Proceedings of the 5th WSEAS International Conference on Signal Processing, Istanbul, Turkey, May 27-29, 2006 (pp93-98)

Adaptive Noise Cancellation with Computational-Intelligence-based Approach

* CHUNSHIEN LI
Department of Computer Science and Information Engineering
National University of Tainan
33, Sec. 2, Shu-Lin St., Tainan, 70005, Taiwan, R.O.C.

** KUO-HSIANG CHENG
Department of Electrical Engineering
Chang Gung University
259, Wen-Hwa 1st Road, Kwei-Shan, Tao-Yuan, 333, Taiwan, R.O.C.

Abstract: A new intelligent noise filtering approach using Computational Intelligence (CI) is proposed for the problem of adaptive noise cancellation (ANC). Since the traditional linear filtering may not be good enough to handle with the noise complexity and time-varying statistic property, a self-constructing neuro-fuzzy system (SCNFS) is used as an adaptive filter to deal with the nonlinearity of noise. A hybrid machine learning algorithm with the methods of both random optimization algorithm (RO) and least square estimation (LSE) is introduced to enable the SCNFS with learning capability. The learning capability includes both the parameter learning and the structure learning. In the parameter learning phase, the premises and the consequents of the SCNFS are updated by RO and LSE, respectively. In the SCNFS structure learning, the system structure can be generated or rearranged using the proposed mechanism with rule-splitting and/or rule-expanding. To demonstrate the feasibility and the capability of the proposed approach, an example of adaptive speech noise cancellation is illustrated. With the experimental results, the SCNFS shows excellent filtering performance for noise cancellation.

Key-Words: - computational intelligence, neuro-fuzzy, learning, random optimization, least square estimation, adaptive noise cancellation.

1. Introduction

In the past decades, the increasing noise influences on engineering applications have encouraged the development of noise canceling. In principle, the problem of adaptive noise filtering is to extract a desired signal from its corrupted version [1]. Generally, noise is usually highly nonlinear and correlated with uncertain factors. Traditional linear filters like FIR or IIR filters may not be good enough to handle with the complexity of noise nonlinearity. Recently, adaptive filtering may provide better solution to noise filtering problem. Adaptive systems as adaptive filters possess the characteristics of self-adjustment, and/or intelligent strategy to practical applications with limited priori information. Recently, engineers and scientists have focused their attention to model-free approaches and the intelligence-based theories such as fuzzy logic [2] and neural network [3]. Since fuzzy inference systems and neural networks have been shown being universal approximators [4, 5], the integration of neuro-fuzzy system (NFS) has widely been recognized a complementary model that allows for low-level learning and computational power and high-level human thinking and knowledge representation. The objective of soft computing based approach for adaptive noise filtering is to attenuate or eliminate noise from the corrupted signal. Generally, establishing an NFS includes the structure complexity and the parameter identification. For structure complexity, the partition of the input-output space and the identification of appropriate rules to achieve the desired performance have received much attention. Several approaches [6, 7, 8-10] have been addressed to achieve compact and effective system structure. However in these approaches the balance between the number of hidden neurons and the system performance is concerned before training. The grid-type partitioning is often suffered from the problem commonly referred to as the “curse of dimensionality”, in which the rule number will become exponentially large if the input dimension increases. In order to avoid the problems of over-parameterization and to ensure structure concision, this depends heavily on experience and a tedious trail-and-error. Thus, it is usually needed to have a “smarter” mechanism for system learning. A new intelligent filtering approach with self constructing neuro-fuzzy system (SCNFS) is proposed for the problem of adaptive noise cancellation (ANC). The goal of noise cancellation is to extract the desired signal from its noise-corrupted version by using the proposed SCNFS as a computational intelligence (CI) based adaptive filter. In the paper, a hybrid learning algorithm with the random optimization algorithm (RO) and the least square estimation (LSE) is used for the NFS to perform structure learning and parameter learning simultaneously. In the parameter learning phase, the RO [11, 12] is combined with the LSE to train the parameters of the SCNFS, where the premises and the consequents of the SCNFS are updated by RO and LSE, respectively. The structure learning is composed of two phases, which are the rule-splitting phase and rule-expanding phase. In the structure learning, appropriate rules can be generated and/or rearranged with the proposed mechanism for rule significance detection. With the proposed hybrid learning, a compact-sized and well-parameterized NFS can be established with good system performance. The proposed SCNFS features the following salient properties: 1) online rule construction, 2) intuitive and derivative-free learning, 3) easy to program, and 4) fast convergence and robustness. The problem of the adaptive noise cancellation is given in section 2. In section 3, the structure of neuro-fuzzy system is presented. In section 4, the RO-LSE hybrid learning algorithm is discussed. In section 5, the self construction scheme for structure learning is discussed. The proposed SCNFS filtering to the noise canceling application is illustrated in section 6. Finally, the paper is concluded.

2. Adaptive Noise Cancellation

The noise-canceling diagram is shown in Fig. 1 [1]. An input signal contaminated by the noise $y(t)$ is transmitted to the receiver. With the diagram, the received signal $m_{rs}(t)$ can be described as follows.

$$m_{rs}(t) = x(t) + y(t) \tag{1}$$
$$= x(t) + f_{channel}(n(t), n(t-1), ..., n(t-k))$$

where $f_{channel}(\cdot)$ represents the noise passage channel, $n(t)$ the noise source, $x(t)$ the clear signal. The basic principle of the adaptive noise cancellation is to estimate the desired signal from the corrupted version. The following assumptions are given.

1) x, y and n, are zero-mean process. (statistical stationary and zero means).
2) x are uncorrelated with n and y.
3) n and y are correlated by the function $f_{channel}(\cdot)$

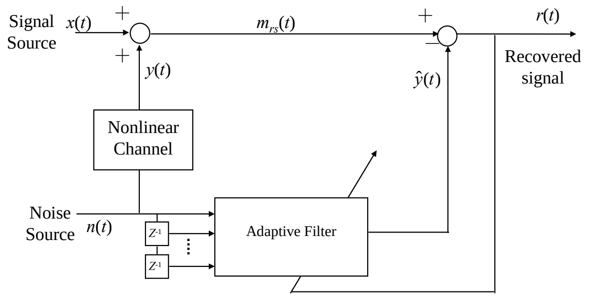

Fig. 1. Schematic diagram of noise-canceling.

The error signal $r(t)$, served as the recovered signal in the adaptive process, is given as follows.

$$r(t) = x(t) + y(t) - \hat{y}(t) \tag{2}$$

where $\hat{y}(t)$ is the estimated output from the filter. By squaring, the expectation is applied on Eq.(2), given as follows.

$$\begin{aligned} \mathrm{E}[r(t)^2] &= \mathrm{E}[x(t)^2] + \mathrm{E}[(y(t) - \hat{y}(t))^2] \\ &\quad + 2\mathrm{E}[x(t)(y(t) - \hat{y}(t))] \end{aligned} \tag{3}$$

where $\mathrm{E}[.]$ represents the expectation operator. Based on the second assumption, the third item of Eq.(3) can be removed.

$$\mathbf{E}[r(t)^2] = \mathbf{E}[x(t)^2] + \mathbf{E}[(y(t) - \hat{y}(t))^2] \tag{4}$$

In the nature of signal power, $\mathbf{E}[x(t)^2]$ remains unaffected while the adaptive filter is used to minimize the power of the recovered signal. In other words, the power difference is minimized between the contaminating noise signal and the filter output, given as follows.

$$\mathbf{E}_{\min}[r(t)^2] = \mathbf{E}[x(t)^2] + \mathbf{E}[(y(t) - \hat{y}(t))^2] \tag{5}$$

From Eq.(5), the conclusion is deducted that the power minimization of the signal $r(t)$ is equivalent to minimizing the power of noise. In the paper, the neuro-fuzzy filter is trained to remove the noise in the adaptive noise cancelling process.

3. SCNFS-Based Adaptive Filter

In this paper, the SCNFS-based adaptive filter receives the input signals from noise source to perform the unknown channel identification. The structure of SCNFS is given as follows.

3.1 Fuzzy inference process of SCNFS

For simplicity, an MISO (multi-input-single-output) fuzzy inference system with the crisp input vector $\mathbf{H}(t)$ measured at time t, composed of M input variables $h_i(t)$, $i = 1,2,\ldots,M$, is considered. The universe of discourse of each input dimension is defined with corresponding linguistic variables denoted as x_i. In this paper, a flexible cluster-based partition method is used to avoid the unnecessary rules, where the partial-connected structure is exploited. Assume that the vector of the linguistic term $\mathbf{V}_i$ belong to the i-th input variable is given as follows.

$$\mathrm{V}_i = \begin{bmatrix} v_{i,1} & v_{i,2} & \cdots & v_{i,R} \end{bmatrix}^T \tag{6}$$

where R is the amount of rules and $v_{i,k}$ the k-th linguistic term of the i-th linguistic input variable. In the proposed SCNFS, the well-known fuzzy T-S model is exploited. Based on the proposed fuzzy inference process, the k-th rule description can be given as follows.

$$\begin{aligned} &\textit{Rule } k: \textit{IF } \mathbf{V}_1 \textit{ is } s_{1,k}(h_1(t)) \textit{ and } \mathbf{V}_2 \textit{ is } s_{2,k}(h_2(t))\ldots \\ &\textit{and } \mathbf{V}_M \textit{ is } s_{M,k}(h_M(t)) \\ &\textit{THEN} \qquad y^k(t) = \mathbf{H_a}(t)\vec{\mathbf{a}}^k \end{aligned} \tag{7}$$

where $s_{i,k}(h_i(t))$ is defined as the k-th fuzzy set for the i-th linguistic variable $\vec{a}^k = \begin{bmatrix} a_0^k & a_1^k & \cdots & a_M^k \end{bmatrix}$ the parameter vector of the consequent of the k-th rule, $\mathbf{H}_a = \begin{bmatrix} 1 & h_1(t) & \cdots & h_M(t) \end{bmatrix}$, and y^k the output of the k-th rule. Based on (7), the inferred result $Y(t)$ at time t is expressed as

$$Y(t) = \frac{\sum_{k=1}^{K} \beta^k(t) \times y^k(t)}{\sum_{k=1}^{K} \beta^k(t)} = \sum_{k=1}^{R} \bar{\beta}^k(t)\mathbf{H}_a\vec{\mathbf{a}}^i \tag{8}$$

where $\bar{\beta}^k$ is the normalized firing strength of the k-th rule.

3.2 Network structure of SCNFS

A five-layer neuro-fuzzy structure is shown in Fig. 2, which is composed of the layers of input (Layer 1), membership (Layer 2), rule (Layer 3), normalization (Layer 4), and output (Layer 5). Layer 1 accepts the input signals. Layer 2 is used to calculate the Gaussian membership values. The nodes of layer 3 represent the fuzzy rules. In layer 4, the normalization process is executed. The network before layer 4 represents the premises of the rules, and that after layer 4 represents the consequents of the rules. Layer 5 is the output layer. Based on the constitution of the T-S fuzzy model, the link weight is given as follows.

$$w_{ko}^5 = \mathbf{H}_a\vec{\mathbf{a}}^k \tag{9}$$

for $k = 1,2,\ldots,R$. The output can be represented as the polynomial functions of the input signals. The premise sets of the SCNFS are collected as follows.

$$\mathbf{m} = \begin{bmatrix} \mathbf{m}_1 & \mathbf{m}_2 & \cdots & \mathbf{m}_R \end{bmatrix} \tag{10}$$

$$\boldsymbol{\sigma} = \begin{bmatrix} \boldsymbol{\sigma}_1 & \boldsymbol{\sigma}_2 & \cdots & \boldsymbol{\sigma}_R \end{bmatrix} \tag{11}$$

where

$$\mathbf{m}_k = \begin{bmatrix} \mathbf{m}_{1k} & \mathbf{m}_{2k} & \cdots & \mathbf{m}_{Mk} \end{bmatrix}^T \tag{12}$$

$$\boldsymbol{\sigma}_k = \begin{bmatrix} \boldsymbol{\sigma}_{1k} & \boldsymbol{\sigma}_{2k} & \cdots & \boldsymbol{\sigma}_{Mk} \end{bmatrix}^T \tag{13}$$

where $(\cdot)^T$ indicates the transpose of $(\cdot)$, m_{ik} and σ_{ik}^2 the mean and the variance of the Gaussian functions associated with the k-th node for the i-th input dimension. The premise parameters set $\mathbf{W}$ is given as follows.

$$\mathbf{W} = \begin{bmatrix} \mathbf{m} & \boldsymbol{\sigma} \end{bmatrix} \tag{14}$$

The consequent parameters are collected to form the matrix .

$$\mathbf{A} = \begin{bmatrix} [a_0^1 & a_1^1 & \cdots & a_M^1]^T \\ [a_0^2 & a_1^2 & \cdots & a_M^2]^T \\ \vdots & \vdots & \ddots & \vdots \\ [a_0^R & a_1^R & \cdots & a_M^R]^T \end{bmatrix} = \begin{bmatrix} \vec{\mathbf{a}}^1 \\ \vec{\mathbf{a}}^2 \\ \vdots \\ \vec{\mathbf{a}}^R \end{bmatrix} \tag{15}$$

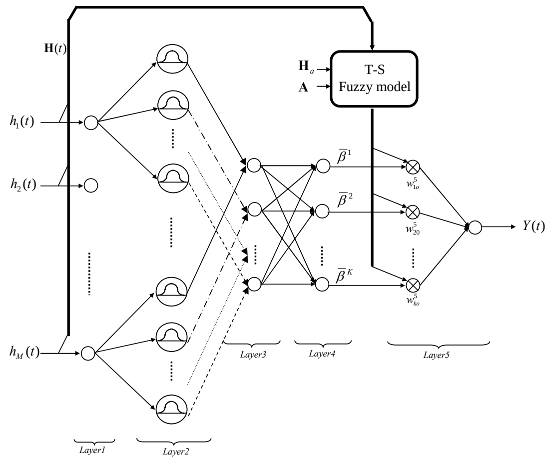

Fig. 2. Neuro-fuzzy Structure of SCNFS.

4. Hybrid Learning Algorithm

To train the proposed SCNFS, the well-known random optimization (RO) is used together with the least square estimation (LSE) for fast convergence of system learning. Without the need for the derivative information of an objective function, the RO algorithm features derivative-free and intuitive exploration in the input space. Moreover, the RO method excels not only at its simplicity and convenience, but also ensures to converge to the global minimum with probability one in a compact set [11]. In this paper, a new version of RO-based learning is depicted as follows. The problem of system identification can be stated as finding the optimal solution and A that minimizes the cost function. In this paper, the cost function is defined

$$E_{cf}(\mathbf{m},\boldsymbol{\sigma},\mathbf{A}) = \left(\frac{1}{Q}\sum_{N=1}^{Q}(D(N) - f_{HDNFS}(\mathbf{W},\mathbf{A},\mathbf{H}(N)))^2\right)^{\frac{1}{2}} \tag{16}$$

Eq. (16) is the root mean square error (RMSE) between the SCNFS output and the desired output from N=1 to N= Q . The flow chart of hybrid RO-LSE learning is illustrated in Fig. 3.

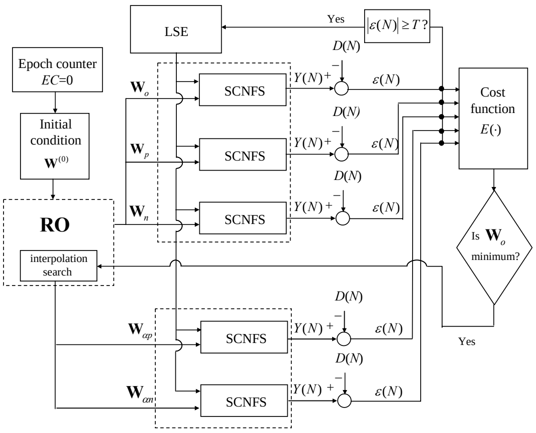

Fig. 3. RO-LSE training.

With the integration of RO-LSE, each candidate point generated by RO is viewed as a potential premise parameter solution. Based on Eq. (8), the relationship between the input vector $\mathbf{H}(N)$ and the desired output can be given as follows.

$$D(N) = \sum_{k=1}^{R} \bar{\beta}^k(N) \times (a_0^k + a_1^k h_1(N) + \ldots + a_M^k h_M(N)) + \varepsilon(N) \tag{17}$$

where $\varepsilon(N)$ is the identification error. Let

$$\mathbf{G}^k(N) = \bar{\beta}^k(N)[1 \quad h_1(N) \quad h_2(N) \quad \ldots \quad h_M(N)] \tag{18}$$
$$= \bar{\beta}^k(N)\mathbf{H}_a(N)$$

$$\mathbf{G}(N) = \left[\mathbf{G}^1(N) \quad \mathbf{G}^2(N) \quad \ldots \quad \mathbf{G}^K(N)\right] \tag{19}$$

Eq. (17) can be represented as follows.

$$D(N) = \mathbf{G}(N)\mathbf{A} + \varepsilon(N) = \sum_{k=1}^{K} \mathbf{G}^k(N)\bar{a}^k + \varepsilon(N) \tag{20}$$

Assuming there are Q training data pairs to be identified as follows.

$$\overbrace{\begin{bmatrix} D(1) \\ D(2) \\ \vdots \\ D(Q) \end{bmatrix}}^{\mathbf{D}} = \overbrace{\begin{bmatrix} \mathbf{G}^1(1) & \mathbf{G}^2(1) & \cdots & \mathbf{G}^K(1) \\ \mathbf{G}^1(2) & \mathbf{G}^2(2) & \cdots & \mathbf{G}^K(2) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{G}^1(Q) & \mathbf{G}^2(Q) & \cdots & \mathbf{G}^K(Q) \end{bmatrix}}^{\bar{\mathbf{B}}} \overbrace{\begin{bmatrix} (\bar{\mathbf{a}}^1) \\ (\bar{\mathbf{a}}^2) \\ \vdots \\ (\bar{\mathbf{a}}^K) \end{bmatrix}}^{\mathbf{A}} + \overbrace{\begin{bmatrix} \varepsilon(1) \\ \varepsilon(2) \\ \vdots \\ \varepsilon(Q) \end{bmatrix}}^{\zeta} \tag{21}$$

In the proposed parameter learning, the LSE can be implemented in recursive way, called recursive LSE (RLSE), by which the update can be implemented with individual training pattern. The algorithm given below is called weighted recursive LSE (WRLSE), which provides with the short-term trace ability to identify time-varying phenomena.

$$\mathbf{P}(N) = \frac{1}{\lambda}(\mathbf{P}(N-1) - \frac{\mathbf{P}(N-1)\mathbf{G}(N)\mathbf{G}(N)^T\mathbf{P}(N-1)}{\lambda + \mathbf{G}(N)^T\mathbf{P}(N-1)\mathbf{G}(N)}) \tag{22}$$

$$\tilde{\mathbf{A}}(N) = \tilde{\mathbf{A}}(N-1) + \mathbf{P}(N)\mathbf{G}(N)(D(N) - \mathbf{G}(N)^T\tilde{\mathbf{A}}(N-1)) \tag{23}$$

where $\mathbf{P}(0) = \alpha I$ is given with a large value α , $\tilde{\mathbf{A}}(0)$ can be initially set to zeros, λ a scalar between 0 and 1. The λ known as the forgetting factor is used to give the significance of importance for the preceding training data. This method of WRLSE can be capable of approximating time-varying system, although the fluctuation caused by noise and disturbance can be potential defectiveness. The value of λ should be problem dependent and usually close to unity. In this paper, a criterion for parameter update is given for the purpose of computational efficiency. With the observation $\{H(N), D(N)\}$, N=1,2,…,Q , the error term $\varepsilon(N)$ given in Eq. (20) is compared to a pre-given threshold T. If $|\varepsilon(N)| \leq T$, the parameter update is omitted and $\tilde{\mathbf{A}}(N) = \tilde{\mathbf{A}}(N-1)$. Otherwise, the parameter is updated with Eqs. (22) and (23). With the criterion the unnecessary computation can be avoided if the current system performance is acceptable.

5. Structure Learning Process

5.1 rule-splitting constitution for SCNFS

In the rule constitution of SCNFS, the mathematical description of the existing rules can be expressed as an M-dimensional cluster, given as follows.

$$\Phi(\mathbf{H}(N))^k = \beta^k(N) = \wedge\{[\exp(-(\frac{h_i(N) - m_{ik}}{\sigma_{ik}})^2)]|_{i=1,2,\ldots,M}\} \tag{24}$$

In Eq. (24), each cluster is regarded as a fuzzy rule and can be viewed as a multi-dimensional fuzzy membership function. In the proposed rule-splitting phase, the rule which has been intensively activated should be split to give advanced I/O interpretation. The normalized values $\bar{\bar{\beta}}^k$ of $\bar{\beta}^k(N)$, $N = 1, 2, \cdots Q$, are calculated.

$$\bar{\bar{\beta}}^k = \frac{\sum_{N=1}^{Q} \bar{\beta}^k(N)}{\sum_{k=1}^{R} \sum_{N=1}^{Q} \bar{\beta}^k(N)} \tag{25}$$

For each rule of SCNFS, the value of $\overline{\overline{\beta}}^k$ is calculated after each parameter training epoch. If $\overline{\overline{\beta}}^k$ is larger than a specific constant $1/\{R(N)+\gamma\}$, then the splitting index of the k-th rule, I_{sp}^k ($I_{sp}^k(0)=1$), will be increased.

$$I_{sp}^k = I_{sp}^k \times \psi \tag{26}$$

where $\psi > 1$. The criterion of splitting a new rule is given as follows.

$$\Delta_{sp}^k = \frac{\{\exp(I_{sp}^k - 1)\} - \{\exp(I_{sp}^k - 1)\}}{\{\exp(I_{sp}^k - 1)\} + \{\exp(I_{sp}^k - 1)\}} \tag{27}$$

If $\Delta_{sp}^k \geq U_{th}$, where $U_{th} \in (0,1)$ a pre-given threshold, then a new cluster should be generated by splitting from the k-th rule. The mean and the variance of the original and the new membership function are given as follows.

$$m_{i(k)} = m_{i(k)} \times \delta_1 \,|_{i=1,2,\dots,M} \tag{28}$$

$$m_{i(new)} = m_{i(k)} \times (1-\delta_1) \,|_{i=1,2,\dots,M} \tag{29}$$

$$\sigma_{i(new)} = \sigma_{i(k)} \,|_{i=1,2,\dots,M} \tag{30}$$

where $0 \leq \delta_1 < 1$ is a pre-given constant.

5.2 Rule-expansion for SCNFS

Although the rule-splitting process has been used to cover the incoming input variables as sufficient as possible, however, some of the generated rules may not well "fired" at current sample time, i. e., the existing clusters are too far away from current input. Thus, the structure redundancy will cause the problem that some rule does not give contribution. In this paper, the rule-expanding process is provided to ensure that each rule is well-contributed. In Eq. (25), if $\overline{\overline{\beta}}^k$ is smaller than the pre-defined constant, then the index for the rule-expansion, denoted as I_{ex}^k ($I_{ex}^k(0)=1$), will be increased.

$$I_{ex}^k = I_{ex}^k \times \psi \tag{31}$$

The criterion of expanding a new rule is described as follows.

$$\Delta_{ex}^k = \frac{\{\exp(I_{ex}^k - 1)\} - \{\exp(I_{ex}^k - 1)\}}{\{\exp(I_{ex}^k - 1)\} + \{\exp(I_{ex}^k - 1)\}} \tag{32}$$

If $\Delta_{ex}^k \leq L_{th}$, where $L_{th} \in (-1,0)$ a pre-given threshold, then the variances of the k-th rule are given.

$$\sigma_{i(k)} = \sigma_{i(k)} \times \delta_2 \,|_{i=1,2,\dots,M} \tag{33}$$

where $\delta_2 > 1$. With the simple concept for determining the importance of the existing rules based on judging the rules is whether appropriately "fired" or not, the proposed rule self-construction has provided an efficient way for optimizing the NFS structure.

Table I. Settings for the SCNFS learning.

η	T	U_{th}	L_{th}	γ	ψ	δ_1	δ_2
0.05	0.001	0.4	-0.4	1	1.05	0.2	1.2

6. Experimental Results

In this section, a speech signal example is given to verify the performance of the SCNFS. 8000 sample data are contained, 400 sample data for the training purpose and the rest for testing. The parameter settings for the RO-LSE algorithm are given in Table I. The inputs to the SCNFS filter are the noise signal $h_1(t)=n(t)$ and the lagged noise $h_2(t)=n(t-1)$. The noise channel transfer function between the noise $n(t)$ and the contaminating signal $y(t)$ is given as follows.

$$y(t) = f(n(t), n(t-1)) = \frac{\sin(n(t))n(t-1)}{1+[n(t-1)^2]} \tag{34}$$

Based on the priori information of received noisy signal data, the neuro-fuzzy filter attempts to identify the relationship of $n(t)$ and $y(t)$ of the nonlinear noise channel. To train the NFS filter, the signal $r(t)$ is generated as the training data for the CI-based adaptive filter. The cost function for training is given in Eq. (16) for RO-LSE learning. The Gaussian noise is assumed with zero mean and with different time-varying variances given as follows.

$$\begin{cases} 0.2, & 0 \leq N < 2000 \\ 0.3, & 2000 \leq N < 4000 \\ 0.15, & 4000 \leq N < 6000 \\ 1.0, & 6000 \leq N < 8000 \end{cases} \tag{35}$$

The cost value generated by Eq. (16) and the rule amount of SCNFS is shown in Fig. 4, where the RMSE is given with 0.0076 and five rules is generated in final for the SCNFS. The source signal $x(t)$ and the corrupted signal $m_{rs}(t)$ of the example are shown in Figs. 5(a-b), where the received signal shown in Fig. 5(c) is more complicate than the noise signal with static variance. The result of recovered signal is shown in Fig. 5(d).

7. Conclusions

The CI-based signal filtering approach has been proposed to the problem of adaptive noise cancellation. The RO-LSE hybrid learning approach for the CI-based filter has been applied on the speech signal example with excellent performance, as shown in Fig. 5. With the machine learning approach, the premises and consequents of the SCNFS are updated complementarily. The RO-LSE hybrid learning algorithm enables the SCNFS to perform the structure learning and the parameter learning simultaneously. With the proposed approach, a compact and well-parameterized NFS filter can be achieved with excellent performance with the RMSE of 0.0076. The proposed SCNFS features the salient properties, which include online rule construction, derivative-free learning, easy in programming, fast convergence, robustness in exposure of different time-varying variances of noise. Through the experimental results, the CI-based approach has shown excellent filtering performance for noise cancellation.

Acknowledgments
The research is funded by the National Science Council, Taiwan, ROC, with the Grand NSC 94-2213-E-024 -007.

References:
[1] B. Widrow and S. D Stearns, *Adaptive signal processing*, Prentice-Hall, New-Jersey, 1985.
[2] B. Kosko, *Fuzzy Engineering*, Englewood Cliffs, NJ: Prentice-Hall, 1997.
[3] S. Haykin, *Neural Networks: A comprehensive Foundation*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1999.
[4] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, pp. 359-366, 1989.
[5] L. X. Wang and J. M. Mendel, "Fuzzy basis functions, universal approximation, and orthogonal least squares learning," *IEEE Trans. Neural Netw.*, vol. 3, no. 5, pp. 807-814, Sep. 1992.
[6] Chunshien Li and C. Y. Lee, "Self-organizing neuro-fuzzy system for control of unknown plants," *IEEE Transactions on Fuzzy Syst.*, vol. 11, no. 1, pp. 135-150, Feb. 2003.
[7]S. Mitra and Y. Hayashi, "Neuro-fuzzy rule generation: Survey in soft computing framework," *IEEE Trans. Neural Netw.*, vol. 11, no. 3, pp. 748-768, May 2000.
[8] C. F. Juang and C. T. Lin, "An online self-constructing neural fuzzy inference network and its applications," *IEEE Transactions on Fuzzy Syst.*, vol. 6, no. 1, pp. 12-32, Feb. 1998.
[9] F. J. Lin, C. H. Lin, and P. H. Shen., "Self-constructing fuzzy neural network speed controller for permanent-Magnet synchronous motor drive," *IEEE Trans Fuzzy Syst.*, vol. 9, no. 5, pp. 751-759, Oct. 2001.
[10]C. S. Ouyang, W. J. Lee, S. J. Lee, "A TSK-type neurofuzzy network approach to system modeling problems," *IEEE Trans. Syst., Man, Cybern.*, vol.35, no. 4, pp. 751-767, Aug. 2005.
[11] Chunshien Li, R. Priemer and K. H. Cheng., "Optimization by Random Search with Jumps," *International Journal for Numerical Methods in Engineering*, vol. 60, pp. 1301-1315, 2004.
[12] Chunshien Li and R. Priemer. "Fuzzy Control of Unknown Multiple-Input-Multiple-Output Plants," *Fuzzy Sets and Syst.*, vol. 104, no. 2, pp. 245-267, 1999.

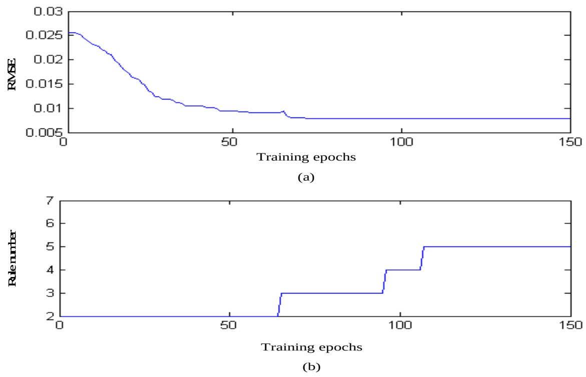

Fig. 4. Machine learning by the SCNFS filter. (a) Learning curve in RMSE. (b) Transition of rule amount in structure learning process.

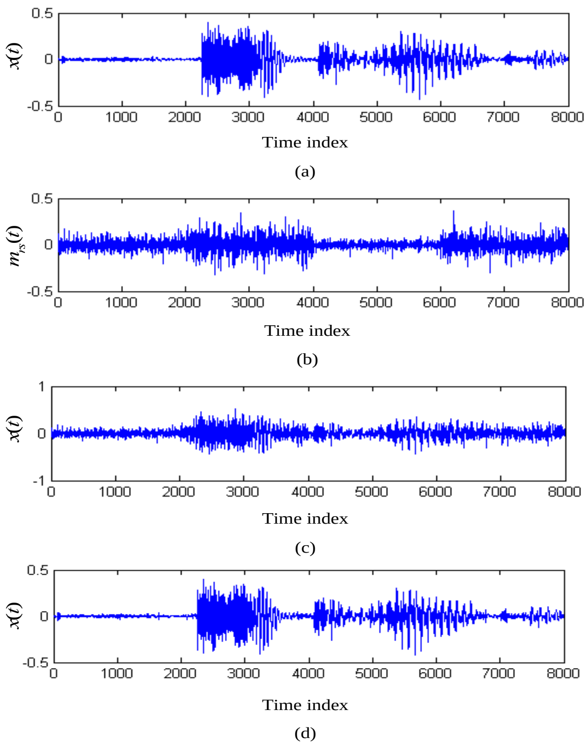

Fig. 5. Experimental results by the CI-based filter. (a) Original speech signal. (b) Noise signal. (c) Corrupted speech signal. (d) Recovered speech signal by the proposed SCNFS.