Simplified Fuzzy Rule-based Systems using Non-parametric Antecedents and Relative Data Density

Plamen Angelov, *Senior Member IEEE*, Ronald Yager, *Fellow IEEE*

Abstract — **In this paper a new method for definition of the antecedent/premise part of the fuzzy rule-based (FRB) systems is proposed. It removes the need to define the membership functions per variable using often artificial parametric functions such as triangular, Gaussian etc. Instead, it strictly follows the real data distribution and in this sense resembles particle filters. In addition, it is in a vector form and thus removes the need to use logical connectives such as AND/OR to aggregate the scalar variables. Finally, it uses the relative data density expressed in a form of a parameter-free (Cauchy type) kernel to derive the activation level of each rule; these are then fuzzily weighted to produce the overall output. This new simplified type of FRB can be seen as the next form after the two popular FRB system types, namely the Zadeh-Mamdani and Takagi-Sugeno. The new type of FRB has a much simplified antecedent part which is formed using data *clouds*. Data clouds are sets of data samples in the data space and differ from clusters significantly (they have no specific shape, boundaries, and parameters). An important specific of the activation level determined by relative density is that it takes directly into account the distance to *all* previous data samples, not just the mean or prototype as other methods do. The proposed simplified FRB types of systems can be applied to off-line, on-line as well as evolving (with adaptive system structure) versions of FRB and related neuro-fuzzy systems. They can also be applied to prediction, classification, and control problems. In this paper examples will be presented of an evolving FRB predictor and of a classifier of one rule per class type which will be compared with the traditional approaches primarily aiming proof of concept. More thorough investigation of the rich possibilities which this innovative technique offers will be presented in parallel publications.**

Index terms — **fuzzy rule-based systems, Mamdani and Takagi-Sugeno fuzzy systems, RLS estimation, data density.**

I. INTRODUCTION

Traditionally, two types of fuzzy rule-based (FRB) systems have been used, respectively called (Zadeh-) Mamdani [1], [2] and Takagi-Sugeno (TS) [3]. It should be noted that there are other types, such as so called relational FRB [4] which are, however, not widely used due to conceptual and computational difficulties. These two well known types of FRB differ by the form of their consequents, but share the same form of antecedents/premise part, which is scalar fuzzy-sets-based. For over two decades to the best of our knowledge there was no attempt to investigate alternative antecedent types. In this paper we propose and demonstrate the advantages of using a much simplified antecedent thus proposing a viable alternative to both traditional types of FRB systems.

In both, Mamdani and TS type FRB systems the antecedent part is determined by a number of fuzzy sets (one per each variable) which are defined by parameterized scalar membership functions. One way to determine these membership functions is by experts. This approach was popular at the dawn of the fuzzy logic theory development and is still popular in certain type of applications such as decision support systems, but is related to numerous difficulties and obstacles, such as the existence of different methods for aggregation of the degrees of membership functions that takes place in order to determine the degree of activation of a fuzzy rule [5]; different types of membership functions exists and their choice and parameterization is not unique; membership functions often differ significantly from the real data distribution.

In this paper we propose an entirely new concept to the way the antecedent part is defined. Based on this, a new simplified type of FRB is proposed as an alternative to both Mamdani and TS type FRB. The core and the strength of the FRB is the interpolation between locally defined and fuzzily connected simple models. Each local sub-system, however, is defined in both the Mamdani and TS type of systems in the same way that is often represented by overlapping clusters [19]-[23],[26],[27]. We propose to revise this widely popular concept and consider a 'bottom-up' approach rather than the existing 'top-bottom' approach where the type of the data distribution (the membership function) is pre-defined and fixed. We propose the membership function to be extracted from the data and thus we introduce and use data *clouds* as opposed to data clusters being currently used.

The overall new simplified type of FRB (which can be called for brevity ANYA by the first letters of the names of the authors) can be seen as an extension of the well known concepts of the case-base reasoning [6] and k nearest neighbors [8] but with a much more sophisticated mathematical underpinning being computationally and conceptually richer (it assumes fuzzy membership of a data sample to more than one data *cloud* at the same time with different degree of association/membership determined by the local density to **all** samples from that *cloud*).

The problems related to the adequacy of the membership functions and their parameterization are well known. The traditional approach [1]-[5],[11] is to use pre-defined and fixed membership functions of triangular, Gaussian, trapezoidal etc. type. In statistical machine learning [8],[9]

978-1-4244-9979-3/11/$26.00 ©2011 IEEE

standard pre-defined kernels, e.g. Gaussian, Cauchy, Epanechnikov etc. also dominated until recently when non-parametric representations focused on the **real** data distribution and density started to be used in particle filters [7]. Similarly to the particle filters, the antecedent part of ANYA FRB is *non-parametric* and represents **exactly** the **real** data density and distribution. In addition, the antecedents of ANYA FRB take into account the density to **all** previous data samples and can be calculated recursively, thus being applicable to on-line, real time, one pass algorithms unlike the particle filters. In ANYA there is no need to define centers/prototypes/focal points of the fuzzy sets and there is no hard requirement to use a particular type of distance, e.g. Euclidean, Mahalonobis, cosine etc. At the same time, transformations to traditional membership functions per variable are still possible for visualization purposes as demonstrated in the next section, but they are not required for computational derivations and manipulations.

The proposed concept touches the very foundations of the complex systems identification and therefore numerous applications and extensions can arise from it in the areas of clustering, modeling, prognostics, classification and time-series prediction. Because of the limitation of space and the introductory nature of this paper, however, we will limit to several numerical examples aiming merely a proof of concept.

II. THE NEW SIMPLIFIED ANTECEDENTS BASED ON RELATIVE DATA DENSITY

Table I provides a comparison of the two existing popular methods for FRB and the newly proposed one. As it is obvious, the Mamdani and TS types of FRB differ by their consequent part and defuzzification technique used while sharing the same type of antecedent part.

TABLE I A COMPARISON OF DIFFERENT TYPES OF FRB

	ANTECEDENT (IF)	**CONSEQ. (THEN)**	**DEFUZZIFICATION**
MAMDANI [1,2]	Fuzzy sets (scalar, parameterized)	Fuzzy sets (scalar, parameterized)	Centre of Gravity
TS [3]		Functional (often linear)	Fuzzily weighted sum
ANYA (THIS PAPER)	Data *clouds* (non-parametric)	Any of the above two types	

The proposed innovation concerns the antecedent part where a non-parametric approach is used that takes into account **all real data** and the relative data distribution. The remainder of the FRB can be the same as the TS type FRB although it can, in principle, also be the same as in Mamdani type FRB.

The formulation of the membership functions needed for the antecedent part is often a stumbling block in the practical design of FRB systems. This is true both in the case when their design relies on real data [19]-[23],[26],[27] as well as when it relies on expert knowledge [1]-[5], [11]. The reason is that defining membership functions per scalar variable and parameterization of all of them requires a very high level of approximation (because the real data distributions and real problems are often not smooth and easy to describe 'per variable' and by Gaussians, triangular or trapezoidal functions). Addressing this important bottleneck of the FRB systems design we propose a simplified and effective new form of antecedent/premise (IF) part that covers various types of systems, including but not limited to MIMO FRB and neuro-fuzzy systems (Fig.1). Moreover, the neuro-fuzzy interpretation of ANYA is much simpler than the respective TS type neuro-fuzzy systems used in approaches such as ANFIS [10], DENFIS [27], eTS+[18], SAFIS [26], ePL [20], SOFNN [19], evolving trees [21], etc. having fewer layers and parameters.

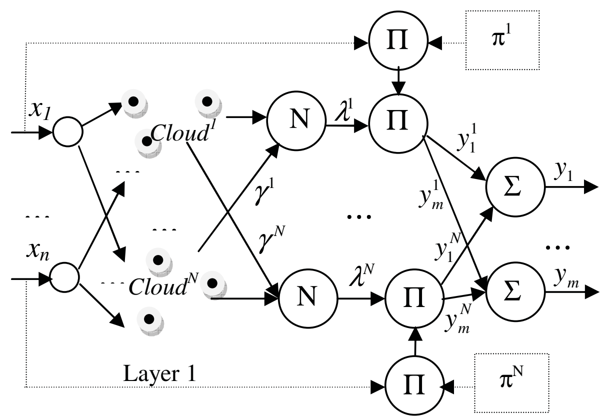

Figure 1 ANYA as a neuro-fuzzy systems.

We propose simplified FRB of the following form:

$$\textbf{ANYA}: \ Rule^i : IF \left(x \sim \aleph^i\right) \quad THEN \left(y^i\right) \tag{1}$$

where ~ denotes the fuzzy membership expressed linguistically as '*is associated with*'; $Rule^i$ denotes the i^{th} fuzzy rule; $i=1,2,\ldots,N$ denotes the number of fuzzy rules; N is the overall number of fuzzy rules; $\aleph^i \quad \aleph \in R^n \quad i=[1,N]$ denotes a *cloud* of input data (a subset of real data with similar properties; $x=[x_1,x_2,\ldots,x_n]^T$ and $y^i=[y_1^i,y_2^i,\ldots,y_m^i]$ denote input and output vectors respectively.

This set of fuzzy rules (1) can describe a complex, generally non-linear, non-stationary, non-deterministic system that can only be observed through its inputs and outputs. The aim is to describe the *IF X THEN Y* dependence based on history of pairs of past observations, $z_j=[x_j^T;y_j^T]^T$, $j=1,2,\ldots k-1$; $z\in R^{n+m}$ plus the current, k^{th} input, x_k^T.

Both, Mamdani and TS type of traditional FRB use antecedents described as:

$$IF\left(\left(x_1 is\ LT_1^i\right) ...AND\ \left(x_n is\ LT_n^i\right)\right) \qquad (2)$$

where $x_j; j=[1,n]$ denotes the j^{th} scalar input variable; $LT_j^i; i=[1,N]; j=[1,n]$ denotes the j^{th} linguistic term (e.g. *Small, Medium, Large* etc.) for the i^{th} fuzzy rule.

There is a variety of aggregation operators that describe mathematically the logical connectives such as AND, OR and NOT [5] used in this traditional FRB and each one leads to somewhat different result. In some practical applications as spectroscopy [12] the number of scalar inputs may be in the order of hundreds which hampers the design and interpretation of such types of antecedents.

The new alternative proposed in this paper is based on data *clouds* formed using relative density in the data space which can be calculated and updated recursively. These data *clouds* are then used as building blocks of the fuzzy rules. Despite some resemblance to the clustering there are several major differences between data *clouds* and clusters, see Figure 2 and Table II.

TABLE II CLOUDS VS CLUSTERS

FEATURE	CLUSTERING	GRANULATION
Boundaries	Defined	None
Centre/prototype	Defined	None
Distance to	Centre/ prototype	**all** data (accumulated)
Membership functions	Scalar	Vector
	Parameterized	non-parametric
	approximation of an ideal	reflects real data distribution

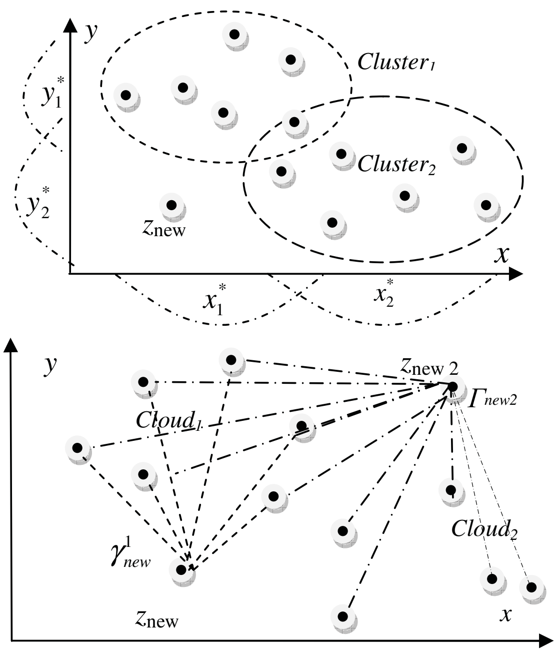

Figure 2 A graphical comparison of clusters and clouds. Top – the traditional partitioning using clustering and membership functions; bottom - ANYA approach using clouds with no boundaries and shapes. Both local (γ) and global densities (Γ) are also visualized.

The main difference is that the proposed approach does not consider and does not require membership functions or fuzzy sets per scalar variable to be formulated *a priori* and to be pre-fixed for the duration of the usage of the FRB system as the traditional approaches do. Therefore, ANYA can also be seen as **type *0* fuzzy sets** (by analogy to the type II fuzzy sets for which the membership functions are defined by a fuzzy set for each point of the membership function). In contrast to the traditional (type I) and to the more complex type II fuzzy sets [29], the proposed ANYA approach **does not require** an **explicit** definition of the membership function or even a prior assumption of its form).

Data *clouds* are sub-sets of previous data samples with common properties (closeness in the observation data space). They **directly and exactly** represent **all** previous data samples**.** Contrast this to the traditional membership functions which usually do **not** represent the true data distributions; instead, they represent some desirable/expected/estimated (often subjectively) preferences. The fuzziness of the proposed method is preserved in the manner of decomposition in the sense that a particular data sample can belong to all *clouds* with different degree, $\gamma \in [0;1]$. Importantly, *clouds* **do not have** or **require boundaries** and, thus, they do not have specific shapes (see Figure 2 bottom plot and compare it to the top plot). A *cloud* is described by the similarity or likeliness of the sub-set of data samples that are associated with it.

The degree of membership to a *cloud* is measured by the normalized relative density for a particular data sample, x_k:

$$\lambda_k^i = \frac{\gamma_k^i}{\sum_{j=1}^{N} \gamma_k^j} \quad i=[1,N] \qquad (3)$$

where γ^i is the *local* density of the i^{th} *cloud* for a particular data sample, which is defined by a suitable kernel over the distance between the current sample, x_k and **all** the other samples from **that** *cloud* (therefore *local*):

$$\gamma_k^i = K\left(\sum_{j=1}^{M^i} d_{kj}^i\right) \quad i=[1,N] \qquad (4a)$$

where M^i is the number of input data samples associated with the i^{th} *cloud* (a sample is associated with the *cloud* with the highest density); d_{kj}^i denotes the distance between the current sample, x_k and any other sample, x_j of that (i^{th}) *cloud*.

Similarly, *global* density Γ for a particular data sample, z_k which is defined by a suitable kernel over the distance between the current observable data sample, z_k and **all** the other previously observed samples (therefore *global*):

$$\Gamma_k = K\left(\sum_{j=1}^{k} d_{kj}\right) \qquad (4b)$$

Different types of distance measures can be used (each having its own advantages and disadvantages [19]) such as Euclidean ($\left[d_{kj}^i\right]_E^2 = \left\|x_k - x_j\right\|^2$ or $\left[d_{kj}\right]_E^2 = \left\|z_k - z_j\right\|^2$), cosine distance, $\left[d_{kj}\right]_{\cos} = \frac{z_k z_j}{\left\|z_k\right\|\left\|z_j\right\|}$, etc.

The kernel is a well known measure of similarity and Cauchy type of kernel is specifically interesting. The *local* density can be defined by a Cauchy type kernel as:

$$\gamma_k^i = \frac{1}{1+\frac{\sum_{j=1}^{M^i}\left(d_{kj}^i\right)^2}{M^i}} = \frac{M^i}{M^i + \sum_{j=1}^{M^i}\left(d_{kj}^i\right)^2} \tag{5}$$

It can be proven that Cauchy type kernel asymptotically tends to Gaussian, but can be calculated recursively [14,15]:

$$\gamma_k^i = \frac{M^i}{M^i\left(z_k^T z_k + 1\right) - 2\alpha_k + \beta_k} \tag{6}$$

where $\alpha_k = z_k^T \xi_k$; $\xi_k = \xi_{k-1} + z_{k-1}$

$$\xi_0 = 0;\ \beta_k = \beta_{k-1} + \left\|z_{k-1}\right\|^2;\ \beta_0 = 0.$$

In a much similar way the **global** density (see Figure 2 bottom plot) can be defined with a Cauchy type kernel as:

$$\Gamma_k = \frac{1}{1+\frac{\sum_{j=1}^{k-1} d_{kj}^2}{k-1}} = \frac{k-1}{k-1+\sum_{j=1}^{k-1} d_{kj}^2} \tag{7}$$

or recursively:

$$\Gamma_k = \frac{k-1}{(k-1)\left(z_k^T z_k + 1\right) - 2\mathrm{A}_k + \mathrm{B}_k} \tag{8}$$

where $\mathrm{A}_k = z_k^T \Xi_k$; $\Xi_k = \Xi_{k-1} + z_{k-1}$

$$\Xi_0 = 0;\ \mathrm{B}_k = \mathrm{B}_{k-1} + \left\|z_{k-1}\right\|^2;\ B_0 = 0$$

It is easy to check that because of the way equation (3) is formulated the degree of fuzzy membership to a data *cloud*, λ^i is normalized, that is:

$$\sum_{i=1}^{N} \lambda^i = 1 \tag{9}$$

In Mamdani type FRB centre-of-gravity defuzzification method is used while in the TS type FRB it is done by fuzzily weighted averaging; for problems such as classification so called 'winner takes all' inference operator is usually used which forms the output based on the rule with the higher firing level [5,11]. ANYA type of FRB systems can work with any of the above as well as some mixed forms such as parameterized defuzzification [13] or 'few winners take all'. In the case of fuzzy weighted average we have:

$$y = \sum_{i=1}^{N} \lambda^i y^i \tag{10}$$

where y^i represents the output of the i^{th} local sub-system.

Returning again to the neuro-fuzzy interpretation of ANYA in comparison to the TS models (which are used for example in ANFIS) from Figure 1 it is obvious that ANYA has a simpler structure using fewer layers. In addition, in layer 1 no scalar parameterized membership functions are required to be pre-defined and pre-fixed (as is the case with TS type FRB, respectively ANFIS); instead, in ANYA a given input data sample, x_k is compared to **all** previous data samples **per data cloud** (note that this can be done recursively) and the *local* density of each *cloud* in terms of this data sample is calculated by (6). The second layer of the network takes as input the density of the respective *cloud*, γ^i and gives as output the normalized firing level of the fuzzy rule (which is the membership to the i^{th} *cloud*), λ^i using (3). The first two layers represent the antecedent part of the fuzzy rules (for comparison ANFIS uses 3 layers to produce the normalised firing level of a particular rule, λ^i). Layer 3 aggregates the antecedent and consequent parts which represent the local sub-system. Finally, layer 4 forms the total output. It performs a weighed summation of local sub-systems according to equation (10).

We will demonstrate with illustrative purpose how a *2D* scalar membership function can be determined per scalar input. On the vertical axis we plot the values of the relative density, $\gamma_{kj}^i \quad i=[1,N];\ j=[1,n]$ for a particular *cloud i* calculated for a particular input, x_j only estimated at the k^{th} time instant. On the horizontal axis the values of this particular input, x_j are depicted.

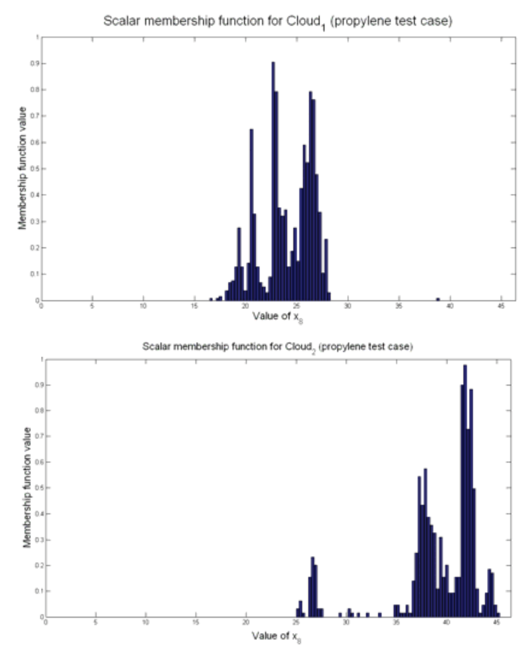

Figure 3 Membership function per cloud can be generated for visualization purposes, but they are not needed explicitly for the inference. Vertical axis denotes the degree of membership to a cloud, horizontal axis - the cardinality of x; top plot - Cloud$_1$; bottom plot – Cloud$_2$. The non-Gaussian character of the membership functions is obvious.

Note that this *2D* visualization is not necessary for the inference but provides an illustration and a link to the traditional fuzzy sets which have scalar *2D* membership functions. It also demonstrates that the proposed approach is much closer to the real data distribution than traditional scalar parameterized membership functions and parameterized distributions which are usually Gaussian, triangular, trapezoidal, etc.

It is also interesting to note that this membership function only takes value of *1* if *all* data of a *cloud* coincide which is extremely rare while traditional membership functions have one or more points with value 1.

III. Designing ANYA FRB Systems

The design and identification of any system consists of two basic parts [11],[16],[17]:

i) structure identification, and
ii) parameter identification.

A. Structure Identification

In the literature, the problem of system structure identification (in general, not only and even not so much for the case of FRB systems) is usually left to the choice of the system designer [11],[16],[17]. This problem started to be treated more intensively since the Mountain clustering approach [22] was proposed to be used to automatically solve the problem of FRB systems design in 1993 by Yager and Filev. This concept was taken further in the modified version of the Mountain clustering widely known as subtractive clustering [23] and, more notably, by the innovative concept of *system structure evolution* [14]-[15], [18]. The ANYA approach is based on the underlying real data density pattern as a driver for the complex system automatic structure identification.

In general, the proposed approach applies to all forms of operation (system designed based on a separate set of training data – off-line; system designed during the data collection and usage – on-line, and system with active structure identification during system design and usage – evolving). In this paper we will represent one application to a single rule per class FRB classifier design which can work off-line or on-line, but is not evolving (system structure is pre-defined with the number of fuzzy rules equal to the number of classes) and to an evolving on-line predictive FRB models design aiming a proof of concept. More extensive study is currently taking place and further results will be published in other publications.

The case of the single rule per class FRB classifier is clear from the system structure point of view – the structure is rigidly fixed and directly linked to the specific problem in the sense of number of possible classes and respectively rules.

For the on-line and evolving case, the following principles are followed [15],[18];

A) good **generalization and summarization**– this is achieved by forming new *clouds* from data samples which have ***high global density***, Γ
B) avoid **excessive overlap**, **old** *clouds* or the ones that are **rarely utilized**
C) Maintain the quality of the *clouds* on-line and remove irrelevant or outdated *clouds*.

1) One rule per class simplified FRB classifier

We consider a very simple yet effective FRB classifier that has a single rule for each class. That means that we assume that all data of a given class form a single data *cloud* (in a more general case one can have more than one *cloud* per class either in an off-line manner or evolving them from data increasing in this way the level of granularity):

$$Rule^i: \ IF \ \left(x \sim Cloud^i\right) THEN \ \left(x \rightarrow Class^i\right) \quad (11)$$

where $i=1,2,\ldots,C$; C is the number of classes;
$Class^i$ denotes the label of the i^{th} class.

This FRB classifier will always have exactly C fuzzy rules and the antecedent of each rule will be formed by a single kernel (unlike in traditional fuzzy sets where the antecedent is an aggregation of fuzzy sets per input feature). The classification itself can be performed based on the well known principle called 'winner takes all' which is often used in classification [35]:

$$Class = \arg\max_{j=1}^{C}\left(\lambda_j\right) \quad (12)$$

FRB classifiers can, generally, be of two types [35]; i) zero order when consequents of the rules constitute of the class labels, and ii) first order when the consequents of the rules are linear. For the former case there are no parameters in the consequent part and for the latter case parameters can be found as described in the next sub-section (for the prediction and estimation case). We will consider in the next section a numerical example of a one-rule per class ANYA FRB of zero order which is fully non-parametric.

It is important to note that this classifier can be used off-line or in an incremental (not evolving!) mode (the data samples can be processed one by one and because of the recursive expression (6) there is no need to memorize previous data samples). It is not evolving, however, because the number of rules is fixed (equal to C). As a remark, often in literature, for example [31], evolving is mixed with incremental. This is a good example to demonstrate that evolving is more than just incremental. It is also important to note that this simplified FRB classifier is a typical incremental classifier that does not require a separate training and validation data sets and iterative procedures. An application of this simple, yet effective classifier will be demonstrated in the next section.

2) Evolving simplified FRB predictor/estimator

In this particular application of ANYA a dynamically evolving FRB for prediction and estimation is considered with a structure starting either from an initially existing set of rules (this may be designed off-line or suggested by an expert) or, if such initial structure does not exist, 'from scratch'. It then forms new *clouds* (evolves the structure of the FRB system) during the usage of the system (on-line). Let us assume the more challenging case of 'starting from scratch'. The very first data sample per class, naturally, starts formation of the first data *cloud*. For all further data samples there are two possibilities:

1) they are associated with the existing *clouds* updating the local density of the nearest one;
2) they initiate a new *cloud*.

The first one is obvious and it invokes update of equation

(6). The second case concerns new data samples for which the *global density* calculated at these points *is higher* than the global density estimated at the initial points of all existing clouds (principle A):

$$\Gamma_k > \Gamma_k^i \quad \forall i \middle| i = [1, N] \tag{13}$$

Note that a new *cloud* is initiated ($z_k \rightarrow z^*$) when condition (13) is satisfied for **all** existing *clouds* ($\forall i$) which is not very often for real data.

Finally, we check for each candidate to start a new *cloud* (one that satisfies (13)) if this data sample's satisfies so called 'one sigma' condition [8] (principle B):

$$\exists i; i = [1, N] \middle| \quad \gamma_k^i > e^{-1} \tag{14}$$

If (14) is also satisfied, a new cloud is NOT formed even if condition (13) is satisfied.

B. *Parameter Identification/Learning*

The problem of parameter identification was traditionally much more studied and usually addressed as an optimization problem [10],[11],[16],[24]. It is important to stress that in ANYA the total number of parameters needed is significantly (can be in orders of magnitude in some problems) lower as compared to the traditional FRB systems. For example, in traditional FRB systems the total number of parameters, TNP can be determined as *TNP=NAP+NCP* (where *NAP* is the number of antecedent parameters, and *NCP* is the number of consequent parameters). If use (which is often the case) Gaussian scalar membership functions *NAP=2*n*N* (where *n* is the number of input variables/features and *N* is the number of rules) and *TNP* is equal to *N*(n+1)*. In total, a traditional FRB requires *N*(3n+1)* parameters to be determined. In ANYA the antecedent part is **parameter-free** (*NAP=0*). This will be demonstrated on real industrial data in the next section.

Therefore, parameter identification only involves learning the consequent part's parameters. This is of principally important, because consequents parameters can be learned locally by approaches such as least squares while the identification of antecedent parameters always involves numerical search procedures which are iterative and with possible convergence problems because of the non-linearity in terms of the antecedent parameters.

The consequent parameters depend on the specific type of consequences used. More often, linear consequences are used such as in TS types FRB (although this should not necessarily be a limitation to the ANYA concept):

$$consequent^i: \; y^i = x_e^T \pi^i \tag{15}$$

where $\pi^i = \begin{bmatrix} a_{01}^i & a_{02}^i & \cdots & a_{0m}^i \\ a_{11}^i & a_{12}^i & \cdots & a_{1m}^i \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1}^i & a_{n2}^i & \cdots & a_{nm}^i \end{bmatrix}$ are the consequent sub-system parameters (in this MIMO system the output is m dimensional, i.e. $y \in R^m$); $x_e^T = [1, x^T]$ denotes the extended inputs vector.

As it was demonstrated for the evolving TS type of FRB systems in [13]-[15], [18], [19] this can be solved successfully as a locally valid RLS estimation problem or fuzzily weighted RLS (fwRLS). First, the overall FRB system output is expressed in a vector form [13]-[15], [18], [19] as:

$$y = \psi^T \theta \tag{16}$$

where $\theta = \left[(\pi^1)^T, \ldots, (\pi^N)^T \right]^T$ is formed by the sub-system parameters; $\psi = [\lambda^1 x_e^T, \ldots, \lambda^N x_e^T]^T$ is a vector of the inputs that are weighted by the normalized activation levels of the rules, λ^i, *i=[1,N]* for the linear consequents, and $\psi = [\lambda^1, \lambda^2, \ldots, \lambda^N]^T$ for the singleton type consequents.

For a given data point, x_k the optimal in LS sense solution, $\hat{\theta}$ that minimizes the following cost function:

$$\left(Y - \Psi^T \theta\right)^T \left(Y - \Psi^T \theta\right) \rightarrow \min \tag{17}$$

can be found applying fuzzy weighted RLS, fwRLS [18]:

$$\hat{\theta}_k = \hat{\theta}_{k-1} + C_k \psi_k (y_k - \psi_k^T \hat{\theta}_{k-1}) \tag{18}$$

$$C_k = C_{k-1} - \frac{C_{k-1}\psi_k \psi_k^T C_{k-1}}{1 + \psi_k^T C_{k-1} \psi_k} \tag{19}$$

where $\hat{\theta}_1 = 0$; C is a $Nn \times Nn$ co-variance matrix; $C_1 = \Omega I$; Ω is a large positive number; I is the identity matrix; *k=2,3,…*

IV. Numerical Examples

To test the newly proposed concept and method we consider couple of 'proof of the concept' type examples evolving a predictive model and a simple single rule per class type of a classifier. Recognizing the limitations of the demonstrative examples we hope that future publications will cover more applications of this technique. For the evolving predictive model we used one data stream from a well known benchmark and one from real industrial processes. The overall performance of the proposed approach was analyzed based on a comparison of the results of applying more established techniques available in Matlab such as ANFIS (off-line) [10], *genfis2* (off-line) [22,23], DENFIS (evolving with off-line initialization) [27] and eTS+ (evolving) [18]. The main reason this particular methods were used for comparison is that they are available as software widely and they were the cornerstone methods for automatic fuzzy and neuro-fuzzy systems design from data in an off-line and in an on-line, evolving manner.

In the test with ANFIS and *genfis2* the data sets were separated into training and validation sub-sets. The training sets were used for the off-line training and the error in prediction was estimated based on the validation sub-sets.

A. *Box-Jenkins gas furnace data*

The Box-Jenkins data set is one of the well established benchmark problems. It consists of *290* pairs of input/output data taken from a laboratory furnace [28]. Each data sample consists of the methane flow rate, *u(k)*, and the CO_2 concentration in off gas, *y(k)*. From different studies the best model structure for this system is:

$$y(k) = f\big(u(k-4), y(k-1)\big) \tag{20}$$

The difficulty is to determine a good (non-linear) function, f both in terms of its structure and parameters. Traditionally, off-line models use *200* data samples for training and *90* for validation. Evolving models (such as DENFIS, eTS+ or ANYA) do not need to separate training and validation data, but in this experiment we did this in order to have the same conditions for comparison with the off-line counterparts. The values of the performance measures (Table III, the time is shown per sample) were calculated for the validation data using non-dimensional error index (NDEI) and root mean square error (RMSE).

From Table III it is seen that using ANYA a ***simple and compact*** fuzzy model of seven fuzzy rules can be extracted from this data stream with significantly smaller number of parameters and better precision.

TABLE III BOX-JENKINS GAS FURNACE DATA

METHOD	ANFIS	GENFIS2	DENFIS	eTS+	**NEW**
Type	Off-line		Evolving		
RMSE	*0.100*	*0.050*	*0.052*	*0.047*	***0.043***
NDEI	*0.605*	*0.311*	*0.322*	*0.291*	***0.272***
# rules	*25*	*3*	*10*	*7*	*7*
#params	*175*	*21*	*70*	*49*	***21***
# inputs	*2*	*2*	*2*	*2*	*2*
time, ms	-	-	*3.1*	*3.4*	***2.7***

For example, rule 1 derived by this method is:

Rule[1]*:* ***IF*** *(x is like Cloud*[1]*)*

$$\textbf{\textit{THEN}}\ (\ y_k^1 = 0.4008 + \begin{bmatrix} -0.4135 \\ 0.6061 \end{bmatrix} \begin{bmatrix} y_{k-1} & u_{k-4} \end{bmatrix})$$

In ANYA there is no need to pre-define membership functions for the antecedent part.

B. Propylene case study

The propylene data set is collected from a chemical distillation process run at The Dow Chemical Co., USA (courtesy of Dr. A. Kordon) [25]). The data set consists of *3000* readings from *23* sensors that are on the plant. They are used to predict the propylene content in the product output from the distillation. The results (Table IV) demonstrate that ***highly compact*** FRB system which consist of only two fuzzy rules can be extracted from the data stream automatically and this simplified FRB system (intelligent sensor) can model the propylene from a real (noisy) data stream with very good precision.

TABLE IV POLYPROPYLENE DATA

METHOD	ANFIS	GENFIS2	DENFIS	eTS+	NEW
Type	Off-line		Evolving		
RMSE	*Can not cope with dimensionality (crash, memory full)*			*0.157*	***0.137***
NDEI				*0.444*	***0.388***
# rules	*N*	*N*	*N*	*6*	***2***
#params	*70*N*	*70*N*	*70*N*	*38*	***8***
# inputs	*23*	*23*	*23*	*2*	***1***
time, ms	-	-	-	*2.38*	***1.44***

In this case, also there is no need to define Gaussian or triangular membership functions for the antecedent part but one can extract scalar membership function for both fuzzy rules (see Fig. 3). The two Clouds that were formed automatically are depicted in Fig.4.

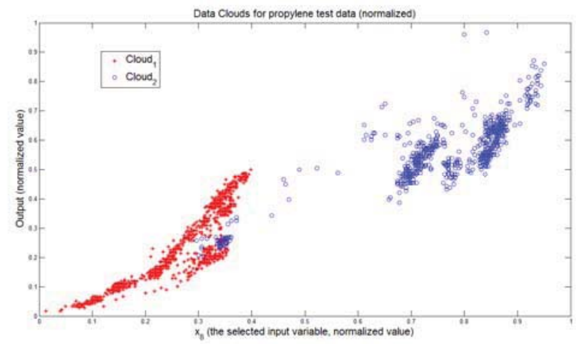

Figure 4 Clouds for the propylene data. Obviously, neither Euclidean (circular shape), nor Mahalonobis (ellipsoidal shape) clusters would correctly and fully represent this real data distribution. The proposed approach, however, will take it fully into account for **all** data samples (see also Figure 3 for the same data samples).

C. ANYA-Class : A simple non-parametric FRB Classifier

We use a well known benchmark data set - wine data [28] to demonstrate the simple FRB classifier. Wine data set contains data from a chemical analysis of wines grown in the same region of Italy but derived from 3 different cultivars (thus, three class labels). The analysis determined the quantities of 13 constituents found in each of the 3 types of wines (the 13 input features/attributes of the classifiers).

Table V and Fig. 5 provide numerical comparison of the results achieved by the proposed simple FRB classifier and other classifiers for the same data set.

TABLE V RESULTS FOR WINE DATA SET CLASSIFICATION

Classifier	Classification rate, %	# of rules
ANYA-Class	***97.44***	***3***
kNN	*96.94*	*3**
eClass 0 [35]	*92.44*	*9.94***
eClass 1 [35]	*97.22*	*6.4****
C4.5	*92.13*	*4.6*

*kNN does not provide any insight of how the result is achieved (model structure) and does not take into account ***all*** data as the newly proposed classifier does; **The rules of *eClass0* have much more complex consequent; ***The rules of *eClass1* have much more complex antecedent and consequent.

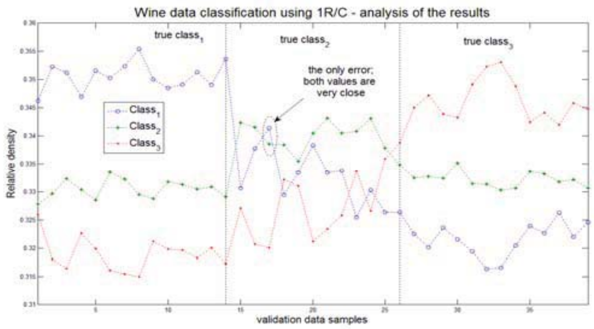

Figure 5 ANYA-Class takes the maximum of the relative density (12) and determines the winning class label. Note the closeness of the values in the validation case *17* (the only one that was misclassified).

It is interesting to note not only a very high classification rate (*97.44%* correct classification), but also the marginal difference between the relative density for the validation sample *17* which was the only sample to be misclassified. The values of the relative density, λ can be provided to the decision maker and can indicate possible problematic cases for further investigation or declaring 'not sure' outcome. It is important to note that the proposed classifier is computationally very light and recursive.

V. Conclusion and Future Direction

In this paper a new method for definition of the antecedent part of the FRB systems is proposed. It removes the need to define the membership functions per variable and strictly follows the *real* data distribution and in this sense resembles particle filters. In addition, it is in a vector form and thus removes the need to use logical connectives such as AND/OR to aggregate the scalar variables. Finally, it uses the relative data density expressed in a form of a parameter-free kernel to derive the activation level of each rule. This new simplified type of FRB called ANYA is using shape- and parameter-free data *clouds* and takes directly into account the distance to *all* previous data samples *exactly*, not just the mean or prototype as other methods do. ANYA can be applied to off-line, on-line as well as evolving versions of FRB or neuro-fuzzy systems. It can be used in prediction, classification as well as control problems. In this paper examples are presented primarily aiming proof of concept. More thorough investigation of the rich possibilities which this innovative technique offers will be presented in future publications.

References

[1] E. H. Mamdani, S. Assilian, An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller, *International Journal of Man-Machine Studies*, vol.7, pp.1-13, 1975.

[2] L. A. Zadeh, Outline of a New Approach to Analysis of Complex Systems and Decision Processes, *IEEE Transactions on Systems, Man and Cybernetics*, vol.1, pp.28-44, 1973.

[3] T. Takagi, M. Sugeno, Fuzzy identification of systems and its application to modeling and control, *IEEE Trans. on Syst., Man & Cybernetics*, vol. 15, pp. 116-132, 1985.

[4] W. Pedrycz, Fuzzy relational equations with generalized connectives and their applications, *Fuzzy Sets and Systems*, vol.10 (1-3), pp. 185-201, 1983.

[5] G. Klir, T. Folger, *Fuzzy Sets, Uncertainty and Information*, Englewood Cliffs, NJ: Prentice Hall, 1988.

[6] I. Watson, Case-based reasoning is a methodology not a technology, *Knowledge-Based Systems*, vol. 12 (5-6), pp.303-308, 1999.

[7] M. S. Arulampalam, S. Maskell, N. Gordon, A Tutorial on Particle Filters for On-line Non- linear/Non-Gaussian Bayesian Tracking, *IEEE Trans. on Signal Processing*, vol. 50 (2), pp. 174-188, 2002.

[8] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Heidelberg, Germany: Springer Verlag, 2001.

[9] V. Vapnik, *Statistical Learning Theory*. Wiley: NY, USA, 1998.

[10] J. S. R. Jang, ANFIS: Adaptive Network-based Fuzzy Inference Systems, *IEEE Trans. on Systems, Man & Cybernetics*, part B – Cybernetics, vol. 23 (3), pp. 665-685, 1993.

[11] R. Yager and D. Filev, *Essentials of Fuzzy Modeling and Control* (John Wiley & Sons, 1994).

[12] J. Kelly et al., Robust classification of low-grade cervical cytology following analysis with ATR-FTIR spectroscopy and subsequent application of self-learning classifier eClass, *Journal of Analytical and Bio-analytical Chemistry*, 2011, in press.

[13] D. Filev and R.R. Yager, Generalized defuzzification method via BADD distribution. *International Journal of Intelligent Systems,* vol. **6**, pp. 687–697, 1991.

[14] P. Angelov, D. Filev, Flexible Models with Evolving Structure, *Proc. IEEE Symposium on Intelligent Systems*, Varna, Bulgaria, 10-12 Sept. 2002, vol.2, pp.28-33

[15] P. Angelov, X. Zhou, On Line Learning Fuzzy Rule-based System Structure from Data Streams, 2008 IEEE Intern. Conf. on Fuzzy Systems, Hong Kong, June 1-6, 2008, pp.915-922.

[16] L. Ljung, *System Identification: Theory for the User*, Prentice Hall, Upper Saddle River, New Jersey, USA, 1999.

[17] R. Babuska, Fuzzy Modelling for Control, Kluwer Verlag, 1998.

[18] P. Angelov, Evolving Takagi-Sugeno Fuzzy Systems from Data Streams (eTS+), In Evolving Intelligent Systems: Methodology and Applications *(Angelov P., D. Filev, N. Kasabov Eds.)*, Willey & IEEE Press, pp. 21-50, ISBN: 978-0-470-28719-4, April 2010.

[19] H.-J. Rong, N. Sundararajan, G.-B. Huang, G. S. Zhao, Extended sequential adaptive fuzzy inference system for classification problems, *Evolving System*, Vol. 2, Jan. 2011, DOI 10.1007/s12530- 010-9023-9.

[20] E. Lima, F. Gomide, R. Ballini, Participatory evolving fuzzy modeling, In: *Proc. 2006 International Symposium on Evolving Fuzzy Systems*, pp.36 -41, IEEE Press, UK, 2006.

[21] A. Lemos, W. Caminhas, F. Gomide, Fuzzy evolving linear regression trees, *Evolving Systems*, Vol.2, Jan. 2011, DOI 10.1007/ s12530-011- 9028-z

[22] R. R. Yager, D. P. Filev, Learning of fuzzy rules by mountain clustering, *Proc. of SPIE Conf. on Application of Fuzzy Logic Technology*, Boston, MA, USA, pp.246-254, 1993.

[23] S. L. Chiu, Fuzzy model identification based on cluster estimation, *Journal of Intelligent and Fuzzy Syst*.vol.2, pp. 267-278, 1994.

[24] G. Box, G. Jenkins (1976) *Time Series Analysis: Forecasting and control*, 2^{nd} edition, Holden-Day, San Francisco, CA, USA.

[25] P. Angelov, A. Kordon, Adaptive Inferential Sensors based on Evolving Fuzzy Models: An Industrial Case Study, *IEEE Trans. on Systems, Man, and Cybernetics, part B - Cybernetics*, Vol.40 (2), pp.529-539, 2010.

[26] G. Leng, T. M. McGinnity, G. Prasad, An approach for on-line extraction of fuzzy rules using a self-organising fuzzy neural network, Fuzzy Sets and Systems, vol. 150, pp.211-243, 2002.

[27] N. Kasabov, Q. Song, DENFIS: dynamic evolving neural-fuzzy inference system and its application for time-series prediction, *IEEE Trans. on Fuzzy Syst.*, Vol.10 (2), pp.144-154, 2002.

[28] UCI Machine Learning Repository, http://www.ics.uci.edu/~mlearn/MLRepository.html, accessed 7 September 2010.

[29] J. M. Mendel, "Type-2 fuzzy sets and systems: an overview," *IEEE Comp. Intelligence Magazine*, vol. 2, pp. 20-29, Feb. 2007.