Forecasting Time-Series for NN GC1 using Evolving Takagi-Sugeno (eTS) Fuzzy Systems with On-line Inputs Selection

Javier Andreu *Student Member, IEEE,* Plamen Angelov *Senior Member, IEEE*

***Abstract*— In this paper we present results and algorithm used to predict 14 days horizon from a number of time series provided by the NN GC1 concerning transportation datasets [1]. Our approach is based on applying the well known Evolving Takagi-Sugeno (eTS) Fuzzy Systems [2-6] to self-learn from the time series. ETS are characterized by the fact that they self-learn and evolve the fuzzy rule-based system which, in fact, represents their structure from the data stream on-line and in real-time mode. That means we used all the data samples from the time series only once, at any instant in time we only used one single input vector (which consist of few data samples as described below) and we do not iterate or memorize the whole sequence. It should be emphasized that this is a huge practical advantage which, unfortunately cannot be compared directly to the other competitors in NN GC1 if only precision/error is taken as a criteria. It is also worth to require time for calculations and memory usage as well as iterations and computational complexity to be provided and compared to build a fuller picture of the advantages the proposed technique offers. Nevertheless, we offer a computationally light and easy to use approach which in addition does not require any user- or problem-specific thresholds or parameters to be specified. Additionally, this approach is flexible in terms not only of its structure (fuzzy rule based and automatic self-development), but also in terms of automatic input selection as will be described below.**

I. Introduction

The time series are modeled by a simple regression model of the form:

$$\hat{x}_k = F(x_{k-1}, x_{k-2}, \dots, x_{k-d}) \tag{1}$$

where $\hat{x}_k$ denotes the estimation of the value of the time series at the k-th time instant; x_{k-1} denotes the value of the time series estimation of the value of the time series at the (k-i)-th time instant; d is obviously the depth of the regression.

The non-linear, non-stationary function, f(x) is approximated by a fuzzy rule-based system of Takagi-Sugeno type [7]:

$$IF\ (x_1 is\ x_1^*) AND \dots (x_d is\ x_d^*) THEN\ \hat{x}_k = a_0 + a_1 x_1 + \dots + a_d x_d \tag{2}$$

where x_i^* are prototypes/focal points around which one can form linguistic terms and $a_0,a_1,..,a_d$ are consequence parameters.

It is well known that fuzzy systems proven to be universal approximations [8] and this approach seems reasonable. The main problem with such an approach is how to design the model structure (how many fuzzy rules to select, where to center them, how to determine the parameters etc.) [9]. Data-driven approaches to designing fuzzy rule-based systems were first introduced in mid-90s of the last century [10]. They did not answer the key problem, however, how to determine the structure of the model. Moreover, if the data stream is non-stationary this structure may need update/evolving itself (as well as the model parameters). This problem was addressed for the first time in the concept of evolving fuzzy [2,9] and neural network systems [11]. We adopt in this paper the first and one of the most popular approaches for evolving fuzzy rule-based systems, eTS [2-6] in its latest from which also includes on-line input variables selection. For the times series that we have in NN GC1 that means modifying the expression (1) into:

$$\hat{x}_k = F(x_{k-i1}, x_{k-i2}, \dots, x_{k-ir}) \tag{1b}$$

Where the set *I={i1,i2,...,ir}* and *r<<d* is determined automatically by the algorithm on-line (taming one sample at a time with no iterations and memorizing the data stream).

Respectively, equation (2) also changes in that it includes a smaller number of premise fuzzy sets, *r* instead of *d*.

In the specific problem we had at hand in NN GC1 it turns out that *r=4* and *i={7,14,21,28}* which means nothing else, but a strong weekly pattern. It was even more impressive that this weekly dependence was discovered by eTS algorithm automatically and without off-line procedures or memorizing the data streams. We also compared this result to an off-line test based on correlation analysis of input-output pairs (Figure 1) and discovered that both approaches provide the same result which will be detailed in the main paper.

In the proposed approach the model structure evolves automatically in an on-line manner through a recursive analysis of each (new at a time of analysis) data sample as compared to an accumulated history of the data stream (time series) which is compressed in the existing rules as well as in a small number of additional recursively updated parameters indicating the data density evolution.

It should be noted that the proposed methodology has already been successfully applied to a wide range of applications such as predictions, classification and control problems. Further advances in the methodology in this century gives the capacity to reduce the computational cost

Both authors are with the Intelligent Systems Research Laboratory, Department of Communication Systems, InfoLab21, Lancaster University, Lancaster, LA1 4WA, UK; e-mail: j.andreu@lancaster.ac.uk

of the algorithm, making feasible to embedded this techniques in low computational devices such as sensors or mobile phones. In addition new on-line capacities made the system much more attractive for real applications including areas such as robotics, advanced industrial process etc. [12,13].

The key difference of eTS in comparison with other approaches is that it does not require a prefixed model structure (number of fuzzy rules, number of inputs) but is fully data-driven method suitable to on-line and real-time applications. The term "evolving" is adopted due to the gradual evolution of the fuzzy rule-based model structure in terms of its components and fuzzy rules. This evolving behavior is achieved by means of an on-line incremental clustering (partitioning of the data space) of the time series in terms of input-output data space (please, see equation (1). In the case of time series, the inputs are the values of the time series different number of steps back. Therefore, starting from a guess of the value d (depth of the history) we are able to down-select a small subset, I of relevant inputs (steps back) based on their contribution to the prediction. This makes the methodology efficient and suitable for online and real time applications.

II. On-line Learning Fuzzy Approach Based On Data Density

The key point of these approaches is the attempt to estimate data recursively which is more computationally efficient. By this aim we are able to get new on-line capacities which are suitable for real time applications. Directly from the data streams and capturing data density variations we can modify the shape and structure of the model. Another approaches deal with the same attempt, for example this is the case of kernels in image processing which estimation is off-line [14], Parzen Windows [15] in statistical learning. A well know approach is also the so called Mountain functions [16] that make use of the so called potential [17]. All of this last tree methods data density estimation is based on Gaussian distribution. We proposed to use a Cauchy function over the sum of distances between several data points. As the Cauchy function is in fact a first order approximation of the Gaussian, both of them have a series of common properties:

- Both are monotonic.
- Its maximum is always 1.
- When the argument tends to infinity, both asymptotically tends to Zero.

Here is the formula of the first order approximation of the Gaussian [14].

To form new clusters in the system the value called potential (data density) P has been used as criteria to form new clusters in the evolving fuzzy clustering approach, eClustering [14]:

$$P\big(x(k)\big) = \frac{1}{1+\frac{1}{k-1}\sum_{i=1}^{k-1}\frac{\|x(k)-x(i)\|^2}{2\sigma^2}} \quad (3)$$

Where k is the current time instant; σ denotes the spread or zone of influence of the cluster.

Points with high potentials are selected to be candidates to form part of the model (focal points or fuzzy rules). Estimation of the data density is not an easy task because we work out this density we need to work out the distance between every single point and other data samples. This fact can be an issue for an on-line mode approach which does not keep in memory in the totality of data samples. Therefore it is required to perform a recursive calculation which was proposed in:

$$P\big(x(k)\big) = \frac{k-1}{(k-1)(a(k)+1)+b(k)-2c(k)} \quad (4)$$

Values *a(k)* and *c(k)* can be only calculated from the current frame:

$$a(k) = \sum_{j=1}^{n+m} x_j^2(k); c(k) = \sum_{j=1}^{n+m} x_j(k)d_j(k) \quad (5)$$

Dimensionalities are represented by n (input) and m (outputs) and $d_j(k)$ is calculated recursively as follows:

The value $b(k)$ is also defined by a recursive expression and accumulated during processing of the following frames individually:

$$b(k) = b(k-1) + a(k-1); b(1) = 0 \quad (6)$$
$$d_j(k) = d_j(k-1) + x_j(k-1); d_j(k) = 0 \quad (7)$$

The area of influence or spread of the clusters σ is updating on-line in a data-driven fashion by means of learning the data distribution and variance [16]:

$$\sigma_{ij}^2(k) = \alpha\sigma_{ij}^2(k-1) + (1-\alpha)\frac{1}{N_i(k)}\sum_{l=1}^{N_i(k)}(x_l(k)-x_l(k))^2 \quad (8)$$

Taking as a initial value $\sigma_j(1) = 0.5$, α denote the learning step (recommended value 0.5); the number of data samples is represented by $N_i(k)$ associated with the number of clusters which belongs to, i^{th}.

In the course of the process of the model, we only keep in the memory the values of the focal points and their potential, all the other values are discarded from the memory. As a result the potential is a representation of the data density regarding all of data samples. Therefore it is required to update each time step (a new data sample being read). Potential of focal points is update even with the new data sample that will appear after a sample is taking as a prototype to form a cluster. The following formula is applied for updating:

$$P\big(x_i^*(k)\big) = \frac{k-1}{k-1+(k-2)\Big(\frac{1}{P(x_i^*(k-1))}-1\Big)+\big\|x_{ij}^*-x_{ij}\big\|^2} \quad (9)$$

In previous formulas (4)-(7) data density is estimated and the spread is adapted by (8). Thereupon we can form a fuzzy rule base in accordance with these basic principles:

1) A data with a high potential is suitable to be a focal point;
2) A data sample coordinates are placed in area where no previous fuzzy rules are covering that space.
3) Overlap and information redundancy in forming new rules must be avoided.
We can represent the first principle, 1) with the following expression [18,19]:

$$P(x(k)) > \max_{i=1}^{R} P(x^*(k)) \tag{10}$$

Z^* denotes a data sample that has been selected to be a focal point (fuzzy rules); R represent the number of fuzzy rules till the current moment k (before condition (10) has been checked).

Regarding the second principle, this is represented by the next expression [20]:

$$P(x(k)) > \min_{i=1}^{R} P(x^*(k)) \tag{11}$$

To accomplish the third principle, 3) we can step by step shrink the cluster radio by the next condition B:

$$\exists i, i = [1, R];\ \mu_{ij}(x(k)) > e^{-1};\ \forall j; j = [1, n] \tag{12}$$

In this formula $x = [x_1,x_2,\ldots,x_n]^T$ represent the input vector values (in the case of classification we would be talking about features); u_{ij} represent a Gaussian type membership function of j^{th} fuzzy set of the i^{th} fuzzy rule:

$$\mu_{ij}(x_k) = e^{\frac{-(x_j(k)-x_{ij}^*)^2}{2\sigma_j^2}} \tag{13}$$

Especially we have to take in consideration the previous point for the focal points of rules that might be formed based on the first principle, 1) following expression (10) which might lies too close to each other. On the 3 principle, we have to say that it makes easy the formation of fuzzy rules, in consideration with another approaches such as VQ [21], ART [22] etc… all of them usually require later so called 'prunning' [23].

The fundaments of the expression (11) are based on the so called 'one-sigma' condition, $|x_j(k) - x^*_{ij}| > \sigma_j$ known from the machine learning literature [15]. Thus, the expression (13) is valid when in the rule base there is implicit a fuzzy rule. The named this i, such that the input vector of the current data sample, x_k is multi-dimensional, represented by j by at least $e^{-1} \approx 0.36$.

On the basis of these principles, we can write a pseudo code of the algorithm to learn on-line from the antecedents of the fuzzy system.

Accordingly of this data density-based on-line clustering we are able to compose antecedent part of the fuzzy rules directly generated from the data stream:

$$R_i\ IF(x_i\ is\ x_{i1}^*) AND\ (x_2 is\ x_{i2}^*) \ldots (x_n is x_{in}^*); i = [1, R] \tag{14}$$

Read data sample *x(k)*
IF (k = 1) THEN
//initialization stage//
//initialize the variables of recursive calculation//
$d_j(k)$ =0;j=[1,n];b(k)=0
//the input part of the first data sample is the focal of the first cluster (rule)//
$X^*_1(1) \leftarrow x(1)$; $P(x^*_1(k)) \leftarrow 1$; $R \leftarrow 1$
//form the antecedent part of the first fuzzy rule//
$Rule_1$ IF(x_{11} is x^*_{11}) AND…AND (X_{1n} is X^*_{1n})
ELSE
Recursively calculate potential of the current data sample, *P(z(k));*
Update the spread of the clusters (membership functions of the respective fuzzy sets) ;
Recursively update the potentials of the existing clusters;
Check
Condition A;
Condition B where,
A) the point is with **high** potential and **covers new area of data space**;
B) the point **overlap** with the previously formed fuzzy rules;
IF (A) THEN (x(k) is new focal point)
$X^*_{R+1}(k) \leftarrow x(k)$ $P(x^*_{R+1}(k)) \leftarrow P(x(k))$; $R \leftarrow R+1$
Assign the new point to the nearest cluster.
Repeat until end of data stream

Algorithm- Recursive data space partitioning based on the data density

The antecedent part may be used in several applications and ways:
1) Stored and Analyzed by an operator;
2) Combined with consequent part identification.
   a. It can be used for prediction at each time.
   b. It can be used also for classification.
3) Clustering the data and various applications, for example in robotics.

III. EVOLVING FUZZY SYSTEM WITH ON-LINE INPUT SELECTION

We considered in the previous section and as far as we have in knowledge, all previous research assumes that the dimensionality of the input (features) vector is pre-defined in each study or problem (Figure 2).

We put forward an approach that on the basis of Takagi-Sugeno (TS) type fuzzy systems is able to select more meaningful input features. The system automatically remove from the system those inputs does not contribute positively to the output. Takagi-Sugeno outputs are locally linear, so the analysis of sensitivity is reduced to just analyze the consequent parameters [5,6].

$$\begin{gathered} THEN(y_i = \theta_{i0} + \textstyle\sum_{j=1}^{n} x_{ij}\theta_{ij}); i = [1, R] \\ R_i\ IF(x_i\ is\ x_{i1}^*)\ AND\ (x_2 is\ x_{i2}^*) \ldots (x_n is\ x_{in}^*) \end{gathered} \tag{15}$$

The overall output of the TS fuzzy system is worked out as the weighted average of the outputs of the local linear model [22]:

$$y = \sum_{i=1}^{R} \lambda_i y_i \tag{16}$$

Where $\lambda_i = \frac{\prod_{j=1}^{n} \mu_{ij}(x)}{\sum_{l=1}^{R} \prod_{j=1}^{n} \mu_{lj}(x)}$ is the firing strength of the i^{th} fuzzy rule.

Each one of the inputs features can be evaluated regarding importance by ratio of the sum of consequent parameters for the specific j^{th} input feature in regard to all n inputs (features) [23]:

$$\omega_{ij}(k) = \frac{T_{ij}(k)}{\sum_{r=1}^{n} T_{ir}(k)}\ ; i = [1, R]; j = [1, n] \qquad (17)$$

Where $T_{ij}(k) = \sum_{l=1}^{k} \left|\theta_{ij}(l)\right|$ represents the accumulated sum of the parameters values of the i^{th} rule.

Inputs, outputs and internal variables of eTS can be normalized online [7] then we may compare between each other. This comparison value gives us the weight value for each one of the input, then on we use this value to remove input features j* which does not contribute or produce unnecessary noise to the overall output. The structure of the next time instant can be determined by [5,6]:

$$\exists j^* \left| \omega_{ij^*}(k) < \varepsilon \sum_{r=1} T_{ir}(k) \right. \qquad (18)$$
$$i = [1, R]; j = [1, n]$$

Where ε_1 is a pre-fixed value represent the tolerable minimum positive weigh of an input (feature). Suggested value is 3 to 5%.

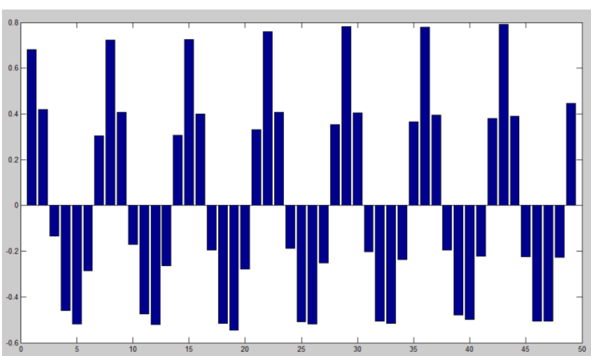

Fig. 1. Vertical axis, correlation values; horizontal axis, number of steps backward.

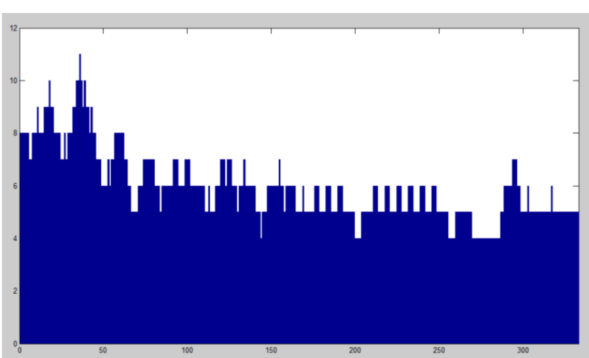

Fig 2. Vertical axis, number of input selected automatically (online input selection); horizontal axis, time instance.

We formulate the condition in terms of the proportion which the weight of a certain input (feature) from the total sum or the maximum of the accumulated sum of parameters. If the value achieved is less than ε this input feature is being insignificant or even hindering the well performance of the system. Manifestly, the removal of this input is not going to affect the output. The two conditions differ, because when the number of inputs (features), *n* (the total sum) may become too large, at the same time, for a small number of features; thus, the sum gives a better representation that the maximum (an averaging effect).

This technique is one of the strongest points of the Evolving Fuzzy system and should not be underestimated. In fact, in a real environment or issue the output selection or feature extraction is a very critical point. The success of the reliability of the system is dependent upon a good selection of inputs. Input selection is oftentimes addressed by approaches such as PCA [15], GP [24], etc. All of these approaches however require a batch set of data and a fixed model structure.

IV. Conclusion And Discussion

A new approach to autonomous generation of fuzzy system from data streams is explained in the paper. We build upon the recently introduced evolving fuzzy Takagi-Sugeno to make predictions over NN GC1 time series which has been submitted to the *Neural Time Series Forecasting Competition for Neural Networks.* Representative figures of predictions are presented below Figure 3 and Figure 4.

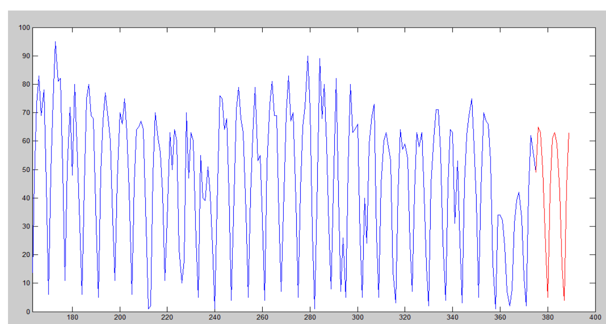

Fig. 3. Prediction (Red) of 14 days ahead in time series number 1 from Dataset E (Daily data).

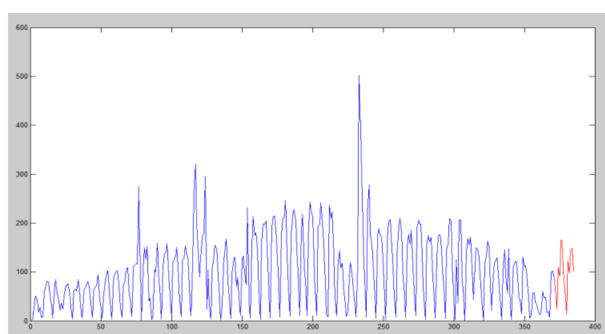

Fig. 4. Prediction (Red) of 14 days ahead in time series number 3 from Dataset E (Daily data).

This combination of techniques allow an extensible, flexible and open structure of fuzzy rule base and fuzzy sets which gives a result and more digested and understandable by humans than common neural networks. On the basis of the flexibility and online structure simplification, a new approach of automatic input selection over time series is presented and explained.

REFERENCES

[1] Time Series Forecasting Grand Competition for Computational Intelligence. 2010 NNG1 Available: http://www.neural-forecasting-competition.com

[2] P. Angelov, "Evolving Rule-based Models: A Tool for Intelligent Adaptation", *Proc. 9th IFSA World Congress*, Vancouver, BC, Canada, 25-28, pp.1062-1067, July 2001.

[3] P. Angelov and D. Filev, "An Approach to On-line Identification of Takagi-Sugeno Fuzzy Models", *IEEE Trans. on System, Man, and Cybernetics, part B - Cybernetics*, vol.34, No. 1, pp. 484-498, 2004.

[4] P. Angelov, "An Approach for Fuzzy Rule-base Adaptation using On-line Clustering", *International Journal of Approximate Reasoning*, vol. 35 , No 3, pp. 275-289, March 2004.

[5] P. Angelov, X. Zhou, On Line Learning Fuzzy Rule-based System Structure from Data Streams, *2008 IEEE Intern. Conf. on Fuzzy Systems, IEEE World Congress on Computational Intelligence*, Hong Kong, June 1-6, 2008, pp.915-922, ISBN 978-1-4244-1821-3/08.

[6] P. Angelov, X. Zhou, Evolving Fuzzy Systems from Data Streams in Real-Time, *2006 Intern. Symp. on Evolving Fuzzy Systems*, 7-9 Sept., 2006, Ambleside, UK, IEEE Press, pp.29-35, ISBN 0-7803-9719-3.

[7] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its application to modelling and control," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 15, No. 1, pp. 116-132, 1985.

[8] L.-X. Wang, "Fuzzy Systems are Universal Approximators", *Proc. of the International Conference on Fuzzy Systems*, San Diego, CA, USA, 1992, pp.1163-1170.

[9] P.P. Angelov, *Evolving Rule-based Models: A Tool for Design of Flexible Adaptive Systems*, Springer-Verlag, Heidelberg, New York, 2002, ISBN 3-7908-1457-1.

[10] R. Babuska, Data-driven Fuzzy Modeling: Transparency and Complexity Issues, *Proc. of the Euro Symposium on Intelligent Technologies ESIT'99*, AB-01, Crete, Greece, June 1999 http://lcewww.et.tudelft.nl/~babuska/bib/index.html.

[11] N. Kasabov, Q. Song "DENFIS: Dynamic Evolving Neural-Fuzzy Inference System and Its Application for Time-Series Prediction," *IEEE Trans. on Fuzzy Systems, Vol.10* (2), 2002, pp. 144-154.

[12] X. Zhou, P. Angelov, "Autonomous Visual Self-localization in Completely Unknown Environment using Evolving Fuzzy Rule-based Classifier", *Proc. IEEE Intern. Conf. on Computational Intelligence Applications for Defense and Security*, Honolulu, USA, 1-5 April 2007, pp. 131-138.

[13] J. J. Macias-Hernandez, P. Angelov, X. Zhou, Soft Sensor for Predicting Crude Oil Distillation Side Streams using Takagi-Sugeno Evolving Fuzzy Models, Proc. 2007 IEEE Intern. Conf. on Systems, Man, and Cybernetics, 7-10 Oct., 2007, Montreal, Canada, ISBN 1-4244-0991-8/07, pp.3305-3310.

[14] A. Elgammal, R. Suraiswami, D. Harwood, and L. Davis, "Background and Foreground modeling using non-parametric Kernel Density Estimation for visual surveillance KDE", Proc. 2002 IEEE Vol. 90 (7), pp. 1151 – 1163.

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

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

[17] S. L. Chiu, "Fuzzy model identification based on cluster estimation", J. of Intel. and Fuzzy Syst. vol. 2, pp. 267-278, 1994.

[18] P. Angelov, "An Approach for Fuzzy Rule-base Adaptation using On-line Clustering", International Journal of Approximate Reasoning, Vol. 35 , No 3, March 2004, pp. 275-289.

[19] P. Angelov, D. Filev, "An Approach to On-line Identification of Takagi-Sugeno Fuzzy Models", IEEE Transactions on System, Man, and Cybernetics, part B - Cybernetics, vol.34, No1, 2004, pp.484-498. ISSN 1094-6977.

[20] P. Angelov, "Evolving Rule-based Models: A Tool for Design of Flexible Adaptive Systems". Heidelberg, Germany: Springer, 2002.

[21] R. Gray, "Vector quantization," IEEE ASSP Magazine, pp. 4–29, 1984.

[22] G. A. Carpenter and S. Grossberg, "Adaptive Resonance Theory (ART)," in The Handbook of Brain Theory and Neural Networks, M. A. Arbib, Ed. Cambridge, MA: MIT Press, 1995, pp. 79–82.

[23] E. Lughofer, E. Huellermeier, and E. Klement, "Improving the interpretability of data-driven evolving fuzzy systems," in Proceedings of EUSFLAT 2005, Barcelona, Spain, 2005, pp. 28–33.

[24] G. Smits, A. Kordon, E. Jordaan . C. Vladislavleva, and M. Kotanchek, Variable Selection in Industrial Data Sets Using Pareto Genetic Programming, In: Yu T., R. Riolo, B. Worzel (Eds): Genetic Programming Theory & Practice III. Springer, NY, pp. 79-92, 2006.