HAL
open science

Prognostic by classification of predictions combining similarity-based estimation and belief functions

Emmanuel Ramasso, Michèle Rombaut, Noureddine Zerhouni

To cite this version:

Emmanuel Ramasso, Michèle Rombaut, Noureddine Zerhouni. Prognostic by classification of predictions combining similarity-based estimation and belief functions. T. Denoeux & M.H. Masson. Belief Functions : Theory and Applications, AISC 164. Proceedings of the 2nd International Conference on Belif Functions. Compiègne - 9-11 May 2012, 164, Springer-Verlag Berlin Heidelberg 2012, pp.61-68, 2012, Advances in Intelligent and Soft Computing. Volume 164 2012, 978-3-642-29460-0. 10.1007/978-3-642-29461-77 . hal-00719583

HAL Id: hal-00719583

https://hal.science/hal-00719583v1

Submitted on 20 Jul 2012

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.

Prognostic by classification of predictions combining similarity-based estimation and belief functions

Emmanuel Ramasso and Michèle Rombaut and Noureddine Zerhouni

Abstract Forecasting the future states of a complex system is of paramount importance in many industrial applications covered in the community of Prognostics and Health Management (PHM). Practically, states can be either continuous (the value of a signal) or discrete (functioning modes). For each case, specific techniques exist. In this paper, we propose an approach called EVIPRO-KNN based on case-based reasoning and belief functions that jointly estimates the future values of the continuous signal and of the future discrete modes. A real datasets is used in order to assess the performance in estimating future break-down of a real system where the combination of both strategies provide the best prediction accuracies, up to 90%.

1 Introduction

Forecasting the future states of a complex system is a complicated task that arised in many industrial applications covered in the community of Prognostics and Health Management (PHM) such as locomotive's health prediction [1], analysis of fleet of vehicles [2] and turbofan engine monitoring [3]. Continuous states generally represent the value of a signal (an observation or a feature) and their prediction can be made by Kalman-like procedures or by neural networks [4, 5], Discrete states generally depict functioning modes reflecting the current degradation and its prediction can be performed by state machines such as Hidden Markov Models [6]. In both cases, data-driven prognostics generally involves a training procedure where

E. Ramasso and N. Zerhouni
FEMTO-ST Institute, UMR CNRS 6174 - UFC / ENSMM / UTBM, Automatic Control and Micro-Mechatronic Systems Department, 24 rue Alain Savary, F-25000 Besançon, France, e-mail: emmanuel.ramasso@femto-st.fr and e-mail: noureddine.zerhouni@ens2m.fr

M. Rombaut
GIPSA-lab, UMR CNRS 5216 - UJF, Signal and Images Department, 38000 Grenoble, France e-mail: michele.rombaut@gipsa-lab.inpg.fr

statistical models of the degradation are built. To cope with the problem of lack of knowledge in PHM, case-based reasoning (CBR) was proposed as an alternative to perform prognostics. For example, the method described in [3] demonstrated better performance than neural network for continuous state prediction in a turbofan engine. For that, historical instances of the system - with condition data and known failure time - were used to create a library of degradation models. Then, for a test instance of the same system, the similarity between it and the degradation models was evaluated generating a set of *Remaining Useful Life* (RUL) estimates which were finally aggregated by a density estimation method. The main problem with the approach described in [3] is the number of parameters that has to be estimated in order to apply it. Moreover, several parts of the algorithm relied on statistical learning procedures requiring large amount of data.

In this paper, we propose an algorithm called EVIPRO-KNN that requires a training dataset composed of trajectories (historical information) plus uncertain knowledge about the possible states and has the following characteristics:

EVIPRO-KNN is a new prognostics approach based on belief functions: A trajectory similarity-based approach based on belief functions is proposed for prognostics. Belief functions were justly proposed to cope with lack of data in data representation, combination and decision-making [7, 8, 9].

EVIPRO-KNN takes into account partial labelling on states: In some applications, the training dataset is composed of continuous trajectories and of a set of labels reflecting the current system state. If these labels are known only partially, then belief functions can be used [10].

EVIPRO-KNN manages trajectories with different temporal length: The weighted sum of trajectories used to compute the prediction of observations requires trajectories with the same length, that is generally false in most of applications. We described two approaches to solve it.

EVIPRO-KNN is able to predict jointly continuous and discrete states: The prediction of the future sequence of states is performed jointly with the prediction of continuous observations. These sequences allow the user to have access to the online segmentation of the current observed data and generate accurate estimate of the Remaining Useful Life (RUL) of the system. As far as we know, the joint prediction of discrete states and of continuous observations was not considered jointly in PHM applications nor in CBR-based prediction.

2 Background

At each time t, an observation vector X_t can be extracted from the observed system. This system can be in one of the possible discrete states ω belonging to a set of S exhaustive and exclusive states $\Omega = \{\omega_1, \ldots, \omega_S\}$. The states can be imprecise and uncertain due to *aleatory uncertainty* induced by the variability in observations and to *epistemic uncertainty* induced by lack of knowledge. For that, we describe the knowledge of states at time t by a belief function [7, 8, 9].

The basis in the theory of belief functions is the basic belief assignment (BBA) defined by: $m_t : 2^{\Omega} \rightarrow [0,1]$, $S \mapsto m_t(S)$, with $\sum_{A \subseteq \Omega} m_t(A) = 1$. The belief mass $m_t(A)$ represents the uncertainty (since $m_t(A) \in [0,1]$) and imprecision (since A is a subset with cardinality $|A| \geq 1$) about the possible state of the system at t. Subset A is composed of unions of singletons ($\omega \in \Omega$) and thus represents explicitly the *doubt* concerning the value of the state.

The training dataset used in EVIPRO-KNN is denoted $\mathscr{L} = \{T_i\}_{i=1}^{N}$ and is composed of N trajectories T_i defined by both a sequence of Q-dimensional observation vectors $X_t \in \Re^Q$ and their associated states $T_i = \{(X_t^i, m_t^i)\}_{t=t_i}^{t_i+|T_i|}$. The i-th (continuous) trajectory begins at time t_i and finishes at time $t_i + |T_i|$ where $|T_i|$ is the length of T_i. With each trajectory T_i is associated a set of blocks $\mathbb{B}_i$ where each block B_i^j in this set corresponds to a sub-trajectory of length W: $B_j^i = \{(X_t^i, m_t^i)\}_{t=c_j}^{c_j+W}$, where $c_j \in [t_i, (t_i + |T_i| - W)]$ is the starting time of the j-th block. The number of blocks (and the range of index j) in the i-th trajectory depends on the length of the latter.

In some applications, the training dataset is composed of features and of a set of labels reflecting the current system's state. If the labels are known only partially, then belief functions can be used [10]. The state can thus be known with uncertainty and imprecision and can be described by a belief mass denoted $m_t^i, \forall i = 1 \ldots N$ and defined on the set of states Ω.

3 EVIPRO-KNN algorithm

Let now consider that a block of data $Y_t \in \Re^Q$ of length W is available (obtained from sensors located on the system). Given the training dataset and this observation, the goal is to predict an observation trajectory $\hat{T}_t = \{(\hat{X}_{t'}, \hat{m}_{t'})\}_{t'=t}^{t+H}$ where H is an horizon of prediction. The value of H will be set automatically as shown in the sequel.

Step 1 - K-best trajectories determination: In this step, the K nearest trajectories to observations Y_t are determined. For that, all trajectories in the training dataset $\mathscr{L}$ are scanned. For each trajectory T_i, the nearest block $B_{j^*}^i \in \mathbb{B}_i$ to the observation block Y_t is found. Index j^* of the best block $B_{j^*}^i$ in the i-th trajectory is given by: $j^* = \text{argmin}_{\,j, B_j^i \in \mathbb{B}_i} \mathscr{D}(Y_t, B_j^i)$. Note that all distances $\mathscr{D}$ are measured using the Euclidean one as in most of the KNN-based algorithms [3]. Let denote c_i^* the starting time of best block $B_{j^*}^i$ in the i-th trajectory. When the best block in each trajectory has been found, all best blocks are sorted by ascending order according to their distance: $\mathscr{D}_{j^*}^i \equiv \mathscr{D}(Y_t, B_{j^*}^i)$. Let $\mathscr{D}_{j^*}^{(i)}$ denote one element of this partial ordering with $\mathscr{D}_{j^*}^{(1)} \leq \mathscr{D}_{j^*}^{(2)} \leq \ldots \mathscr{D}_{j^*}^{(i)} \leq \ldots \mathscr{D}_{j^*}^{(N)}$. Finally, the K best trajectories $T_k, k = 1 \ldots K$ are simply the ones associated to the K best and sorted blocks: $\mathscr{D}_{j^*}^{(1)} \leq \mathscr{D}_{j^*}^{(2)} \leq \ldots \mathscr{D}_{j^*}^{(k)} \leq \ldots \mathscr{D}_{j^*}^{(K)}$. The K selected trajectories $T_k = \{(X_t^k, m_t^k)\}_{t=c_k}^{|T_k|}, k = 1 \ldots K$ are composed of both a set of features $X_t \in \Re^Q$ and

knowledge m_t about the state. The next steps of the algorithm consists in aggregating trajectories $T_k, k = 1 \dots K$ where two problems arised: 1) How to aggregate the features $\{X_t^k\}_{t=c_k}^{|T_k|}, k = 1 \dots K$ in order to obtain a predicted set of features $\hat{X}_t$ (Step 2)?, and 2) How to aggregate the knowledge about states $\{m_t^k\}_{t=c_k}^{|T_k|}, k = 1 \dots K$ in order to obtain a predicted knowledge $\hat{m}_t$ (Step 3)?

Step 2 - Predicted observation trajectory: A simple and usual way to define a predicted observation trajectory $\hat{X}_t$ linked to the observation block Y_t is to compute the weighted average of the K sets of features:

$$\hat{X}_{t+h} = \sum_{k=1}^{K} F^k \cdot X_l^k, l = c_k \dots |T_k|, h = 1 \dots \mathscr{P} \tag{1}$$

where $\mathscr{P} = |T_k| - c_k + 1$ defines the set of instants of prediction. The normalized weights F^k are obtained by the softmax function of the sorted distances:

$$F^k = \frac{\exp\left(-\mathscr{D}_{j^*}^{(k)}\right)}{\sum_{k'=1}^{K} \exp\left(-\mathscr{D}_{j^*}^{(k')}\right)}, k = 1 \dots K \tag{2}$$

Equations 1 and 2 are directly used if the length of trajectories $T_k, k = 1 \dots K$ are the same. If it is not the case (and generally it is not), one can use a strategies consisting in selecting an horizon of prediction equal to the length of the smallest trajectory. For that, first, the trajectory with the smallest size is found: $H_t = \min_{k=1}^{K} |T_k|$, where H_t can be seen as the horizon of prediction at time t. Then, for all trajectories, only samples from c_k to H_t are kept. After removal of samples located beyond H_t, Equations 1 and 2 can be directly used:

$$\hat{X}_{t+h}^{CS} = \sum_{k=1}^{K} F^k \cdot X_l^k, l = c_k \dots H_t, h = 1 \dots H_t \tag{3}$$

where CS stands for "Cautious Strategy" and X_h^k is the value of features in trajectory T_k taken at time h. The value of F^k is given by Eq. 2. The main advantage of this strategy is simplicity and efficiency since the horizon is gene rally shortened (to the smallest trajectory) and thus providing more reliable predictions. The main drawback is that the horizon of prediction is justly made shorter and therefore reducing forecasting capability.

At the end of step 2, the prediction of observation trajectory $\hat{X}_t$ is known according to the observation block Y_t and to the training dataset $\mathscr{L}$. Note that exponential smoothing using past prediction ($\hat{X}_{t-1}$) can be performed to improve temporal consistency [1] (not used in this paper).

Step 3 - Predicted sequence of states: It is concerned by the prediction of future states. Two strategies are proposed: 1) Classification of predictions (CPS) and 2) Direct projection of future state sequence (DPS).

Classification of predictions strategy (CPS): This strategy consists in classifying the predicted observations given by step 2 into states. It requires the training of classifiers able to discriminate the different states. For the sake of simplicity, we consider the multiclass classifier called Evidential K-nearest neighbours (EvKNN) [11]. This classifier is able to generate a belief mass on the possible states in Ω given an observation. The main feature of this classifier is the possibility to manage belief functions m_t^i provided in the training dataset $\mathscr{L}$ (partially-supervised classification).

Given both a block of data $\hat{B}_h$ centered around the predicted observation $\hat{X}_{t+h}$ and the training dataset $\mathscr{L}$, the classifier provides a belief mass on the possible states:

$$m_{t+h}^{CPS} \leftarrow \text{EvKNN classifier}(\mathscr{L}, \hat{B}_h) \tag{4}$$

From this belief mass, a *hard* decision can be made to estimate the state of the current block by using the pignistic transform [9] which computes a probability distribution (suited for decision-making) from the belief mass m_{t+h}^{CPS}. Repeating this process on blocks composing the predicted observation $\hat{X}_t$, one simply obtains a sequence of states.

Direct projection of future state sequence (DPS): In order to avoid the dependency between state sequence prediction to observation prediction as in CPS, we propose to exploit another strategy that is the *direct projection of future state sequence*. This second strategy draws benefits directly from the training dataset. The main idea is to apply a similar reasoning as for features X_t but now for belief mass m_t. To go further in details, let consider the set of belief masses for the K nearest neighbours, i.e. $m_t^k, k = 1 \dots K, t = c_k \dots |T_k|$. These K belief masses can be considered as coming from distinct pieces of evidence so that the conjunctive rule of combination $\oplus$ can be used:

$$\hat{m}_{t+h}^{DPS} = \oplus_{k=1}^{K} m_l^k, l = c_k \dots |T_k|, h = 1 \dots \mathscr{P} \tag{5}$$

where *DPS* stands for "direct projection strategy" and $\mathscr{P} = |T_k| - c_k + 1$. To decrease the amount of conflict during the fusion process, we propose to use a *discounting* using the weights estimated in the KNN. The highest the weight, the less the discount, meaning that the related BBA is trusted. Once the BBAs have been discounted, the estimated belief mass at time t in DPS is given by Eq. 5.

Step 4 - Remaining Useful Life estimation: *CPS and DPS fusion:* To draw benefits from both CPS and DPS approaches, the BBAs m_{t+h}^{CPS} (Eq. 4) and m_{t+h}^{DPS} (Eq. 5) are combined and the resulting BBA is converted into a probability distribution from which a decision can be made [12]. Dempster's rule is not adapted for the fusion of CPS and DPS's BBAs because m_{t+h}^{CPS} and m_{t+h}^{DPS} can not be considered as coming from distinct bodies of evidence. Indeed: 1) CPS is a classification of predictions resulting from the weighted combination of continuous predictions, and 2) DPS generates belief masses discounted by the weights, and therefore, both approaches depend on the weights. Moreover, both rely on the BBAs in the training dataset $\mathscr{L}$. Thus, the fusion may be performed using the cautious rule [13]:

$$\hat{m}_{t+h} = m_{t+h}^{CPS} \circledwedge m_{t+h}^{DPS} \tag{6}$$

from which a decison concerning the state at time $t+h$ can be made and the result is the estimation of a sequence of states $\hat{\omega}_{t+h}$. Note that the neutral element is not always the vacuous BBA [13], except for separable BBAs as the ones used in evidential KNN exploited in CPS. In this case, if BBAs in the training dataset are vacuous, then the fusion equals CPS.

RUL estimates: Let consiser this sequence of states but also all previous predicted sequences. Since each sequence is composed of possible transitions between some states q and r, the set of time instants of transitions between both states is: $I_{q\to r} = \{t : \hat{\omega}_{t-1} = q \text{ and } \hat{\omega}_t = r\}$. To estimate the Remaining Useful Life (RUL) of the system, it is sufficient to determine the location of the critical transition from state $q =$ "degrading state" to state $r = q+1 =$ "fault state":

$$\text{transition } q \to r \text{ critical } \Rightarrow RUL = \mu_{q,r} - t \tag{7}$$

where $\mu_{q,r}$ is the estimated time from t to the transition between the degrading state q and the faulty state r that can be computed by a median. It can be associated to a dispersion $\sigma_{q\to r}$ that we computed using the interquartile range:

$$\begin{aligned} \mu_{q\to r} &= \text{median } (I_{q\to r}) \\ \sigma_{q\to r} &= Q_3 - Q_1 \end{aligned} \tag{8}$$

where Q_i is the i-th quartile and $n_I = |I_{q\to r}|$ is the number of elements in the set of time instants of transition $I_{q\to r}$.

Therefore, both methods for sequence prediction, CPS (classification) and DPS (direct projection), assume that each trajectory in the training dataset is made of *at least* two states, say "normal state" and "abnormal state", and knowledge on these states can be uncertain and imprecise and represented by belief functions.

4 First results, Conclusion and further work

Illustration : We considered the PHM'08 challenge data [14] that we segmented into four states (available at `http://www.femto-st.fr/~emmanuel.ramasso/PEPS_INSIS_2011_PHM_by_belief_functions.html`). The first features and the segmentation are depicted in Fig. 1 which underlines the difficulty of using a statistical approach based on durations for degradation modelling [15].

Figure 2 depicts the sensitivity of the EVIPRO-KNN algorithm with respect to the parameters K (number of neighbours) and W (window's size). With $K = 3$ and $W = 30$, one can expect results close to 90% on the considered dataset. The prediction was considered as correct when falling in the interval $[-10, +13]$ around the ground truth, and the beginning of the prediction was taken as the time-instant corresponding to 75% of the length of the analysed trajectory (e.g. if the trajectory's length is equal to 240 then the starting time of the prediction was set to 180).

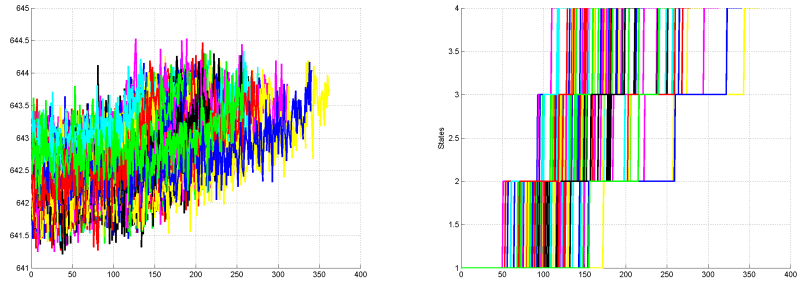

Fig. 1 Left: Evolution of the first feature for all trajectories in the training dataset, and right: the state sequences after decision-making based on the belief masses.

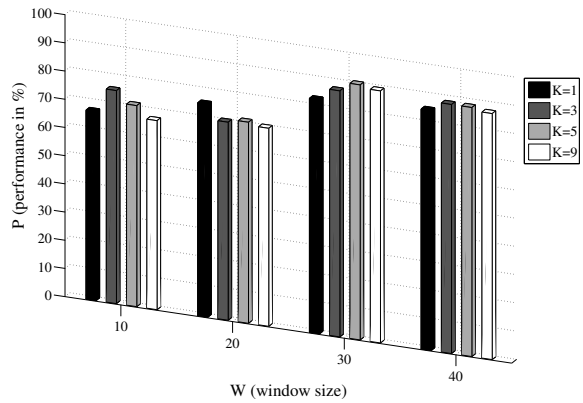

Fig. 2 Left: Sensitivity to W and K.

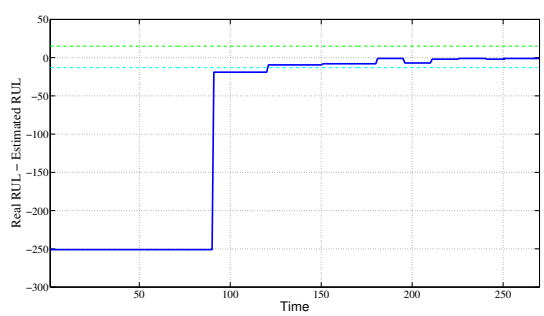

Fig. 3 The prediction appears in continuous bold line, while the real value is in dashed line.

Figure 3 illustrates the evolution of the differences at each time-step between the estimated RUL and the real RUL for $W = 30$ and $K = 3$ where a convergence to the real value is observed as expected: a good estimate of the RUL (in interval $[-10, 13]$) is obtained at $t = 90$, so 180 time-units in advance.

Conclusion and further work : EVIPRO-KNN is an online algorithm for prognostics and health detection working as case-based reasoning but managing uncertain knowledge about the states that could be provided as belief functions in the training dataset. EVIPRO-KNN can predict sequence of continuous observations jointly with discrete states enabling the user to have access to the online segmentation of the current observed data and of predictions which is then used to estimate the RUL.

Acknowledgment: This work is supported by a PEPS-INSIS-2011 grant from the French National Center for Scientific Research (CNRS) under the administrative authority of France's Ministry of Research.

References

1. P. Bonissone, A. Varma, and K. Aggour, "A fuzzy instance-based model for predicting expected life: A locomotive application," in *IEEE Int. Conf. on Computational Intelligence for Measurement Systems and Applications*, 2005, pp. 20–25.
2. A. Saxena, B. Wu, and G. Vachtsevanos, "Integrated diagnosis and prognosis architecture for fleet vehicles using dynamic case-based reasoning," in *Autotestcon*, 2005, pp. 96–102.
3. T. Wang, "Trajectory similarity based prediction for remaining useful life estimation," Ph.D. dissertation, University of Cincinnati, 2010.
4. C. Bishop, *Pattern Recognition and Machine Learning*. Springer, August 2006.
5. K. P. Murphy, "Dynamic Bayesian networks: Representation, inference and learning," Ph.D. dissertation, UC Berkeley, 2002.
6. E. Ramasso and R. Gouriveau, "Prognostics in switching systems: Evidential Markovian classification of real-time neuro-fuzzy predictions," in *IEEE Int. Conf. on Prognostics and System Health Management*, Macau, China, 2010, pp. 1–10.
7. A. Dempster, "Upper and lower probabilities induced by multiple valued mappings," *Annals of Mathematical Statistics*, vol. 38, pp. 325–339, 1967.
8. G. Shafer, *A mathematical theory of Evidence*. Princeton University Press, Princeton, NJ, 1976.
9. P. Smets and R. Kennes, "The Transferable Belief Model," *Artificial Intelligence*, vol. 66, no. 2, pp. 191–234, 1994.
10. E. Come, L. Oukhellou, T. Denoeux, and P. Aknin, "Learning from partially supervised data using mixture models and belief functions," *Pattern Recognition*, vol. 42, no. 3, pp. 334–348, 2009.
11. T. Denoeux, "A k-nearest neighbor classification rule based on Dempster-Shafer theory," *IEEE Trans. on Systems, Man and Cybernetics*, vol. 5, pp. 804–813, 1995.
12. P. Smets, "Decision making in the TBM: The necessity of the pignistic transformation," *Int. Jour. of Approximate Reasoning*, vol. 38, pp. 133–147, 2005.
13. T. Denoeux, "Conjunctive and disjunctive combination of belief functions induced by non distinct bodies of evidence," *Artificial Intelligence*, vol. 172, pp. 234–264, 2008.
14. A. Saxena, K. Goebel, D. Simon, and N. Eklund, "Damage propagation modeling for aircraft engine run-to-failure simulation," in *Int. Conf. on Prognostics and Health Management*, Denver, CO, USA, 2008, pp. 1–9.
15. M. Dong and D. He, "A segmental hidden semi-markov model (HSMM)-based diagnostics and prognostics framework and methodology," *Mechanical Systems and Signal Processing*, vol. 21, pp. 2248–2266, 2007.