A HYBRID APPROACH TO LOCALLY OPTIMIZED INTERPRETABLE PARTITIONS OF FUZZY NEURAL MODELS

Wanqing Zhao[1], Kang Li[1], George W. Irwin[1] and Qun Niu[2]
[1] *Intelligent Systems and Control, Queen's University Belfast, Stranmillis Road, Belfast, U.K.*
[2] *School of Mechatronical Engineering and Automation, Shanghai University, Shanghai, China*

Keywords: Fuzzy neural systems, Interpretable model, Differential evolution, Weighted fast recursive algorithm, ANFIS.

Abstract: Many learning methods have been proposed for Takagi-Sugeno-Kang fuzzy neural modelling. However, despite achieving good global performance, the local models obtained often exhibit eccentric behaviour which is hard to interpret. The problem here is to find a set of input space partitions and, hence, to identify the corresponding local models which can be easily understood in terms of system behaviour. A new hybrid approach for the construction of a locally optimized, functional-link-based fuzzy neural model is proposed in this paper. Unlike the usual linear polynomial models used for the rule consequent, the functional link artificial neural network (FLANN) is employed here to achieve a nonlinear mapping from the original model input space. Our hybrid learning method employs a modified differential evolution method to give the best fuzzy partitions along with the weighted fast recursive algorithm for the identification of each local FLANN. Results from a motorcycle crash dataset are included to illustrate the interpretability of the resultant model structure and the efficiency of the new learning technique.

1 INTRODUCTION

Takagi-Sugeno-Kang (TSK) fuzzy neural systems have been widely applied to the modelling of nonlinear dynamic systems from noisy datasets. In fact, a TSK fuzzy model can be viewed as a number of local models valid in different input spaces. However, due to the linear polynomial form of the rule consequent, this representation may not capture fully all the information contained in the original input space. A functional-link-based neural fuzzy network, consisting of nonlinear combinations of the model inputs has therefore been proposed for the rule consequent (Lin et al., 2009).

The most important issue for nonlinear modelling is to optimize the fuzzy neural system parameters using the given data. To deal with possible local minima, heuristic algorithms can either be directly applied, or else integrated with more conventional methods to enhance the training accuracy. For example the random optimization approach has been successfully used to update the premise parameters (Cheng, 2009). Given sufficient rules and training data, there are many techniques available for producing an accurate TSK fuzzy model. However, the interpretability of the local models obtained cannot always be guaranteed.

The goal of this paper is to construct a locally optimized fuzzy neural model, consisting of a set of fully-interpretable local models. Unlike (Lin et al., 2009), the parameters to be learned are divided into two subsets, corresponding to the premise and consequent parts of the model. A simple modification to differential evolution (DE) is described and then utilized to optimize the nonlinear parameters in the rule premises. A weighted version of the fast recursive algorithm (FRA) (Li et al., 2005) is also derived for determining the structure and identifying the linear parameters locally in the rule consequents. An application study and comparison with ANFIS illustrated the interpretability of the resultant model structure and the efficiency of the new learning technique.

2 FUNCTIONAL-LINK-BASED FUZZY NEURAL SYSTEMS

Functional-link-based fuzzy neural systems are represented by the following:

$$\begin{aligned} R_i : \texttt{IF } x_1(t) = A_{i,1} \texttt{ AND } x_2(t) = A_{i,2} \texttt{ AND} \ldots \\ \texttt{AND } x_n(t) = A_{i,n}, \texttt{ THEN } \hat{y}_i(t) = f_i(\mathbf{Z}_i(t);\boldsymbol{\theta}_i) \end{aligned} \quad (1)$$

Zhao W., Li K., W. Irwin G. and Niu Q..
A HYBRID APPROACH TO LOCALLY OPTIMIZED INTERPRETABLE PARTITIONS OF FUZZY NEURAL MODELS.
DOI: 10.5220/0003626304610465
In *Proceedings of the International Conference on Evolutionary Computation Theory and Applications* (FCTA-2011), pages 461-465
ISBN: 978-989-8425-83-6

where i is the rule index, $\mathbf{X}(t) = [x_1(t), \ldots, x_n(t)]^{\mathrm{T}} \in \Re^n$ represents the n model inputs, $A_{i,j}$ is the fuzzy set associated with the ith rule corresponding to the input x_j, $\hat{y}_i(t)$ is the local model output realized using the output of a functional link artificial neural network $f_i(\cdot)$, $\mathbf{Z}_i(t) = [z_{i,1}(t), \ldots, z_{i,q_i}(t)]^{\mathrm{T}} \in \Re^{q_i}$ defines the q_i most significant basis functions of the input variables selected for the ith output of the FLANN, and $\boldsymbol{\theta}_i$ is the corresponding coefficient vector.

In a FLANN, the input vector $\mathbf{X}(t)$ is functionally enhanced by a set of linearly independent functions chosen from an orthonormal basis set, to give the output $s_i(t)$:

$$s_i(t) = \sum_{j=1}^{q} \omega_{i,j} p_j(\mathbf{X}(t)) \tag{2}$$

where q denotes the total number of functional expansions for all inputs, $\theta_{i,j}$ is the weight parameter relating to the ith output of the FLANN and p_j represents the jth basis function. As with (Lin et al., 2009), trigonometric expansions are used for the expanders. Now the ith output of FLANN is taken as the consequent part for the ith rule. To give a compact local model, and to enhance the understandability, a subset q_i of the most significant combination of terms selected from (2) form the ith rule consequent. Thus

$$f_i(\mathbf{Z}_i(t); \boldsymbol{\theta}_i) = \sum_{j=1}^{q_i} \theta_{i,j} z_{i,j}(t);\ j = 1, \ldots, q_i \tag{3}$$

Here $\boldsymbol{\theta}_i = [\theta_{i,1}, \ldots, \theta_{i,q_i}]^{\mathrm{T}} \in \Re^{q_i}$ are the coefficients associated with the ith output of the FLANN. For N data samples, $\mathbf{z}_{i,j} = [z_{i,j}(1), \ldots, z_{i,j}(N)]^{\mathrm{T}} \in \Re^N$ and $\mathbf{p}_j(\mathbf{X}) = [p_j(\mathbf{X}(1)), \ldots, p_j(\mathbf{X}(N))]^{\mathrm{T}} \in \Re^N$, then $\mathbf{Z}_i = [\mathbf{z}_{i,1}, \ldots, \mathbf{z}_{i,q_i}]$ will be a subset of $\mathbf{P} = [\mathbf{p}_1(\mathbf{X}), \ldots, \mathbf{p}_q(\mathbf{X})]$.

Assuming Gaussian membership functions are used in the fuzzy sets, the firing strength of the ith fuzzy rule can be computed from the T-norm:

$$\mu_i(\mathbf{X}(t); \mathbf{w}_i) = \prod_{j=1}^{n} \exp\left\{ -\frac{1}{2} \left(\frac{x_j(t) - c_{i,j}}{\sigma_{i,j}} \right)^2 \right\} \tag{4}$$

where $c_{i,j}$ and $\sigma_{i,j}$ denote the centre and standard deviation of $A_{i,j}$, $\mathbf{w}_i = [\mathbf{c}_i^{\mathrm{T}}, \boldsymbol{\sigma}_i^{\mathrm{T}}]^{\mathrm{T}} \in \Re^{2n}$, $\mathbf{c}_i = [c_{i,1}, \ldots, c_{i,n}]^{\mathrm{T}} \in \Re^n$, and $\boldsymbol{\sigma}_i = [\sigma_{i,1}, \ldots, \sigma_{i,n}]^{\mathrm{T}} \in \Re^n$. The normalized firing strength of the ith rule which determines its region of validity is now given by

$$N_i(\mathbf{X}(t); \mathbf{W}) = \mu_i(\mathbf{X}(t); \mathbf{w}_i) / \sum_{i=1}^{N_r} \mu_i(\mathbf{X}(t); \mathbf{w}_i) \tag{5}$$

where $\mathbf{W} = [\mathbf{w}_1^{\mathrm{T}}, \ldots, \mathbf{w}_{N_r}^{\mathrm{T}}]^{\mathrm{T}}$ and N_r is the number of fuzzy rules. A weighted-average-defuzzification can be employed to give the output of the fuzzy system:

$$f(\mathbf{X}(t); \boldsymbol{\Theta}) = \sum_{i=1}^{N_r} N_i(\mathbf{X}(t); \mathbf{W})\, f_i(\mathbf{Z}_i(t); \boldsymbol{\theta}_i) \tag{6}$$

3 HYBRID LEARNING METHOD

A modified differential evolution (DE) algorithm (Storn and Price, 1997) is performed to globally optimize the nonlinear parameters $\mathbf{W}$. The network structure and associated parameters for each FLANN in (3) are locally determined by a weighted version of fast recursive algorithm (FRA) (Li et al., 2005).

3.1 DE for Optimizing Rule Premises

The DE algorithm (Storn and Price, 1997) is employed to optimize the fuzzy partitions for the rule premise represented by the parameter vector $\mathbf{W}$ as follows.

1) **Mutation.** For each individual solution $\mathbf{W}_\beta(g)$, three other indexes $\alpha1$, $\alpha2$, and $\alpha3$ are randomly selected between 1 and the size of the population N_p (with $\beta \neq \alpha1 \neq \alpha2 \neq \alpha3$). A new trial vector $\mathbf{V}_\beta(g+1)$ is created as

$$DE/rand/1 : \mathbf{V}_\beta(g+1) = \mathbf{W}_{\alpha3}(g) + \mathrm{F}(\mathbf{W}_{\alpha1}(g) - \mathbf{W}_{\alpha2}(g))$$

where g represents the generation and F is a mutation control parameter. Another mutation strategy is (Qin and Suganthan, 2005):

$$DE/cur-best/2 : \mathbf{V}_\beta(g+1) = \mathbf{W}_\beta(g) + \mathrm{F}(\mathbf{W}_{best}(g) - \mathbf{W}_\beta(g)) + \mathrm{F}(\mathbf{W}_{\alpha1}(g) - \mathbf{W}_{\alpha2}(g))$$

where $\mathbf{W}_{best}(g)$ is the best solution in the gth generation.

2) **Crossover.** Binomial crossover is now performed on the mutated trial vector $\mathbf{V}_\beta(g+1)$. For each gene in the vector, a random number $\Upsilon_j (j = 1, \ldots, 2nN_r)$ within the range [0, 1] is generated.

$$\mathbf{V}_{\beta j}(g+1) = \begin{cases} \mathbf{V}_{\beta j}(g+1); & \Upsilon_j \leq CR \ \ or \ \ j = I_j \\ \mathbf{W}_{\beta j}(g); & otherwise \end{cases} \tag{7}$$

In (7), I_j is a randomly chosen integer in the set I, i.e., $I_j \in \mathrm{I} = \{1, \ldots, 2nN_r\}$ and j represents the jth component of the trial vector. CR is the crossover constant lying in the range [0, 1].

3) **Selection.** The objective function used to evaluate an individual solution is given by

$$J = \frac{1}{N} \sum_{t=1}^{N} (y(t) - \hat{y}(t))^2 \tag{8}$$

where $y(t)$ is the desired output and $\hat{y}(t)$ is the output of the overall fuzzy model. To decide whether the vector $\mathbf{V}_\beta(g+1)$ should be included at the next generation, it is compared with the corresponding vector $\mathbf{W}_\beta(g)$ as follows:

$$\mathbf{V}_\beta(g+1) = \begin{cases} \mathbf{V}_\beta(g+1); & J(\mathbf{V}_\beta(g+1)) < J(\mathbf{W}_\beta(g)) \\ \mathbf{W}_\beta(g); & otherwise \end{cases} \tag{9}$$

Steps 1)-3) are repeated until the specified termination criterion is reached.
4) **Modified Differential Evolution.** To balance the exploration ability and the stability of the algorithm, it is proposed that the two mutation variants in the mutation part are combined. Here, the trial vectors produced by DE/rand/1 occur with a probability M_p and those that from DE/cur-best/2 occur with a probability 1-M_p.

$$\mathbf{V}_\beta(g+1) = \begin{cases} DE/rand/1; & rand < M_p \\ DE/cur-best/2; & otherwise \end{cases} \quad (10)$$

3.2 Weighted FRA for Identification of Rule Consequents

A weighted version of FRA (Li et al., 2005) is now derived and applied to determine the structure of the rule consequents and to identify the associated linear parameters in (3). Instead of estimating all the parameters simultaneously, N_r separate local estimations are carried out for each local model. The output of each local FLANN can be expressed as

$$\hat{\mathbf{y}}_i = \mathbf{Z}_i\boldsymbol{\theta}_i \quad (11)$$

where $\mathbf{Z}_i = [\mathbf{Z}_i(1),\ldots,\mathbf{Z}_i(N)]^{\mathrm{T}}$. To estimate all the local linear models, weighted least-squares is employed using the cost function

$$E_i(\boldsymbol{\theta}_i) = \mathbf{e}_i^{\mathrm{T}}\mathbf{Q}_i\mathbf{e}_i \quad (12)$$

where $\mathbf{e}_i = [e_i(1),\ldots,e_i(N)]^{\mathrm{T}}$ and $e_i(t) = y(t) - \hat{y}_i(t)$ denotes the ith model errors for data sample $\{\mathbf{Z}_i(t),y(t)\}$. The weighting matrix $\mathbf{Q}_i = diag(N_i(\mathbf{X}(1),\mathbf{W}),\ldots,N_i(\mathbf{X}(N),\mathbf{W}))$, has the advantage that knowledge about the confidence in each data sample which is determined by each fuzzy partition.

Assuming there are N_r fuzzy partitions, the full regression matrix is chosen as $\mathbf{Z}_i = \mathbf{P}$ with each column denoted as $\mathbf{Z}_i = [\mathbf{p}_1,\ldots,\mathbf{p}_q]$, and $\mathbf{p}_k = [p_k(1),\ldots,p_k(N)]^{\mathrm{T}}$, $k = 1,\ldots,q$. Letting $\mathbf{Z}_i^{(k)} = [\mathbf{z}_{i,1},\ldots,\mathbf{z}_{i,k}] \in \Re^{N\times k}$, the local model output weights are thus computed as

$$\boldsymbol{\theta}_i = (\mathbf{Z}_i^{(k)\mathrm{T}}\mathbf{Q}_i\mathbf{Z}_i^{(k)})^{-1}\mathbf{Z}_i^{(k)\mathrm{T}}\mathbf{Q}_i\mathbf{y} \quad (13)$$

The accuracy criterion in (12) is now given by

$$E_i^{(k)}(\mathbf{Z}_i^{(k)}) = \mathbf{y}^{\mathrm{T}}\mathbf{Q}_i^{1/2}\mathbf{R}_i^{(k)}\mathbf{Q}_i^{1/2}\mathbf{y} \quad (14)$$

where $\mathbf{R}_i^{(k)}$ $(0 < k \le q)$, referred to as the residual matrix is defined as

$$\mathbf{R}_i^{(k)} = \mathbf{I} - \mathbf{Q}_i^{1/2}\mathbf{Z}_i^{(k)}(\mathbf{Z}_i^{(k)\mathrm{T}}\mathbf{Q}_i\mathbf{Z}_i^{(k)})^{-1}\mathbf{Z}_i^{(k)\mathrm{T}}\mathbf{Q}_i^{1/2} \quad (15)$$

Specifically, the following properties hold for $\mathbf{R}_i^{(k)}$:

$$\mathbf{R}_i^{(k+1)} = \mathbf{R}_i^{(k)} - \frac{\mathbf{R}_i^{(k)}\mathbf{Q}_i^{1/2}\mathbf{z}_{i,k+1}\mathbf{z}_{i,k+1}^{\mathrm{T}}\mathbf{Q}_i^{1/2}\mathbf{R}_i^{(k)}}{\mathbf{z}_{i,k+1}^{\mathrm{T}}\mathbf{Q}_i^{1/2}\mathbf{R}_i^{(k)}\mathbf{Q}_i^{1/2}\mathbf{z}_{i,k+1}} \quad (16)$$

$$\mathbf{R}_i^{(k)\mathrm{T}} = \mathbf{R}_i^{(k)};\ \mathbf{R}_i^{(k)}\mathbf{R}_i^{(k)} = \mathbf{R}_i^{(k)};\ k = 0,\ldots,q \quad (17)$$

$$\mathbf{R}_i^{(k)}\mathbf{R}_i^{(j)} = \mathbf{R}_i^{(j)}\mathbf{R}_i^{(k)} = \mathbf{R}_i^{(k)};\ k \ge j;\ k,j = 0,\ldots,q \quad (18)$$

$$\mathbf{R}_i^{(k)}\mathbf{Q}_i^{1/2}\mathbf{p}^{(i)} = \mathbf{0};\ \ rank[\mathbf{z}_{i,1},\ldots,\mathbf{z}_{i,k},\mathbf{p}^{(i)}] = k \quad (19)$$

The derivation of the above properties is similar to (Li et al., 2005). Letting $\mathbf{p}^{(i)}$ denote those terms relating to the ith model which are still not included in the regression matrix $\mathbf{Z}_i^{(k)}$, the net contribution to the cost function from adding $\mathbf{p}^{(i)}$ to the model is

$$\Delta E_i^{(k+1)}(\mathbf{Z}_i^{(k)};\mathbf{p}^{(i)}) = E_i^{(k)}(\mathbf{Z}_i^{(k)}) - E_i^{(k+1)}(\mathbf{Z}_i^{(k)};\mathbf{p}^{(i)}) \quad (20)$$

Using (16), this net contribution can be more easily calculated as:

$$\Delta E_i^{(k+1)}(\mathbf{Z}_i^{(k)};\mathbf{p}^{(i)}) = \frac{\mathbf{y}^{\mathrm{T}}\mathbf{Q}_i^{1/2}\mathbf{R}_i^{(k)}\mathbf{Q}_i^{1/2}\mathbf{p}^{(i)}\mathbf{p}^{(i)\mathrm{T}}\mathbf{Q}_i^{1/2}\mathbf{R}_i^{(k)}\mathbf{Q}_i^{1/2}\mathbf{y}}{\mathbf{p}^{(i)\mathrm{T}}\mathbf{Q}_i^{1/2}\mathbf{R}_i^{(k)}\mathbf{Q}_i^{1/2}\mathbf{p}^{(i)}} \quad (21)$$

For efficient computation of $\Delta E_i^{(k+1)}(\mathbf{Z}_i^{(k)};\mathbf{p}^{(i)})$, the following quantities are defined:

$$\begin{cases} a_{k+1,c}^{(i)} = \mathbf{z}_{i,k+1}^{\mathrm{T}}\mathbf{Q}_i^{1/2}\mathbf{R}_i^{(k)}\mathbf{Q}_i^{1/2}\mathbf{z}_{i,c} \\ a_{k+1,y}^{(i)} = \mathbf{y}^{\mathrm{T}}\mathbf{Q}_i^{1/2}\mathbf{R}_i^{(k)}\mathbf{Q}_i^{1/2}\mathbf{z}_{i,k+1} \\ (k = 0,\ldots,q-1;\ c = k+1,\ldots,q) \end{cases} \quad (22)$$

and successively using (16), gives

$$\begin{cases} a_{k+1,c}^{(i)} = \mathbf{z}_{i,k+1}^{\mathrm{T}}\mathbf{Q}_i\mathbf{z}_{i,c} - \sum_{j=1}^{k} a_{j,k+1}^{(i)}a_{j,c}^{(i)}/a_{j,j}^{(i)} \\ a_{k+1,y}^{(i)} = \mathbf{y}^{\mathrm{T}}\mathbf{Q}_i\mathbf{z}_{i,k+1} - \sum_{j=1}^{k} a_{j,y}^{(i)}a_{j,k+1}^{(i)}/a_{j,j}^{(i)} \end{cases} \quad (23)$$

Now, suppose a number of k regressors have been added into $\mathbf{Z}_i^{(k)}$ from the full regression matrix $\mathbf{Z}_i$, A variable $k\times q$ upper triangular matrix $\mathbf{A}$ is defined as

$$\mathbf{A} = [a_{r,c}^{(i)}]_{k\times q}$$
$$a_{r,c}^{(i)} = \begin{cases} 0; & c < r \\ \mathbf{z}_{i,r}^{\mathrm{T}}\mathbf{Q}_i^{1/2}\mathbf{R}_i^{(r-1)}\mathbf{Q}_i^{1/2}\mathbf{z}_{i,r}; & r \le c \le k \\ \mathbf{z}_{i,r}^{\mathrm{T}}\mathbf{Q}_i^{1/2}\mathbf{R}_i^{(r-1)}\mathbf{Q}_i^{1/2}\mathbf{p}_c^{(i)}; & k < c \le q \end{cases} \quad (24)$$

Finally $\mathbf{z}_{i,k+1} = \arg\max \Delta E_i^{(k+1)}(\mathbf{Z}_i^{(k)};\mathbf{p}_r^{(i)})$, which means that the $\mathbf{p}_r^{(i)}$ giving the *maximum* value of $\Delta E_i^{(k+1)}(\mathbf{Z}_i^{(k)};\mathbf{p}_r^{(i)})$ can be selected as the $(k+1)$th basis vector in $\mathbf{Z}_i^{(k+1)}$ for the ith local model. Akaike's information criterion (AIC) is adopted as the stopping condition here. To avoid linear correlated terms in $\mathbf{Z}_i^{(k+1)}$, small values for the diagonal entries $\mathbf{A}$ must not be generated. In this situation, the term leading to the *second* largest reduction in $\Delta E_i^{(k+1)}(\mathbf{Z}_i^{(k)};\mathbf{p}^{(i)})$ is employed. Assuming that q_i terms have been selected for the ith rule consequent by the proposed algorithm, the solution for each local model is then given by:

$$\begin{cases} \hat{\theta}_{i,j} = (a_{j,y}^{(i)} - \sum_{c=j+1}^{q_i} \hat{\theta}_{i,c}a_{j,c}^{(i)})/a_{j,j}^{(i)} \\ (i = 1,\ldots,N_r;\ j = q_i,\ldots,1) \end{cases} \quad (25)$$

4 MOTORCYCLE DATASET

A simulated motorcycle crash dataset (Silverman, 1985) is used as a illustrative example. This dataset consists of a series of measurements of head acceleration in a simulated motorcycle accident. A total of 133 one-dimensional time-series accelerometer readings were recorded experimentally. Note that the time points are not regularly spaced, and there are multiple observations at some instants. The interest here is to determine the general nature of the underlying acceleration as a function of time soon after an impact by using a locally optimized, functional-link-based fuzzy neural model with time and acceleration taken as the input and output respectively. Modelling was done using 67 readings for training and the remaining 66 records as a test dataset. All the samples were normalized to lie in the range [0, 1], thus limiting the centres and widths of the Gaussian membership functions to the same range.

The hybrid learning approach involved 300 iterations with 40 individuals in each population. The values of F, CR, and M_p were set as 0.7, 0.9, and 0.5 respectively, and the learning process was repeated for 10 runs. The number of evaluations for each run was therefore 40(individuals)×300(iterations) = 12,000. As with ANFIS, the number of rules was determined by trial-and-error, four fuzzy rules finally being adopted in this application. The overall mean sum-squared error of the best fuzzy neural model obtained was 6.34×10^{-3} on the training data and 15.6×10^{-3} on the test data, both comparable with the corresponding ANFIS results to be discussed later. The final fuzzy model with the best performance overall was defined by the following rule:

$$
\begin{aligned}
R_1 &: \texttt{IF } x_1 \textit{ is } \mu_{1,1}(0.2982; 0.0453) \\
&\quad \texttt{THEN } f_1 \textit{ is } -2.9995\cos(\pi x_1) \\
&\quad -5.5678\sin(\pi x_1) + 6.4335 \\
R_2 &: \texttt{IF } x_1 \textit{ is } \mu_{2,1}(0.1028; 0.0481) \\
&\quad \texttt{THEN } f_2 \textit{ is } 0.0418 + 0.5836\cos(\pi x_1) \\
&\quad +0.1290\sin(\pi x_1) \\
R_3 &: \texttt{IF } x_1 \textit{ is } \mu_{3,1}(0.4815; 0.0458) \\
&\quad \texttt{THEN } f_3 \textit{ is } 1.8445 x_1 - 5.4456 + 5.2852\sin(\pi x_1) \\
R_4 &: \texttt{IF } x_1 \textit{ is } \mu_{4,1}(0.7959; 0.0967) \\
&\quad \texttt{THEN } f_4 \textit{ is } 0.3436 + 0.6079\cos(\pi x_1) + 0.9580 x_1
\end{aligned} \tag{26}
$$

where $\mu_{i,1}(c_{i,1}; \sigma_{i,1})$ denotes the centre $c_{i,1}$ and standard deviation $\sigma_{i,1}$ of the ith membership function with regard to the input, used to partition the time axis into different local regions, and f_i represents the corresponding local predicted acceleration ($i = 1, 2, 3, 4$).

To obtain a visual understanding of each fuzzy local model over the test dataset, the behaviour of each has been characterized in its corresponding working region as shown in Fig. 1. (Here only the dominant rule for both ANFIS and our method is shown for each local region). The rule premises are used to generate these working regions and the behaviour within each is defined by the rule consequents. It can be seen that the predicted acceleration of each local model matches well the measured value in each local time interval. As required, the locally optimized models can therefore be individually interpreted as a description of the identified nonlinear behaviour within the regime represented by the corresponding rule premise. These properties thus allow one to gain insight into the model behaviour and thus to improve its interpretability as required. Furthermore, in (26) the f_i (i=1,2,3,4) also show the relative order of importance of the combinations of FLANN terms included. This could be helpful in understanding the evolution of the head acceleration over time.

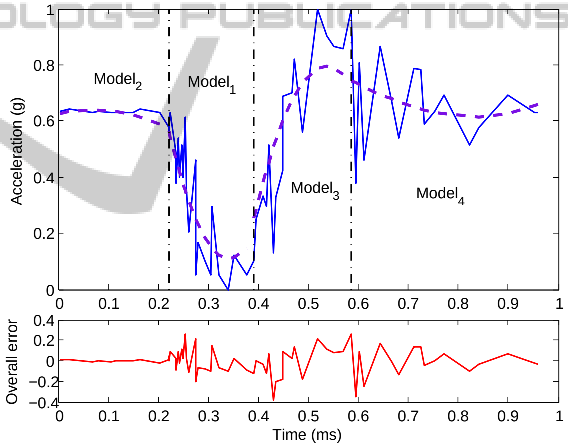

Figure 1: Proposed method over the test dataset (The dotted line represents each local model behaviour as distinguished by the vertical dashed line, the solid line is the original test data, and the bottom curve stands for the testing error between original data and overall model output).

For comparison, the well-known ANFIS model trained by another two-stage method combining steepest descent with least-squares was applied. The overall training error and testing error were now 6.79×10^{-3} and 16.3×10^{-3} respectively. Both the ANFIS model and ours are capable of producing good accuracy in terms of the error between the measured acceleration and that produced by the model. The local models produced by ANFIS are as shown in Fig. 2. In this case its clear that these local models are less successful, particular so around the time interval [0.29 0.41]. To evaluate the local models and to compare ANFIS and our method, the mean sum-squared

Table 1: Comparison of local performance of ANFIS and our method.

Model/Method	ANFIS (training)	Proposed (training)	ANFIS (testing)	Proposed (testing)
Local Model 1	63.6×10^{-3}	7.5×10^{-3}	55.5×10^{-3}	12.0×10^{-3}
Local Model 2	31.1×10^{-3}	0.2×10^{-3}	73.5×10^{-3}	0.3×10^{-3}
Local Model 3	349.5×10^{-3}	7.4×10^{-3}	276.8×10^{-3}	25.0×10^{-3}
Local Model 4	1.5×10^{-3}	7.4×10^{-3}	2.7×10^{-3}	22.2×10^{-3}

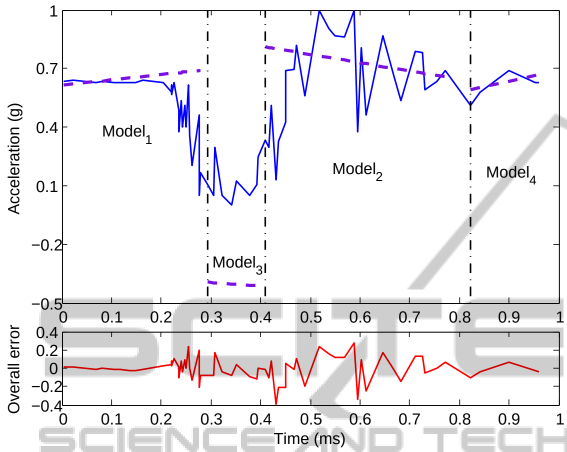

Figure 2: ANFIS model over the test dataset.

error of each dominant local model in its corresponding time interval are listed in Table 1. This shows that the local models obtained by ANFIS did not always give an accurate approximation within the local regions. Note especially the eccentric behaviour of submodel three (see Fig. 2 and Table 1). Fig. 1 and the results in Table 1 confirm that the method proposed here led to an interpretable fuzzy neural model with respect to all the local model behaviours. This was not the case for the ANFIS method. Notice also that the fourth local ANFIS model gave good results due to the small (5) number of data points coupled with the low noise in this time interval. Due to the local characteristic of the weighted FRA, the hybrid approach was also able to find interpretable partitions in a more complex problem than the one above. This has not been included due to lack of space.

5 CONCLUSIONS AND FUTURE WORK

In contrast to existing learning methods for fuzzy neural models which focus on the overall accuracy of the model, and which may lead to uninterpretable eccentric behaviour in the local models, the aim here has been to obtain a set of interpretable input space partitions and the corresponding local models. Taking into account the two different parameter types corresponded to the rule premise and rule consequent, a new hybrid learning approach has been proposed. This employed a modified differential evolution method to give the best fuzzy partitions, along with a weighted fast recursive algorithm for identification of each local FLANN. An application study and comparison with ANFIS illustrated the interpretability of the resultant model structure and the efficiency of the new learning technique. Further improvements might accrue from investigating other evolutionary strategies to optimize the rule premises.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the UK-China Science Bridge grant EP/G042594/1, EP/F021070/1, and the China Scholarship Council.

REFERENCES

Cheng, K. H. (2009). CMAC-based neuro-fuzzy approach for complex system modeling. *Neurocomputing*, 72(7-9):1763–1774.

Li, K., Peng, J. X., and Irwin, G. W. (2005). A fast nonlinear model identification method. *IEEE Transactions on Automatic Control*, 50(8):1211–1216.

Lin, C. J., Chen, C. H., and Lin, C. T. (2009). A hybrid of cooperative particle swarm optimization and cultural algorithm for neural fuzzy networks and its prediction applications. *IEEE Transactions on Systems, Man, and Cybernetics*, 39(1):55–68.

Qin, A. K. and Suganthan, P. N. (2005). Self-adaptive differential evolution algorithm for numerical optimization. In *Proceedings of the IEEE congress on evolutionary computation*, volume 2, pages 1785–1791.

Silverman, B. W. (1985). Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society Series B*, 47(1):1–52.

Storn, R. and Price, K. (1997). Differential evolution - A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341–359.