# E2GK-pro: An Evidential Evolving Multimodeling Approach for Systems Behavior Prediction

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

## ABSTRACT

Nonlinear dynamic systems identification and nonlinear dynamic behavior prediction are important tasks in several areas of industrial applications. Multiple works proposed multimodel-based approaches to model nonlinear systems. Multimodeling permits to blend different model types together to form hybrid models. It advocates the use of existing, well known model types within the same model structure. Recently, a multimodeling strategy based on belief functions theory was developed based on a fuzzy rule based system. We propose a different approach of this latter taking advantage of new efficient evidential clustering algorithms for the determination of the local models and the assessment of the global model. In particular, the algorithm called E2GK-pro relies on an online procedure based on the Evidential Evolving Gustafsson-Kessel (E2GK) algorithm that ensures an evolving partitioning of the data into clusters corresponding to operating regions of the global system. Thus the estimation of the local models is dynamically performed by upgrading and modifying their parameters while the data arrive. Each local model is weighted by a belief mass provided by E2GK, and the global model (multimodel) is a combination of all the local models.

## 1. INTRODUCTION

Dealing with nonlinear systems behavior identification and prediction is a widely encountered problem in real world applications in engineering, industry, time series analysis, prediction and fault diagnosis. Modeling an a priori unknown dynamic process from observed data is a hard task to perform. Among the large variety of proposed approaches taking into account nonlinearity, one can cite Fuzzy logic based models (Takagi & Sugeno, 1985) and especially neural network based approaches, which applications during the last decades are numerous in dynamical system modeling, and in particular in prognosis applications (El-Koujok, Gouriveau, & Zerhouni, 2011). Usually, the models consist of a set of functional relationships between the elements of a set of variables. Multiple works propose multimodel-based approaches to avoid difficulties (modeling complexity) related to nonlinearity (P. Angelov, Lughofer, & Zhou, 2008; Madani, Rybnik, & Chebira, 2003).

Multimodeling permits to blend different model types together to form hybrid models, offering a unified view toward modeling with well known model types instead of promoting a singular model type which is insufficient to model large scale systems. In a general way, in multimodel-based approaches, a set of models, corresponding to a set of operating ranges of the system, contributes to identify the whole system. Such an approach can be seen as a weighted contribution of a set of models approximating the whole system's behavior, each of which is valid in a well defined interval which corresponds to operating region of the system or covers a part of the whole feature space of the problem to be solved. The description of the global system's behavior is made by combination of all the local models. The contribution of each local model in the assessment of the multimodel's output is quantified by an *activation degree*.

One of the most popular models is the TSK fuzzy model that showed great performances in many applications on prediction (El-Koujok et al., 2011). A first order Takagi-Sugeno model can be seen as a multimodel structure consisting of linear models. It is based on a fuzzy decomposition of the input space to describe the inherent structure for a concrete problem by partitioning each input variable range into fuzzy sets. For each part of the state space, a fuzzy rule can be constructed to make a linear approximation of the input, and the global output is a combination of all the rules. Then, the parameters of the models (non-linear parameters of membership degrees and linear parameters for the consequent of each rule) are tuned in an appropriate learning procedure. Usually, the identification of the linear parameters is addressed by some gradient descent variant, e.g., the least squares algorithm, whereas non-linear parameters are determined by some clustering method on the input space. This kind of approach has been applied to build a Neuro-Fuzzy predictor in the context of prognosis application by (El-Koujok et al., 2011).

It was based on the evolving extended Takagi-Sugeno system (exTS) proposed by Angelov (P. P. Angelov & Filev, 2004).

Recently, a multimodeling strategy based on belief functions theory was developed based on a TSK fuzzy model (Ramdani, Mourot, & Ragot, 2005). The basic idea was to consider a fuzzy rule based system with a belief structure as output. The focal elements of each rule were formed by a subset of a collection of functional models each of which was constructed based on a fuzzy model of Takagi-Sugeno type. In this paper we investigate this method and we introduce some modification taking advantage of new efficient evidential clustering algorithms for the determination of the local models and the assessment of the global model. In particular, we propose an online procedure using the Evidential Evolving Gustafsson-Kessel (E2GK) (Serir, Ramasso, & Zerhouni, 2011) algorithm that ensures an evolving partitioning of the data into clusters that correspond to operating regions of the global system. Thus the estimation of the local models is dynamically performed by upgrading and modifying their parameters while the data arrive. Each local model is weighted by a belief mass provided by E2GK, and the global model (multimodel) is a combination of all the local models.

The paper is organized as follows: Section 2 is dedicated to the necessary background for our approach. In section 3, the existing approach will be first presented (Section 3.1), analyzed (Section 3.2) in order to introduce the proposed model (Section 3.3). Results will finally be presented in Section 4.

## 2. Background

A brief description of belief functions is first given. Then, ECM algorithm is presented followed by E2GK algorithm as the basis of the prediction algorithm.

### 2.1 Belief Functions

Dempster-Shafer theory of evidence, also called belief functions theory, is a theoretical framework for reasoning with partial and unreliable information. Ph. Smets proposed the *Transferable Belief Model* (TBM) (Smets & Kennes, 1994) as a general framework for uncertainty representation and combination of various pieces of information without additional priors. In particular, TBM offers the possibility to explicitly emphasize doubt, that represents ignorance, and conflict, that emphasizes the contradiction within a fusion process. We give here some of the basic notions of the theory and refer the reader to (Smets & Kennes, 1994) for a more complete description.

The central notion of the theory of belief functions is the *basic belief assignment* (BBA), also called *belief mass assignment* that represents the *belief* of an agent in subsets of a finite set $\Omega$, called the frame of discernment. It is defined by:

$$\begin{array}{rccc} m: & 2^{\Omega} & \rightarrow & [0,1] \\ & A & \mapsto & m(A) \end{array}, \tag{1}$$

with $\sum_{A \subseteq \Omega} m(A) = 1$. A belief mass can not only be assigned to a singleton ($|A| = 1$), but also to a *subset* ($|A| > 1$) of variables *without assumption concerning additivity*. This property permits the explicit modelling of doubt and conflict, and constitutes a fundamental difference with probability theory. The subsets $A$ of $\Omega$ such that $m(A) > 0$, are called the *focal elements* of $m$. Each focal element $A$ is a set of possible values of $\omega$. The quantity $m(A)$ represents a fraction of a unit mass of belief allocated to $A$. Complete ignorance corresponds to $m(\Omega) = 1$, whereas perfect knowledge of the value of $\omega$ is represented by the allocation of the whole mass of belief to a unique singleton of $\Omega$, and $m$ is then said to be *certain*. In the case of all focal elements being singletons, $m$ boils down to a probability function and is said to be *bayesian*.

A positive value of $m(\emptyset)$ is considered if one accepts the *open-world assumption* stating that the set $\Omega$ might not be complete, and thus $\omega$ might take its value outside $\Omega$. The conflict is then interpreted as a mass of belief given to the hypothesis that $\omega$ might not lie in $\Omega$. This interpretation is useful in clustering for outliers detection (Masson & Denoeux, 2008).

### 2.2 Evidential C-Means

In 2008, Masson and Denoeux (Masson & Denoeux, 2008) proposed a clustering algorithm based on the concept of *Credal Partition* (Masson & Denoeux, 2008). Similar to the concept of fuzzy partition, but more general, it particularly permits a better interpretation of the data structure and makes it possible to code all situations, from certainty to total ignorance. Considering a set of $N$ data $x_1, \ldots, x_n$ to be grouped in $c$ clusters, a credal partition is constructed by assigning a BBA to each possible subset of clusters. Partial knowledge regarding the membership of an data point $i$ to a class $j$ is represented by a BBA $m_{ij}$ on the set $\Omega = \{\omega_1, \ldots, \omega_c\}$. ECM is an optimization based clustering algorithm whose objective function is given by:

$$J_{ECM}(M,V) = \sum_{i=1}^{N} \sum_{\{j/A_j \neq \emptyset, A_j \subseteq \Omega\}} |A_j|^{\alpha} m_{ij}^{\beta} d_{ij}^2 + \sum_{i=1}^{N} \delta^2 m_i(\emptyset)^{\beta} \tag{2}$$

subject to

$$\sum_{\{j/A_j \neq \emptyset, A_j \subseteq \Omega\}} m_{ij} + m_i(\emptyset) = 1 \quad \forall i = 1, \ldots, N \ , \tag{3}$$

where:

- $\alpha$ is used to penalize the subsets of $\Omega$ with high cardinality,
- $\beta > 1$ is a weighting exponent that controls the fuzziness of the partition,
- $d_{ij}$ denotes the Euclidean distance between object $i$ and prototype $v_j$,
- $\delta$ controls the amount of data considered as outliers.

The $N \times 2^c$ partition matrix $M$ is derived by determining, for each object $i$, the BBAs $m_{ij} = m_i(A_j)$ , $A_j \subseteq \Omega$ such that $m_{ij}$ is low (resp. high) when the distance $d_{ij}$ between data $i$ and focal element $A_j$ is high (resp. low). The matrix $M$ is computed by the minimization of criterion (2) and was shown to be (Masson & Denoeux, 2008), $\forall i = 1 \ldots n$,

$\forall j / A_j \subseteq \Omega$, $A_j \neq \emptyset$:

$$m_{ij} = \frac{|A_j|^{-\alpha/(\beta-1)}\, d_{ij}^{-2/(\beta-1)}}{\sum\limits_{A_k \neq \emptyset} |A_k|^{-\alpha/(\beta-1)}\, d_{ik}^{-2/(\beta-1)} + \delta^{-2/(\beta-1)}} \ , \tag{4}$$

and $m_i(\emptyset) = 1 - \sum_{A_j \neq \emptyset} m_{ij}$. Centers of clusters are optimized by minimizing criterion (2). The distance between an object and any non empty subset $A_j \subseteq \Omega$ is then defined by computing the center of each subset $A_j$. This latter is the barycenter $\overline{v_j}$ of the centers of clusters composing $A_j$.

From the credal partition, the classical clustering structures (possibilistic, fuzzy and hard partitions) can be recovered (Masson & Denoeux, 2008). One can also summarize the data by assigning each object to the set of clusters with the highest mass. One then obtains a partition of the points in at most $2^c$ groups, where each group corresponds to a set of clusters. This makes it possible to find the points that unambiguously belong to one cluster, and the points that lie at the boundary of two or more clusters. Moreover, points with high mass on the empty set may optionally be rejected as outliers.

### 2.3 E2GK: Evidential Evolving Gustafson-Kessel Algorithm

In (Serir et al., 2011), an online clustering method, the evidential evolving Gustafson-Kessel algorithm (E2GK), was introduced in the theoretical framework of belief functions. The algorithm enables an online partitioning of data streams based on two existing and efficient algorithms: Evidantial $c$-Means (ECM) and Evolving Gustafson-Kessel (EGK) (Georgieva & Filev, 2009). E2GK makes it possible to compute, online, a credal partition as data gradually arrive. We summarize in the following the main steps of the algorithm:

*Step 1 – Initialization:* At least one cluster center should be provided. Otherwise, the first point is chosen as the first prototype. If more than one prototype is assumed in the initial data, the Gustafsson-Kessel (Gustafson & Kessel, 1978) or ECM algorithm can be applied to identify an initial partition matrix. The result of the initialization phase is a set of $c$ prototypes $v_i$ and covariance matrices $F_i$.

*Step 2 – Decision making:* The boundary of each cluster is defined by the cluster radius $r_i$, defined as the *median* distance between the cluster center $v_i$ and the points belonging to this cluster with membership degree larger or equal to a given threshold $u_h$:

$$r_i = \text{median}_{\forall x_j \in \textit{ i-th cluster and } P_{ij} > u_h} \|v_i - x_j\|_{A_i} \ . \tag{5}$$

where $P_{ij}$ is the confidence degree that point $j$ belongs to $\omega_i \in \Omega$ and can be obtained by three main process: either by using the belief mass $m_j(\omega_i)$, or the pignistic transformation (Smets & Kennes, 1994) that converts a BBA into a probability distribution, or by using the plausibility transform (Cobb & Shenoy, 2006). We propose to choose the belief mass for which the computation is faster.

The minimum membership degree $u_h$ - initially introduced in (Georgieva & Filev, 2009) and required to decide whether a data point belongs or not to a cluster - can be difficult to assess. It may depend on the density of the data as well as on the level of cluster overlapping. Thus $u_h$ is automatically set to $1/c$ in order to reduce the number of parameters while ensuring a natural choice for its value.

*Step 3 – Computing the partition matrix:* Starting from the resulting set of clusters at a given iteration, the partition matrix $M$ is built as in ECM. The Mahalanobis-like distance $d_{ik}$ is considered assuming that each cluster volume $\rho_i$ is one as in standard GK algorithm:

$$d_{ik}^2 = \|x_k - v_i\|_{A_i}^2 = (x_k - v_i) A_i (x_k - v_i)^T \ , \tag{6a}$$

$$A_i = [\rho_i \cdot \det(F_i)]^{1/n}\, F_i^{-1} \ , \tag{6b}$$

$$F_i = \frac{\sum_{k=1}^{N} (m_{ik})^{\beta} (x_k - v_i)^T (x_k - v_i)}{\sum_{k=1}^{N} (m_{ik})^{\beta}} \ . \tag{6c}$$

where $F_i$ is the fuzzy covariance matrix.

Storing the whole partition is not efficient. Indeed, only the belief masses on singletons need to be stored in order to make the decision concerning the radius. As shown in Eq. 4, values on singletons are easy to compute but the problem is to estimate the normalization factor. To overcome this problem, all values of masses have to be computed but not stored. This little trick exponentially decreases memory comsumption.

*Step 4 – Adapting the structure:* Given a new data point $x_k$, two cases are considered:

- *Case 1: $x_k$ belongs to an existing cluster, thus a clusters' update has to be performed.* Data point $x_k$ is assigned to the closest cluster $p$ if the distance $d_{pk}$ is less or equal to the radius $r_p$. Then, an update of the $p$-th cluster has to be performed as follows:

  $$v_{p,new} = v_{p,old} + \theta \cdot \Delta \ , \tag{7}$$

  where

  $$\Delta = x_k - v_{p,old} \ , \tag{8}$$

  and

  $$F_{p,new} = F_{p,old} + \theta \cdot \left( \Delta^{\mathrm{T}} \Delta - F_{p,old} \right) \ , \tag{9}$$

  where $\theta$ is a learning rate (and can be set in $[0.05, 0.3]$), $v_{p,new}$ and $v_{p,old}$ denote respectively the new and old values of the center, and $F_{p,new}$ and $F_{p,old}$ denote the new and old values of the covariance matrix.
- *Case 2: $x_k$ is not within the boundary of any existing cluster (i.e. $d_{pk} > r_p$), thus a new cluster may be defined and a clusters' update has to be performed.* The number of clusters is thus incremented: $c = c + 1$. Then, the incoming data $x_k$ is accepted as a center $v_{new}$ of the new cluster and its covariance matrix $F_{new}$ is initialized with the covariance matrix of the closest cluster $F_{p,old}$. In order to quantify the credibility of the estimated clusters, a parameter $P_i$ has been introduced in (Georgieva & Filev, 2009) to assess the number of points belonging to the $i$-th cluster. The authors suggested a threshold parameter $P_{tol}$ to guarantee the validity of the covariance matrices and to improve the robustness. This parameter corresponds to the desired minimal amount of points falling within the boundary of each cluster. The threshold value is context determined due to the specificity of the considered data set. The new created cluster is then rejected if it contains less than $P_{tol}$ data points.

After creating a new cluster, the data structure evolves. However, the new cluster may contain data points previously assigned to another cluster. Thus, the number of data points in previous clusters could change. After the creation of a new cluster, E2GK verifies that all clusters have at least the required minimum amount of data points ($P_{tol}$ or more). If clusters don't satisfy this condition, the cluster with the minimum number of points is removed.

The overall algorithm is presented in Alg. 1.

**Algorithm 1** E2GK algorithm

```
Initialization: Take the first point as a center or apply the off-line GK algorithm to get the initial number of clusters $c$ and the corresponding matrices $V$ and $F_i$, $i = 1 \cdots c$
Calculate $\overline{v_j}$, the barycenter of the cluster centers composing $A_j \subseteq \Omega$
Calculate the credal partition $M$ using Eq. 4 (store only singletons and normalize)
for all new data point $x_k$ do
    Find the closest cluster $p$
    Calculate the radius $r_p$ of the closest cluster (Eq. 5)
    if $d_{pk} \leq r_p$ then
        Update the center $v_p$ (Eq. 7)
        Update the covariance matrix $F_p$ (Eq. 9)
    else
        Create a new cluster: $v_{c+1} := x_k$ and $F_{c+1} := F_p$
        Keep it if the number of points in this cluster is $\geq P_{\text{tol}}$
    end if
    Recalculate the credal partition $M$
    Check the new structure: estimate the number of points within each cluster and remove one cluster for which the latter is $\leq P_{tol}$
end for
```

## 3. Modeling Dynamics

In this section, the existing approach is first presented (Section 3.1) and then analyzed (Section 3.2). Finally we introduce the proposed model based on belief functions (Section 3.3).

### 3.1 The existing approach

In (Ramdani et al., 2005), a multi-modeling strategy based on belief function theory was developed for modeling complex nonlinear mappings by combination of simpler functional models. It was based on the TSK fuzzy model. The basic idea was to consider a fuzzy rule based system with a belief structure as output. The focal elements of each rule were formed by a subset of a collection of functional models. Each functional model is constructed based on a fuzzy model of Takagi-Sugeno type in two steps: structure identification and parameters estimation. In the first step, the antecedent and consequent variables of the model are determined. From the available training data that contain input-output samples, a regression matrix and an output vector are constructed. In the second step, the number of rules $K$, the antecedent fuzzy sets, and the parameters of the rule consequents are identified. The system behaviour is approximated by local linear models of the different operating regions that are represented by clusters. The Gustafsson-Kessel fuzzy clustering algorithm (Gustafson & Kessel, 1978) is applied on the product-space of input and output variables to discover the potential regions of the rules and capture the interaction between the input and output variables. Thus, a certain number $c$ of functional relationships between input and output variables, denoted by $f^j(x), j = 1, \ldots, c$, are assumed and form the frame of discernment $\Omega$:

$$\Omega = \left\{ \left\{ f^1 \right\}, \ldots, \left\{ f^c \right\} \right\} \quad , \tag{10}$$

where $\left\{ f^j \right\}$ is the hypothesis that corresponds to the functional model $f^j(x)$. The authors consider the case where the number of input prototypes (or rules) is equal to the number of functional prototypes ($K = c$). In order to predict an output value $y$ for a given input vector $x$, each of the $K$ rules (determined in the second step (Ramdani et al., 2005)) provides a piece of evidence concerning the value of the unknown output $y$, which can be represented by a belief mass $m^i, i = 1, \ldots, K$:

$$\left\{ \begin{array}{lcl} m^i \left( \left\{ f^j \right\} | x \right) & = & \phi_i \left( x \right), j = 1, \ldots, J \left( i \right) \\ m^i \left( \Omega | x \right) & = & 1 - \phi_i \left( x \right) \\ m^i \left( A | x \right) & = & 0 \;\; \forall A \in F^{\Omega} - F^i \end{array} \right. \tag{11}$$

where $F^{\Omega}$ is the power set of $\Omega$, $F^i$ are the focal sets of $m^i$, and The function $\phi_i \left( x \right)$ is related to the input domain (domain of expertise) of the $i$th rule. We refer the reader to (Ramdani et al., 2005) for more details. This method of constructing belief masses is based on a method proposed by T. Denoeux (Denoeux, 2000) in the context of classification.

In order to make a decision, the outputs of the different rules which are belief structures, are combined using Dempster's rule of combination giving the overall belief structure $m$, which is a vector of $c + 1$ elements:

$$m = \oplus_{i=1}^{K} m^i, \tag{12}$$

It is then normalized providing a belief structure: $m^*_j = \frac{m_j}{\sum_{q=1}^{c+1} m_q} \;\; j = 1, \ldots, c + 1$.

The overall multimodel is then defined as a combination of the functional prototypes with an additional model representing the frame of discernment, denoted by $f^{\Omega}$:

$$\hat{y} = \sum_{i=1}^{c} m^* \left( \left\{ f^i \right\} \right) f^i \left( x \right) + m^* \left( \Omega \right) f^{\Omega} \left( x \right) \quad . \tag{13}$$

Here, the authors associate the mass of total ignorance to a general model $f^{\Omega} \left( x \right)$, which is a convex combination of local linear functions whose parameters are identified globally by a single least squares equation (Eq.18, (Ramdani et al., 2005)). This formulation emphasizes the doubt concerning the model. On the other hand, le linear models $f^i \left( x \right)$ are identified by the weighted least squares (Eq.19 in (Ramdani et al., 2005)).

### 3.2 Analysis of the existing approach

*Problem 1 – Determining the belief masses:* In (Ramdani et al., 2005), the approach relies on fuzzy modeling using belief functions based on two existing approaches. The first approach was proposed by Yager (Yager & Filev, 1995) in the context of fuzzy modeling. This strategy allows the integration of probabilistic uncertainty in fuzzy rule based systems. The output of the rules is a belief structure whose focal elements are fuzzy sets among the output variable linguistic terms. The second approach is the evidential k-nearest neighbours proposed by Denoeux (Denoeux, 2000) in the context of classification and later applied in regression analysis (Petit-Renaud & Denoeux, 1999). For a given input query vector, the output variable is obtained in the form of a fuzzy belief assignment (FBA), defined as a collection of fuzzy sets of values with associated masses of belief. In (Petit-Renaud & Denoeux, 1999), the output FBA is computed nonparametrically on the basis of the training samples in the neighbourhood of the query point. In this approach, the underlying principle is that the neighbours of the query point are considered as sources of partial information on the response variable; the bodies of evidence are discounted as a function of their distance to the query point, and pooled using the Dempster's rule of combination.

*Contribution 1:* We propose a simpler and more efficient method than the one described in section 3.1 to generate the masses of belief directly from the data at the clustering step. Indeed, in 2004, the authors of (Ramdani et al., 2005) couldn't yet benefit from new efficient clustering algorithms exclusively based on belief functions. In 2008, the first clustering algorithm, the Evidential c-Means algorithm (ECM) (described in section 2.2), based on belief functions was proposed by M-H. Masson and T. Denoeux (Masson & Denoeux, 2008). In the approach proposed by (Ramdani et al., 2005), applying ECM to the set of learning data would directly provide the BBA.

*Problem 2 – Modeling doubt regarding the global model:* In (Ramdani et al., 2005), the authors define the overall model as a combination of the functional prototypes with a single model representing the frame of discernment denoted by $f^{\Omega}(x)$. This particular model is associated to the mass of total ignorance (Eq.13). Doing so, the authors claimed to emphasize the doubt concerning the global model. We believe that the global model $f^{\Omega}(x)$ as proposed in (Ramdani et al., 2005), which is a convex combination of local linear functions, doesn't bring significant additional information to the model. Indeed, it is very similar to the local linear models as shown in their experiments.

*Contribution 2:* ECM assigns masses of belief to singletons but also to unions of clusters representing doubt regarding the general model. As a unit mass is distributed among all possible subsets of $\Omega$, the masses on singletons are computed taking into account the doubt regarding the global model. Thus, we propose a different formulation of the overall multimodel, where we no longer have to combine the functional prototypes with a model representing the frame of discernment:

$$\hat{y} = \sum_{i=1}^{c} m^{*}\left(\left\{f^{i}\right\}\right) f^{i}(x) \quad . \tag{14}$$

Contrary to the original approach where the doubt concerning the global model is emphasized by taking into account an additional model representing the frame of discernment, we develop an approach where doubt is emphasized directly based on E2GK.

*Problem 3 – Evolving Modeling:* The previous approach is suitable for a fixed set of data supplied in batch mode and under the assumption that the model structure remains unchanged. When the training data are collected continuously, some of them will reinforce and confirm the information contained in the previous data, while others could bring new information. This new information could concern a change in operating conditions, development of a fault or simply more significant change in the dynamic of the process. They may provide enough new information to form a new local model or to modify or even delete an existing one. Thus an adaptation of the model structure is necessary. To do so, an on-line clustering of the inputoutput data space with gradually evolving regions of interest should be used.

*Contribution 3:* We propose to use the recently proposed online clutering method (Serir et al., 2011) E2GK (Evidential Evolving Gustafson-Kessel) that enables online partitioning of data streams and adapts the clusters' parameters along time. As presented in section 2.3, E2GK uses the concept of credal partition of ECM, offering a better interpretation of the data structure. The resulting BBAs can then be used in Eq.14.

### 3.3 The proposed model (E2GK-pro)

Based on the same general idea, we propose to construct a model for approximating nonlinear functional mappings. As in (Ramdani et al., 2005), the system behaviour is approximated by local linear models of the different operating regions that are represented by clusters.

Compared to the original approach, we propose the following methodology :

1. Use the online evidential clustering algorithm E2GK that is capable of generating the belief masses and adapt the clusters' parameters along time;
2. For each cluster discovered by E2GK, construct a linear local model and update with the new incoming data;
3. Predict the new output $\hat{y}$ by the linear combination of the local models as in Eq.14.

*Initialization:* The first data point $x_1$ is chosen as the first prototype. At the moment not enough data are available to construct the first model. As discussed in section (2.3), a new cluster is created if it contains at least $P_{tol}$ data points. We will consider the same threshold for the necessary amount of data to construct a new model. Basically, the initialization step is the same as in E2GK.

*Adapting the structure:* At each new incoming data point $x_k$, an update of the clusters' parameters is performed by E2GK. Either $x_k$ belongs to an existing cluster, thus a clusters' update has to be performed. Or, $x_k$ is not within the boundary of any existing cluster, thus a new cluster may be defined and a clusters' update has to be performed. After the creation of a new cluster, E2GK verifies if all clusters have at least the required minimum amount of data points ($P_{tol}$ or more) and suppresses the clusters that fail to satisfy this condition. To each cluster $i$ corresponds a local model $f^i$ such that:

$$f^{i}(x) = \theta_{i0} + \theta_{i1}x_1 + \cdots + \theta_{ir}x_r \tag{15}$$

where $x = [x_1, \dots x_r]^T$ is the vector of data composing the $i$th cluster and $\theta_i^T = [\theta_{i0}, \theta_{i1}, \dots, \theta_{ir}]$ is the vector of the parameters of $f^i$.

Then either a simple or a weighted recursive least squares estimation (RLS) (P. Angelov et al., 2008) could be used to identify the parameters of these linear sub-models.

Three cases are considered: 1) *a new cluster is created*, then the partition matrix changes and a learning step has to be performed for both the new local model and the previously generated local models; 2) *a cluster is removed*, then the partition matrix also changes and a new learning step has to be performed to update the existing local models; 3) Nothing happens.

*Predicting the new output:* Once the parameters of the local models are identified, the new output (prediction at $t+1$) is estimated by Eq. 14.

We summarize the general approach in Alg. 2:

**Algorithm 2** General Approach

```
Require: $x_k$ a new data point and E2GK parameters
Ensure: $\hat{x}^{k+1}$ and E2GK parameters update
if a new prototype is created then
    Add a new model
    Estimate parameters of the new model
    Update parameters of existing models
end if
Predict the new output $\hat{x_{k+1}}$ (Eq.14)
```

## 4. Experiment

The proposed EG2K-pro algorithm is designed for the prediction at $t+1$. Further predictions requires other developments which are under study. So we are in the same case as in (Ramdani et al., 2005) where we assess the algorithm for the prediction of signals at $t+1$.

Experiments were conducted on three applications:

- the 1-D Mackey-Glass chaotic time series,
- a multidimensional case: the PHM 2008 challenge data,
- a multidimensional case: the PRONOSTIA platform.

### 4.1 A benchmark 1-D problem

As a first example of application, we consider the Mackey-Glass chaotic time series:

$$x(t) = \frac{a \cdot x(t-\tau)}{1 + x^{10}(t-\tau)} - b \cdot x(t) \ , \qquad (16)$$

with $a = 0.3, b = 0.1, \tau = 20, x_0 = 1.2$ and 100 points.

E2GK parameters were set to $\delta = 10$, $\alpha = 1$, $\beta = 2$, $\theta = 0.01$ and $P_{\text{tol}} = 10$, and inputs were composed of $[t \;\; x(t-2) \;\; x(t-1) \;\; x(t)]$.

Figure 1 depicts the prediction at $t+1$, with a mean-squared error (MSE) of $2.10^{-2}$. One interesting feature of the proposed algorithm is the degree of conflict (Fig. 1, bottom). As expected the degree of conflict is low around prototypes and increases when data points are far from the latter. The increasing of conflict is mainly encountered in non-linear parts. For example, around $t = 200$ and $t = 275$, the increasing is much more important than in interval $[1, 150]$.

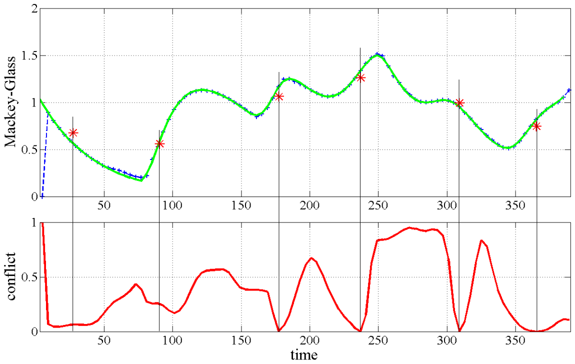


Figure 1. Top: Predictions (crosses), truth (continuous line) and segmentation (stars). Bottom: Conflict evolution.

Let consider the maximum degree of belief generated by the clustering phase at instant (data-point) $t$:

$$s_t = \max_k m_{tk} \qquad (17)$$

If the maximum is low, then the confidence in the prediction should be also low. This quantity is illustrated in Figures 2 (E2GK) and 3 (EGK[1]) where at each instant, the value of $1 - s_t$ appears as an error bar.

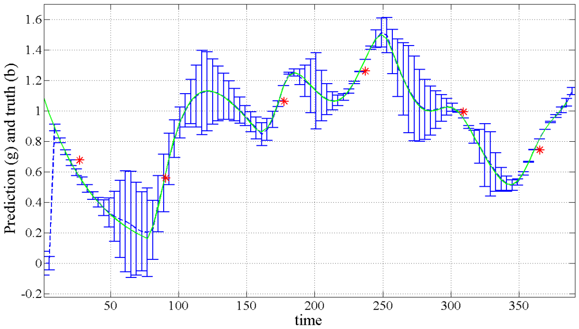


Figure 2. Illustration of $1 - s_t$ (Eq. 17) to quantify uncertainty around predictions by E2GK. These uncertainties appear on the figure with error bars around predictions. The continuous line corresponds to the truth.

In both methods (EGK and E2GK), the maximum degree is close to 1 when points are located near clusters and therefore the values of $1 - s_t$ are close to 0. The main difference is that values for E2GK present more contrast than the ones for EGK. For example, in $[175, 275]$, the error is almost constant for EGK while in E2GK it increases as the distance to clusters increase. Moreover, for E2GK, high values are generally encountered in non-linearities. In these

[1] We implemented the prediction based on EGK algorithm (Georgieva & Filev, 2009) in order to assess the difference with E2GK.

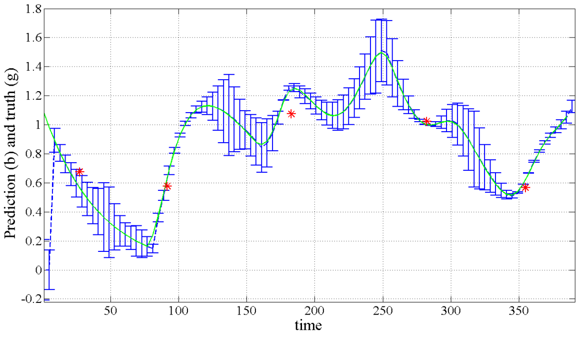


Figure 3. Illustration of $1 - s_t$ (Eq. 17) to quantify uncertainty around predictions by EGK (to be compared with Figure 2). The continuous line corresponds to the truth.

areas, the value of conflict is generally high because points can belong to several clusters. These two figures also show the predictions (continuous line).

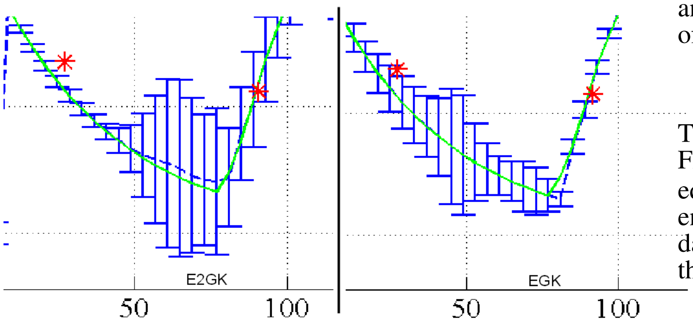


Figure 4. Close-up view to emphasize the differences between EG2K-pro (left) and EGK-based predictions (right). See text for comments.

Figure 4 depicts the difference between both approaches for the predictions around time instant $t = 80$. For E2GK-pro (left-hand side), the predictions are farther from the truth than EGK-based predictions (right-hand side) before $t = 80$, whereas E2GK-pro provides better predictions after $t = 80$ than EGK. However, when E2GK-pro is predicting far from the truth, it is able to provide a degree of ignorance that is high (big error bars), therefore it warns the user about the possible errors. On the contrary, EGK-based predictions provides low values of ignorance meaning that the algorithm is certain about its (wrong) predictions.

### 4.2 A multi-dimensional case: the PHM 2008 challenge data

We considered the challenge dataset concerning diagnostics and prognostics of machine faults from the first Int. Conf. on Prognostics and Health Management (2008) (Saxena, Goebel, Simon, & Eklund, 2008). 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 preselected features $(3, 4, 5, 7, 9)$ was considered.

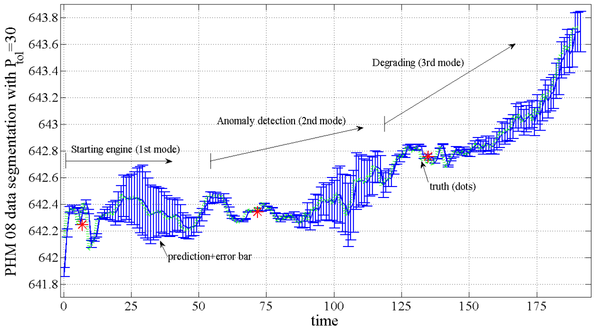


Figure 5. Segmentation (stars) by E2GK with $P_{\text{tol}} = 30$ obtained on PHM 2008 data. Predictions appear with crosses and the truth with dots. Error bars represent the opposite of the degree of support in each model.

The automatic segmentation obtained by E2GK is given in Figure 5. This figure also depicts the prediction (with MSE equal to $1.6.10^{-4}$) and error bars representing the difference between 1 and the maximum degree of belief for each data point (Eq. 17). E2GK parameters were the same as in the previous section, except $P_{\text{tol}} = 30$.

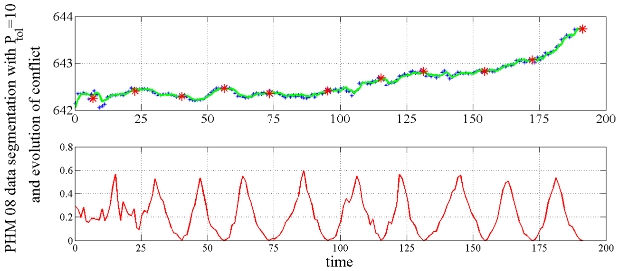


Figure 6. Top: Segmentation (stars) by E2GK with $P_{\text{tol}} = 10$ obtained on PHM 2008 data (to be compared to Figure 5). Predictions appear with crosses and the truth with the continuous line. Bottom: The opposite of the maximum degree of belief (the lower the conflict the higher the confidence in predictions)

These data are highly noisy in some parts, for example in the starting / normal mode. In these cases, E2GK-pro provides smooth predictions with high error bars, for example in $[20, 50]$. After $t = 125$, noise is reduced and the system is degrading. In this area, predictions are close to the truth and uncertainty (Eq. 17) is low meaning high confidence.

In comparison, Figure 6 is the result for $P_{\text{tol}} = 10$ where the segmentation is finer as expected.

### 4.3 A multi-dimensional case: the PRONOSTIA platform

#### Description of PRONOSTIA

PRONOSTIA is an experimentation platform (Figure 7) dedicated to the test and validation of the machinery prognosis approaches, focusing on bearing prognostics. It was developed at FEMTO-ST institute ("Franche-Comté Electronics, Mechanics, Thermal Processing, Optics - Science and Technology") in particular in AS2M department (Automatic control and Micro-Mechatronic Systems).

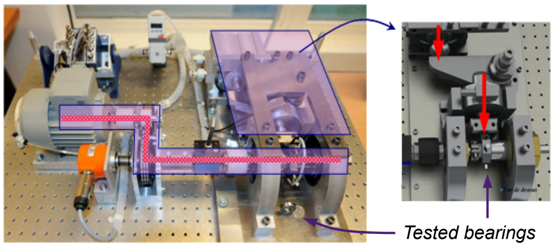


Figure 7. PRONOSTIA platform.

The main objective of PRONOSTIA is to provide real experimental data that characterise the degradation of a ball bearing along its whole operational life (until fault/failure). The collected data are vibration and temperature measurements of the rolling bearing during its functioning mode.

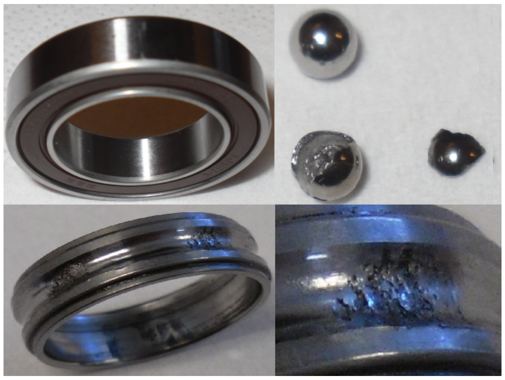


Figure 8. Bearing degradation.

The internal bearing ring is put in rotation, while the external bearing ring is maintained fixed. A radial load is applied on the external bearing ring in order to simulate its functioning. To speed up the degradation, the load exceeds the maximal load recommended by the supplier. The originality of this experimental platform lies not only in the conjunction of the characterization of both the bearing functioning (speed, torque and radial force) and its degradation (vibrations and temperature), but also in the possibilities, offered by the platform, to make the operating conditions of the bearing vary during its useful life. Figure 8 depicts a bearing before and after the experiment.

The bearing operating conditions are determined by instantaneous measures of the radial force applied on the bearing, the rotation speed of the shaft handling the bearing, and of the torque inflicted on the bearing. During a test, the rolling bearing starts from its nominal mode until the fault state. The bearing behavior is measured using different types of sensors (Figure 9) such as miniaturized acceleration sensors and temperature probe.

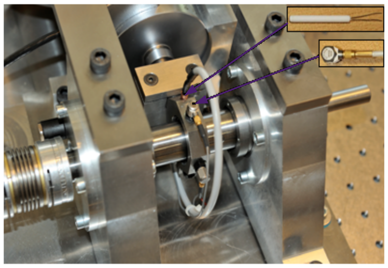
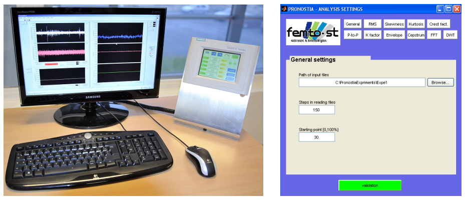


Figure 9. Sensors for degradation measurement.

The raw signals provided by the sensors are processed in order to extract relevant information concerning bearings states. Several techniques have been implemented and gathered in a signal processing toolbox with Matlab (Fig. 10): time-domain methods (RMS, skewness and kurtosis, crest factor, K-factor, Peak-to-Peak), frequency-domain methods (spectral and cepstrum analysis, envelope detection), time-frequency domain (short-time fourier transform) and wavelets (discrete transform).



General settings

Path of input files: C:\PronostiaExpriments\Expe1

Steps in reading files: 150

Starting point [0,100%]: 30

Figure 10. (left) Labview VI for raw signal visualization and (right) the graphical user interface to set the optional parameters (if required) of the signal processing algorithms.

#### Application of E2GK-pro on PRONOSTIA

E2GK parameters were the same as in the first section, except $P_{\text{tol}} = 20$. The prediction results are given in Figure 11 with a MSE equal to $6.10^{-5}$. In the first subfigure, predictions appear with crosses while the truth is given by a continuous line. At bottom, the signal represents the value of Eq. 17 at each instant. As expected, the highest values are encountered in the transition phase (in $[100, 115]$).

The obtained segmentation (in clusters) is provided in Figure 12. In comparison, Figure 13 is the result of segmentation for $P_{\text{tol}} = 10$. For this value, EGK was not able to provide a segmentation. As expected, the number of clusters is greater for this latter value and "over"-segmentation appears mainly in areas with changes.

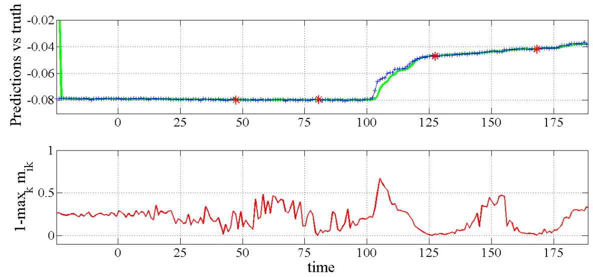


Figure 11. Top: The predictions for PRONOSTIA's data (continuous line). Bottom: The opposite of the maximum degree of belief.

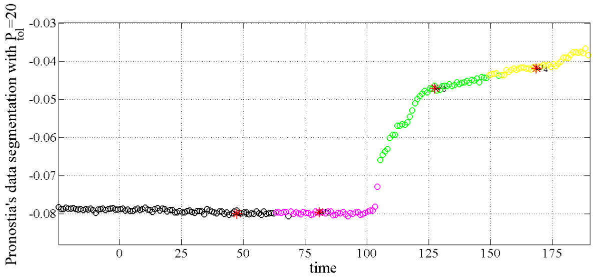


Figure 12. Segmentation (stars) by E2GK with $P_{\text{tol}} = 20$ obtained for a bearing in PRONOSTIA's platform.

## 5. Conclusion

E2GK-pro is an evidential approach proposed for detecting, adapting and combining local models in order to analyse complex systems behavior. The approach relies on three main processes: 1) an online clustering called E2GK that generates belief functions and adapts its structure gradually, 2) the creation, adaptation or removing of models which are locally computed for each cluster, and 3) prediction of the future evolution.

Experiments were done on three datasets: one simulated and two real-world problems, in particular the PRONOSTIA platform. Results demonstrate the ability of the proposed method for online segmentation of multidimensional time-series and to build provide predictions for the next iteration. We also proposed a confidence value attached to predictions.

Future work is mainly focused on the validation of the proposed methodology for long term prediction and to its comparison to Angelov's methodology (P. Angelov et al., 2008).

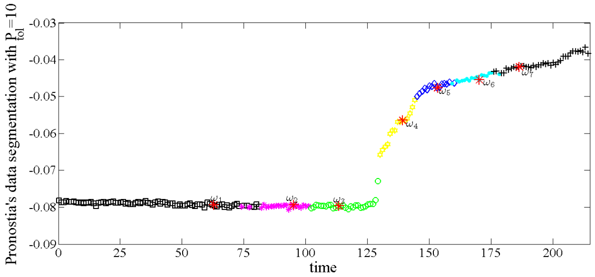


Figure 13. Segmentation (stars) by E2GK with $P_{\text{tol}} = 10$ for a bearing in PRONOSTIA's platform.

## References

Angelov, P., Lughofer, E., & Zhou, X. (2008). Evolving fuzzy classifiers using different model architectures. *Fuzzy Sets and Systems*, 3160-3182.

Angelov, P. P., & Filev, D. P. (2004). An approach to online identification of Takagi-Sugeno fuzzy models. *IEEE Trans Syst Man Cybern B Cybern*, *34*, 484-98.

Cobb, B. R., & Shenoy, P. P. (2006). On the plausibility transformation method for translating belief function models to probability models. *International Journal of Approximate Reasoning*, *41*, 314-330.

Denoeux, T. (2000). A Neural network classifier based on Dempster-Shafer theory. *IEEE Trans. Syst., Man, Cybern*, *30*, 131-150.

El-Koujok, M., Gouriveau, R., & Zerhouni, N. (2011). Reducing arbitrary choices in model building for prognostics: An approach by applying parsimony principle on an evolving neuro-fuzzy system. *Microelectronics Reliability*, *51*, 310-330.

Georgieva, O., & Filev, D. (2009). Gustafson-Kessel Algorithm for Evolving Data Stream Clustering. In *International Conference on Computer Systems and Technologies - CompSysTech 09.*

Gustafson, E., & Kessel, W. (1978). Fuzzy clustering with a fuzzy covariance matrix. In *IEEE Conference on Decision and Control.*

Madani, K., Rybnik, M., & Chebira, A. (2003). Non Linear Process Identification Using a Neural Network Based Multiple Models Generator. *LNCS series*, 647-654.

Masson, M.-H., & Denoeux, T. (2008). ECM: An evidential version of the fuzzy c-means algorithm. *Pattern Recognition*, *41(4)*, 1384 - 1397.

Petit-Renaud, S., & Denoeux, T. (1999). Regression analysis using fuzzy evidence theory. *Proceedings of FUZZ-IEEE*, *3*, 1229-1234.

Ramdani, M., Mourot, G., & Ragot, J. (2005). A Multi-Modeling Strategy based on Belief Function Theory. In *CDC-ECC '05.*

Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008). Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation. In *IEEE Int. Conf. on Prognostics and Health Management.*

Serir, L., Ramasso, E., & Zerhouni, N. (2011). E2GK: Evidential Evolving Gustafsson-Kessel Algorithm For Data Streams Partitioning Using Belief Functions. In *11th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty.*

Smets, P., & Kennes, R. (1994). The Transferable Belief Model. *Artificial Intelligence*, *66*, 191-234.

Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its application to modeling and control. *IEEE Trans. On Systems Man and Cyberneticc*, *15*, 116-132.

Yager, R. R., & Filev, D. P. . (1995). Including probabilistic uncertainty in fuzzy logic controller modeling using Dempster-Shafer theory. *IEEE Trans. Syst., Man, Cybern.*, *25*, 1221-1230.