# ON-LINE EVOLUTION OF TAKAGI-SUGENO FUZZY MODELS

**Plamen Angelov[1], José Victor[2,3], António Dourado[3] and Dimitar Filev[4]**

[1]*Department of Communications Systems*
*Lancaster Universiy, UK*
*p.angelov@lancaster.ac.uk*

[2]*School of Technology and Management*
*Polytechnic Institute of Leiria, Portugal*
*zevictor@estg.ipleiria.pt*

[3]*Adaptive Computation Group - CISUC*
*University of Coimbra, Portugal*
*{zevictor, dourado}@dei.uc.pt*

[4]*Ford Motor Co.,USA*
*Dfilev@Ford.com*

Abstract: Evolving Takagi-Sugeno (eTS) fuzzy models and the method for their on-line identification has been recently introduced for both MISO and MIMO case. In this paper, the mechanism for rule-base evolution, one of the central points of the algorithm together with the recursive clustering and modified recursive least squares (RLS) estimation, is studied in detail. Different scenarios are considered for the rule base upgrade and modification. The radius of influence of each fuzzy rule is considered to be a vector instead of a scalar as in the original eTS approach, allowing different areas of the data space to be covered by each input variable. Simulation results using a well-known benchmark (Mackey-Glass chaotic time-series prediction) are presented. 

Keywords: evolving Takagi-Sugeno fuzzy models, rule-base evolution, recursive clustering, RLS algorithm.

## 1. INTRODUCTION

For several centuries the so-called first principles models have dominated the natural sciences. However, for a number of practical engineering problems they are difficult or even impossible to build (Angelov, 2002; Yager and Filev, 1994). Another alternative is to use so-called "black-box" models (polynomial, regression models, neural networks). They can fit the data with arbitrary precision, but they are not transparent enough: their coefficients and structure is not directly related to the system being modelled (Yager and Filev, 1994).

Fuzzy rule-based models and especially Takagi-Sugeno (TS) fuzzy models have gained significant impetus due to their flexibility and computational efficiency (Takagi and Sugeno, 1985; Yager and Filev, 1994). They have a quasi-linear nature and use the idea of approximation of a nonlinear system by a collection of fuzzily mixed local linear models. The TS fuzzy model is attractive because of its ability to approximate nonlinear dynamics, multiple operating modes and significant parameter and structure variations (Takagi and Sugeno, 1985).

On-line learning of TS fuzzy models involves recursive, non-iterative clustering responsible for model structure (rule base) learning and recursive consequent parameter estimation (Angelov, 2002; Angelov and Filev, 2004). eTS is based on the assumption that the model structure evolves gradually instead of being known *a priori* (Angelov and Filev, 2004). It is important to note that this evolution is much slower than the evolution of the model parameters. For the eTS the notion of informative potential of the new data sample (accumulated spatial proximity measure) is very important. It has been first introduced in the mountain clustering approach (Yager and Filev, 1993) and then refined in the subtractive clustering approach (Chiu, 1994). It is used as a trigger to update the rule-base (Angelov, 2002; Angelov and Filev, 2004). It is a great advantage of this approach that the learning can start without *a priori* information and only a single data sample. This interesting feature makes the approach potentially very useful in autonomous, robotic, and smart adaptive systems (Angelov, 2002).

## 2. IDENTIFICATION OF TS FUZZY MODELS

TS fuzzy models have been originally introduced by Takagi and Sugeno (1985) as the first systematic method for identification of fuzzy models. They are a group of rule-based models with fuzzy antecedents and functional consequent (1),

$$\Re_i : \textbf{IF } \left(x_1 \text{ is } \aleph_1^i\right) \textbf{AND} \ldots \textbf{AND } \left(x_n \text{ is } \aleph_n^i\right) \textbf{ THEN } (y^i = x_e^T \pi^i);\ i = 1,2,\ldots,R \quad (1)$$

where $\Re_i$ denotes the $i^{th}$ fuzzy rule; $R$ is the number of fuzzy rules; $x_e^T$ is the extended input vector, $x_e^T = [1, x^T]$, which is formed by appending the input vector $x = [x_1, x_2, \ldots, x_n]^T$ with 1 (allowing a free parameter); $\aleph_i^j$ denotes the antecedent fuzzy sets, $j = 1,2,\ldots,n$; $y^i$ is the output of the $i^{th}$ linear subsystem; $\pi^i = [a_0^i, a_1^i, \ldots, a_n^i]^T$ are its parameters.

The basic notion of the TS method is the fuzzy separation of the data space into local regions (Takagi and Sugeno, 1985). Each region is associated with a linear sub-model, which is valid with a degree, proportional to the distance to the region's center. In this way, the overall nonlinear system is represented by a fuzzy weighted combination of locally valid linear models. Usually Gaussians are used to represent the antecedent fuzzy sets. This ensures greatest possible generalization of the description (2),

$$\mu^i(x) = e^{-4\left\|\frac{x - x^{i*}}{r}\right\|^2};\ i = 1,2,\ldots,R \quad (2)$$

where $r$ is a positive constant, which defines the radius of the antecedent and the zone of influence of the $i^{th}$ model; $x^{i*}$ is the focal point of the $i^{th}$ rule antecedent.

In the eTS fuzzy model, which is based on TS fuzzy models and considers their on-line identification subject to *gradually evolving* rules, a small number of parameters is needed to be predefined. One of the few such parameters is the radius $r$. As it will be shown, it can be used as a leverage for a trade-off between the model complexity and precision. As a general guidance, too large values of $r$ lead to averaging, too small - to over-fitting. A value of $r$ in the range of [0.3; 0.5] has been recommended (Angelov, 2002; Angelov and Filev, 2004). Since the parameters of the linear models are boundless, there is no general need for normalization of the inputs. However, when the values of inputs differ significantly a vector representation of the radius will give more flexibility and can compensate the weights of the projections of the distance between a data point and a rule center on different inputs. We consider here a vector definition of the radii, $r = [r_1, r_2, \ldots, r_n]^T$ in the form (3),

$$r = \bar{r}(\bar{x} - \underline{x}) \quad (3)$$

where $\bar{x} = [\bar{x}_1, \bar{x}_2, \ldots \bar{x}_n]^T$ and $\underline{x} = [\underline{x}_1, \underline{x}_2, \ldots \underline{x}_n]^T$ are respectively the vector of expected maximums and minimums of the inputs. It should be noted that in on-line mode one could only expect the range of each of the inputs, but the precise values of $\bar{x}$ and $\underline{x}$ are not critically important as they are compensated to some extend by the value of $\bar{r}$, and ultimately by the boundless consequent parameters.

Using different radii for each of the input variables we can denote the membership function of the antecedent fuzzy sets as (4).

$$\mu^i(x) = e^{-4\sum_{j=1}^{n}\frac{(x_j - x_j^{i*})^2}{r_j^2}} \quad (4)$$

The overall model output is calculated by weighted averaging of individual rules' contributions (5),

$$y = \sum_{i=1}^{R} \lambda^i(x) y^i = \sum_{i=1}^{R} \lambda^i(x) x_e^T \pi^i \quad (5)$$

where $\lambda^i(x) = \dfrac{\mu^i(x)}{\sum_{j=1}^{R} \mu^j(x)}$ is the normalized firing level of the $i^{th}$ rule. In a vector form we can represent it as (6),

$$y = \psi^T \theta \quad (6)$$

where $\theta = [(\pi^1)^T, (\pi^2)^T, \ldots, (\pi^R)^T]^T$ is a vector composed of the linear model parameters; $\psi = [\lambda^1 x_e^T, \lambda^2 x_e^T, \ldots, \lambda^R x_e^T]^T$ is a vector of the inputs that are weighted by the normalized firing levels of the rules.

Both locally and globally optimal solutions can be estimated using the RLS algorithm. The globally optimal one does not guarantee locally adequate behavior of the sub-models that form the TS fuzzy model. Locally meaningful sub-models could be found using weighted RLS (wRLS) algorithm associated with each rule (Angelov and Filev, 2004).

## 3. ON-LINE LEARNING OF eTS MODELS

On-line learning of eTS fuzzy models includes recursive clustering and forming of a gradually evolving rule-base and weighted recursive least squares estimation. The basic stages of the procedure for on-line learning will be briefly recalled. For more details the reader is directed to (Angelov and Filev, 2004).

### *A. Rule-base initialization*

The on-line learning procedure starts with initialization of the rule-base. The first data point is established as the focal point of the first cluster $(i = 1)$. Its coordinates are used to form the antecedent part of the fuzzy rule (1) using for example Gaussian membership functions (2). Its potential, $P$, is assumed equal to 1. Parameters of the local linear model associated to this rule are also

initialized to 0. The covariance matrix, *C*, is initialized with large values, $\Omega$, in the main diagonal (7),

$$k=1; R=1; x^{1^*}=x_k; P_1(z^{1^*})=1;\ \theta_1=\pi_1=0; C_1=\Omega I \quad (7)$$

where $z^{1^*}$ is the first cluster center; $x^{1^*}$ is the focal point of the first rule being a projection of $z^{1^*}$ on the axis $\boldsymbol{x}$; *I* is the identity matrix.

*B. Reading the next data point*

The time step is updated $(k:=k+1)$ and the potential of the new data points $(z_k)$ is calculated recursively. Please note that *k* now denotes the new (next) time step.

*C. Calculation of the potential of the new data point*

The potential of a data point is measured by a Cauchy type function (8) (Angelov and Filev, 2004),

$$P_k(z_k)=\frac{1}{1+\frac{1}{(k-1)}\sum_{i=1}^{k-1}\sum_{j=1}^{n+1}(d_{ik}^j)^2};\ k=2,3,\ldots \quad (8)$$

where $P_k(z_k)$ denotes the potential of the data point $(z_k)$ calculated at time *k*; $d_{ik}^j=z_i^j-z_k^j$, denotes the projection of the distance between two data points ($z_i^j$ and $z_k^j$) on the axis $z^j$ ($x^j$ for $j=1,2,\ldots,n$ and on the axis *y* for $j=n+1$).

The potential of the new data sample is calculated recursively by (9) (Angelov and Filev, 2004).

$$P_k(z_k)=\frac{k-1}{(k-1)(\vartheta_k+1)+\sigma_k-2\nu_k} \quad (9)$$

where

$$\vartheta_k=\sum_{j=1}^{n+1}(z_k^j)^2;\ \sigma_k=\sum_{l=1}^{k-1}\sum_{j=1}^{n+1}(z_l^j)^2;\ \nu_k=\sum_{j=1}^{n+1}z_k^j\beta_k^j;\ \beta_k^j=\sum_{l=1}^{k-1}z_l^j$$

Parameters $\vartheta_k$ and $\nu_k$ in (9) are calculated from the current data point $z_k$, while $\beta_k^j$ and $\sigma_k$ are recursively updated by (10).

$$\sigma_k=\sigma_{k-1}+\sum_{j=1}^{n+1}(z_{k-1}^j)^2;\ \beta_k^j=\beta_{k-1}^j+z_{k-1}^j \quad (10)$$

*D. Updating the potentials of the centers*

Each new data point influences the potentials of the centers of the clusters ($z^{l^*}$, $l=1,2,\ldots,R$), because by definition the potential depends on the distance to **all** data points, including the new ones (the sum in the denominator by *i* in (8) has an increasing number of components). The input part $x^{l^*}$ of these centers, $z^{l^*}$, $l=1,2,\ldots,R$ is used as focal points of the existing rules.

The potentials of the focal points of the existing clusters are updated recursively (11) (Angelov and Filev, 2004),

$$P_k(z^{l^*})=\frac{(k-1)P_{k-1}(z^{l^*})}{k-2+P_{k-1}(z^{l^*})+P_{k-1}(z^{l^*})\sum_{j=1}^{n+1}\left(d_{k(k-1)}^j\right)^2} \quad (11)$$

where $P_k(z^{l^*})$ is the potential of the $l^{th}$ cluster $z^{l^*}$, which is a prototype of the $l^{th}$ rule at time *k*.

*E. Rule base evolution*

At this stage the potential of the new data point, calculated at stage C, is compared to the potential of the centers of the existing clusters updated at stage D and the important decision whether to modify or upgrade the rule-base is taken. The evolution of the rule-base is driven by the following two basic principles:

**Principle 1** (MODIFY):

**IF** (MODIFY condition)
**THEN** the new data point $(z_k)$ **replaces** this center (let us suppose that it has index *j*) (12).

$$x^{j^*}=x_k;\ P_k(z^{j^*})=P_k(z_k) \quad (12)$$

**Principle 2** (UPGRADE):

**IF** (UPGRADE condition)
**THEN** the new data point is **added** to the rule-base as a new center and a new rule is formed with a focal point based on the projection of this center on the axis $\boldsymbol{x}$ (13).

$$R=R+1;\ x^{R^*}=x_k;\ P_k(z^{R^*})=P_k(z_k) \quad (13)$$

Generally, the MODIFY condition includes the UPGRADE condition plus the **check for the closeness** of the candidate new rule center to the already existing rule centers. Originally, two thresholds $(\bar{\varepsilon}=50\%;\underline{\varepsilon}=15\%)$ based on the maximum of the potentials of the existing rules, $\overline{P}_k$, have been considered as UPGRADE condition (Angelov, 2002), then the condition was simplified (Angelov and Filev, 2004) to a comparison to $\overline{P}_k$ directly. A number of different scenarios have been investigated and they will be summarized and analyzed later. The closeness to the existing rule centers is measured with the Euclidean distance from the new data point to the closest of the existing rule centers (14).

$$\delta^i=\left\|z_k-z^{i^*}\right\| \quad (14)$$

*F. Estimation of the local sub-models parameters*

In eTS fuzzy models the rule-base gradually evolves (Angelov, 2002; Angelov and Filev, 2004) leading to a change of the normalized firing strengths of the

rules ($\lambda^i$), which effects all the data (including the data collected before time of the change). Therefore, the straightforward application of the RLS algorithm or wRLS is not correct. It has been proposed to reset the covariance matrices and initialize the parameters of the RLS each time a new rule is added to the rule base. This $(R+1)^{th}$ rule is estimated as a weighted average of the respective covariance and parameters of the remaining *R* rules (Angelov and Filev, 2004).
The following RLS procedure (15) (16) is globally optimal (Angelov and Filev, 2004),

$$\hat{\theta}_k = \hat{\theta}_{k-1} + C_k \psi_k \left(y_k - \psi_k^T \hat{\theta}_{k-1}\right);\ k = 2,3,\ldots \quad (15)$$

$$C_k = C_{k-1} - \frac{C_{k-1}\psi_k \psi_k^T C_{k-1}}{1+\psi_k^T C_{k-1}\psi_k} \quad (16)$$

where $\hat{\theta}_1 = \left[(\hat{\pi}^1)^T, (\hat{\pi}^2)^T, \ldots, (\hat{\pi}^R)^T\right]^T = 0;\ C_1 = \Omega I$.

When the rule-base is upgraded (UPGRADE principle is active) the RLS algorithm is reset in the following way (Angelov and Filev, 2004):

a) Parameters of the new rule, $(R+1)^{th}$, are determined by the weighted average of the parameters of the rest $(R)$ of the rules with weights equal to the normalized firing levels of the existing rules $\lambda^i$. Parameters for the rest of the rules are taken from the previous step without change (17) (Angelov and Filev, 2004),

$$\hat{\theta}_1 = \left[(\hat{\pi}_{k-1}^1)^T, (\hat{\pi}_{k-1}^2)^T, \ldots, (\hat{\pi}_{k-1}^R)^T, (\hat{\pi}_{k-1}^{R+1})^T\right]^T \quad (17)$$

where

$$\hat{\pi}_k^{R+1} = \sum_{i=1}^{R} \lambda^i \hat{\pi}_{k-1}^i \quad (18)$$

b) Covariance matrices are reset as (19),

$$C_k = \begin{bmatrix} \rho\varsigma_{11} & \cdots & \rho\varsigma_{1R(n+1)} & 0 & \ldots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \rho\varsigma_{R(n+1)1} & \cdots & \rho\varsigma_{R(n+1)R(n+1)} & 0 & \ldots & 0 \\ 0 & 0 & 0 & \Omega & \ldots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \ldots & \Omega \end{bmatrix} \quad (19)$$

where $\varsigma_{ij}$ is an element of the covariance matrix, $i = 1,2,\ldots,R(n+1);\ j = 1,2,\ldots,R(n+1);\ \rho = \frac{R^2+1}{R^2}$ is a coefficient (Angelov and Filev, 2004).
When a rule is replaced with another one (MODIFY principle is active), the covariance matrices are taken from the previous step without change.
Similarly, there is a locally optimal procedure, which is based on the wRLS (20) (21) (Angelov and Filev, 2004),

$$\hat{\pi}_k^i = \hat{\pi}_{k-1}^i + c_k^i x_{ek-1} \lambda^i(x_{k-1})\left(y_k - x_{ek-1}^T \hat{\pi}_{k-1}^i\right);\ k = 2,3,\ldots \quad (20)$$

$$c_k^i = c_{k-1}^i - \frac{\lambda^i(x_{k-1}) c_{k-1}^i x_{ek-1} x_{ek-1}^T c_{k-1}^i}{1+\lambda^i(x_{k-1}) x_{ek-1}^T c_{k-1}^i x_{ek-1}} \quad (21)$$

with initial conditions $\hat{\pi}_1^i = 0;\ c_1^i = \Omega I;\ i = 1,2,\ldots,R$.
In this case, the covariance matrices are separate for each rule and have smaller dimensions ($c_k^i \in R^{(n+1)\times(n+1)};\ i = 1,2,\ldots,R$). Parameters of the newly added rule (UPGRADE principle) are determined as weighted average of the parameters of the rest *R* rules by (18). Parameters of the rest *R* rules are not changed ($\pi_k^i = \pi_{k-1}^i;\ i = 1,2,\ldots,R$). When a rule is replaced by another rule (MODIFY principle) then parameters of all rules are not changed ($\pi_k^i = \pi_{k-1}^i;\ i = 1,2,\ldots,R$). The covariance matrix of the newly added rule (UPGRADE) is initialized by $c_k^{R+1} = \Omega I$. The covariance matrices of the rest *R* rules are inherited ($c_k^i = c_{k-1}^i;\ i = 1,2,\ldots,R$).

*G. Next output prediction*

Having the estimated parameters of the local sub-models the next value of the output is predicted (22).

$$\hat{y}_{k+1} = \psi_k^T \hat{\theta}_k;\ k = 2,3,\ldots \quad (22)$$

The first value to be predicted is the output of the *3rd* data sample, $\hat{y}_3$. The procedure for rule-base evolution of eTS fuzzy models repeats the stages from B to G in a loop over all of the existing data in real time (theoretically an infinite loop), Fig. 1.

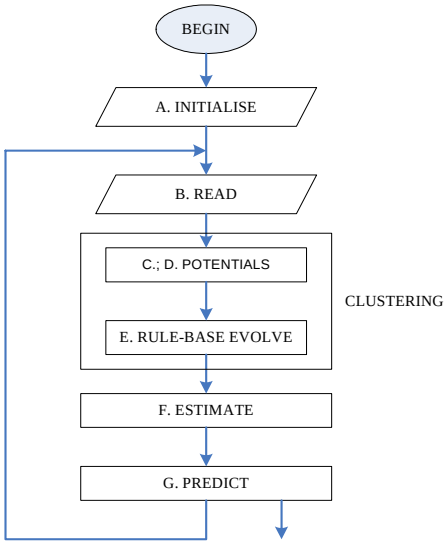


Fig. 1. Flowchart of the eTS learning algorithm.

After the initialization at stage A, and reading the new data at the next time step, stage B, the recursive clustering starts (stages C-E). Stage C is very important because it offers the possibility to apply the approach in real time avoiding the problem with the continuously growing number of data. The stage of rule-base evolution, E, is a very important part of

the overall algorithm, because it defines the model structure (rule-base) evolution. In fact, stages C and D are supportive for this stage, which will be considered in more detail in the next section. In the next stage, F, another important operation is performed, the recursive estimation of the parameters of local sub-models. The last stage, G, is summative, where the prediction is made.

## 4. RULE BASE EVOLUTION

In this section different scenario for the rule-base evolution will be studied. They are summarised in Table 1. The original scenario is adopted by Angelov and Filev (2004) while scenarios D, E, F and G have been proposed by Victor and Dourado (2003).

Table 1 Scenarios for rule-base evolution.

| Scenario | UPGRADE | Closeness check |
|---|---|---|
| Original | $P_k(z_k) > \overline{P}_k$ | $\frac{P_k(z_k)}{\max_{i=1}^{R} P_k(z^{i^*})} - \frac{\delta_{\min}}{r} > 1$ |
| D | $P_k(z_k) > \overline{P}_k$ | $\delta_{\min} < r/2$ |
| E | $P_k(z_k) > \overline{P}_k$ or $0.5\overline{P}_k < P_k(z_k) < 0.675\overline{P}_k$ | $\delta_{\min} < r/2$ |
| F | $P_k(z_k) > \overline{P}_k$ or $P_k(z_k) < \underline{P}_k$ | $\delta_{\min} < r/2$ |
| G | $P_k(z_k) > \overline{P}_k$ or $P_k(z_k) < \underline{P}_k$ | $\delta_{\min} < r/2$ or $\delta_{\min} < 0.85r$ |

where $\underline{P}_k = \min_{i=1}^{R} P_k(z^{i^*})$ and $\overline{P}_k = \max_{i=1}^{R} P_k(z^{i^*})$ denotes respectively the smallest and the biggest of the potentials of the existing rule centers.

Since we consider vector radii in this paper, instead of *r* we will use the norm of the radii, $\|r\| = \sqrt{\frac{1}{n}\sum_{i=1}^{n} r_i^2}$, same as in (Angelov, *et al.*, 2004).

These different scenarios for rule-base evolution have been tested on a benchmark problem: prediction of the Mackey-Glass chaotic time-series, which is generated from the differential delay equation defined by (Angelov and Filev, 2004; Chiu, 1994; Kasabov and Song, 2002; Victor and Dourado, 2003):

$$\dot{v}(t) = \frac{0.2v(t-17)}{1+v^{10}(t-17)} - 0.1v(t);\ v(0) = 1.2 \qquad (23)$$

The aim is using the past values of *v* to predict future values of *v*. The value of the signal 85 steps ahead $y = v(t+85)$ is predicted based on the values of the signal at the current moment, 6, 12, and 18 steps back: $x = [v(t-18), v(t-12), v(t-6), v(t)]^T$.

The following experiment was conducted: 3000 data points, for $t = 201:3200$, are extracted from the time series and used as training data, Fig. 2; 500 data points, for $t = 5001:5500$, are used as testing (validation) data. All the data is put in the same file, i.e. training data plus testing data. The data set has 3500 data samples and the learning mechanism is always active, even for the testing data.

To evaluate the performance of the models we use the RMSE and the NDEI (Non-Dimensional Error Index) (24), the ratio of the root mean square error over the standard deviation of the target data.

$$NDEI = \frac{RMSE}{std(y(t))} \qquad (24)$$

The values of the performance measures will be calculated separately for the training and testing data.

For the scenarios described before the results for a particular value of the constant radii, $r = 0.4$, and initialization parameter for the RLS algorithm $\Omega = 750$ are tabulated in Table 2 (Victor and Dourado, 2003).

Table 2 Results for the different scenarios.

| Scenario | *R* | RMSE Training | RMSE Testing | NDEI Training | NDEI Testing |
|---|---|---|---|---|---|
| Original | 19 | 0.09359 | 0.09853 | 0.37055 | 0.38891 |
| D | 4 | 0.11815 | 0.11481 | 0.46775 | 0.45315 |
| E | 7 | 0.11607 | 0.11063 | 0.45952 | 0.43663 |
| F | 6 | 0.11314 | 0.10900 | 0.44791 | 0.43019 |
| G | 5 | 0.11505 | 0.11178 | 0.45548 | 0.44119 |

From this table it is obvious that for the same value of *r* very different values for the number of rules created and modified are obtained. The performance of the models, measured by the RMSE and NDEI, also varies significantly. We have also considered the use of the different scenario for creating the number of rules. The results for $R = 9$ are presented in Table 3 (Victor and Dourado, 2003).

Table 3 Results for a similar number of rules.

| Scenario | *r* | RMSE Training | RMSE Testing | NDEI Training | NDEI Testing |
|---|---|---|---|---|---|
| D | 0.17 | 0.10723 | 0.10526 | 0.42454 | 0.41545 |
| E | 0.25 | 0.10720 | 0.10388 | 0.42441 | 0.41001 |
| F | 0.25 | 0.10089 | 0.09640 | 0.39942 | 0.38048 |
| G | 0.20 | 0.10539 | 0.10085 | 0.41726 | 0.39806 |

One can see that different radii are necessary to achieve the same rule-base size using the different scenario. From Table 3 we conclude that the performance is better for scenario F. It must be also stated that in scenario E the definition of the relevant threshold, the upper one (see Table 1), was made after some experimentation, which makes this conclusion problem-dependent.

It is very illustrative to compare the rule-base evolution for the different scenario. This is shown for the first 500 data samples to the original scenario on Fig.2 and Fig. 3 (Victor and Dourado, 2003).

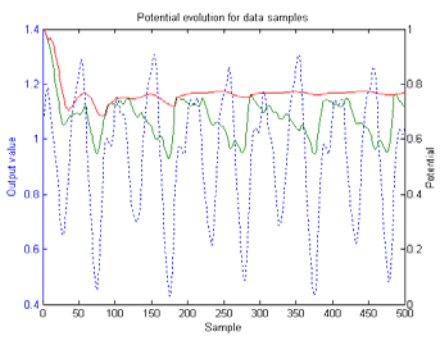


Fig. 2. Potential evolution: $\overline{P}_k$ - red, $P_k(z_k)$ - green.

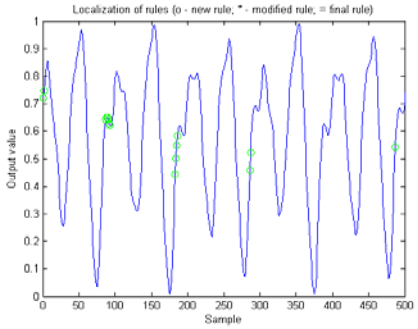


Fig. 3. Rule base evolution for the original scenario.

Similarly, for the scenario F (the best performing overall (Victor and Dourado, 2003)) we have:

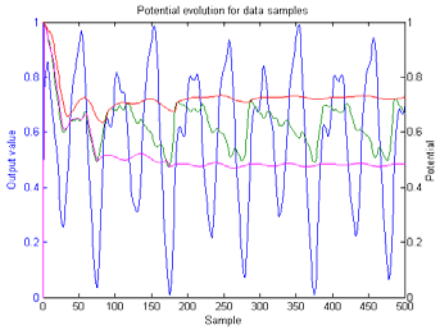


Fig. 4. Potential evolution: $\overline{P}_k$, $P_k(z_k)$, $\underline{P}_k$ - magenta.

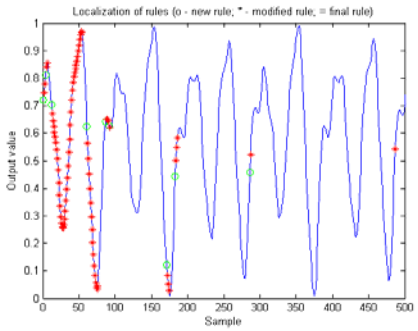


Fig. 5. Rule base evolution for scenario F.

It should be noted that scenario D-F have a very large number of replaced rules (the MODIFY principle is active very often, and the rule base in less stable). The number of replaced rules has been 16, 48, 83, and 85 respectively for the case represented in Table 2, and 11, 46, 80, and 80 for the case represented in the Table 3 (Victor and Dourado, 2003). However, as a whole the eTS method gives a significantly more compact, and hence more transparent rule base than the similar neuro-fuzzy approaches, Table 4.

Table 4 Comparative analysis.

| Methods | Rules/Units | NDEI |
|---|---|---|
| RAN (Kasabov and Song, 2002) | 113 units | 0.373 |
| ESOM (Kasabov and Song, 2002) | 114 units | 0.320 |
| EfuNN (Kasabov and Song, 2002) | 193 rule nodes | 0.401 |
| DENFIS (Kasabov and Song, 2002) | 58 fuzzy rules | 0.276 |
| eTS (Angelov and Filev, 2004) | 113 fuzzy rules | 0.095 |
| eTS (Victor and Dourado, 2003) | 9 fuzzy rules | 0.380 |

## 5. CONCLUDING REMARKS

Different scenarios are considered for the rule base evolution. The radius of influence of each fuzzy rule is considered to be a vector instead of a scalar, allowing different areas of the data space to be covered by each input variable. Simulation results using a well-known benchmark (Mackey-Glass chaotic time-series prediction) are considered in this paper. Further investigation concern the application of eTS to predictive modelling of the speech spectrum magnitude, classification of multi-channel source modulation.

## REFERENCES


Angelov, P. (2002). *Evolving Rule-Based Models: A Tool for Design of Flexible Adaptive Systems*. Springer-Verlag, Heidelberg.

Angelov, P. and D. Filev (2004). An Approach to Online Identification of Takagi-Sugeno Fuzzy Models. *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, **34**(1), 484-498.

Angelov P., C. Xydeas and D. Filev (2004). On-line identification of MIMO evolving Takagi-Sugeno models. *FUZZ-IEEE Conference*, to appear.

Chiu, S. (1994). Fuzzy Model identification Based on Clustering Estimation. *Journal of Intelligent and Fuzzy Systems*, **2**, 267-278.

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

Victor, J. and A. Dourado (2003). Evolving Takagi-Sugeno Fuzzy Models. *Technical Report*, CISUC, http://cisuc.dei.uc.pt/acg/view_pub.php?id_p=760.

Takagi, T. and M. Sugeno (1985). Fuzzy Identification of Systems and Its Applications to Modeling and Control. *IEEE Transactions on Systems, Man, and Cybernetics*, **15**(1), 116-132.

R. Yager and D. Filev (1993). Learning of Fuzzy Rules by Mountain Clustering, *Proc. SPIE Conf. on Appl. of Fuzzy Logic Technology*, Boston, MA, USA, 246-254.

Yager, R. and D. Filev (1994). *Essentials of Fuzzy Modeling and Control*. John Wiley, NY.