



HAL
open science

Time-Sliced temporal evidential networks: the case of evidential HMM with application to dynamical system analysis.

Lisa Serir, Emmanuel Ramasso, Nouredine Zerhouni

► To cite this version:

Lisa Serir, Emmanuel Ramasso, Nouredine Zerhouni. Time-Sliced temporal evidential networks: the case of evidential HMM with application to dynamical system analysis.. IEEE International Conference on Prognostics and Health Management, PHM'11., Jun 2011, Denver - Colorado, United States. 10 p. hal-00605846

HAL Id: hal-00605846

<https://hal.science/hal-00605846>

Submitted on 4 Jul 2011

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.

Time-Sliced Temporal Evidential Networks: the case of Evidential HMM with application to dynamical system analysis

Lisa Serir, Emmanuel Ramasso and Noureddine Zerhouni
FEMTO-ST Institute, UMR CNRS 6174 - UFC / ENSMM / UTBM,
Automatic Control and Micro-Mechatronic Systems Department, 25000, Besançon, France
(lisa.serir, emmanuel.ramasso, noureddine.zerhouni)@femto-st.fr

Abstract—Diagnostics and prognostics of health states are important activities in the maintenance process strategy of dynamical systems. Many approaches have been developed for this purpose and we particularly focus on data-driven methods which are increasingly applied due to the availability of various cheap sensors. Most data-driven methods proposed in the literature rely on probability density estimation. However, when the training data are limited, the estimated parameters are no longer reliable. This is particularly true for data in faulty states which are generally expensive and difficult to obtain. In order to solve this problem, we propose to use the theory of belief functions as described by Dempster, Shafer (Theory of Evidence) and Smets (Transferable Belief Model). A few methods based on belief functions have been proposed for diagnostics and prognostics of dynamical systems. Among these methods, Evidential Hidden Markov Models (EvHMM) seems promising and extends usual HMM to belief functions. Inference tools in EvHMM have already been developed, but parameter training has not fully been considered until now or only with strong assumptions. In this paper, we propose to complete the generalization of HMM to belief functions with a method for automatic parameter training. The generalization of this training procedure to more general Time-Sliced Temporal Evidential Network (TSTEN) is discussed paving the way for a further generalization of Dynamic Bayesian Network to belief functions with potential applications to diagnostics and prognostics. An application to time series classification is proposed.

I. INTRODUCTION

Fault diagnosis and prognosis are two key elements of effective Condition-Based Maintenance. Today's concept of machine diagnosis comprises the automated detection and classification of faults, whereas machine prognosis is the automated estimation of how soon and likely a failure will occur [1]. One approach to prognosis is to use a physics-based model of the system. Another approach is the expert system approach. A third approach, on which we focus our interest, is the data-driven approach, which uses historical data to automatically learn a model of the system's behavior. Algorithms that use the data-driven approach to prognosis learn models directly from the data, rather than using a model based on human expertise. Artificial intelligence techniques, such as expert system, neural network, fuzzy logic and genetic algorithm, have been employed to assist the diagnostic task to correctly interpret the fault data. Neural network technique has gained popularity over other techniques as it is efficient

in discovering similarities among large bodies of data.

Another common solution is to use state sequence representation, learning and inference algorithms. A state generally represents a stationarity or a functioning mode in the data and describes the dynamical system at a given time, while transitions represent its dynamics. In practical applications, states are hidden and only features are observable. An additional modelling step is thus required to relate features to states. To manage imperfections, the modelling should cope with uncertainty using probability theory, possibility theory or evidence theory [2], the latter being more general.

Hidden Markov model (HMM) [3] is one commonly used method for state sequence representation and recognition. They represent an appropriate model for analyzing event and condition monitoring data together. An HMM consists of two stochastic processes: a Markov chain with a finite number of states describing an underlying mechanism and an observation process depending on the hidden state. Early applications of HMM in fault classification and diagnostics treated the real machine faulty states and the machine normal state as the hidden states of the HMM. The use of HMM in bearing fault prognosis was investigated by Zhang *et al.* [4] where one HMM was trained to recognise one type of cone-and-cup bearing faults based on the corresponding vibration data. The HMMs were also trained to estimate the fault states. The similarity between current state and failure state was used as the bearing degradation index, which was then extrapolated to estimate the time of exceeding a predetermined failure threshold. Dong and He [5] have proposed a segmental Hidden Semi-Markov Model (HsMM) which generates a segment of observations where durations are estimated from training data.

For the problem of state sequence modelling and recognition, we propose here to use the theory of belief functions as described by Dempster and Shafer's Theory of Evidence [6], [7] and Smets' Transferable Belief Model [8], [9]. Belief functions cope with some limitations of usual HMMs [10]:

- The use of belief functions allows the modelling of information on states without priors. Vacuous belief functions are used when no prior knowledge is available.
- HMMs requires probability density estimation, which can be no longer reliable when the training data are limited. This is particularly true for data in faulty states which are

expensive and difficult to obtain. Belief functions takes this lack of information into account.

- In some cases, observations provided by sensors are reliable only in some context and for some specific subset of states. The theory of belief functions allows modelling of this case easily.
- Emission models used in HMM are based only on probability. However, there are some cases where some knowledge can be generated using fuzzy or belief functions formalisms. The use of belief functions provides tools to combine information in a common framework, using specific operators for information fusion with non-distinct sources [11].

Although inference tools in EvHMM have been developed [12], parameter training has not been considered without the use of strong assumptions [13], [14]. In this paper, we first propose to complete the generalization of HMM to belief functions with a method for automatic parameter training. The generalization of this training procedure to more general Time-Sliced Temporal Evidential Network (TSTEN) is discussed, paving the way for a further generalization of Dynamic Bayesian Network (a generalization of HMM) to belief functions with potential applications to diagnostics and prognostics.

The paper is organized as follows: Section II describes HMM basics and is followed by a brief description of belief functions in Section III. EvHMM inference mechanisms are then presented. Compared to [12] where canonical weights are used, we here present the procedures using the *communality* function to fulfill the requirements described in [15], [16] for belief propagations. Section IV and parameter training are finally presented in Section V.

II. PROBABILISTIC HMM BACKGROUND

The notations and basic mechanisms of HMMs described in the tutorial of Rabiner [3] are presented in this section.

A sequence is defined by a set of N states in the set Ω_t called the *frame of discernment* with:

$$\Omega_t = \{s_1, s_2, \dots, s_N\} \quad (1)$$

and $s_i \in \Omega_t$ is the i -th state at time t . States are said hidden because only - continuous - observations $\mathbf{O}_t = [O_1 \ O_2 \ \dots \ O_F]$, $t \in \{1..T\}$ (F is the dimension) are actually measured on the dynamical system. Each observation \mathbf{O}_t generates a degree of confidence on each state s_i that is generally represented by a likelihood $P(\mathbf{O}_t|s_i)$, denoted by $b_i(\mathbf{O}_t)$. The latter are assessed by a modelling technique \mathbb{B} such as a mixture of Gaussians [3]. At each time, the transition from a state s_i (at $t-1$) to a state s_j (at t) has a probability $a_{ij} = P(s_j|s_i)$. The $N \times N$ stochastic matrix $\mathbb{A} = [a_{ij}]$, $i \in \{1..N\}$, $j \in \{1..N\}$ such as $\sum_j a_{ij} = 1$ is called *the transition matrix*. At time $t=1$, the a priori on states is represented by the initial distribution on states π_i , $i \in \{1..N\}$. An HMM $\lambda = \{\mathbb{A}, \mathbb{B}, \Pi\}$ is thus characterized by three elements: the transition matrix \mathbb{A} , the observation models \mathbb{B} (generating likelihoods $b_i(\mathbf{O}_t)$) and the initial distributions π_i .

The *forward* and *backward* propagations are the basic mechanisms of the HMM that ensure the temporal coherence and compute useful quantities for inference and learning:

- Forward / filtered estimate: $\alpha_t(s_j|\lambda) = P(\mathbf{O}_1, \mathbf{O}_2 \dots \mathbf{O}_t, s_j|\lambda) = \sum_{\forall s_i \in \Omega_{t-1}} \alpha_{t-1}(s_i) \cdot a_{ij} \cdot b_j(\mathbf{O}_t)$, used in particular for classification since the log-likelihood of a sequence of observations given a HMM λ_r is:

$$\mathcal{L}_p(\lambda_r) = \frac{1}{T} \sum_{t=1}^T \log \left(\sum_{s_j \in \Omega_t} \alpha_t(s_j|\lambda_r) \right) \quad (2)$$

and the best model is $\lambda^* = \arg\max_r \mathcal{L}_p(\lambda_r)$.

- Backward: $\beta_t(s_i) = P(\mathbf{O}_T, \mathbf{O}_{T-1} \dots \mathbf{O}_{t+1}|s_i, \lambda) = \sum_{\forall s_j \in \Omega_{t+1}} a_{ij} \cdot b_j(\mathbf{O}_{t+1}) \cdot \beta_{t+1}(s_j)$,
- γ -variable / smooth estimate: $\gamma_t(s_j) = P(s_j|\mathbf{O}_{1:T}, \lambda) \propto \alpha_t(s_j) \cdot \beta_t(s_j)$, used in particular to determine both the most probable state at t and model's parameters \mathbb{B} ,
- ξ -variable: $\xi_t(s_i, s_j) = P(s_i, s_j | \mathbf{O}_{1:T}, \lambda) \propto \alpha_t(s_i) \cdot a_{ij} \cdot b_j(\mathbf{O}_{t+1}) \cdot \beta_{t+1}(s_j)$ used to estimate the expected probabilities of a transition by:

$$a_{ij} \propto \sum_{t=1}^{T-1} \xi_t(s_i, s_j) \quad (3)$$

- *Viterbi* metric δ : computed similarly as α but replacing the sum a maximum (max-product algorithm).

III. TBM BACKGROUND

The *Transferable Belief Model* (TBM) [9] is a general framework for uncertainty representation and combination of various pieces of information without additional priors. In particular, doubt and conflict are explicitly emphasized. Typically, the former represents ignorance and the latter emphasizes the contradiction within a fusion process. This section reviews some basic notions.

A. Belief mass

The belief of an agent in subsets of the frame of discernment Ω_t can be represented by a *basic belief assignment* (BBA), also called *belief mass assignment*:

$$m^{\Omega_t} : 2^{\Omega_t} \rightarrow [0, 1], A \rightarrow m^{\Omega_t}(A) \quad (4)$$

$$\sum_{A \subseteq \Omega_t} m^{\Omega_t}(A) = 1$$

A belief mass can not only be assigned to a single state ($|A| = 1$), but also to a *subset* ($|A| > 1$) of states *without assumption concerning additivity*. This property permits the explicit modelling of doubt between states (composing A) and constitutes a fundamental difference with probability theory. Elements A for which $m^{\Omega_t}(A) > 0$ are called *focal sets*.

A normalised BBA is such as the degree of conflict is nil:

$$\begin{cases} m_*^{\Omega_t}(A) &= \frac{m^{\Omega_t}(A)}{1 - m^{\Omega_t}(\emptyset)} \\ m_*^{\Omega_t}(\emptyset) &= 0 \end{cases} \quad (5)$$

This process is called *Dempster normalisation*. Other sound redistribution rules for conflict normalisation exist [17].

B. Other belief functions

Several functions - in *one-to-one correspondance* [9] - can be computed from a BBA. Depending on the context of application, these functions are used to represent the knowledge and to make the combination rules easier to compute.

In the following, the *communality* q will be used and is defined by:

$$q^{\Omega_t}(B) = \sum_{C \supseteq B} m^{\Omega_t}(C) \quad (6)$$

The *plausibility* is also an important function:

$$pl^{\Omega_t}(B) = \sum_{C \cap B \neq \emptyset} m^{\Omega_t}(C) \quad (7)$$

In particular $pl^{\Omega_t}(\Omega_t) = 1 - m^{\Omega_t}(\emptyset)$ links the plausibility of “everything” (considered in the frame Ω_t) to the opposite of the belief mass of “nothing”. This relation is actually a bridge from likelihood maximization principle to minimum of conflict principle, also called unlikelihood [18].

C. Combination

The *combinations* of several sources of belief can be computed by four main rules [11]. In the sequel we will mainly used the *conjunctive rule of combination* (CRC, \odot):

$$q_{1 \odot 2}^{\Omega_t}(B) = q_1^{\Omega_t}(B) \cdot q_2^{\Omega_t}(B) \quad (8)$$

The CRC is a simple product in the q -space while it implies several summations and products in the m -space:

$$m_{1 \odot 2}^{\Omega_t}(B) = \sum_{C \cap D = B} m_1^{\Omega_t}(C) \cdot m_2^{\Omega_t}(D) \quad (9)$$

If the CRC is used and if the obtained belief mass (using the inverse transform from q to m) is normalised (Eq. 5) then we obtain *Dempster rule of combination* [19].

D. Conditional BBA

A *conditional BBA* can be used to represent the state of knowledge. For example $m^{\Omega_t | \Omega_{t-1}}(\cdot | S_i)$ is a BBA defined on Ω_t conditionally to subset $S_i \subseteq \Omega_{t-1}$. Conditioning is also a process which consists in combining conjunctively (using the CRC, Eq. 9) a BBA m^{Ω_t} with a *categorical BBA* defined by $m_2^{\Omega_t}(A) = 1$, i.e. the latter has the particularity to be nil except for one single element $A \subset \Omega_t$ (“ A ” is the condition).

Therefore, after conditioning, the focal sets - elements for which the mass is greater than zero - have a non-empty intersection with A which becomes the restricted frame of discernment. The key point is that the mass on the empty set increases and is given by the following definition.

Definition 1: After conditioning of a BBA m^{Ω_t} by a subset A , the mass on the empty set is equal to:

$$1 - m^{\Omega_t}(\emptyset | A) = pl^{\Omega_t}(A) \quad (10)$$

and:

- when $|A| = 1$, only two focal sets remain: \emptyset and A ,
- when $|A| > 1$, there may be more than two focal sets, including \emptyset .

E. Generalized Bayes theorem

The Generalized Bayesian Theorem (GBT) is an extension of Bayesian Theorem in the TBM framework [20]. The GBT alleviates the problem of priors, since belief functions make the representation of total ignorance possible. Expression of the GBT varies according to the type of belief functions. For example, in the sequel we will use the $q - pl$ form:

$$q_a^{\Omega_t | \Omega_{t+1}}(S_i | S_j) = \prod_{s_i \in S_i} pl_a^{\Omega_{t+1} | \Omega_t}(S_j | s_i) \quad (11)$$

F. Law of total plausibility

The law of total plausibility will be used for projection and is given by:

$$pl^{\Omega_{t+1}}(S_j) = \sum_{S_i \subseteq \Omega_t} pl^{\Omega_{t+1} | \Omega_t}(S_j | S_i) \cdot m^{\Omega_t}(S_i) \quad (12)$$

This equation remains valid with m or q instead of pl [20].

G. Decision making

After combination of multiple sources of belief, a resulting BBA m^{Ω_t} is obtained. *Decision making* in the TBM framework consists in the choice of the best hypothesis from the *pignistic probability* distribution [9] or the plausibility transform [21].

IV. EVIDENTIAL HMM

Compared to [12] where canonical weights are used, we here present the procedures using the *communality* function in order to fulfill the required axioms described in [15], [16] for belief propagations [22]. Canonical weights can be used but only with the product operator which is equivalent to use the communalities.

A. Classification in EvHMM

Given an observation sequence $\mathbf{O}_{1:T}$ (length T) and a set of EvHMMs $\lambda_{1..R}$, the goal of the classification process is to choose the EvHMM that best fits observations. The classification criterion is given by [12]:

$$\mathcal{L}_e(\lambda_r) = \frac{1}{T} \sum_{t=1}^T \log pl_{\alpha}^{\Omega_t}(\Omega_t | \lambda_r) \quad (13a)$$

$$\lambda^* = \underset{r}{\operatorname{argmax}} \mathcal{L}_e(\lambda_r) \quad (13b)$$

and is a generalization of HMM criterion to belief functions. This criterion relies on the opposite of the conflict value (which is a plausibility) computed in the *forward* propagation. Therefore, it has a natural interpretation: the less the conflict, the more likely is the model λ . The EvHMM classification process is presented in Algorithm 1.

The computation of the evidential forward variable at t satisfies a “prediction – update” mechanism. The prediction of proposition S_j is first computed using the law of total plausibility (Eq. 12). The prediction is then combined with observations in order to update belief on states:

$$q_{\alpha}^{\Omega_t}(S_j) = \sum_{S_i \subseteq \Omega_{t-1}} m_{\alpha}^{\Omega_{t-1}}(S_i) \cdot q_a^{\Omega_t}(S_j | S_i) \cdot q_b^{\Omega_t}(S_j) \quad (14)$$

The prior on states at $t = 1$ can be “vacuous” and this state of knowledge is easily represented by $q_{\alpha}^{\Omega_1}(S_i) = 1, \forall S_i \subseteq \Omega_1$ meaning total ignorance. This equation generalizes the probabilistic forward variable when transitions, observations, and prior are all Bayesian (i.e. the belief mass is nil except for singleton hypotheses) and with Dempster normalisation (Section III-C). In the worst case, the evidential version has a complexity is equal to $T \cdot N \cdot 2^N$, with N the number of states and T the sequence length (vs $T \cdot N^2$ in HMM). In this equation, $q_a^{\Omega_t}(\cdot|S_i)$ are the conditional communalities defined on Ω_t conditionally to subsets $S_i \subseteq \Omega_{t-1}$ and representing transitions between subsets at $t-1$ and t , and $q_b^{\Omega_t} \equiv q_b^{\Omega_t}(\cdot|\mathbf{O}_t)$ are communalities obtained from current observations.

Remark 1: The conflict resulting from the conjunctive combination between observations and prediction must be cancelled out (by normalisation) at each iteration because it is absorptive by the conjunctive rule (embedded within the forward propagation). As in probabilistic HMM, where the normalisation process consists in redistributing uniformly $1 - \sum_j \alpha_t(j)$ to each state at t , belief function framework also provides a similar process (Eq. 5) [17] called Dempster normalisation.

Algorithm 1 Algorithm EvHMM Classification

Require: model $\lambda = \{\mathbb{A}_e, \mathbb{B}_e, \Pi_e\}$ and s {Evidential transitions, evidential models generating masses from observations, and evidential prior on the first state. Also s if a generalized cautious rule is used (Sect. III-C).}

Ensure: Evidential likelihood \mathcal{L}_e {Used in classification}

Ensure: Evidential $\alpha - \beta - \gamma - \xi$ {Filtering/smoothing}

- 1: **for all** instants $t = 1$ to T **do**
 - 2: $\alpha =$ Forward propagation {Eq. 14}
 - 3: $\alpha_* =$ Normalise α {Eq. 5}
 - 4: **end for**
 - 5: Compute \mathcal{L}_e {Eq. 13}
 - 6: **for all** instants $t = T$ to **1 do**
 - 7: $[\beta \ \gamma \ \xi] =$ Backward propagation {Eq. 15 and 16. If ξ is required then use Eq. 22 of [23] or see Section V-B for an approximation}
 - 8: **end for**
-

B. Hidden state sequence recognition using smoothing

As in probabilistic HMM, one can use the γ -variable (posterior probability of states given observations) for state recognition. In EvHMM, a similar process can be used. For that, the evidential backward variable has to be computed:

$$q_{\beta}^{\Omega_t}(S_i) = \sum_{S_j \subseteq \Omega_{t+1}} m_{\beta \ominus b}^{\Omega_{t+1}}(S_j) \cdot q_a^{\Omega_t}(S_i|S_j) \quad (15)$$

which follows also a prediction-fusion scheme. The BBA $m_{\beta}^{\Omega_{t+1}}$ is the backward variable defined on subsets of states at $t + 1$, $pl_a^{\Omega_{t+1}}(\cdot|S_i)$ is the set of conditional plausibilities representing transitions from subsets of states at t and $t + 1$, and $m_b^{\Omega_{t+1}} \equiv m_b^{\Omega_{t+1}}(\cdot|\mathbf{O}_{t+1})$ is the observed BBA. The

BBA $m_{\beta \ominus b}^{\Omega_{t+1}}$ on Ω_{t+1} is first computed by the conjunctive combination of the available BBAs at $t + 1$ ($m_{\beta}^{\Omega_{t+1}}$ and $m_b^{\Omega_{t+1}}$). Then a projection towards Ω_t is carried out. For this, given transitions in the pl -space, the posterior conditional communalities $q_a^{\Omega_t}(\cdot|S_j)$ are obtained from these plausibilities by applying the GBT (Eq. 11). Note that unknown prior at T is modelled by a vacuous BBA ($q_{\beta}^{\Omega_T}(S_i) = 1, \forall S_i \subseteq \Omega_T$) and Eq. 15 reduces to the probabilistic case if all BBAs are Bayesian and if Dempster rule is used.

The evidential γ variable is then computed by the conjunctive combination of both forward and backward variables:

$$q_{\gamma}^{\Omega_t}(S_j) = q_{\alpha}^{\Omega_t}(S_j) \cdot q_{\beta}^{\Omega_t}(S_j) \quad (16)$$

The γ -variable is then used for the *detection of the best state s_t^** at a given time t . This state can be found by maximizing the pignistic probability [9] or the plausibility [21]. The γ -variable is not always well-suited for state recognition because the decision is too “local”. In [12], a Viterbi-like procedure was proposed to find the best sequence of hidden states. This procedure is not optimal but draw benefits of belief functions. In this paper, we propose an optimal version.

C. Hidden state sequence recognition using a Viterbi procedure

The problem is to find the best sequence of states. A solution is to maximize the state sequence plausibility. For that, let consider a sequence of singletons states $\{s^1, s^2, \dots, s^T\}$ with $|s^t| = 1$. At $t = 1$, the available information are: the prior $m_{\pi}^{\Omega_1}$ and observations $m_b^{\Omega_1}$. They have to be combined by the CRC rule and then conditioned on s^1 . Since conditioning is made by a singleton, the BBA has two focal sets: \emptyset and s^1 . Therefore, at $t = 1$, the combination leads to $pl_{\delta}(s^1) = pl_{\pi}(s^1) \cdot pl_b(s^1)$.

Then at time $t = 2$, one observes that the true state is s^2 . Given the transition BBA $m_a^{\Omega_2|\Omega_1}[s^1]$, conditioning by s^2 gives $m_a^{\Omega_2|\Omega_1}[s^1, s^2](\emptyset)$ and $m_a^{\Omega_2|\Omega_1}[s^1, s^2](s^2)$, where the latter is equal to $pl_a^{\Omega_2|\Omega_1}[s^1](s^2)$ (after conditioning since s^2 is a singleton). Observations are also conditioned by s^2 and the combination of all these information provides: $pl_{\delta}(s^1, s^2) = pl_{\pi}(s^1) \cdot pl_b(s^1) \cdot pl_a^{\Omega_2|\Omega_1}[s^1](s^2) \cdot pl_b(s^2)$.

Applying the same reasoning for the whole sequence, we have the following definition.

Definition 2: In EvHMM, the plausibility of a sequence of singleton states $S = \{s^1, s^2, \dots, s^T\}$, $|s^t| = 1$ is given by:

$$pl_{\delta}(S) = pl_{\pi}(s^1) \cdot \prod_{t=2}^T pl_a^{\Omega_t|\Omega_{t-1}}[s^{t-1}](s^t) \cdot \prod_{t=1}^T pl_b(s^t) \quad (17)$$

Proposition 1: Eq. 17 remains valid for subsets (which are not considered in this paragraph). To show this property, it is sufficient to consider the mass on the empty set after conditioning which is equal to the opposite of the plausibility of the conditioning subset.

Given all possible state sequences, one can apply Eq. 17 and then choose the best sequence S^* by:

$$S^* = \operatorname{argmax}_{S=\{s^1, s^2, \dots, s^T\}} pl_{\delta}(S) \quad (18)$$

which is intractable because the number of possible sequences is N^T , which leads to the following proposition.

Proposition 2: The Viterbi algorithm defined in probabilistic HMM can be applied using as inputs the plausibilities on singletons: $pl_\pi(s_i), s_i \in \Omega_1$ (prior), $pl_a^{\Omega_t|\Omega_{t-1}}[s_{t-1}](s_t), s_{t-1} \in \Omega_{t-1}, s_t \in \Omega_t$ (transitions) and $pl_b(s_t), s_t \in \Omega_t$ (observations). Finding the best sequence is then performed in $N \cdot T$.

If no prior information is available at $t = 1$, then $pl_\pi(s_i) = 1, \forall s_i \in \Omega_1$ (i.e. the related BBA is vacuous). The Viterbi metric is thus a plausibility.

Proposition 3: In the maximization, it is not relevant to consider all possible sequences of subsets $S^t \subseteq \Omega_t$ (even though Eq. 17 remains valid) because the plausibility increases as the cardinality of the subsets increases and therefore the resulting optimal sequence would be $S^ = \{\Omega^1, \Omega^2, \dots, \Omega^T\}$ which brings no information.*

D. The case of Evidential Markov Chains (EMC)

In Markov chains, the state $s^t \in \Omega_t$ is observed. In Evidential Markov Chains (EMC), the originality is that the state can be a subset $S^t \subseteq \Omega_t$. In signal and image processing, observations of subsets can be useful in *transitions* where the state is naturally imprecise.

Thus, in EMC, we have a sequence $S = \{S^1, S^2, \dots, S^T\}, S^t \subseteq \Omega_t$, and one is interested in assessing the plausibility of this sequence. Reasoning similarly as in the previous Viterbi procedure but conditioning by the subsets in the sequence S and considering that the true subset is perfectly known ($pl_b(S^t) = 1$), we have the following definition.

Definition 3: In an EMC, the plausibility of a sequence $S = \{S^1, S^2, \dots, S^T\}, S^t \subseteq \Omega_t$ is

$$pl_\delta(S) = pl_\pi(S^1) \cdot \prod_{t=2}^T pl_a^{\Omega_t|\Omega_{t-1}}[S^{t-1}](S^t) \quad (19)$$

EMC were first proposed in [14]. However, the authors propose to use the BBA (cf. Definition 4.1 in [14]) in order to assess the support given to a sequence. We here justify theoretically the use of plausibilities instead. In particular, if one uses only belief masses on subsets then pieces of information are lost.

V. TRAINING IN EVHMM

The problem is to learn transitions (m_a) and models (\mathbb{B}) in EvHMM. As underlined in [12], applying an iterative procedure as in EM-based HMM training is not relevant. Indeed, successive forward and backward propagations imply conjunctive combinations which has a tendency to generate specific BBAs focused on singletons, therefore losing the interest of using belief functions. We rather propose two distinct processes: one for observation models (called RCGI) and one for transitions (called ITS).

A. RCGI: Observations models training

The proposed training process of observation models can be decomposed in two steps: Clustering data into M clusters (called components in the sequel) and Regrouping the M components into N states. The main features of this algorithm (Alg. 3) are represented in Figure 1.

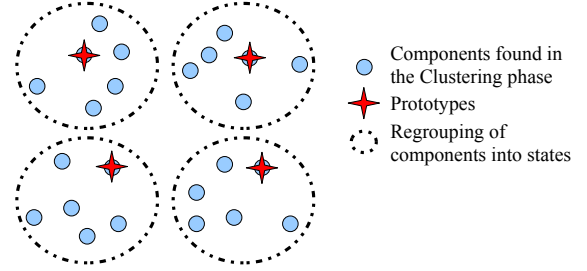


Fig. 1. RCGI steps with $N = 4$ and $M = 6$. Filled circles are the components, stars represent the first component called prototypes (obtained in the initialisation step), dotted circles are the final regrouping.

1) *Step 1 - Clustering:* The first step consists in paving the feature space by first finding $M \times N$ components in the data (see filled circles in Fig. 1):

$$\Omega_0 \leftarrow \text{find } M \times N \text{ components using a clusterer} \quad (20)$$

This phase can be performed by any clustering approach. If there are enough data, an EM algorithm can be used by estimating, for example, the parameters of a mixture of $M \times N$ Gaussian components. In the presence of imperfect data, a Gustafson-Kessel Fuzzy-C-Means [24], an Evidential C-Means (ECM) [25] or an evidential EM algorithm [26] can be used. One can also use online clustering such as [27].

2) *Step 2 - Regrouping:* In probabilistic HMM, a set of states (N) and a number of components for each state (M) has to be chosen. Then Baum-Welch algorithm finds the parameters of each component in each state:

$$\mathbb{B} \leftarrow \text{Baum-Welch algo. (data, } M \text{ comp., } N \text{ states)} \quad (21)$$

Regrouping of components into states is made automatically by maximizing likelihood, and a relevant regrouping implies a better recognition of states.

We propose to adapt this algorithm for EvHMM as follows: let $M \times N$ components found by the Clustering phase (Eq. 20), we then need to find N states, each one composed of M components. For that, we propose a procedure called RCGI (Regrouping Components with Geometrial Interaction) described in Algorithm 3. Note that the optimal way is intractable because it requires to consider all possible permutations between components.

We denote Ω_0 the set of $M \times N$ components provided by the Clustering phase. The N sets of states are denoted $\Omega_i, i = 1 \dots N$ such as:

$$\cap_i \Omega_i = \emptyset \text{ and } \cup_i \Omega_i = \Omega_0 \quad (22)$$

The cardinality $|\Omega_i|$ can be different for each state but, for the sake of simplicity, we consider here the same cardinality. RCGI thus fills a $M \times N$ association matrix A with:

$$A(i, j) = \begin{cases} 1 & \text{if component } j \text{ is assigned to state } i \\ 0 & \text{otherwise} \end{cases} \quad (23)$$

a) *Initialisation*: RCGI first requires one component for each state. This step is critical since components will be added gradually based on this initialisation. We propose the following procedure (Alg. 2):

- 1) First, compute distances between all components. The result $Z = [\mathcal{D}(i, j)]$ is a $N \times M$ triangular square matrix where elements are the distances between components i and j (we here use the Euclidean distance).

$$\mathcal{D}(i, j) \leftarrow \text{Distance between comp. } i \text{ and } j \quad (24)$$

- 2) Then, find the farthest component from all others:

$$C_1 = \operatorname{argmax}_j \sum_i \mathcal{D}(i, j) \quad (25)$$

- 3) The third step consists in finding the farthest component from C_1 :

$$C_2 = \operatorname{argmax}_{j, j \neq C_1} \mathcal{D}(\text{comp. } C_1, j) \quad (26)$$

- 4) At this stage, we have two states, each with one component. In order to find the first component for the remaining $N-2$ states, we consider the distance between C_1 and C_2 and divide it into $N-1$ segments of equal-length. Let denote \hat{C}_i the estimated component for state $i = 3 \dots N$. Therefore, C_i is given by the closest component to \hat{C}_i :

$$C_i = \operatorname{argmin}_{j, j \neq C_k, k > i} \mathcal{D}(\text{comp. } \hat{C}_i, j), i = 3 \dots N \quad (27)$$

Example 1: Let consider the data of Figure 4. It represents a set of $N = 4$ states, each one being corrupted by a $M = 3$ components additive noise (different for each state). Ideally, there are 12 components. Assume that the components are characterized by the following center means: $\mu = [4.2 \ 3.2 \ 2.2 \ 1.2 \ 1.6 \ 2.7 \ 0.7 \ 3.4 \ 3.7 \ 0.8 \ 3.6 \ 2.3]$. Thus, criterion (24) gives the following values: $\mathcal{D}(i, j) = [51.68 \ 22.08 \ 16.48 \ 34.88 \ 24.64 \ 16.28 \ 53.08 \ 26.08 \ 33.88 \ 48.96 \ 31.04 \ 15.96]$. Therefore $C_1 = 7$ ($\mu_7 = 0.7$) and $C_2 = 1$ ($\mu_1 = 4.2$). Then the segment length is $(4.2 - 0.7)/3 = 1.1667$ thus $\hat{C}_3 = 3.033$ and $\hat{C}_4 = 1.8667$ leading to $C_3 = 2$ (with $\mu_2 = 3.2$) and $C_4 = 5$ (with $\mu_5 = 1.6$). Finally, the first components of each state are 7, 1, 2 and 5.

In Fig. 1, the result of the initialisation step is represented by the stars on the chosen components.

b) *Association*: A component j in Ω_0 is associated to a state i if the latter is the closest state to j .

A representation of this assignment is depicted by dotted circles in Fig. 1. Since a state can be composed of several components, it is necessary to adapt the distance measure

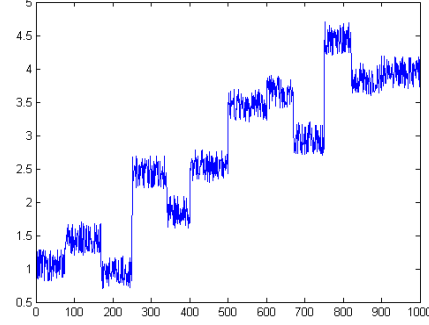


Fig. 2. Signal to be segmented.

\mathcal{D}' in order to compare a single component (j) to a set of components (composing state i):

$$\begin{aligned} j^* &= \operatorname{argmin}_j \mathcal{D}'(\text{component } j, \text{state } i) \\ A(i, j^*) &= 1 \end{aligned} \quad (28)$$

For distribution-based clusterers (such as Gaussian mixtures models), we propose to use the Kullback-Leibler (KL) divergence between both the distribution $p_j \equiv p(y|j)$ of data points y in component j and the distribution $p_i \equiv p(y|i)$ of data points y in the mixture of components composing state i :

$$\mathcal{D}'(j, i) = \text{KL}(p_i || p_j) \quad (29)$$

For mixtures of continuous densities, the KL divergence does not have a closed-form but can be estimated by Monte-Carlo sampling. Samples are thus drawn from the mixture associated to p_i and given a set of i.i.d. sampled points $y_1 \dots y_n \dots y_{N_s}$, we can approximate the KL by its Monte-Carlo estimate:

$$\hat{\text{KL}} = \frac{1}{N_s} \sum_n \log \left(\frac{p(y_n|i)}{p(y_n|j)} \right) \xrightarrow{N_s \rightarrow \infty} \text{KL}(p_i || p_j) \quad (30)$$

where we used $N_s = 1e5$ samples.

Algorithm 2 ONE_STATE_RCGI

Require: Set of components Ω_0

Require: Number of states N {assume the same number of components for each state}

Ensure: Find N prototypes: $A(j) = 1, j = 1 \dots |\Omega_0|$ if component j is a prototype

- 1: Compute distances between all components ($[\mathcal{D}(i, j)]$)
 - 2: Find the farthest component: $C_1 \Rightarrow A(C_1) = 1$
 - 3: Find the farthest component from C_1 : $C_2 \Rightarrow A(C_2) = 1$
 - 4: Find $N-2$ components between C_1 and C_2 as described in the text: assign $A(C_i) = 1, i = 3 \dots N$
-

Example 2: (continue) RCGI is applied on the data described in the previous example. It finds a set of $N = 4$ states, with $M = 3$ components each. The resulting association is $[7 \ 10 \ 4]$ for state 1, $[1 \ 9 \ 11]$ for state 2, $[2 \ 8 \ 6]$ for state 3 and $[5 \ 3 \ 12]$ for state 4. This association is depicted on Figure 3

Algorithm 3 RCGI

Require: Set of components Ω_0 {characterized by some parameters}

Require: Number of states N $\{M = |\Omega_0|/N$ since we assume the same number of components for each state}

Ensure: Association matrix $A(i, j) = 1$ if component j is assigned to state i

- 1: $A(:, 1) \leftarrow \text{ONE_STATE_RCGI}(\Omega_0, N)$ (Alg. 2) {Initialisation, then remove the prototypes from Ω_0 .}
 - 2: **for** states $i = 1$ To N **do**
 - 3: **while** $\sum_j A(i, j) < M$ **do**
 - 4: **for all** remaining components j in Ω_0 **do**
 - 5: Compute the distance $\mathcal{D}'(i, j)$ between state i and component j {See comments in text}
 - 6: **end for**
 - 7: $A(i, j^*) = 1$ with $j^* = \text{argmin}_j \mathcal{D}'(i, j)$ {Assign a component to state i }
 - 8: $\Omega_0 \leftarrow \Omega_0 - \{j^*\}$ {Update remaining components}
 - 9: **end while**
 - 10: **end for**
-

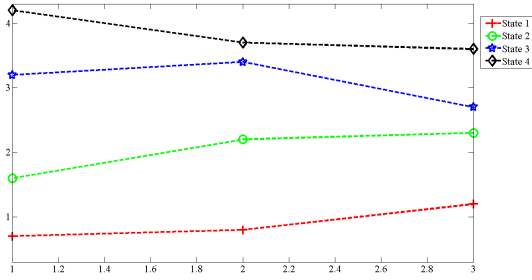


Fig. 3. Visualisation of regrouping (one line for each state).

and the obtained segmentation is given on Figure 4 after reordering of states according to their norm over components.

Compared to probabilistic HMM, RCGI is deterministic (if the number of samples is sufficient to compute the KL) and thus allows to obtain a stable result while HMM are sensitive to initialisation. Note that RCGI can also be applied on clustering results obtained in HMM.

In signal processing, we believe that RCGI can be useful since the coherence obtained has a real meaning on the signal (for temporal data).

B. Transition estimation

The problem is to estimate belief concerning transitions between states. As in Baum-Welch algorithm, we propose an iterative process described in Algorithm 4 and called ITS (Iterative Transition Specialization).

ITS algorithm starts after RCGI and requires a first estimation of the transition matrix. When there is no prior information on transitions, we use [23]:

$$m_{\hat{a}_0}^{\Omega_t \times \Omega_{t+1}} \propto \sum_{t=1}^{T-1} \left(m_b^{\Omega_t \uparrow \Omega_t \times \Omega_{t+1}} \odot m_b^{\Omega_{t+1} \uparrow \Omega_t \times \Omega_{t+1}} \right) \quad (31)$$

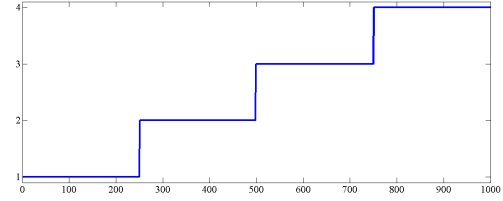


Fig. 4. Segmentation after RCGI.

up to a constant $\frac{1}{T-1}$ and where $m_b^{\Omega_t \uparrow \Omega_t \times \Omega_{t+1}}$ is the *vacuous extension* [9] of the belief mass $m_b^{\Omega_t}(\cdot | \mathbf{O}_t)$ (provided by observations) on the cartesian product defined by:

$$m_b^{\Omega_t \uparrow \Omega_t \times \Omega_{t+1}}(B) = m_b^{\Omega_t}(C) \text{ if } C \times \Omega_{t+1} = B \quad (32)$$

and 0 otherwise.

Eq. 31 is a generalization of HMM transition estimate to belief functions when there is no prior information on transitions. It is also a simplification of Eq. 22 in [23].

Once the first estimate $m_{\hat{a}_0}$ has been estimated, the evidential forward algorithm is applied. The resulting α variable is then used in Eq. 31 (instead of m_b) to compute transitions $m_{\hat{a}_1}$ used in the second iteration. In iteration k , the evidential forward produces the α variable used again in Eq. 31 to compute transitions $m_{\hat{a}_k}$. This process iterates and ITS stops when the conflict (unlikelihood) converged. The final transition matrix is denoted $m_{\hat{a}_*}$.

Algorithm 4 Algorithm ITS (Iterative Transition Specialization)

Require: Belief given observations $\{m_b^{\Omega_t}, \forall t = 1 \dots T\}$

Require: The threshold for convergence ϵ {e.g. 1e-5}

Ensure: Transition estimate $m_{\hat{a}_*}$

- 1: $\text{logpl}^{\text{old}} \leftarrow -\infty$
 - 2: $\text{converged} \leftarrow \infty$
 - 3: $m_{\hat{a}_0} \leftarrow$ Estimation of transitions using observations {First estimate of transitions by Eq. 31}
 - 4: **while** $|\text{converged}| > \epsilon$ **do**
 - 5: $[\alpha \text{ logpl}^{\text{new}}] \leftarrow$ Forward propagation {Eq. 14 and 13, Inference process}
 - 6: $m_{\hat{a}_*} \leftarrow$ Estimation of transitions using filtered belief estimate {use α in Eq. 31 instead of m_b }
 - 7: $\text{converged} \leftarrow \text{logpl}^{\text{new}} - \text{logpl}^{\text{old}}$ {Convergence}
 - 8: $\text{logpl}^{\text{old}} \leftarrow \text{logpl}^{\text{new}}$
 - 9: **end while**
 - $m_{\hat{a}_*} \leftarrow (m_{\hat{a}_*} + m_{\hat{a}_0})/2$ {Consensus, avoid low belief on doubt}
-

As expected [12], consecutive applications of the forward algorithm provides an evidential transition matrix with high values on singletons and low values on doubt. This effect is due to conjunctive operations embedded in propagations and is increased if we use the smoothed estimate (γ) instead of the filtered estimate (α). In order to solve this problem, we simply take the value of $(m_{\hat{a}_*} + m_{\hat{a}_0})/2$ as the final transition matrix. This provides a consensus between the two transitions.

Figure 5 depicts the value of the likelihood at each iteration of ITS (given the data of the previous example).

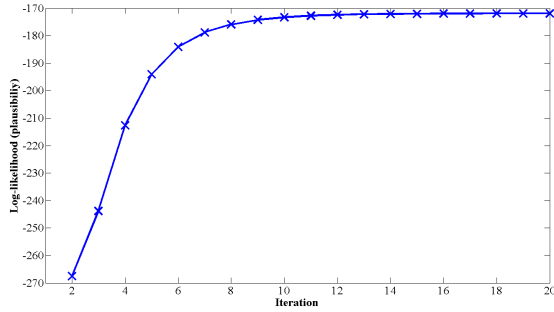


Fig. 5. Evolution of EvHMM's log-plausibility at each iteration of ITS.

C. Towards RIL in TSTEN

Previous sections introduced Evidential HMMs. We now consider more complex models that we call TSTEN, standing for Time-Sliced Temporal Evidential Networks (TSTEN), as a generalization of Dynamic Bayesian Network to belief functions. TSTEN are graphical models representing conditional independencies between variables within and across positions in a sequence. As emphasized in [28], *modelling of time-series data is natural by directed graphical models, which can capture the fact that time flows forward. Arcs within a time-slice can be directed or undirected, since they model "instantaneous" correlation.* We rather use "time-sliced" instead of "dynamic" because the latter often means the graph parameters and structure change over time.

Generalizing probabilistic *Representation - Inference - Learning* (RIL) is a necessary condition for developing TSTEN.

1) *Inference in TSTEN*: The RI-problem (Representation and Inference) was treated by Shafer, Shenoy [15], [16] and Smets using Valuation-Based Network and Directed Evidential Network [20]. To our knowledge, inference and representation were treated in the static case. Their procedures also applies in the temporal case and the first application was proposed in Evidential HMM.

EvHMM is a TSTEN where a time-slice is composed of two nodes: one discrete (X_t) and one continuous (Y_t). The temporal link is represented by an arc from X_t and X_{t+1} , i.e. two discrete nodes across successive slices.

The main particularity of EvHMM is that it considers hidden states X_t , that means only Y_t is observable and we need to infer X_t . Inference tools were formulated in this paper and used for classification and state sequence recognition.

Inference in TSTEN is carried out as follows:

- Within a time-slice: Apply a local propagation scheme¹ to update belief masses on all variables within a time-slice.

¹Such as Smets algorithm [20] which requires less computations than Shafer-Shenoy algorithm but that is more restricted than the latter because limited to directed networks.

- Between time-slices: Apply inference tools proposed in this paper to project and update belief masses on variables linked between two time-slices.
- Apply the classification algorithm (Alg. 1) proposed in this paper to classify a sequence.

2) *Training in TSTEN*: To our knowledge, the problem of parameter training in evidential networks was not considered in the past, and the same holds for TSTEN. There are three categories of parameters that have to be estimated:

- Transitions between time-slices which are the conditional BBAs of elements in X_t given elements in X_{t-1} ,
- Conditional BBAs within a time-slice and representing the link between elements in a variable and elements in another connected variable,
- Parameters of models for hidden variables.

a) *Observable variables in TSTEN*: It is the simplest case because when all variables are observable, one can compute distribution of observations. For transitions, one can also use the proposed ITS algorithm.

b) *Hidden variables in TSTEN*: This case was considered only in EvHMM. In this paper, we proposed an algorithm called RCGI that links observations to hidden states. The same procedure can be applied in more general TSTEN for all hidden variables. Besides, ITS algorithm can be applied for transition estimation.

c) *Training all parameters jointly*: We proposed two disjoint procedures to estimate the parameters of transitions (ITS) and of observation models (RCGI). However, it should be more adapted to use an algorithm that could estimate transition and models jointly.

These parameters should be chosen such as to minimize the overall conflict given the data, which is equivalent to maximize the overall plausibility. For a TSTEN with parameters θ , we can use the following iterative process:

- 1) Perform inference (see previous section),
- 2) Perform classification (Alg. 1),
- 3) Update parameters θ such as to increase the overall plausibility (Eq. 17) for the next iteration (equivalent to decrease conflict).

VI. A SYNTHESIS OF METHODS FOR DYNAMICAL SYSTEMS ANALYSIS BASED ON BELIEF FUNCTIONS

To our knowledge, the problem of state sequence recognition based on belief functions was first stated by Rombaut et al. [29] in 1999. They used a simple singly-connected tree to represent a sequence of states. The tree is initialised with a vacuous belief function to reflect total ignorance. The belief mass then flows down in the tree until the last state according to transition truthfulness. Sequence classification was done using the last state's plausibility. The main problem of this method is the sensitivity of transitions because they depend on observations which may be noisy.

In 2000, Fouque et al. [30] proposed the first method for image analysis mixing Markovian modelling and belief functions. This is the basis of Pieczynski's work on evidential

Markovian models (see [14] and references inside). However, the author generally makes probabilistic assumptions. This was also the case in [13] where the authors generalized HMM to fuzzy measures. Although the final model is very close to HMM, the generalization capability is decreased by assumptions which are very similar to HMM's ones (e.g. combinations are made using simple product). Moreover, it is not proved whether belief functions are covered.

In 2003, IDRES system was proposed for driving situation recognition [31]. The system is dedicated to a physical system and thus is difficult to generalize. In 2005, the paper of Smets on conflict analysis [32] is also a good introduction for Markovian modelling with belief functions.

In 2004, Smets and Ristic proposed to study the Kalman filter for joint tracking and classification in belief function framework [33]. They conclude that Kalman equations are quite similar to the probabilistic ones but they also showed that belief functions had a better capability to represent implication rules in the classification phase.

The next work in 2007 includes the development of a Temporal Evidential Filter (TEF) [18] which is used for belief functions filtering. This work was then used in [34] (2010) for deterministic state sequence classification. In 2007 was also proposed the first steps in the generalization of HMM to belief functions (EvHMM [23]) without probabilistic assumption. In [12] (2009), a formulation is proposed to represent imprecise knowledge on states in the training set used in probabilistic HMM training.

In 2010, an algorithm for Belief State Estimation (BSE) [35] of continuous state using interval analysis and belief functions is described.

Finally, a first step towards the generalization of inference mechanisms in Dynamic Bayesian Network to belief functions was proposed in [36]. However, their proposition is not general and is actually a particular case of EvHMM.

VII. EXPERIMENTS

As an example of RCGI algorithm for classification, we considered the challenge dataset concerning diagnostic and prognostics of machine faults from the first Int. Conf. on Prognostics and Health Management [37]. The dataset is a multiple multivariate time-series (26 variables) with sensor noise. 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.

The first experiment (*train_FD001.txt*) with five pre-selected features (3, 4, 5, 7, 9) was considered. Each time series was manually segmented into four functioning states: normal mode, transition mode, degrading mode and fault mode. The segmentation is available on demand. For each state a HMM and an EvHMM were built using 8 time-series. Testing (classification) was performed on 32 other time series. In RCGI, the clusterer was a mixture of Gaussians (GMM) with adaptive

HMM	56%	59%	100%	90%	$GA = 76\%$
HMM+RCGI	56%	78%	100%	94%	$GA = 82\%$
EvHMM 1	81%	81%	100%	91%	$GA = 88\%$
EvHMM 2	78%	78%	100%	94%	$GA = 87\%$
GMM	53%	66%	84%	81%	$GA = 71\%$

TABLE I
EvHMM 1 = GMM+RCGI+ITS, EvHMM 2 = WITH VACUOUS TRANSITIONS, GMM: CLUSTERING ONLY, GA IS THE GLOBAL ACCURACY.

number of components [38], [39] where the parameters were estimated using an EM algorithm.

For normal, transition, degrading and fault modes respectively, the results of state detection are given in table I.

RCGI algorithm greatly improved the detection results. When used within HMM, an average improvement of +6% was obtained (lines 1 and 2). The improvement is also important (+10%) when using EvHMM (lines 1 and 3). In comparison, classification using only GMM leads to a global accuracy of 71% (vs. 88%, lines 2 and 5).

Results obtained by EvHMM is explained by RCGI algorithm which was combined with an efficient clustering method where the number of components was set automatically. This number was adaptively computed according to the distribution of the data which depends on the number of data points used in the training set. Therefore, the proposed algorithm is well suited for limited training sets.

Line 4, EvHMM was applied with vacuous transitions leading to a satisfying classification accuracy of 87%. Actually, during experiments on this dataset, transitions were shown to have a limited impact on the detection accuracy. This is mainly due to the evidential transitions matrix generated by ITS which were composed of too many parameters ($N \cdot 2^N$) and thus their influence were limited.

VIII. CONCLUSION

An original review on methods for state sequence recognition based on belief functions was proposed and emphasized the need of a general tool for stochastic state sequence recognition. The generalization to belief functions of inference mechanisms used in HMM was then presented and illustrated. They are the basis of Evidential HMM (EvHMM) which represents a Time-Sliced Temporal Evidential Network (TSTEN).

Training algorithms for EvHMM parameters estimation were then developed. The first algorithm called RCGI (Re-grouping Components with Geometrical Interaction) allows to generate belief functions on states. RCGI first performs a clustering step to pave the feature space. Then clusters are gathered into states (using here the Kullback-Leibler divergence). The second algorithm called ITS (Iterative Transition Specialization) allows to estimate transitions. The first results are encouraging on a diagnostics task. In particular, results provided by RCGI are easily interpreted and this algorithm can also be used in probabilistic HMM. The performance of EvHMM for classification was shown on an example and compared with probabilistic HMM.

The proposed methodology is adapted with poor training sets and more experiments are under way to thoroughly validate the approach for general diagnostics and prognostics tasks.

ACKNOWLEDGMENT

We thank the French Ministry of Advanced Education and Research for supporting this work.

REFERENCES

- [1] A. Heng, S. Zhang, A. Tan, and J. Mathew, "Rotating machinery prognostics: State of the art, challenges and opportunities," *Mechanical Systems and Signal Processing*, vol. 23, pp. 724–739, 2009.
- [2] G. Klir and M. Wierman, *Uncertainty-based information. Elements of generalized information theory*. Physica-Verlag, 1999, ch. Studies in fuzzyness and soft computing.
- [3] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proc. of the IEEE*, vol. 77, pp. 257–285, 1989.
- [4] X. Zhang, R. Xu, C. Kwan, S. Y. Liang, Q. Xie, and L. Haynes, "An integrated approach to bearing fault diagnostics and prognostics," in *American Control Conf.*, Portland, OR, USA, 2005.
- [5] M. Donga 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.
- [6] A. Dempster, "Upper and lower probabilities induced by multiple valued mappings," *Annals of Mathematical Statistics*, vol. 38, pp. 325–339, 1967.
- [7] G. Shafer, *A mathematical theory of Evidence*. Princeton University Press, 1976.
- [8] P. Smets, "Advances in the dempster-shafer theory of evidence - what is dempster-shafer's model ?" *Wiley*, pp. 5–34, 1994.
- [9] P. Smets and R. Kennes, "The transferable belief model," *Artificial Intelligence*, vol. 66, pp. 191–234, 1994.
- [10] J. Bilmes, "What HMM can do," Technical Report of University of Washington, Department of Electrical Engineering, Tech. Rep. UWEETR-2002-0003, 2002.
- [11] T. Denoeux, "Conjunctive and disjunctive combination of belief functions induced by non distinct bodies of evidence," *Artificial Intelligence*, vol. 172, pp. 234–264, 2008.
- [12] E. Ramasso, "Contribution of belief functions to hidden markov models," in *IEEE Workshop on Machine Learning and Signal Processing*, Grenoble, France, October 2009, pp. 1–6.
- [13] M. Mohamed and P. Gader, "Generalized HMM - part 1: Theoretical frameworks," *IEEE Trans. on Fuzzy Systems*, vol. 8, pp. 67–81, 2000.
- [14] W. Pieczynski, "Multisensor triplet markov chains and theory of evidence," *Int. Journal of Approximate Reasoning*, vol. 45, pp. 1–16, 2007.
- [15] P. Shenoy, "Valuation-based systems: A framework for managing uncertainty in expert systems," *Fuzzy Logic for the Management of Uncertainty*, pp. 83–104, 1992.
- [16] P. Shenoy and G. Shafer, "Axioms for probability and belief function propagation," *Uncertainty in Artificial Intelligence*, vol. 4, pp. 169–198, 1990.
- [17] P. Smets, "Analyzing the combination of conflicting belief functions," *Info. Fusion*, vol. 8, pp. 387–412, 2005.
- [18] E. Ramasso, M. Rombaut, and D. Pellerin, "State filtering and change detection using TBM conflict," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 17, pp. 944–949, 2007.
- [19] A. Dempster, "A generalization of bayesian inference," *Journal of the Royal Statistical Society*, vol. 30, pp. 205–247, 1968.
- [20] P. Smets, "Beliefs functions: The disjunctive rule of combination and the generalized bayesian theorem," *Int. Journal of Approximate Reasoning*, vol. 9, pp. 1–35, 1993.
- [21] B. Cobb and P. Shenoy, "On the plausibility transformation method for translating belief function models to probability models," *Int. journal of approximate reasoning*, vol. 41, no. 3, pp. 314–330, 2006.
- [22] F. Pichon, "Belief functions: canonical decompositions and combination rules," Ph.D. dissertation, University of Technology of Compiègne, 2009.
- [23] E. Ramasso, M. Rombaut, and D. Pellerin, "Forward-backward-viterbi procedures in TBM for state sequence analysis using belief functions," in *Europ. Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, 2007, pp. 405–417.
- [24] E. Gustafson and W. Kessel, "Fuzzy clustering with a fuzzy covariance matrix," in *IEEE Int. Conf. on Decision and Control*, 1978.
- [25] M.-H. Masson and T. Denoeux, "ECM: An evidential version of the fuzzy c-means algorithm," *Pattern Recognition*, pp. 1384–1397, 2008.
- [26] E. Côme, L. Oukhellou, T. Denoeux, and P. Aknin, "Learning from partially supervised data using mixture models and belief functions," *Pattern Recognition*, vol. 42, pp. 334–348, 2009.
- [27] L. Serir, E. Ramasso, and N. Zerhouni, "Evidential evolving gustafsson-kessel algorithm for data streams partitioning using belief functions," in *The 11th European Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, vol. LNAI 6717, 2011.
- [28] K. Murphy, "Dynamic bayesian networks: Representation, inference and learning," Ph.D. dissertation, UC Berkeley, Computer Science Division, 2002.
- [29] M. Rombaut, I. Jarkass, and T. Denoeux, "State recognition in discrete dynamical systems using Petri nets and Evidence theory," in *Europ. Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, 1999, pp. 352–361.
- [30] L. Fouque, A. Appriou, and W. Pieczynski, "Evidential markovian model for data fusion and unsupervised image classification," in *Int. Conf. on Information Fusion*, vol. TuB4, 2000, pp. 25–31.
- [31] J. Nigro and M. Rombaut, "IDRES: A rule-based system for driving situation recognition with uncertainty management," *Information Fusion*, vol. 4, pp. 309–317, 2003.
- [32] P. Smets, "Analyzing the combination of conflicting belief functions," *Information Fusion*, vol. 8, pp. 387–412, 2004.
- [33] B. Ristic and P. Smets, "Kalman filters for tracking and classification and the transferable belief model," in *Int. Conf. on Information Fusion*, 2004.
- [34] E. Ramasso, C. Panagiotakis, M. Rombaut, and D. Pellerin, "Belief scheduler based on model failure detection in the TBM framework. application to human activity recognition," *Int. Journal of Approximate Reasoning*, vol. 51, 2010.
- [35] G. Nassreddine, F. Abdallah, and T. Denoeux, "State estimation using interval analysis and belief-function theory: application to dynamic vehicle localization," *IEEE Trans. on Syst., Man, and Cybernetics, Part B: Cybernetics*, vol. 40, 2010.
- [36] E. Pollard, M. Rombaut, and B. Pannetier, "Bayesian networks vs. evidential networks. an application to convoy detection," in *Int. Conf. on Information Processing and Management of Uncertainty*, 2010.
- [37] A. Saxena, K. Goebel, D. Simon, and N. Eklund, "Damage propagation modeling for aircraft engine run-to-failure simulation," in *IEEE Int. Conf. on Prognostics and Health Management*, 2008.
- [38] D. Ververidis and C. Kotropoulos, "Gaussian mixture modeling by exploiting the mahalanobis distance," *IEEE Trans. on Signal Processing*, vol. 56, pp. 2797–2811, 2008.
- [39] M. Figueiredo and A. Jain, "Unsupervised learning of finite mixture models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, pp. 381–396, 2002.