Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Adaptive Noise Cancellation Using Partially Recurrent Fuzzy System

Meng Joo Er and Zhengrong Li

Abstract— In this paper, a partially recurrent fuzzy system is developed to function as an adaptive noise canceller. In order to cancel noise distorting the information signal, the temporal information (dynamics) underlying the noise source and the distorting noise, which is generated by the noise source passing through some unknown channels, should be captured accurately. For this purpose, short-term memory is embedded into the input layer of the fuzzy system for handling local time information and internal feedback is introduced into the consequent part for processing global time information by virtue of a partially recurrent mechanism. A novel adaptive algorithm is proposed to tune the parameters of the premise and consequent part online. Simulation studies show that the proposed fuzzy system can cancel noise cancellation successfully for nonlinear dynamic channels.

I. Introduction

An adaptive noise canceller should be capable of capturing the dynamics of the channel which the noise source passes through. In other words, it should handle temporal information underlying the noise source and the additive noise that distorts information signal.

Normally, there are two kinds of distinguished temporal information, normally local and global according to the reference time interval. Local time information refers to a part of the time series with fixed length (time window), whereas global information is related to the entire time series up to a certain (usually the current) point in time.

It is possible to use a feedback structure to form a recurrent structure so as to handle global time information [1]. In [2], a fuzzy identification algorithm, under the fully recurrent mechanism, is proposed for a single-input-single-output continuous-time nonlinear dynamic system. Moreover, convergence analysis using the GD-based method is investigated. In [3], a class of architectures called Nonlinear AutoRegressive models with eXogenous inputs (NARX) recurrent neural networks is adopted to lessen long-term dependencies between input and output data. Moreover, most of the fuzzy inference systems need to assign fuzzy rules in advance according to some a priori knowledge or by using clustering-based methods [4]–[6] and they employ offline training algorithms. These methods are not applicable for the case of adaptive noise cancellation because the actual desired signal (original information) is not available due to distortion from the additive noise. Therefore, it is desired to develop a new training strategy so that fuzzy rules can be generated and optimized during the online training process.

In this paper, a partially recurrent fuzzy system is proposed to function as an adaptive noise canceller. A short-term memory structure is embedded into the input layer to form a focused Time-Lagged Feedforward Network (TLFN) in order to handle local time information from the input sequence. The internal feedback forms partially recurrency in the consequent part in order to deal with the long-term dynamics (global time information) underlying input/output sequences. Based on the linear structure in consequent part, an appropriate training algorithm is proposed.

The rest of the paper is organized as follows. Section II discusses the structure of the proposed partially recurrent fuzzy system. An adaptive strategy, which adopts a potential measurement of temporal-spatial proximity for tuning the premise part and an improved recursive method for determining free parameters of the consequent part, is developed in Section III. Simulation studies in Section IV show that the partially recurrent fuzzy system can function as an adaptive noise canceller effectively and remove additive noise by capturing the channel's dynamics. Section V concludes the paper.

II. Architecture of Partially Recurrent Fuzzy Systems

The proposed partially recurrent fuzzy system, whose consequent part consists of linear dynamics submodels, is implemented by an Ellipsoidal-Basis-Function Network (EBFN). Its block diagram is shown in Fig. 1. The input layer is in the form of TLFN which embeds a short-term memory structure in the form of a tapped delay line (see [7] for more details).

The functions of each layer and its nodes are described below:

Layer 1: Each node in layer 1 is an input node. These nodes simply transmit input signals (input signal and their lagged versions) to the next layer directly. In this layer, we have

$$X(k) = [x(k), x(k-1), ..., x(k-M+1)]^T \qquad (1)$$

where M is the order of the lagged inputs.

Layer 2: Nodes in this layer stand for input terms associated with the input variables. In this layer, each input variable is characterized by

$$A_i = [a_{i1}, a_{i2}, ..., a_{ir}] \qquad i = 1, 2, ...M \qquad (2)$$

Meng Joo Er is with Intelligent Systems Centre, 50, Nanyang Drive, 7th Storey, Research TechnoPlaza, BorderX Block, Singapore, 637533 emjer@ntu.edu.sg

Zhengrong Li is with School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, 639798 lizhengrong@pmail.ntu.edu.sg

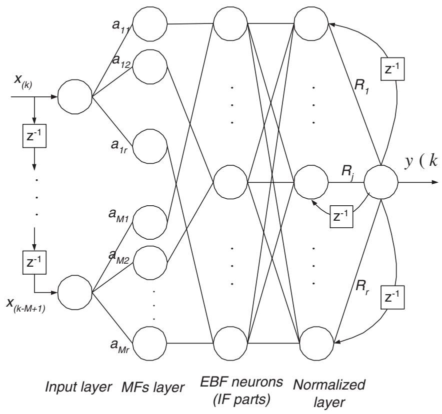

Fig. 1. EBFN-based TSK fuzzy system

where A_i is the input term set associated with the ith input variable $x(k-i+1)$, r is the number of fuzzy rules and a_{ij} is a fuzzy number with a one-dimensional membership function (MF) which is a Gaussian function of the following form:

$$\mu_{ij}(x(k-i+1)) = exp[-\frac{(x(k-i+1)-c_{ij})^2}{\sigma_{ij}^2}] \quad (3)$$

where $j = 1, 2, ...r$, c_{ij} and σ_{ij} are the center and width of the jth Gaussian membership function of the ith lagged input $x(k-i+1)$ respectively.

Layer 3: Each node in layer 3 is an EBF neuron which represents the premise part of a fuzzy IF-THEN rule. For the jth EBF neuron, i.e., the jth fuzzy rule, its firing strength is

$$\begin{aligned} \phi_j &= \prod_{i=1,2,...M} \mu_{ij} \\ &= exp[-\sum_{i=1}^{M} \frac{(x(k-i+1)-c_{ij})^2}{\sigma_{ij}^2}] \quad (4) \end{aligned}$$

Equivalently, the firing strength can be measured in the sense of Mahalanobis distance as follows:

$$d_j = \sqrt{[(X(k)-C_j)\Sigma_j^{-1}][(X(k)-C_j)\Sigma_j^{-1}]^T} \quad (5)$$

where $C_j = [c_{1j}, c_{2j}, ..., c_{Mj}]^T$ and $\Sigma_j = diag[\sigma_{1j}, \sigma_{2j}, ..., \sigma_{Mj}]$. Therefore, the firing strength of the jth fuzzy rule is given by

$$\phi_j = exp[-d_j^2] \quad (6)$$

Layer 4: Nodes in this layer are employed for the purpose of normalization. The output of the normalized nodes is given by

$$\psi_j = \frac{\phi_j}{\sum_{j=1}^{r} \phi_j} \quad (7)$$

Layer 5: Each node in this layer represents an output variable which is the interpolation of multiple dynamic models. For a Multi-Input-Single-Output (MISO) system, its output is given by

$$y(k) = \sum_{j=1}^{r} y_j(k)\psi_j(k) \quad (8)$$

where y is the value of the output variable and y_j is the fuzzy inferred output of the jth fuzzy rule.

The internal feedback is introduced in the consequent part of the proposed fuzzy system in order to have dynamic processing capability. The feedback only exists in the consequent part, different from other fully recurrent fuzzy systems which introduce the feedback from the output layer to the input layer. Therefore, the jth fuzzy rule R_j is described as follows: R_j : $IF\ (x(k)\ is\ a_{1j}\ and\ x(k-1)\ is\ a_{2j}\ ...\ x(k-M+1)\ is\ a_{Mj}),\ THEN\ (y\ is\ y_j(k))$
where $y_j(k) = \sum_{n=1}^{N} w_{j(M+n)} y_j(k-n) + \sum_{m=1}^{M} w_{jm} x(k-m+1) + w_{j0}$ and w_{j0} is the DC value of the jth fuzzy rule. As a matter of fact, the proposed partially recurrent fuzzy system is a nonlinear time-varying IIR filter from the point of view of filter design.

III. Adaptive Algorithm

A. Partition of the Input Space

In the proposed EBFN-based partially recurrent fuzzy system, partitioning the input space is essentially the generation of EBF units. Therefore, it is assumed that the proposed fuzzy system starts with no fuzzy rules. The EBF units will be generated in the sequential training process hierarchically. Therefore, when a new incoming sample $X(k)$ is applied, compute (5) and find

$$d_{min} = min\{d_j(X(k))\} \qquad j = 1, 2, ..., r \quad (9)$$

If $d_{min} > d_{def}$, an EBF unit will be generated standing for the IF part in the premise part and the corresponding THEN is also added into the consequent part. In (9), d_{def} is a predefined parameter.

Geometrically, the cluster is characterized by C_j and Σ_j which represents the centers and variances (widths) respectively. Therefore, when a new EBF neuron is generated, the center and variance are initialized as follows:

$$\begin{aligned} C_j &= X(k) \quad (10) \\ \Sigma_j &= \frac{1}{\alpha \times d_{def}}[f_c(x(k)), f_c(x(k-1)), \\ & ..., f_c(x(k-M+1))] \quad (11) \end{aligned}$$

where α is a constant slightly smaller than 1 and the function $f_c(.)$ is described as follows:

$$\begin{aligned} f_c(x(k-i+1)) &= max(\|x(k-i+1)-c_i^1\|, \\ & \|x(k-i+1)-c_i^2\|) \quad (12) \end{aligned}$$

where c_i^1 and c_i^2 are the centers of neighboring $x(k-i+1)$ in the set $\{c_{i1}, c_{i2}, ..., c_{ir}\}$ in the sense of Euclidean distance.

After the initialization procedure which is described by (10) and (11), the centers and widths of the subspace will be adjusted based on a redefined temporal and spatial potential measurement of the newly incoming data points.

The potential of a cluster center at time k is defined as follows:

$$P_{ij}(k) = \alpha_1 P_{ij}^s(k) + \alpha_2 P_{ij}^t(k) \tag{13}$$

where α_1, α_2 are the predefined constants and $\alpha_1 + \alpha_2 = 1$.

The term $P_{ij}^s(k)$ is a potential measurement from the spatial proximity between the cluster center c_{ij} and other existing centers on the dimension of $x(k-i+1)$, and it is defined as follows:

$$P_{ij}^s(k) = \frac{1}{r-1} \sum_{j'=1, j' \neq j}^{r} exp[-\frac{(c_{ij} - c_{ij'})^2}{\sigma_{ij'}^2}] \qquad r > 1 \tag{14}$$

where $P_{ij}^s(k) = 1$ when there is only one fuzzy rule in the rule base.

The term $P_{ij}^t(k)$ is a potential measurement from the temporal proximity between the cluster center c_{ij} and those data points which fall in the fuzzy set a_{ij} characterized by the center c_{ij} and σ_{ij}. The potential of temporal proximity is defined as follows:

$$P_{ij}^t(k) = \frac{1}{k} \sum_{k'=1}^{k} exp[-\frac{(c_{ij} - x(k'-i+1))^2}{\sigma_{ij}^2}] \tag{15}$$

It is noted that the $P_{ij}^t(k)$ can be rewritten as follows:

$$\begin{aligned} P_{ij}^t(k) &= \frac{1}{k} exp[-\frac{(c_{ij} - x(k-i+1))^2}{\sigma_{ij}^2}] + \\ &\quad \frac{1}{k} \sum_{k'=1}^{k-1} exp[-\frac{(c_{ij} - x(k'-i+1))^2}{\sigma_{ij}^2}] \\ &\approx \frac{1}{k} exp[-\frac{(c_{ij} - x(k-i+1))^2}{\sigma_{ij}^2}] \\ &\quad + P_{ij}^t(k-1) \end{aligned} \tag{16}$$

where $x(k'-i+1) = 0$ if $k'-i+1 \leq 0$.

Equation (16) implies that the potential of temporal proximity $P_{ij}^t(k)$ can be calculated recursively which means there is no need to memorize the past incoming data points. Therefore, potential measurement of the spatial and temporal proximity between the newly applied input data point and the existing cluster centers will be calculated online in order to judge whether the existing centers should be adjusted according to the new information brought by the current input data point.

When a new incoming data point is applied to the fuzzy system at time $k+1$, for the ith input variable, we have

$$L = arg\ max(exp[-\frac{(x((k+1)-i+1) - c_{ij})^2}{\sigma_{ij}^2}]) \tag{17}$$

where $j = 1, 2, ..., r$. Equation (17) determines which cluster the newly input data point will fall in. The cluster will be named as the leading cluster which means it provides the biggest representation degree for the input data point.

The potential measurement of the spatial proximity between the newly incoming data point and the leading cluster center is given by

$$P_x^s(k+1) = \frac{1}{r-1} \sum_{j'=1, j' \neq L}^{r} exp[-\frac{(x((k+1)-i+1) - c_{ij'})^2}{\sigma_{ij'}^2}]$$

The potential measurement of the temporal proximity between the newly incoming data point and the past data points which are lie in the leading cluster is updated as follows:

$$\begin{aligned} P_x^t(k+1) &= \frac{1}{k+1} exp[-\frac{(x((k+1)-i+1) - c_{iL})^2}{\sigma_{iL}^2}] \\ &\quad + P_{ij}^t(k)|_{\ j=L} \end{aligned} \tag{18}$$

The total potential measurement of the new incoming data point of the ith input variable is given by:

$$P_x(k+1) = \alpha_1 P_x^s(k+1) + \alpha_2 P_x^t(k+1) \tag{19}$$

Therefore, we have a similarity measurement which is given by

$$S = \frac{\|P_x(k+1) - P_{ij}(k)\|}{P_{ij}(k)}|_{\ j=L} \tag{20}$$

The index S stands for the similarity between the newly data point which brings new information into the fuzzy system and the leading cluster center which remains the past information of the data partition. The smaller S indicates the higher similarity that the new incoming data point has with respect to the leading cluster center.

In order to exploit the information brought by the newly incoming data point, if

$$S < S^* \tag{21}$$

the leading cluster center should be adjusted where S^* is a predefined parameter.

The leading cluster center will be adjusted as follows:

$$c'_{iL} = c_{iL} + \alpha_c(1-S)[x((k+1)-i+1) - c_{iL}] \tag{22}$$

where α_c is a learning rate.

After the leading cluster center is adjusted, the potential measurement of other existing cluster centers will be updated due to the change of the leading cluster's position. In other words, the potential of other existing cluster centers being updated is as follows:

$$P_{ij}^s(k+1) = \frac{1}{r-1} \sum_{j'=1, j' \neq j}^{r} exp[-\frac{(c_{ij} - c_{ij'})^2}{\sigma_{ij'}^2}] \tag{23}$$

where $j = 1, 2, ..., r$ and $j \neq L$.

B. Weighted Backward Adaption of Cluster Shape

In order to tune the entire fuzzy system optimally and comply with the purpose of adaptive noise cancellation, the cost function of the proposed fuzzy system is defined as follows:

$$E = \frac{1}{2}\sum_{k=1}^{n}[d(k) - y(k)]^2 = \frac{1}{2}\sum_{k=1}^{n}[e(k)]^2 \quad (24)$$

where n is the number of input data points and $e(k)$ is denoted as the global error at time k of the proposed fuzzy system which is given by

$$e(k) = d(k) - y(k) \quad (25)$$

As a matter of fact, each EBF unit contributes differently in each time step due to the diversity of firing strength. Intuitively, the adjustments of centers and widths are related to the contribution of EBF units. That is, the update terms are weighted by the instantaneous contribution of each EBF unit (i.e. firing strength). Therefore, the synaptic weights of EBF units (i.e. the centers and widths of the subspace), which determine the IF part of fuzzy rules, will be updated as follows:

$$\begin{bmatrix} C_j(k+1) \\ \Sigma_j(k+1) \end{bmatrix} = \begin{bmatrix} C_j(k) \\ \Sigma_j(k) \end{bmatrix} + \frac{1}{2}\vec{\mu}_\psi \begin{bmatrix} \mu_C \\ \mu_\Sigma \end{bmatrix} \begin{bmatrix} \triangle C_j(k) \\ \triangle \Sigma_j(k) \end{bmatrix} \quad (26)$$

where $\vec{\mu}_\psi = diag[\frac{\mu_{1j}}{\psi_j}, \frac{\mu_{2j}}{\psi_j}, ..., \frac{\mu_{Mj}}{\psi_j}]$.

The update term $\triangle C_j$ is given by

$$\begin{aligned} \triangle C_j &= -\frac{\partial E}{\partial C_j} = -\frac{\partial E}{\partial y}\frac{\partial y}{\partial \phi_j}\frac{\partial \phi_j}{\partial C_j} \quad (27) \\ &= 2\psi_j \triangle_j \quad (28) \end{aligned}$$

where $\triangle_j$ is defined as follows

$$\triangle_j = \frac{1}{\Sigma_j} e e_j d_j \quad (29)$$

Therefore, the centers of the clusters associated with the jth fuzzy rule could be rewritten as follows:

$$C_j(k+1) = C_j(k) + \mu_C \triangle_j \vec{\mu} \quad (30)$$

where $\vec{\mu} = diag[\mu_{1j}, \mu_{2j}, ..., \mu_{Mj}]$.

The updates for the widths of clusters associated with the jth fuzzy rule, $\triangle\Sigma_j$, are given by

$$\triangle\Sigma_j = -\frac{\partial E}{\partial \Sigma_j} = -\frac{\partial E}{\partial y}\frac{\partial y}{\partial \phi_j}\frac{\partial \phi_j}{\partial \Sigma_j} \quad (31)$$

It should be noted that only the term $\frac{\partial \phi_j}{\partial \Sigma_j}$ will be re-computed compared with the update term of $\triangle C_j$. Therefore, we have

$$\frac{\partial \phi_j}{\partial \Sigma_j} = \frac{\partial \phi_j}{\partial d_j}\frac{\partial d_j}{\partial \Sigma_j} = \frac{2}{\Sigma_j} d_j^2 \phi_j \quad (32)$$

Similarly, the update term of the widths can be written as follows:

$$\triangle\Sigma_j = 2\psi_j \triangle_j d_j \quad (33)$$

Therefore, the widths will be updated as follows:

$$\Sigma_j(k+1) = \Sigma_j(k) + \mu_\Sigma \triangle_j d_j \vec{\mu} \quad (34)$$

The update pair of centers and widths of (26), can be rewritten as follows:

$$\begin{bmatrix} C_j(k+1) \\ \Sigma_j(k+1) \end{bmatrix} = \begin{bmatrix} C_j(k) \\ \Sigma_j(k) \end{bmatrix} + \triangle_j \begin{bmatrix} \mu_C \\ \mu_\Sigma \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & d_j \end{bmatrix} \vec{\mu} \quad (35)$$

Equation (35) implies the centers and widths of the subspaces are "weighted" adjusted according to the current membership functions' value μ_{ij} respectively. The cluster which has bigger membership function value will learn more from the backward process.

C. Parameter Determination of the Consequent Part

Free parameters in the THEN part of the jth fuzzy rule are given by

$$W_j = [w_{j0}, w_{j1}, ..., w_{jM}, w_{j(M+1)}, ..., w_{j(M+N)}]^T \quad (36)$$

The individual (local) output of the jth fuzzy rule can be written as follows:

$$y_j(k) = W_j^T B_j \quad (37)$$

where $B_j = [1, x(k), ..., x(k-M+1), y_j(k-1), ..., y_j(k-N)]^T$. Substituting (37) into (8), we have the following global inferred output

$$y(k) = \sum_{j=1}^{r} W_j^T B_j \psi_j \quad (38)$$

It should be highlighted that ψ_j is a scalar so that we have $q_j = B_j\psi_j$. The inferred output can be expressed as

$$y(k) = W^T Q \quad (39)$$

where $W = [W_1^T, W_2^T, ..., W_r^T]^T$ and $Q = [q_1^T, q_2^T, ..., q_r^T]^T$.

From (8) and (25), we have

$$e(k) = d(k) - \sum_{j=1}^{r} y_j(k)\psi_j(k) \quad (40)$$

It should be noted that $\psi_j(k)$ is a normalized scaler and has the following property $\sum_{j=1}^{r}\psi_j(k) = 1$. Therefore, (40) can be rewritten as follows:

$$\begin{aligned} e(k) &= \sum_{j=1}^{r}[d(k) - y_j(k)]\psi_j(k) \quad (41) \\ &= \sum_{j=1}^{r} e_j(k)\psi_j(k) \quad (42) \end{aligned}$$

During the online training, new parameters will be optimized in order to capture the dynamics brought by the newly generated fuzzy rule. An improved recursive algorithm is proposed for adapting the parameters quickly. Suppose there are j fuzzy rules in the rule base at time $k-1$, the free parameters of the consequent part are denoted as $W(k-1) = [W_1(k-1) \;\; W_2(k-1) \;\; ... \;\; W_j(k-1)]$. When a new fuzzy rule (the $(j+1)$th fuzzy rule) is generated at time k, the corresponding parameter in the consequent part will be given by

$$W(k) = [W(k-1) \quad W^o_{j+1}(k)] \tag{43}$$

where $W^o_{j+1}(k)$ is initialized orthogonally by the Gram-Schmidt transformation as follows:

$$\begin{aligned} W^o_1(k-1) &= W_1(k-1) \\ o_{i'j'} &= \frac{W^{oT}_{i'}(k-1)W_{j'}(k-1)}{W^{oT}_{i'}(k-1)W^o_{i'}(k-1)} \; 1 \le i' < j' \\ W^o_{j'}(k-1) &= W_{j'}(k-1) - \sum_{i'=1}^{j'-1} o_{i'j'} W^o_{i'}(k-1) \end{aligned}$$

where $j' = 2, 3, ..., j$. In other words, for the newly generated fuzzy rule, we have

$$W^o_{j+1}(k) = W_{j+1}(k) - \sum_{i'=1}^{j} o_{i'(j+1)} W^o_{i'}(k-1) \tag{44}$$

where $W_{j+1}(k) = \delta_W I$ and δ_W is a small positive constant.

At the same time, we introduce a learning matrix

$$\begin{aligned} L_W(k) = \; & diag[1 + \mu_W(l_1)^{k-k_1} I_1, ..., 1 + \mu_W(l_j)^{k-k_j} I_1, \\ & ..., 1 + \mu_W(l_{j+1})^{k-k_{j+1}} I_1] \end{aligned} \tag{45}$$

where $\mu_W > 1$ is the initial learning-control rate, $0 < l_1 < ... < l_j < l_{j+1} < 1$ are the various learning rates of the individual coefficient vectors $\{W_1, ..., W_j, W_{j+1}\}$ in the consequent part, $\{k_1, ..., k_j, k_{j+1}\}$ are the time points when the corresponding fuzzy rules are generated and $I_1 \in \Re^{(M+N+1)\times(M+N+1)}$ is a unit matrix.

At time k, the inverse correlation matrix and the gain vector are calculated and updated as follows:

$$K(k) = \frac{P(k-1)Q(k)}{1 + Q^T(k)P(k-1)Q(k)} \tag{46}$$

$$P(k) = P(k-1) - K(k)Q(k)P(k-1) \tag{47}$$

Complying with the principle of using the individual errors, at time k, when the $(j+1)$th fuzzy rule is generated, we have

$$\xi(k) = d(k) - \sum_{j'=1}^{j+1} y_{j'}(k)\psi_{j'}(k) = \sum_{j'=1}^{j+1} e_{j'}(k)\psi_{j'}(k) \tag{48}$$

where $\xi(k)$ is the a priori estimation error because it is the estimation error between the desired response $d(k)$ and the old least-squares estimate of the free parameters in the consequent part that was made at time $k-1$. It should be noted that the part of W_{j+1} of $W(k)$ is initialized and not learned from the current training data pair. Rewrite (48), we have

$$\xi(k) = \vec{\xi}(k)\vec{\psi}^T(k) \tag{49}$$

where $\vec{\xi}(k) = [e_1(k), e_2(k), ..., e_j(k), e_{j+1}(k)]$ and $\vec{\psi}(k) = [\psi_1(k), \psi_2(k), ..., \psi_j(k), \psi_{j+1}(k)]$.

The free parameters $W(k)$ of the consequent part will be updated as follows:

$$W'(k) = W(k) + L_W(k)K(k)\vec{\xi}(k) \tag{50}$$

The a posteriori estimation error (global error) is essentially given by

$$e(k) = d(k) - W'^T(k)\vec{\psi}(k)Q(k) \tag{51}$$

IV. Simulation Studies and Results

The objective of adopting the partially recurrent fuzzy system as an adaptive noise canceller is to minimize the error measure $E[e^2(k)]$ by capturing the dynamics underlying the data pairs. In the following simulation study, the channel that the noise source goes through is nonlinear and dynamic. The performance of using the proposed fuzzy system as an adaptive noise canceller is validated on the basis of the MSE criterion and a series of tests based on the correlation between the input $x(k)$, and the residual $e(k)$ [8]. In [9], it was shown that for an efficient noise canceller, the following model validity test should be satisfied:

$$\Psi_{xe}(t) = E[x(k)e(k+t)] = 0 \tag{52}$$

$$\Psi_{x^2e}(t) = E[(x^2(k) - E[x^2(k)])e(k+t)] = 0 \tag{53}$$

$$\Psi_{x^2e^2}(t) = E[(x^2(k) - E[x^2(k)])e^2(k+t)] = 0 \tag{54}$$

In practice, the model will be regarded as adequate if all the tests by (52), (53) and (54) fall within the 95% confidence bands at approximately $\pm 1.96/\sqrt{n}$, where n is the number of samples.

In the simulation example, the noise source $x(k)$ passes through a nonlinear dynamic channel producing the additive noise $n(k)$ which interferes with the information signal. The passage's dynamics is simulated by a second-order nonlinear auto-regressive model with exogenous inputs (NARX) as follows:

$n(k) = 0.25n(k-1) + 0.1n(k-2) + 0.5x(k-1) + 0.1x(k-2) - 0.2x(k-3) + 0.1x^2(k-2) + 0.08x(k-2)n(k-1)$ where $x(k)$ is a uniformly distributed white noise source varying in the range of $[-2, 2]$.

The information signal $s(k)$ is a saw-tooth signal of unit magnitude, 50 samples per period. It is distorted by the additive noise $n(k)$ so that the measurable part is essentially $d(k)$. The training data sets consist of 12,000 pairs $[x(k), d(k)]$. The noise source $x(k)$ will be applied to the partially recurrent fuzzy system as the input signal. With the Time Delay Line (TDL) in the input layer, the delayed counterparts of the noise source forms an input vector $[x(k-1), x(k-2), x(k-3)]$ according to the passage's dynamics. Online recovered information signal and online reproduction error (both in the last 500 samples) are shown in Fig. 2 respectively. The saw-tooth information signal is recovered in a qualified waveform.

Another 1000 samples are generated for the purpose of testing and the testing result will be evaluated by the model validity test described by (52), (53) and (54).

Fig. 3 shows that the correlation falls within the 95% confidence bands which means the proposed fuzzy system cancel the noise successfully by capturing the nonlinear dynamics of the passage due to the IIR-based consequent part.

Some numerical indexes for comparative analysis between the proposed fuzzy system with other noise cancellation filters, including IIR filter, Adaptive-Network-Based Fuzzy Inference System (ANFIS) of [10], Dynamic-Fuzzy Neural Constrained Optimization Method (D-FUNCOM) of [11], are given in Table I.

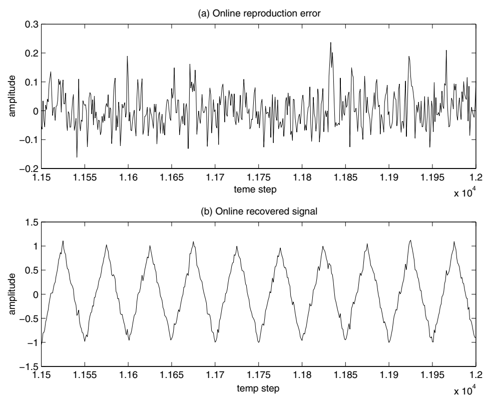

Fig. 2. (a). Online reproduction error. (b) Online recovered signal.

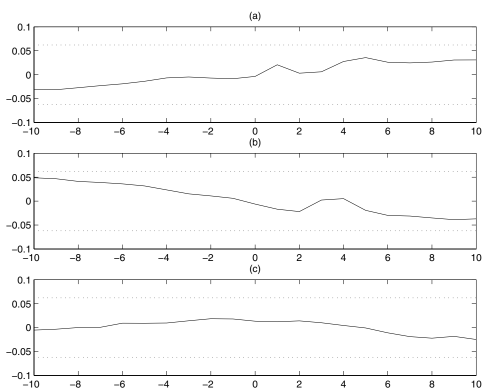

Fig. 3. Model validity test. The dotted lines correspond to the 95% confidence bands. (a) $\Psi_{xe}(t)$. (b)$\Psi_{x^2e}(t)$. (c)$\Psi_{x^2e^2}(t)$.

TABLE I

Comparison with other noise cancellation filters

Approaches	MSE_{trn}	MSE_{tst}	No. of parameters
ANFIS	0.0980	0.1151	135
IIR	0.0157	0.0169	87
D-FUNCOM	0.0136	0.0131	135
Our approach	0.0018	0.0055	48

V. Conclusions

In this paper, a partially recurrent fuzzy system is proposed to work as an adaptive noise canceller. It has the following functions: (1) Partially recurrent structure. There is no feedback from the output layer to the input layer; only internal feedback at the consequent part of the fuzzy system is needed. As a consequence, the dimension of the input layer is reduced so that the network size is parsimonious. (2) The corresponding adaptive algorithm is efficient. The input space is partitioned based on a potential measurement from temporal-spatial proximity. The number of fuzzy rules is determined during the training process. The free parameters in the consequent part are determined in the context of Least Square Error (LSE) by an enhanced recursive algorithm. Therefore, long-term dependencies of the input/output data could be learned and latched correctly without using the gradient descent algorithm. The entire system dynamics are implemented by the individual dynamics regression models in the consequent part. Simulation studies shows that the proposed fuzzy system effectively deal with adaptive noise cancellation for a nonlinear dynamic channel.

References

[1] F.-J. Lin, R.-J. Wai, K.-K. Shyu, and T.-M. Liu, "Recurrent fuzzy neural network control for piezoelectric ceramic linear ultrasonic motor drive," IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, vol. 48, pp. 900–913, 2001.

[2] W. Xie and A. Rad, "Fuzzy on-line identification of siso nonlinear systems," Fuzzy Sets and Systems, vol. 107, pp. 323–334, 1999.

[3] T. Lin, B. Horne, P. Tiňo, and C. Giles, "Learning long-term dependencies in narx recurrent neural networks," IEEE Trans. on Neural Networks, vol. 7, pp. 1329–1338, 1996.

[4] L. Wang and R. Langari, "Building sugeno-type models using fuzzy discretization and orthogonal parameter estimation techniques," IEEE Trans. on Fuzzy Systems, vol. 3, pp. 454–458, 1995.

[5] R. Dave, "Robust fuzzy clustering algorithms," in Fuzzy Systems, 1993., Second IEEE International Conference on, vol. 2, (San Francisco, CA, USA), pp. 1281–1286, 1993.

[6] M. Sugeno and G. T. Kang, "Structure identification of fuzzy model," Fuzzy Sets and Systems, vol. 28, pp. 15–33, 1988.

[7] S. Haykin, Adaptive filter theory. Englewood Cliffs, N.J, Prentice-Hall, 1986.

[8] I. J. Leontaritis and S. A. Billings, "Model selection and validation methods for nonlinear systems," Int. J. Contr., vol. 45, pp. 311–341, 1987.

[9] S. A. Billings and F. A. Alturki, "Performance menitoring in nonlinear adaptive noise cancellation," J. Sound Vibr., vol. 157, pp. 161–175, 1992.

[10] J.-S. R. Jang, "Anfis: Adaptive-network-based fuzzy inference system," IEEE Trans. on Systems, Man, and Cybernetics, vol. 23, pp. 665–685, 1993.

[11] P. A.Mastorocostas and J. B.Theocharis, "A recurrent fuzzy-neural model for dynamic system identifation," IEEE Trans. on Systems, Man, and Cybernetics, vol. 32, pp. 176–190, 2002.