

# Overcoming Time-Varying Co-Channel Interference Using Type-2 Fuzzy Adaptive Filters

Qilian Liang and Jerry M. Mendel, *Fellow, IEEE*

**Abstract**—This paper presents a method for overcoming time-varying co-channel interference (CCI) using type-2 fuzzy adaptive filters (FAF). The type-2 FAF is realized using an unnormalized type-2 Takagi–Sugeno–Kang fuzzy logic system. A clustering method is used to adaptively design the parameters of the FAF. We use transversal equalizer and decision feedback equalizer structures to eliminate the CCI. Simulation results show that the equalizers based on type-2 FAFs perform better than the nearest neighbor classifiers or the equalizers based on type-1 FAFs when the number of co-channels is much larger than 1.

**Index Terms**—Co-channel interference, equalizers, time-varying channels, type-2 fuzzy adaptive filters, type-2 fuzzy logic systems.

## I. INTRODUCTION

CELLULAR MOBILE communication systems rely on an intelligent allocation and reuse of channels throughout a coverage region. The reuse of channels is realized by frequency reuse. Frequency reuse implies that in a given coverage area there are several cells that use the same set of frequencies. These cells are called co-channel cells, and the interference between signals from these cells is called *co-channel interference* (CCI) [18].

With the limitation of available signal spectrum, one way to incorporate more subscribers is to increase frequency reuse via reducing cell size, which introduces more CCI. For cellular communication systems, the radio link performance is usually limited by interference rather than noise and, therefore, by CCI. The effect of CCI on the radio link performance depends on the ability of the radio receiver to reject it [20]. In [2], an adaptive radial basis function (RBF) network is used to overcome CCI. In [22], polynomial perceptrons were used for fading channel equalization and co-channel interference suppression. In [13], an adaptive fractionally spaced decision feedback equalizer (DFE) which exploits the correlation of the cyclostationary interference is used to eliminate co-channel interference in a multipath fading environment. In [5], a Bayesian DFE was used to overcome CCI. In [8], a functional-link neural network-based DFE was used for overcoming CCI. Recently, Patra, and Mulgrew [17] used a fuzzy adaptive filter (FAF) to eliminate CCI (we call it “type-1 FAF” because its membership functions are

type-1 fuzzy sets), but the channel and co-channel models are time-invariant.

The statistical signal processing-based approach (e.g., Bayesian decision rule) is based on a probability model (e.g., Gaussian distribution), whereas the FAF-based approach is model free. As noted in [15], a shortcoming to model-based statistical signal processing is “... the assumed probability model, for which model-based statistical signal processing results will be good if the data agrees with the model, but may not be so good if the data does not.”

In this paper, we assume that the co-channel models are time-varying and the number of co-channels is much larger than 1. For these situations, we demonstrate that a type-1 FAF should be extended to a type-2 FAF in order to overcome CCI. In a type-2 FAF, the antecedent or consequent membership functions are type-2 fuzzy sets.

The concept of type-2 fuzzy sets was introduced by Zadeh [24] as an extension of the concept of an ordinary fuzzy set, i.e., a type-1 fuzzy set. Type-2 fuzzy sets have grades of membership that are themselves fuzzy [7]. A type-2 membership grade can be any subset in  $[0, 1]$ , the *primary membership*. Corresponding to each primary membership, there is a *secondary membership* (which can also be in  $[0, 1]$ ) that defines the possibilities for the primary membership. Type-2 fuzzy sets allow us to handle linguistic uncertainties, as typified by the adage “words can mean different things to different people [14].” Karnik and Mendel (see [9] and [10]) established a complete type-2 fuzzy logic systems (FLS) theory to handle linguistic and numerical uncertainties. Liang and Mendel [12] proposed a type-2 FAF and applied it to time-varying channel equalization.

In this paper, we interpret CCI as an uncertain disturbance added to the channel states. Theoretical analysis shows that this interpretation matches the reason of existence for a type-2 FAF, namely, to handle unknown uncertainties, and motivates us to use a type-2 FAF to overcome CCI.

In Section II, we provide some preliminaries that are needed for the rest of this paper, i.e., we review an unnormalized output type-1 TSK FLS, and summarize the concept of upper and lower membership functions (MFs) of a type-2 fuzzy set. In Section III, we introduce a type-2 FAF. In Section IV, a communication system with co-channel interference (CCI), intersymbol interference (ISI), and additive Gaussian noise (AGN) is reviewed, and we explain why type-2 fuzzy sets are needed to represent the uncertain channel states. In Section V, we apply a type-2 FAF as a transversal equalizer (TE) for overcoming CCI and ISI. In Section VI, we apply type-2 FAFs to a decision feedback equalizer (DFE) for eliminating CCI

Manuscript received September 1999; revised September 2000. This paper was recommended by Associate Editor C. Cowan.

The authors are with the Signal and Image Processing Institute, Department of Electrical Engineering-Systems, University of Southern California at Los Angeles, Los Angeles, CA 90089-2564 USA (e-mail: mendel@sipi.usc.edu).

Publisher Item Identifier S 1057-7130(00)11027-4.

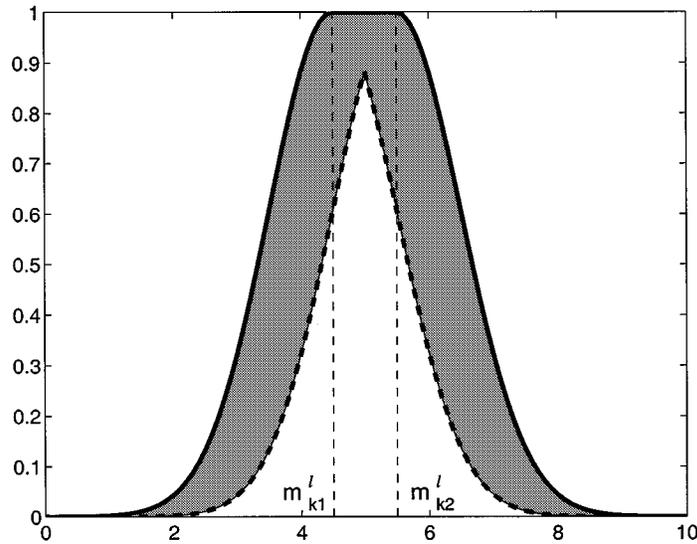


Fig. 1. The type-2 MF for Example 1. The thick solid lines denote upper MFs, and the thick dashed lines denote lower MFs. The shaded regions are the footprints of uncertainty for interval secondaries. The centers of Gaussian MFs vary from 4.5 to 5.5.

and ISI. Conclusions and future research directions are given in Section VII.

In this paper,  $A$  denotes a type-1 fuzzy set, and the membership grade of  $x \in X$  in  $A$  is  $\mu_A(x)$ , which is a crisp number in  $[0, 1]$ . A type-2 fuzzy set in  $X$  is  $\tilde{A}$ , and the membership grade of  $x \in X$  in  $\tilde{A}$  is  $\mu_{\tilde{A}}(x)$ , which is a type-1 fuzzy set in  $[0, 1]$ . The elements of the domain of  $\mu_{\tilde{A}}(x)$  are called *primary memberships* of  $x$  in  $\tilde{A}$  and the memberships of the primary memberships in  $\mu_{\tilde{A}}(x)$  are called *secondary memberships* of  $x$  in  $\tilde{A}$ . The latter defines the possibilities for the primary membership.  $\mu_{\tilde{A}}(x)$ , can be represented, for each  $x \in X$ , as  $\mu_{\tilde{A}}(x) = \int_u f_x(u)/u, u \in J_x \subseteq [0, 1]$  when the secondary MFs are type-1 interval sets, we call the type-2 set an *interval type-2 set*.  $\sqcap$  denotes *meet* operation while  $\sqcup$  denotes *join* operation. Meet and join are defined and explained in great detail in [9]. In this paper, we only use interval type-2 sets.

## II. PRELIMINARIES

In this section, we provide some preliminaries that are needed for the rest of the paper. We review an unnormalized output type-1 TSK FLS, and summarize the concept of upper and lower MFs of a type-2 fuzzy set.

### A. Unnormalized Output Type-1 TSK FLS

A type-1 TSK FLS is described by fuzzy IF-THEN rules which represent input-output relations of a system. The most widely used type-1 TSK FLS (the one we direct our attention at) is a first-order type-1 TSK FLS. It has a rule base of  $M$  rules, each having  $p$  antecedents, where the  $i$ th rule,  $R^i$ , is expressed as

$$\begin{aligned} R^i: & \text{IF } x_1 \text{ is } F_1^i \text{ and } x_2 \text{ is } F_2^i \text{ and } \dots \text{ and } x_p \text{ is } F_p^i \\ & \text{THEN } y^i = c_0^i + c_1^i x_1 + c_2^i x_2 + \dots + c_p^i x_p \end{aligned}$$

in which  $i = 1, 2, \dots, M$ ;  $c_j^i$  ( $j = 0, 1, \dots, p$ ) are the consequent parameters;  $y^i$  is the output of the  $i$ th IF-THEN rule; and,  $F_k^i$  ( $k = 1, 2, \dots, p$ ) are type-1 fuzzy sets. Given an input

$(x_1, x_2, \dots, x_p)$ , the final output of the unnormalized first-order type-1 TSK model is inferred as [21]

$$y = \sum_{i=1}^M f^i y^i \quad (1)$$

where  $f^i$  are rule firing strengths defined as

$$f^i = \mathcal{T}_{k=1}^p \mu_{F_k^i}(x_k) \quad (2)$$

and  $\mathcal{T}$  denotes a  $t$ -norm (minimum or product).

When Gaussian MFs are used, i.e.,

$$\mu_{F_k^i}(x_k) = \exp \left[ -\frac{1}{2} \left( \frac{x_k - m_k^i}{\sigma_k^i} \right)^2 \right] \quad (3)$$

and product  $t$ -norm is used, then (1) can be expressed as

$$y = \sum_{i=1}^M y^i \prod_{k=1}^p \exp \left[ -\frac{1}{2} \left( \frac{x_k - m_k^i}{\sigma_k^i} \right)^2 \right]. \quad (4)$$

Observe that (4) is identical to the output formula for a radial basis function (RBF) network [3] when Gaussian MFs are used as the RBFs. This kind of RBF network has been applied to Bayesian equalization [3], [4]. Later in this paper, the unnormalized output type-1 TSK FLS in (4) will be used as a type-1 FAF equalizer (for suppressing CCI and ISI), and its performance will be compared with that of a type-2 FAF.

### B. Upper and Lower MFs of Type-2 MFs

For convenience in defining the upper and lower MFs of a type-2 MF, we first give the definition of *footprint of uncertainty of a type-2 MF*.

*Definition 1 (Footprint of Uncertainty of a Type-2 MF):* Uncertainty in the primary membership grades of an interval type-2 MF consists of a bounded region that we call the *footprint of uncertainty* of that MF (e.g., see Fig. 1). It is the union of all primary membership grades.

*Definition 2 (Upper and Lower MFs):* An upper MF and a lower MF are two type-1 MFs which are bounds for the footprint of uncertainty of an interval type-2 MF. The upper MF is a

subset which has the maximum membership grade of the footprint of uncertainty; and, the lower MF is a subset which has the minimum membership grade of the footprint of uncertainty.

We use an overbar (underbar) to denote the upper (lower) MF. For example, the upper and lower MFs of the interval type-2 fuzzy set  $\mu_{\tilde{F}_k^l}(x_k)$  (used in the next section) are  $\bar{\mu}_{\tilde{F}_k^l}(x_k)$  and  $\underline{\mu}_{\tilde{F}_k^l}(x_k)$ , and,  $\mu_{\tilde{F}_k^l}(x_k)$  can be expressed as

$$\mu_{\tilde{F}_k^l}(x_k) = \int_{w^l \in [\underline{\mu}_{\tilde{F}_k^l}(x_k), \bar{\mu}_{\tilde{F}_k^l}(x_k)]} 1/w^l. \quad (5)$$

*Example 1: Gaussian Primary MF with Uncertain Mean:* Consider the case of a Gaussian primary MF having a fixed standard deviation,  $\sigma_k^l$ , and an uncertain mean that takes on values in  $[m_{k1}^l, m_{k2}^l]$ , i.e.,

$$\mu_k^l(x_k) = \exp\left[-\frac{1}{2}\left(\frac{x_k - m_k^l}{\sigma_k^l}\right)^2\right], \quad m_k^l \in [m_{k1}^l, m_{k2}^l] \quad (6)$$

where:  $k = 1, \dots, p$ ;  $p$  is the number of antecedents;  $l = 1, \dots, M$ ; and,  $M$  is the number of rules. The upper MF,  $\bar{\mu}_k^l(x_k)$ , is (see Fig. 1)

$$\bar{\mu}_k^l(x_k) = \begin{cases} \mathcal{N}(m_{k1}^l, \sigma_k^l; x_k), & x_k < m_{k1}^l \\ 1, & m_{k1}^l \leq x_k \leq m_{k2}^l \\ \mathcal{N}(m_{k2}^l, \sigma_k^l; x_k), & x_k > m_{k2}^l \end{cases} \quad (7)$$

where, for example,  $\mathcal{N}(m_{k1}^l, \sigma_k^l; x_k) \triangleq \exp(-(1/2)(x_k - m_{k1}^l/\sigma_k^l)^2)$ . The lower MF,  $\underline{\mu}_k^l(x_k)$ , is (see Fig. 1)

$$\underline{\mu}_k^l(x_k) = \begin{cases} \mathcal{N}(m_{k2}^l, \sigma_k^l; x_k), & x_k \leq \frac{m_{k1}^l + m_{k2}^l}{2} \\ \mathcal{N}(m_{k1}^l, \sigma_k^l; x_k), & x_k > \frac{m_{k1}^l + m_{k2}^l}{2} \end{cases} \quad (8)$$

□

We use the results of this example later in Sections V and VI.

### III. TYPE-2 FAF: AN OVERVIEW

In [12], a type-2 FAF for channel equalization is obtained by generalizing the unnormalized output type-1 TSK FLS to a type-2 TSK FLS. In a type-2 FAF with a rule base of  $M$  rules, where each rule has  $p$  antecedents, the  $i$ th rule,  $R^i$ , is denoted as

$$R^i: \quad \text{IF } x_1 \text{ is } \tilde{F}_1^i \text{ and } x_2 \text{ is } \tilde{F}_2^i \text{ and...and } x_p \text{ is } \tilde{F}_p^i \\ \text{THEN } y^i = c_0^i + c_1^i x_1 + c_2^i x_2 + \dots + c_p^i x_p$$

where  $i = 1, 2, \dots, M$ ;  $c_j^i$  ( $j = 0, 1, \dots, p$ ) are the consequent parameters that are crisp numbers;  $y^i$  is an output from the  $i$ th IF-THEN rule, which is a crisp number (because it will be determined by the channel state category); and, the  $\tilde{F}_k^i$  ( $k = 1, 2, \dots, p$ ) are type-2 fuzzy sets. Other more general type-2 TSK FLSs have been proposed by Liang and Mendel in [11].

Given an input  $\mathbf{x} = [x_1, x_2, \dots, x_p]^T$ , the firing strength of the  $i$ th rule is [9], [10]

$$F^i = \mu_{\tilde{F}_1^i}(x_1) \sqcap \mu_{\tilde{F}_2^i}(x_2) \sqcap \dots \sqcap \mu_{\tilde{F}_p^i}(x_p). \quad (9)$$

The final output of the type-2 FAF is obtained by applying the *Extension Principle* [24] to (1), i.e.,

$$Y(F^1, \dots, F^M) = \int_{f^1} \dots \int_{f^M} \mathcal{T}_{i=1}^M \mu_{F^i}(f^i) \left/ \sum_{i=1}^M f^i y^i \right. \quad (10)$$

where

- $M$  is the number of rules fired,  $f^i \in F^i$ , and,
- $\mathcal{T}$  indicates the chosen  $t$ -norm.
- $Y$  is called an *extended weighted average* as it reveals the uncertainty at the output of a type-2 FLS due to antecedent uncertainties, and is itself a type-1 fuzzy set.

Here we focus on the very practical case when interval type-2 sets are used in the antecedents, which means  $\mu_{\tilde{F}_k^i}(x_k)$ , ( $k = 1, \dots, p$ ) is an interval set, and we denote

$$\mu_{\tilde{F}_k^i}(x_k) = [\underline{\mu}_{\tilde{F}_k^i}(x_k), \bar{\mu}_{\tilde{F}_k^i}(x_k)] \triangleq [f_k^i, \bar{f}_k^i]. \quad (11)$$

Our type-2 FAF is then computed using results in the following: *Theorem 1* [12]:

- 1) In an interval type-2 FAF with meet under minimum or product  $t$ -norm, the firing strength in (9) for rule  $R^i$  is an interval set,  $F^i = [f^i, \bar{f}^i]$ , where ( $i = 1, \dots, M$ )

$$\underline{f}^i = \underline{\mu}_{\tilde{F}_1^i}(x_1) \star \dots \star \underline{\mu}_{\tilde{F}_p^i}(x_p) = \mathcal{T}_{k=1}^p \underline{f}_k^i \quad (12)$$

and

$$\bar{f}^i = \bar{\mu}_{\tilde{F}_1^i}(x_1) \star \dots \star \bar{\mu}_{\tilde{F}_p^i}(x_p) = \mathcal{T}_{k=1}^p \bar{f}_k^i. \quad (13)$$

- 2) The extended weighted average  $Y(F^1, \dots, F^M)$  is also an interval set,  $[y_l, y_r]$ , where

$$y_r = \sum_{i=1}^M \bar{f}^i y^i \quad (14)$$

$$y_l = \sum_{i=1}^M \underline{f}^i y^i \quad (15)$$

and

$$y^i = c_0^i + c_1^i x_1 + c_2^i x_2 + \dots + c_p^i x_p. \quad (16)$$

- 3) The defuzzified output of this type-2 FAF is

$$y = \sum_{i=1}^M y^i (\underline{f}^i + \bar{f}^i) / 2. \quad (17)$$

The proof of this theorem is given in [12].

### IV. A COMMUNICATION SYSTEM WITH CCI

The discrete-time model of a communication system that is subject to CCI, intersymbol interference (ISI) and additive Gaussian noise (AGN) is shown in Fig. 2 [1], [5], where  $s(k)$  is the symbol to be transmitted;  $c(k)$  is the noise; the CCI  $u(k)$  comes from  $N$  co-channels; the channel order is  $n$  ( $n + 1$  taps), and we assume that the tap coefficients are time-varying, because in today's communications, such as wireless communications, the coefficients  $a_i(k)$  ( $i = 0, 1, \dots, n$ ) are time-varying, hence,  $r(k)$  can be represented as

$$r(k) = \hat{r}(k) + u(k) + c(k) \\ = \sum_{i=0}^n a_i(k) s(k-i) + u(k) + c(k). \quad (18)$$

Here, we assume that  $s(k)$  is binary, i.e., it is either  $+1$  or  $-1$  with equal probability. Assuming the number of taps in an equalizer is  $p$ , we let

$$\hat{\mathbf{r}}(k) \triangleq [\hat{r}(k), \dots, \hat{r}(k-p+1)]^T \quad (19)$$

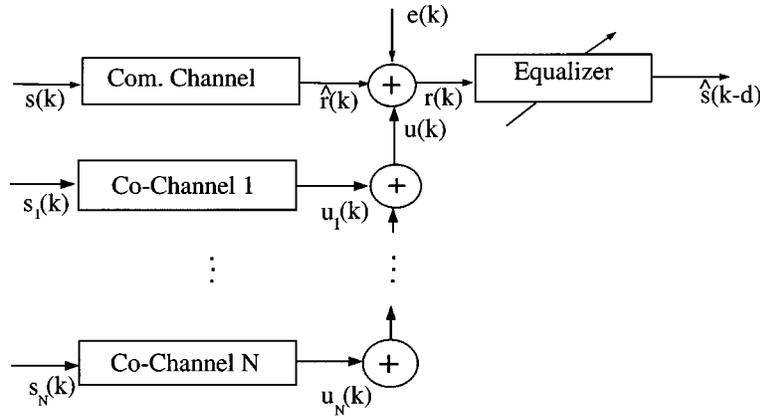


Fig. 2. Discrete time model of communication system subject to CCI, ISI, and AGN.

where  $\hat{\mathbf{r}}(k)$  is called *channel state*. Observe from (18), that  $\mathbf{r}(k)$  depends on the channel input sequence  $\mathbf{s}(k)$  (an  $(n+p) \times 1$  vector), where

$$\mathbf{s}(k) = [s(k), s(k-1), \dots, s(k-n-p+1)]^T \quad (20)$$

so each of the  $n_s = 2^{n+p}$  combinations of the channel input sequence  $\mathbf{s}(k)$  generates one  $\hat{\mathbf{r}}(k)$ , which we denote as  $\hat{\mathbf{r}}_i(k)$ , where  $\hat{\mathbf{r}}_i(k) = [\hat{r}_i(k), \dots, \hat{r}_i(k-p+1)]^T$ . Hence, the number of channel states is  $n_s = 2^{n+p}$ , and each channel state has a probability of occurrence equal to  $1/n_s$ .

Assume the order of the  $j$ th co-channel is  $n_j$  ( $n_j+1$  taps), and time-varying tap coefficients are  $b_{ij}$  ( $i = 0, 1, \dots, n_j$ ); then, CCI  $u(k)$  can be expressed as

$$u(k) = \sum_{j=1}^N u_j(k) = \sum_{j=1}^N \sum_{i=0}^{n_j} b_{ij}(k) s_j(k-i) \quad (21)$$

where  $s_j$  is also binary, but  $s_j$  is always blind to the equalizer, even in the training period. As in (19), the co-channel states of the  $j$ th co-channel are

$$\mathbf{u}_j(k) \triangleq [u_j(k), u_j(k-1), \dots, u_j(k-p+1)]^T \quad (22)$$

so similarly to (20), there are  $2^{n_j+p}$  channel states in the  $j$ th co-channel. The total co-channel state is

$$\mathbf{u}(k) \triangleq [u(k), u(k-1), \dots, u(k-p+1)]^T \quad (23)$$

$$= \sum_{j=1}^N \mathbf{u}_j(k) \quad (24)$$

where  $\mathbf{u}_j(k)$  ( $j = 1, 2, \dots, N$ ) are independent, so the number of total co-channel states is  $n_{co} = \prod_{j=1}^N 2^{n_j+p} = 2^{pN + \sum_{j=1}^N n_j}$ . This leads to a large number of co-channel states, especially when  $N \gg 1$ , e.g., if there are 6 co-channels (i.e.,  $N = 6$ ), and each co-channel has 3 taps (i.e.,  $n_j = 2$ ), and the equalizer has 2 taps (i.e.,  $p = 2$ ), then there will be  $2^{24} = 16\,777\,216$  co-channel states. So when  $N \gg 1$ , it is not possible to perform an exact analysis of the co-channel states.

The signal-to-noise ratio (SNR), signal-to-interference ratio (SIR), and signal-to-interference noise ratio (SINR) in decibels are defined as (see Fig. 2)

$$\text{SNR} = 10 \log_{10} \frac{\sigma_{\hat{\mathbf{r}}}^2}{\sigma_e^2} \quad (25)$$

$$\text{SIR} = 10 \log_{10} \frac{\sigma_{\hat{\mathbf{r}}}^2}{\sigma_u^2} \quad (26)$$

$$\text{SINR} = 10 \log_{10} \frac{\sigma_{\hat{\mathbf{r}}}^2}{\sigma_e^2 + \sigma_u^2} \quad (27)$$

in which  $\sigma_e$  and  $\sigma_u$  denote the standard deviation (std) of the Gaussian additive noise  $e(k)$  and CCI  $u(k)$ , respectively.

In a time-invariant communication system with ISI and AWGN, but no CCI, the discriminant function of a Bayesian equalizer is [3]

$$f(\mathbf{r}(k)) = \sum_{i=1}^{n_s} \prod_{l=0}^{p-1} w_i \exp \left[ -\frac{1}{2} \frac{[r(k-l) - \hat{r}_i(k-l)]^2}{\sigma_e^2} \right] \quad (28)$$

where  $w_i$  equals either  $+1$  or  $-1$  as determined by the channel state category. Observe that (28) is structurally identical to (4). Hence, an unnormalized output type-1 TSK FLS can be used to implement a Bayesian equalizer for a time-invariant channel.

In contrast, in a time-invariant communication system with CCI, ISI and AWGN, the discriminant function of the Bayesian equalizer is [17]

$$f(\mathbf{r}(k)) = \sum_{i=1}^{n_s} \sum_{m=1}^{n_{co}} \prod_{l=0}^{p-1} w_i \times \exp \left[ -\frac{1}{2} \frac{[r(k-l) - \hat{r}_i(k-l) - u^m(k-l)]^2}{\sigma_e^2 + \sigma_u^2} \right] \quad (29)$$

where  $u^m(k-l)$  is the  $l$ th element of the  $m$ th co-channel state. As we discussed above,  $n_{co}$  is a very large number if  $N \gg 1$ . To simplify the problem, when SIR is high, we assume that the

term  $\hat{r}_i(k-l) + u^m(k-l)$  in (29) is an uncertain value which is dominated by  $\hat{r}_i(k-l)$ . Thus, (29) can be approximated by

$$f(\mathbf{r}(k)) \approx \sum_{i=1}^{n_s} \prod_{l=0}^{p-1} w_i \exp \left[ -\frac{1}{2} \frac{[r(k-l) - \hat{r}_i(k-l)]^2}{\sigma_e^2 + \sigma_u^2} \right] \quad (30)$$

where  $\hat{r}_i(k-l)$  is uncertain. This point of view motivates us to use a type-2 FAF for CCI elimination because type-2 FAFs can handle the uncertainties. In addition, the channel states are time-varying in a time-varying channel, which is another reason to treat  $\hat{r}_i(k-l)$  as an uncertain value. Observing (30), we see that a Gaussian MF with an uncertain mean (see Fig. 1) is an appropriate choice for the MFs in a type-2 TSK FAF.

## V. TRANSVERSAL EQUALIZER FOR OVERCOMING CCI

In [23], the following nonlinear time-invariant channel model was used

$$r(k) = a_1(k)s(k) + a_2(k)s(k-1) - 0.9[a_1(k)s(k) + a_2(k)s(k-1)]^3 + e(k) \quad (31)$$

where  $a_1(k) = 1$  and  $a_2(k) = 0.5$ . We choose two taps in the transversal equalizer (TE), so this channel has eight channel states, i.e.,  $n_s = 8$ . In this section, we illustrate the design of a type-2 FAF for this channel, but we focus on the case when the channel is time-varying, i.e., when  $a_1(k)$  and  $a_2(k)$  in (31) are time-varying coefficients, each simulated, as in [6], by using a second-order Markov model in which a white Gaussian noise source drives a second-order Butterworth low-pass filter (LPF). In our simulations below, we used the function *butter*, provided by the Matlab Signal Processing Toolbox, to generate a second-order low-pass digital Butterworth filter with cut-off frequency 0.1. The function *filter* was then used to generate a colored Gaussian sequence which was used as a time-varying channel coefficient. Note that we centered  $a_1(k)$  about 1, and  $a_2(k)$  about 0.5. The input to the Butterworth filter was a white Gaussian sequence with standard deviation (std)  $\beta$ .

So that readers may replicate our simulations, we provide the source code for the time-varying coefficients with length 1000

```
[B,A]=butter(2,0.1); % B (numerator) and
A (denominator) of LPF
a1=1+filter(B,A,beta*randn(1,1000));
a2=0.5+filter(B,A,beta*randn(1,1000)).
```

We also assumed that this communication system had five co-channels ( $N = 5$ ), where

$$H_{co1}(z) = \lambda(b_{11}(k) + b_{12}(k)z^{-1}) \quad (32)$$

$$H_{co2}(z) = \lambda(b_{21}(k) + b_{22}(k)z^{-1} + b_{23}(k)z^{-2}) \quad (33)$$

$$H_{co3}(z) = \lambda(b_{31}(k) + b_{32}(k)z^{-1} - b_{33}(k)z^{-2}) \quad (34)$$

$$H_{co4}(z) = \lambda(b_{41}(k) + b_{42}(k)z^{-1} + b_{43}(k)z^{-2}) \quad (35)$$

$$H_{co5}(z) = \lambda(b_{51}(k) + b_{52}(k)z^{-1} + b_{53}(k)z^{-2}). \quad (36)$$

In our simulations, we assumed that the co-channel coefficients were also time-varying, and their nominal values are:  $b_{11}(k) = 1$ ,  $b_{12}(k) = 0.2$ ;  $b_{21}(k) = 0.4084$ ,  $b_{22}(k) = 0.8164$ ,  $b_{23}(k) = 0.4084$ ;  $b_{31}(k) = 0.407$ ,  $b_{32}(k) = 0.815$ ,  $b_{33}(k) = 0.407$ ;  $b_{41}(k) = 0.3482$ ,  $b_{42}(k) = 0.8704$ ,  $b_{43}(k) = 0.3482$ ; and,

$b_{51}(k) = 0.5$ ,  $b_{52}(k) = 0.81$ ,  $b_{53}(k) = 0.31$ . They were simulated the same way as we described for simulating  $a_1(k)$  and  $a_2(k)$ . The std of white Gaussian noise used for generating co-channel coefficients was fixed at 0.1.  $\lambda$  was determined by SIR.

### A. Designing the Type-2 FAF

In our type-2 FAF design for (31) with five co-channels, there are eight rules (each rule corresponds to one channel state) where the  $l$ th rule,  $R^l$ , is expressed [see (30)] as

$$R^l: \text{IF } r(k) \text{ is } \tilde{F}_1^l \text{ and } r(k-1) \text{ is } \tilde{F}_2^l \text{ THEN } y^l = w_l$$

in which  $\tilde{F}_1^l$  and  $\tilde{F}_2^l$  are type-2 Gaussian MFs with uncertain means, and  $w_l$  is a crisp value of +1 or -1 as determined by the channel state category, which equals  $s(k-d)$ . Observe from this rule that the consequent is a constant, i.e., it does not depend on  $r(k)$  and  $r(k-1)$ . Hence, the consequent is a special case of the consequent in Section 3.

We used (17) to compute the output of the type-2 FAF, where  $y^l = w_l$  ( $l = 1, \dots, 8$ ) equals 1 or -1,  $\underline{f}^l$  is obtained from (12), and  $\bar{f}^l$  is obtained from (13). As in (6), we chose

$$\mu_{\tilde{F}_k^l}(x_k) = \exp \left[ -\frac{1}{2} \frac{(x_k - m_k^l)^2}{\sigma_e^2 + \sigma_u^2} \right], \quad m_k^l \in [m_{k1}^l, m_{k2}^l] \quad (37)$$

(see Fig. 1) and  $k = 1, 2$ . In order to specify the MFs  $\tilde{F}_1^l$  and  $\tilde{F}_2^l$ , we had to specify their parameters, namely,  $[m_{k1}^l, m_{k2}^l]$  and  $\sigma_e^2 + \sigma_u^2$ . Below, we let  $\mathbf{m}_1^l = [m_{11}^l, m_{21}^l]^T$  and  $\mathbf{m}_2^l = [m_{12}^l, m_{22}^l]^T$ .

We used a clustering approach to estimate  $\mathbf{m}_1^l$  and  $\mathbf{m}_2^l$ , because it is computationally simple [3]. Here we briefly summarize this approach. Suppose the number of training prototypes,  $(\mathbf{s}(k), \mathbf{r}(k))$ , is  $N$ . As we see from (20) and (19),  $\mathbf{s}(k)$  determines which cluster  $\mathbf{r}(k)$  belongs to. So, the  $N$   $\mathbf{r}(k)$  are classified into  $n_s = 2^{n+p}$  clusters, where, in our example,  $2^{n+p} = 2^{2+1} = 8$ . Suppose  $N_l$  training prototypes belong to the  $l$ th cluster, ( $l = 1, \dots, 8$ ), and the mean and std of these  $\mathbf{r}(k)$ , ( $k = 1, \dots, N_l$ ), are denoted  $\mathbf{m}_r^l$  ( $2 \times 1$  vector) and  $\sigma_r^l = [\sigma_1^l, \sigma_2^l]^T$ , respectively. We let

$$\mathbf{m}_1^l \triangleq \mathbf{m}_r^l - \sigma_r^l \quad (38)$$

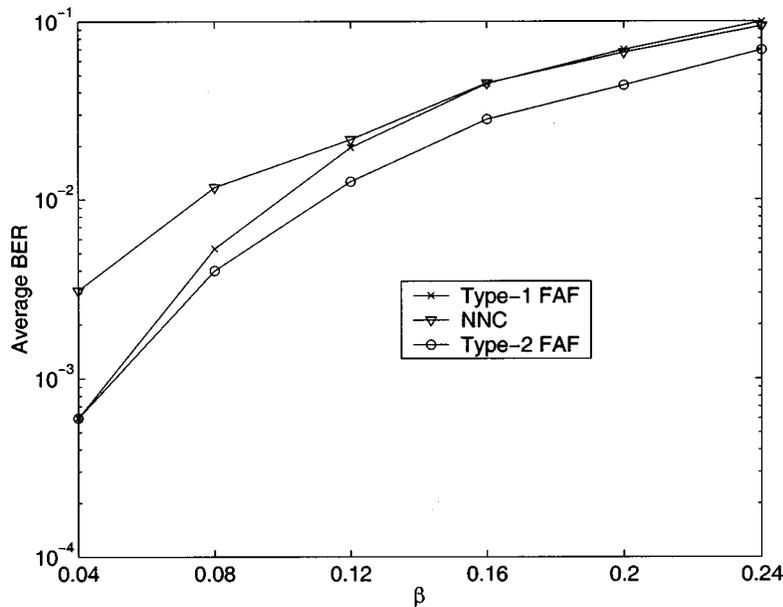
$$\mathbf{m}_2^l \triangleq \mathbf{m}_r^l + \sigma_r^l \quad (39)$$

where  $l = 1, 2, \dots, 8$ . Doing this assumes that each cluster is centered at  $\mathbf{m}_r^l$ . Consequently,  $[m_{11}^l, m_{12}^l]$  is the range of the mean of the type-2 antecedent Gaussian MF,  $\mu_{\tilde{F}_1^l}$ , and  $[m_{21}^l, m_{22}^l]$  is the range of the mean of  $\mu_{\tilde{F}_2^l}$ . In our simulations, we compare our type-2 FAF with a type-1 FAF in which we used  $\mathbf{m}_r^l$  (i.e.,  $[m_{r1}^l, m_{r2}^l]^T$ ) as the centers of its type-1 Gaussian antecedent MFs.

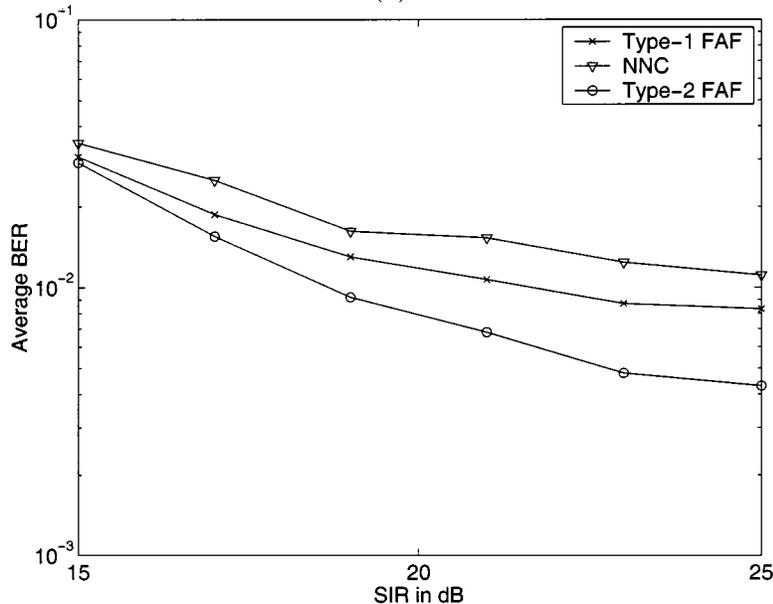
To complete the specification of the MFs in (37), we also need the value of  $\sigma_e^2 + \sigma_u^2$ . We let  $\sqrt{\sigma_e^2 + \sigma_u^2}$  be the average of the std components of all clusters, i.e.,

$$\sqrt{\sigma_e^2 + \sigma_u^2} \triangleq \frac{1}{p2^{n+p}} \sum_{l=1}^{2^{n+p}} \sum_{i=1}^p \sigma_i^l. \quad (40)$$

In this simulation,  $n = 1$  and  $p = 2$ .



(a)



(b)

Fig. 3. Average BER of type-1 FAF, NNC, and type-2 FAF for 100 MC realizations when SNR = 20 dB and the number of training prototypes is 121. (a) Average BER versus  $\beta$  when SIR = 20 dB. (b) Average BER versus SIR when  $\beta = 0.1$ .

### B. Simulations

We compared our type-2 FAF with an unnormalized type-1 FAF (the latter is identical to an RBF network [3] in its output formula) and a nearest neighbor classifier (NNC) [19] for ISI and CCI elimination of the time-varying channel in (31) and co-channels (32)–(36). In our simulations, we chose the number of taps of the equalizer,  $p = 2$ , and the number of rules equal to the number of clusters, i.e., 8. We used six independent sequences  $s(k), s_1(k), s_2(k), s_3(k), s_4(k), s_5(k)$ , each of length 1000 for our experiments. The first 121 symbols in  $s(k)$  were used for training, and the remaining 879 in  $s(k)$  were used for testing. The training sequence established the parameters of the antecedent MFs, as described in Section V-A. After training,

the parameters of the type-1 and type-2 FAFs were fixed, and then testing was performed. In our simulations in this section, we fixed SNR = 20 dB.

In our first experiment, we fixed SIR at 20 dB and ran simulations for six different  $\beta$ , ranging from  $\beta = 0.04$  to  $\beta = 0.24$  (0.04:0.04:0.24), and we set decision delay  $d = 0$ . We performed 100 Monte Carlo (MC) simulations for each  $\beta$  value. In Fig. 3(a), we plot the average bit error rate (BER) for the 100 MC realizations. In a second experiment, we fixed  $\beta = 0.1$  and ran simulations for six different SNR values ranging from SNR = 15 dB to SNR = 25 dB (15:2:25). We again performed 100 MC simulations for each SNR value. In Fig. 3(b), we plot the average BER for the 100 MC realizations. Observe

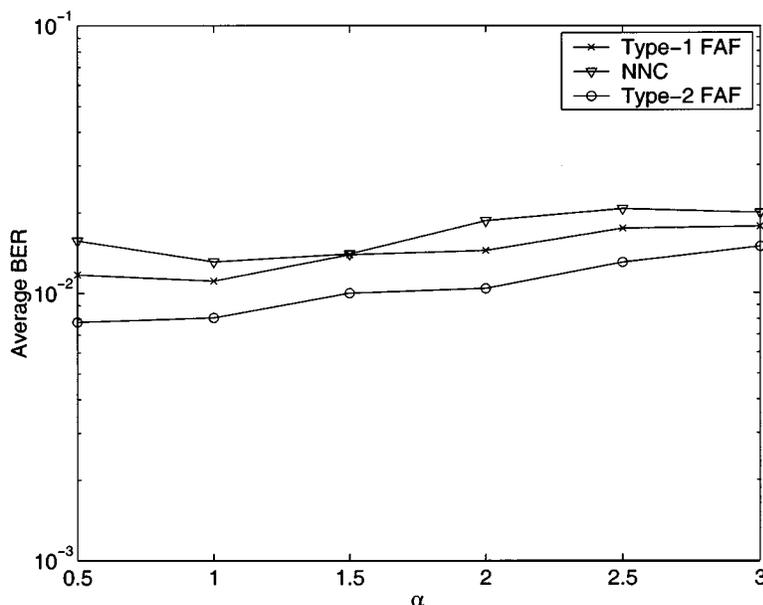


Fig. 4. Average BER versus  $\alpha$  in (41) for type-1 FAF, NNC, and type-2 FAF using 100 MC realizations when SNR = 20 dB and SIR = 20 dB, and the number of training prototypes is 121. The CCI in (41) was introduced at time-index 500.

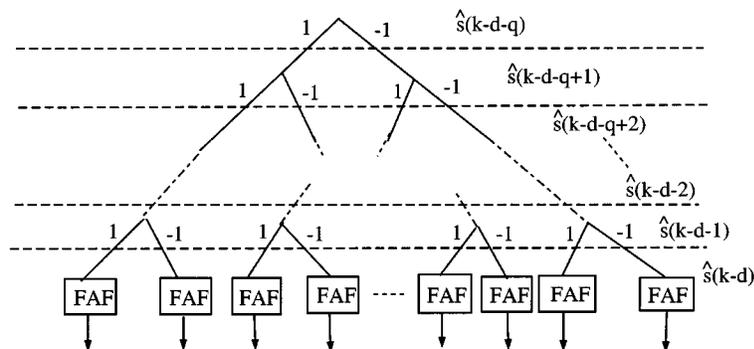


Fig. 5. For a channel with  $n + 1$  taps and a decision delay of  $d$ , the general architecture of a DFE with  $p$  ( $p = d + 1$ ) feedforward taps and  $q$  ( $q = n$ ) feedback taps. The DFE consists of a decision tree and  $2^q$  FAFs, where each FAF has  $2^p$  rules.

from these figures that the type-2 fuzzy filter performs better than both the NNC and type-1 FAF.

To show the robustness of the equalizer in overcoming CCI, we introduced another co-channel,  $H'_{co}(z)$ , during the testing period, beginning at time index  $k = 500$ , and remaining through  $k = 1000$ , where

$$H'_{co}(z) = \alpha\lambda(0.33 - 0.4z^{-1} + 0.5z^{-2} + 0.69z^{-3}). \quad (41)$$

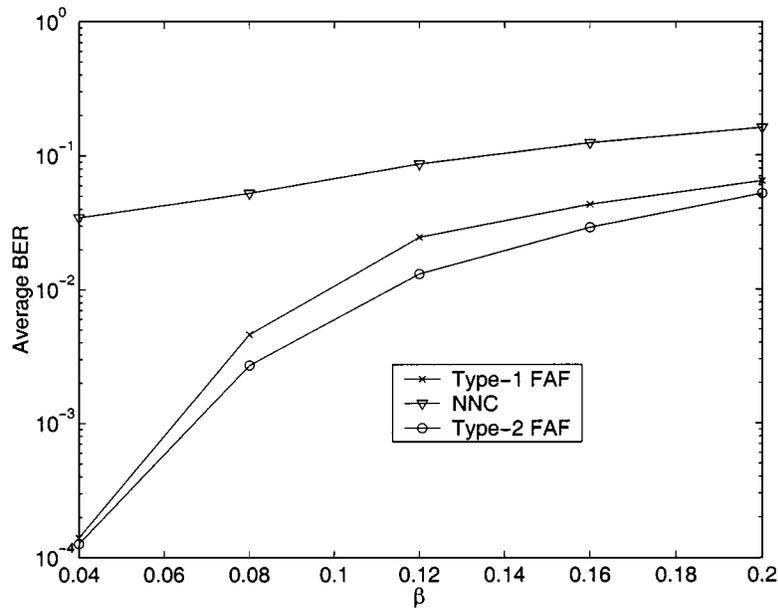
The value of  $\lambda$  is the same as in (32)–(36); changing  $\alpha$  changes the CCI strength in  $H'_{co}(z)$ . In this experiment, we fixed SNR = 20 dB and SIR = 20 dB, and ran simulations for 6 different  $\alpha$ , ranging from  $\alpha = 0.5$  to  $\alpha = 3$  ( $0.5 : 0.5 : 3$ ). We performed 100 MC simulations for each  $\alpha$  value. In Fig. 4, we plot the average BER for the 100 MC realizations. Observe, from this figure, that all three equalizers are very robust. However, the type-2 FAF maintains better performance than both the NNC and type-1 FAF over the entire range of  $\alpha$  values.

In real world communications such as the global systems for mobile (GSM) communications, unique words (training samples) are sent in each frame, so all parameters in our type-2 FAF

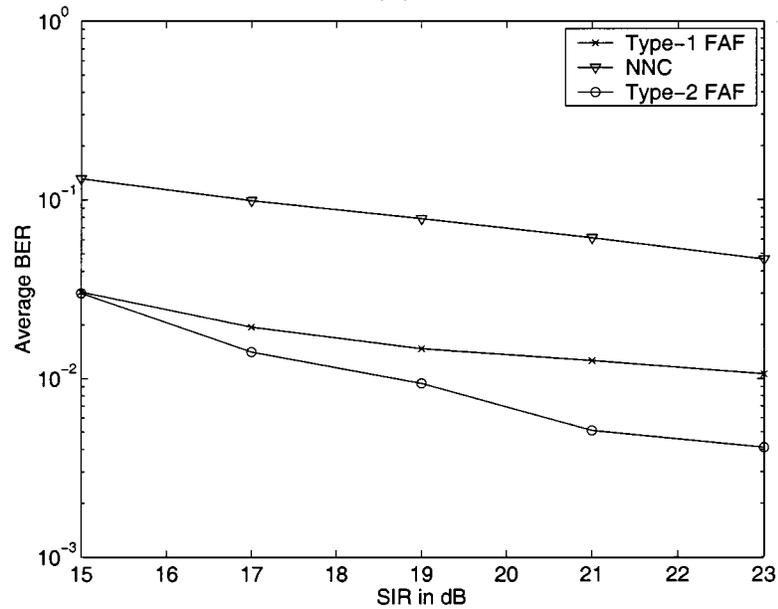
can be determined based on clustering the training samples in the current frame, which means our type-2 FAF can track the system very well. Since we do not need to use tuning algorithms such as the least-mean-square (LMS) algorithm to optimize the parameters, there is no convergence problem in our type-2 FAF.

As shown in Theorem 1, the computation complexity of a type-2 FAF is equivalent to that of two type-1 FAFs. However, a nearest neighbor classifier (NNC) has much higher computation complexity than either a type-1 or type-2 FAF because generally the number of training samples is much larger than the number of rules. To detect an information symbol, NNC needs to compute the Euclidean distances between the information symbol and all training samples, and then sort all distances and make a decision. However, a type-1 or type-2 FAF can detect an information symbol just based on computing the fired rules.

As described in [12], the number of rules for a TE is  $n_s = 2^{n+p}$  (recall that  $n + 1$  is the number of channel taps, and  $p$  is the number of antecedents), e.g., for  $n = 4$ ,  $p = 5$ , we need 512 rules. This causes huge computational complexity when the channel order is high. As in [12], we can use a decision feedback



(a)



(b)

Fig. 6. Average BER of type-1 FAF, NNC, and type-2 FAF for 100 MC realizations when  $\text{SNR} = 25$  dB and the number of training prototypes is 289. (a) Average BER versus  $\beta$  when  $\text{SIR} = 20$  dB. (b) Average BER versus SIR when  $\beta = 0.1$ .

equalizer (DFE) which can reduce the number of rules in a FAF to overcome CCI.

## VI. DFE FOR ELIMINATING CCI

The structure of a general DFE is specified by the decision delay  $d$ , and number of channel taps,  $n + 1$ .  $d$  is chosen by the designer, and increasing  $d$  improves performance, but it is required that  $d \leq n$  [4]. It has been shown in [4] that choosing the number of feedforward taps as  $p = d + 1$  (reducing  $d$  reduces the number of antecedents), and the number of feedback taps as  $q = n$ , is sufficient for a DFE to achieve all the performance

potential (i.e., a DFE with  $p = d + 1$  has the same performance as a DFE with  $p > d + 1$  taps) for a given  $d$  and  $n$ .

An architecture which can eliminate rule explosion in a FAF is proposed in [12]. In Fig. 5, we depict this general structure for a DFE. It consists of a decision tree and  $2^q$  FAFs, where each FAF has only  $2^p$  rules. Observe that only one FAF is activated at a time to obtain the value of  $\hat{s}(k - d)$ . This structure reduces the number of FAF rules, and makes it easy to design each of the FAFs, e.g., if a channel has five taps ( $n = 4$ ), delay  $d = 4$ , and we choose  $p = d + 1 = 5$  and  $q = n = 4$ , then we only need to design  $2^q = 16$  FAFs, each having  $2^p = 32$  rules (although there are 16 FAFs, only one is activated at any time).

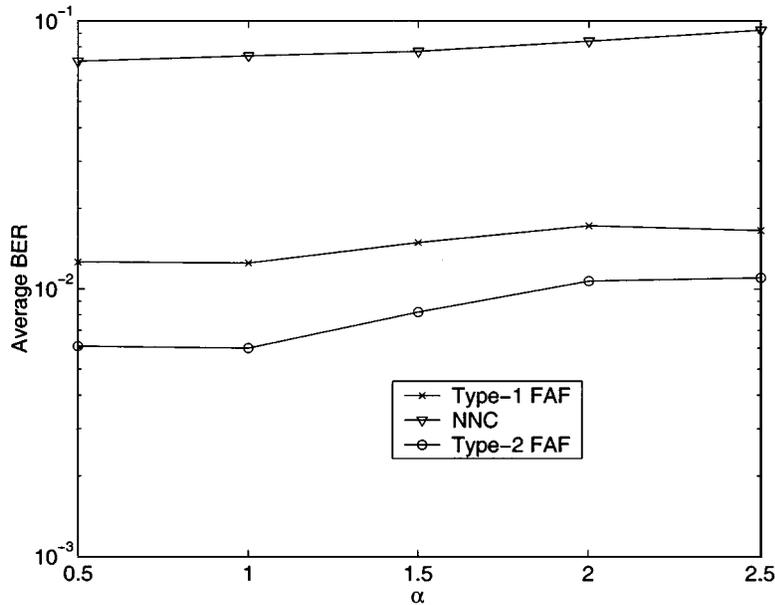


Fig. 7. Average BER versus  $\alpha$  in (41) for type-1 FAF, NNC, and type-2 FAF using 100 MC realizations when SNR = 25 dB and SIR = 20 dB, and the number of training prototypes is 289. The CCI in (41) was introduced at time-index 500.

We used the following nonlinear time-varying channel in our simulations of a DFE [12]:

$$r(k) = a_1(k)s(k) + a_2(k)s(k-1) + a_3(k)s(k-2) - 0.7[a_1(k)s(k) + a_2(k)s(k-1) + a_3(k)s(k-2)]^3 + c(k) \quad (42)$$

where nominal values for the channel coefficients are:  $a_1 = 0.3482$ ,  $a_2 = 0.8704$ , and  $a_3 = 0.3482$ . We used the method given in Section V to simulate the time-varying natures of  $a_1(k)$ ,  $a_2(k)$ , and  $a_3(k)$ . There were six co-channels;  $H_{co1}(z), \dots, H_{co5}(z)$  are the same as those in (32)–(36), and

$$H_{co6}(z) = \lambda(b_{61}(k) + b_{62}(k)z^{-1} + b_{63}(k)z^{-2} + b_{64}(k)z^{-3} + b_{65}(k)z^{-4}) \quad (43)$$

where the nominal values of  $b_{61}(k) = -0.2052$ ,  $b_{62}(k) = -0.5131$ ,  $b_{63}(k) = 0.7183$ ,  $b_{64}(k) = 0.3695$ , and  $b_{65}(k) = 0.2052$ .

We assumed a decision delay  $d$  of one. Since channel order  $n = 2$  (we assumed that  $n$  is known), then it is sufficient to design a DFE with  $p = 2$  and  $q = 2$ , which means the decision tree has  $2^2 = 4$  leaves (FAFs), and each leaf (FAF) has  $2^2 = 4$  rules. As we see from Fig. 5, in this case  $\hat{s}(k-3)$  and  $\hat{s}(k-2)$  (during training  $s(k-3)$  and  $s(k-2)$ , because they are available) determine which FAF will be chosen. Each chosen FAF has  $2^2$  rules, which means  $2^2$  clusters are needed. Designing the rules in each of the four FAFs is the same as that of designing a transversal fuzzy equalizer, as described in Section V-A. In our simulations, we used (40) to determine  $\sigma_e^2 + \sigma_u^2$ , where  $p = 2$  and  $n = 2$ .

Simulations were performed for channel (42) with six co-channels, and seven independent sequences  $s(k), s_1(k), s_2(k), s_3(k), s_4(k), s_5(k), s_6(k)$  were used, each of length 1000 for our experiments. The first 289 symbols were for training, and the remaining 711 symbols were for

testing. After training, the parameters in the four type-2 FAFs were fixed and then testing was performed. In all simulations, we fixed SNR at 25 dB.

In our first experiment, we fixed SIR at 20 dB and ran simulations for five different  $\beta$  ranging from  $\beta = 0.04$  to  $\beta = 0.2$  (0.04 : 0.04 : 0.20). We performed 100 MC simulations for each  $\beta$  value. In Fig. 6(a), we plot the average BER for the 100 MC realizations. In a second experiment, we fixed  $\beta = 0.1$ , and ran simulations for five different SIRs ranging from SIR = 15 dB to SIR = 23 dB (15 : 2 : 23). We again performed 100 MC simulations for each SIR value. In Fig. 6(b), we plot the average BER for these 100 MC realizations. From these figures, we see that the DFE based on our four type-2 FAFs performs better than the NNC and the DFE based on four type-1 FAFs (each is an unnormalized type-1 TSK FLS). The NNC cannot work well in such a complicated channel because there are 16 channel states and an NNC typically needs more training prototypes than we have used.

To show the robustness of the equalizer in overcoming CCI, we introduced another co-channel, the one in (41), during the testing period beginning at time index  $k = 500$ , and remaining through  $k = 1000$ . The value of  $\lambda$  was the same as in the other six co-channels, and changing  $\alpha$  changes the CCI strength in  $H'_{co}(z)$ . In this experiment, we fixed SNR = 25 dB and SIR = 20 dB, and ran simulations for five different  $\alpha$ , ranging from  $\alpha = 0.5$  to  $\alpha = 3$  (0.5 : 0.5 : 2.5). We performed 100 MC simulations for each  $\alpha$  value. In Fig. 7, we plot the average BER for the 100 MC realizations. Observe, from this figure, that all the equalizers are fairly robust. However, the type-2 FAF again maintains better performance than both the NNC and type-1 FAF.

Our DFE architecture can eliminate the rule explosion in one TE FAF and reduce computational complexity. It appears to be very promising for implementation using hardware for high-order channels.

## VII. CONCLUSIONS AND FUTURE WORKS

We applied type-2 FAFs to eliminate co-channel interference in nonlinear and time-varying channels. TE and DFE structures were implemented for overcoming co-channel interference. Simulation results showed that both the type-2 TE and DFE FAFs performed better than either a type-1 FAF, or a nearest neighbor classifier. Since no tuning procedure was used in the design of either type-2 FAF-based equalizer, real-time information processing is guaranteed.

Both TE and DFE have their own advantages and disadvantages. A TE has higher computational complexity, and for some channels, some channel states with different categories cannot be classified because it is possible that different  $s$  leads to the same  $\hat{r}$  value. A DFE reduces computational complexity, and increases the performance of classification. However, as we see from Fig. 5, earlier decision errors will lead to a current detection error, known as error accumulation. A TE does not have this problem.

The data modulation used in this paper is BPSK. However, quadrature modulations are often used in today's communication systems, such as in spread-spectrum communications, since they are more difficult to detect using feature detectors [16]. How to apply our type-2 FAFs to CCI and ISI elimination for channels with QAM signal constellations, using a sequence detection approach, is presently under study.

## REFERENCES

- [1] J. C. Campbell, A. J. Gibbs, and B. M. Smith, "The cyclostationary nature of crosstalk interference from digital signals in multipair cables," *IEEE Trans. Commun.*, vol. COM-31, pp. 629–649, May 1983.
- [2] S. Chen and B. Mulgrew, "Overcoming co-channel interference using an adaptive radial basis function equalizer," *Signal Processing*, vol. 28, pp. 91–107, July 1992.
- [3] S. Chen, B. Mulgrew, and S. McLaughlin, "A clustering technique for digital communications channel equalization using radial basis function network," *IEEE Trans. Neural Networks*, vol. 4, pp. 570–579, July 1993.
- [4] —, "Adaptive Bayesian equalizer with decision feedback," *IEEE Trans. Signal Processing*, vol. 41, pp. 2918–2927, Sept. 1993.
- [5] S. Chen, S. McLaughlin, B. Mulgrew, and P. M. Grant, "Bayesian decision feedback equalizer for overcoming co-channel interference," *Proc. Inst. Elect. Eng.*, vol. 143, pp. 219–225, Aug. 1996.
- [6] C. F. N. Cowan and S. Semnani, "Time-variant equalization using a novel nonlinear adaptive structure," *Int. J. Adaptive Control Signal Processing*, vol. 12, no. 2, pp. 195–206, 1998.
- [7] D. Dubois and H. Prade, *Fuzzy Sets and Systems: Theory and Applications*. New York: Academic, 1980.
- [8] A. Hussain, J. J. Soraghan, and T. S. Durrani, "A new adaptive functional-link neural-network-based DFE for overcoming co-channel interference," *IEEE Trans. Commun.*, vol. 45, pp. 1358–1362, Nov. 1997.
- [9] N. N. Karnik and J. M. Mendel. (1998) Introduction to Type-2 Fuzzy Logic Systems. Univ. Southern California. [Online]. Available: <http://sipi.usc.edu/~mendel/report>
- [10] N. N. Karnik, J. M. Mendel, and Q. Liang, "Type-2 fuzzy logic systems," *IEEE Trans. Fuzzy Syst.*, vol. 7, pp. 643–658, Dec. 1999.
- [11] Q. Liang and J. M. Mendel, "An introduction to type-2 TSK fuzzy logic systems," in *Proc. IEEE FUZZ'99*, Seoul, Korea.
- [12] —, "Equalization of nonlinear time-varying channels using type-2 fuzzy adaptive filters," *IEEE Trans. Fuzzy Syst.*, vol. 8, pp. 551–563, Oct. 2000.
- [13] N. W. Lo, D. D. Falconer, and A. U. Sheikh, "Adaptive equalization for co-channel interference in a multipath fading environment," *IEEE Trans. Commun.*, vol. 43, pp. 1441–1453, Apr. 1995.
- [14] J. M. Mendel, "Computing with words when words can mean different things to different people," presented at the Int'l. ICSC Congress on Computational Intelligence: Methods & Applications, Third Annual Symposium on Fuzzy Logic and Applications, Rochester, NY, June 22–25, 1999.
- [15] —, "Uncertainty, fuzzy logic, and signal processing," *Signal Proc.*, vol. 80, no. 6, pp. 913–933, June 2000.
- [16] D. L. Nicholson, *Spread Spectrum Signal Design: LPE and AJ Systems*. Rockville, MD: Computer Science, 1988.
- [17] S. K. Patra and B. Mulgrew, "Fuzzy implementation of Bayesian equalizer in the presence of intersymbol and co-channel interference," *Proc. Inst. Elect. Eng.*, vol. 145, pp. 323–330, 1998.
- [18] T. S. Rappaport, *Wireless Communications*. Englewood Cliffs, NJ: Prentice-Hall, 1996.
- [19] P. Savazzi, L. Favalli, E. Costamagna, and A. Mecocci, "A suboptimal approach to channel equalization based on the nearest neighbor rule," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1640–1648, Dec. 1998.
- [20] G. L. Stuber, *Mobile Communication*. Norwell, MA: Kluwer, 1998.
- [21] K. Tanaka, M. Sano, and H. Watanabe, "Modeling and control of carbon monoxide concentration using a neuro-fuzzy technique," *IEEE Trans. Fuzzy Syst.*, vol. 3, pp. 271–279, Aug. 1995.
- [22] Z. Xiang, G. Bo, and T. Le-Ngoc, "Polynomial perceptrons and their applications to fading channel equalization and co-channel interference suppression," *IEEE Trans. Signal Processing*, vol. 42, pp. 2470–2480, Sept. 1994.
- [23] L.-X. Wang and J. M. Mendel, "Fuzzy adaptive filters, with application to nonlinear channel equalization," *IEEE Trans. Fuzzy Systems*, vol. 1, pp. 161–170, Aug. 1993.
- [24] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning—I," *Inform. Sci.*, vol. 8, pp. 199–249, 1975.



**Qilian Liang** received the B.S. degree from Wuhan University, China, in 1993, M.S. degree from Beijing University of Posts and Telecommunications, China, in 1996, and Ph.D. degree from University of Southern California, Los Angeles, in May 2000, all in electrical engineering.

He was a Research Assistant in the Department of Electrical Engineering-Systems, University of Southern California, Los Angeles, from Sept. 1997 to May 2000. His research interests include wireless communications, broadband networks, fuzzy logic systems, and video traffic classification. He is currently a Systems Engineer in Hughes Network Systems Inc. at San Diego.



**Jerry M. Mendel** (S'59–M'61–SM'72–F'78) received the Ph.D. degree in electrical engineering from the Polytechnic Institute of Brooklyn, Brooklyn, NY.

Currently he is Professor of Electrical Engineering and Associate Director for Education of the Integrated Media Systems Center at the University of Southern California in Los Angeles, where he has been since 1974. He has published over 380 technical papers and is author and/or editor of eight books, including *Uncertain Rule-based Fuzzy Logic Systems: Introduction and New Directions* (Prentice-Hall, 2001). His present research interests include: type-2 fuzzy logic systems and their applications to a wide range of problems.

Dr. Mendel is a Fellow of the IEEE and a Distinguished Member of the IEEE Control Systems Society. He was President of the IEEE Control Systems Society in 1986. Among his awards are the 1983 Best Transactions Paper Award of the IEEE Geoscience and Remote Sensing Society, the 1992 Signal Processing Society Paper Award, a 1984 IEEE Centennial Medal, and an IEEE Third Millennium Medal.