t, x_1(t), x_1(t-1), x_1(t-2)$	F 5	$t, x_5(t)$

F 2	$t, x_2(t), x_2(t-1), x_2(t-2)$	F 6	$t, x_6(t)$
F 3	$t, x_3(t), x_3(t-1), x_3(t-2)$	F 7	$t, x_7(t), x_7(t-1)$
F 4	$t, x_4(t), x_4(t-1)$	F 8	$t, x_8(t), x_8(t-1)$

For the classification step, a four states EvHMM was then built where the evidential transition matrix was estimated as proposed in this paper (using the training set). For belief propagations, we used the CRC rule. The results of the evidential Viterbi-like decoder for state detection are given in Table 2 (“trans.” means “transition state”, “degrad.” means “degrading state”, “critic.” means “critical state”).

Table 2: State detection using EvHMM

	STEADY	TRANS.	DEGRAD.	CRITIC.
STEADY	87	43	7	0
TRANS.	9	52	23	10
DEGRAD.	4	5	54	18
CRITIC.	0	0	16	72

The results show good performance in detecting states, in particular the critical state and the steady states. The main problems appeared in detecting both the transition and degrading states: the former were of short duration and thus difficult to detect while the latter was highly evolving.

In order to compare our approach with probabilistic HMM, we simply converted the evidential transition matrix into probabilities (using the pignistic transformation) and used the likelihoods directly (without the GBT). Results of the detection are shown in Table 3.

Table 3: State detection using HMM

	STEADY	TRANS.	DEGRAD.	CRITIC.
STEADY	79	45	7	0
TRANS.	17	46	28	16
DEGRAD.	4	9	48	24
CRITIC.	0	0	17	60

The overall detection was slightly better with the evidential version than with the probabilistic HMM (an example is given in figure 9). In particular the detection of “transition” and “degrading” states were better with the evidential approach. This is mainly due to the (sub-optimal) Viterbi-like decoder which allows to propagate decisions carefully. Actually, this algorithm postpones the decision until the last instants thanks to the conditioned forward propagation.

The results presented in this paper are less good than in [25]. However in [25], the classification was done at each instant while in this paper the classification relied on predictions with a relatively long horizon because the average duration of the time series is close to 180 (time units) and thus the average value of the prediction horizon was close to 130 (time units). So the results of the proposed architecture are significative.

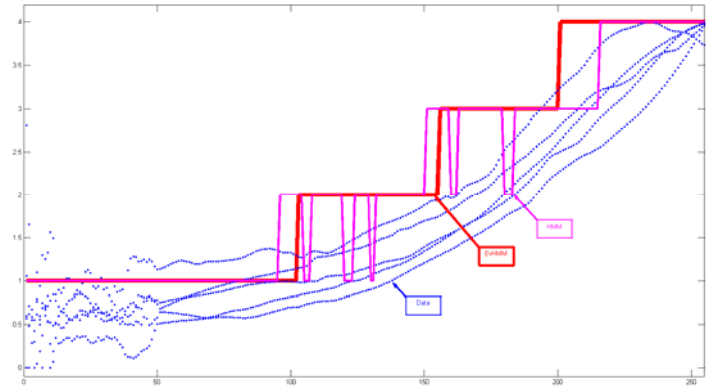


Figure 9: A detection result. Blue dots: data normalized in [0,4] for visualization. Magenta: HMM, Red: EvHMM detection.

VII CONCLUSIONS

An original and efficient architecture was proposed for diagnostics and prognostics of a system health. This architecture was composed of three modules: an information theory-based feature selection process, an exTS for reliable multi-step ahead predictions and on an evidence theory-based Markovian classifier for classification.

The efficiency of the proposed architecture was showed on a real data set concerning an engine health. In particular, the average horizon for predictions used in experiments was close to ± 130 time units and despite this challenging condition, the overall performance of the evidential classification of states was close to 70%. A comparison with probabilistic HMM for state classification clearly showed the efficiency of the proposed approach.

References

1. Brotherton, TT., Jahns, G., Jacobs, J. and Wroblewski, D. "Prognosis of faults in gas turbine engines". Proceedings IEEE Int. Conf. on Aerospace, Vol. 6 (2000), pp. 163-171.
2. Byington, C., Roemer, M., Kacprzynski, G. Galie, T. "Prognostic enhancements to diagnostic systems for improved condition-based maintenance". Proceedings of IEEE Int. Aerospace Conference, Big Sky, USA, 2002.
3. Vachtsevanos, G. *et al.*, Intelligent Fault Diagnostic and Prognosis for Engineering Systems, Wiley & Sons, (2006).
4. Wang, WQ., Golnaraghi, MF. and Ismail, F. "Prognosis of machine health condition using neuro-fuzzy systems", Mech. Syst. and Sign. Proc., Vol. 18 (2004), pp. 813-831.
5. ISO, 13381-1. Condition monitoring and diagnostics of machines, prognostics - Part1: General guidelines. International Standard, ISO (2004).
6. Lebold, M. and Thurston, M. "Open standards for condition-based maintenance and prognostics systems". Proc. of 5th A. Maintenance and Reliability Conf, Gatlinburg, USA, 2001.
7. Jullien, S., Valet, L., Mauris, G., Bolon, Ph. and Teyssier, S. "An attribute fusion system based on the Choquet Integral to evaluate the quality of composite parts". IEEE Trans. on Instrumentation and Measurement, Vol. 57 (2008), pp. 755-762.
8. Angelov, P. and Filev, D. "An approach to online identification of takagi-sugeno fuzzy models". IEEE Trans. Syst. Man Cybern. - Part B: Cybernetics", Vol. 34 (2004), pp. 484-498.

9. Smets, Ph. and Kennes, R. "The Transferable Belief Model", Vol 66 (1994), pp. 191-234.
10. A. P. Dempster. "Upper and lower probabilities induced by a multivalued mapping", *Annals of Mathematical Statistics*, Vol. 38 (1967), pp. 325-339.
11. Shafer, G. A mathematical theory of evidence, Princeton University Press, Princeton N.J (1976).
12. De Gooijer, JG. and Hyndman, R.J. "25 years of time series forecasting". *International Journal of Forecasting*, Vol. 22 (2006), pp. 443-473.
13. El-Koujok, M., Gouriveau, R. and Zerhouni N. "Error estimation of a neuro-fuzzy predictor for prognostic purpose". *Proc. 7th Ifac Int. Sym. SAFE PROCESS*, 2009.
14. Yam, RCM., Tse, P.W., Li, L. and Tu, P. "Intelligent predictive decision support system for condition-based maintenance, *International Journal of Advanced Manufacturing Technology*, Vol. 17 (2001), pp. 383-391.
15. Angelov, P. and Zhou, X. "Evolving fuzzy systems from data streams in real-time". *Proc. Int. Symp. On Evolving Fuzzy Systems* (2006), pp. 26-32.
16. Dong, M. and He, D. "A segmental hidden semi-Markov model (HSMM)-based diagnostics and prognostics framework and methodology", *Mechanical Systems and Signal Processing*, Vol. 21 (2007), pp. 2248-2266.
17. Denoeux, T. and Smets, Ph. "Classification using belief functions: the relationship between the case-based and model-based approaches", *IEEE Transactions on Systems, Man and Cybernetics B*, Vol. 36 (2006), pp. 1395-1406.
18. Smets, Ph. "Belief functions: the disjunctive rule of combination and the generalized Bayesian theorem", *International Journal of Approximate Reasoning*, Vol. 9 (1993), pp. 1-35.
19. Rombaut, M., Jarkass, I. and Denoeux, T. "State recognition in discrete dynamical systems using Petri nets and Evidence theory", *European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty* (1999), pp. 352-361.
20. Mohamed, M.A. and Gader, P. "Generalized HMM - I: Theoretical frameworks", *IEEE Tr. on Fuzzy Systems*, Vol. 8 (2000), pp. 67-81.
21. Fouque, L., Appriou, A. and Pieczynski, W. "Evidential markovian model for data fusion and unsupervised image classification", *Int. Conf. on Information Fusion*, Vol. TuB4 (2000), pp. 25-31.
22. Pieczynski, W. "Multisensor triplet Markov chains and theory of evidence", *International Journal of Approximate Reasoning*, Vol. 45 (2007), pp. 1-16.
23. Ramasso, E., Panagiotakis, C., Rombaut, M., Pellerin, D. and Tziritas, G. "Human shape-motion analysis in athletics videos for coarse to fine action/activity recognition using Transferable Belief Model", *Electronic Letters on Computer Vision and Image Analysis Journal*, Vol. 4 (2009), pp. 32-50.
24. Ramasso, E., Rombaut, M. and Pellerin, D. "State filtering and change detection using TBM conflict", *IEEE Trans. on Circuit and Systems for Video Technology*, Vol. 17 (2007), pp. 944-949.
25. Ramasso, E. "Contribution of belief functions to Hidden Markov Models with an application to fault diagnosis", *IEEE Int. Workshop on Machine Learning and Signal Processing*, Vol (2009).
26. Denoeux, T. "Conjunctive and Disjunctive Combination of Belief Functions Induced by Non Distinct Bodies of Evidence", *Artificial Intelligence*, Vol. 172 (2008), pp. 234-264.
27. Delmotte, F. and Smets, Ph. "Target identification based on the Transferable Belief Model interpretation of Dempster-Shafer model", *IEEE Transactions on Systems, Man and Cybernetics A*, Vol. 34 (2004), pp. 457-471.
28. Ben Yaghlane, B. and Mellouli, K. "Inference in directed evidential networks based on the Transferable Belief Model", *International Journal of Approximate Reasoning*, Vol. 48 (2008), pp. 399-418.
29. Côme, E., Oukhellou, L., Denoeux, T. and P. Aknin. "Learning from partially supervised data using mixture models and belief functions", *Pattern Recognition*, Vol. 42 (2009), pp. 334-348.
30. Saxena, A., Goebel, K., Simon, D. and Eklund, N. "Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation", *IEEE Prognostics and Health Management* (2008), Denver CO, USA.

Biography

Dr. Emmanuel Ramasso earned both his engineer degree in Automatic Control and Computing and his M.Sc. in Computer Science from Polytech'Savoie in 2004 (Annecy, France). His Ph.D. obtained in 2007 from the University of Grenoble (France) was focused on signal and image processing and analysis and on computer vision. He joined the national high school of mechanics and microtechniques of Besançon (ENSMM) as an Associate Professor in 2008. His current research is focused on prognostics and health management using methods based on artificial intelligence and in particular evidence theory and graphical models.

Dr. Rafael Gouriveau received his engineer degree from National Engineering School of Tarbes (ENIT) in 1999. He then got his MS (2000) and his Ph.D. in Industrial Systems in 2003, both from the Toulouse National Polytechnic Institute (INPT). During his PhD, he worked in the field of risk management and dependability analysis. In September 2005, he joined the national high school of mechanics and microtechniques of Besançon (ENSMM) as Associate Professor. His main teaching activities are concerned with production, maintenance, manufacturing, and informatics domains. Nowadays, his research interests concern the development of industrial prognostics systems by using neuro-fuzzy methods, and the investigation of reliability modeling by using possibility theory.