An Extended Kalman Filter Based Fuzzy Adaptive Equalizer For Powerline Channel

Wai Kit Wong, Heng Siong Lim
Faculty of Engineering and Technology, Multimedia University, Jln Ayer Keroh Lama, 75450 Melaka, Malaysia. Tel:+60 06-2523258
Email:wkwong@mmu.edu.my;hslim@mmu.edu.my

Abstract
Fuzzy logic is the principles of imprecise knowledge. Fuzzy adaptive equalizers are adaptive equalizers that apply the concepts of fuzzy logic. The main merit of applying fuzzy adaptive equalizers in powerline channel equalization is that linguistic information (fuzzy IF-THEN rules) and numerical information (input-output pairs) can be combined into the equalizers. The adaptive algorithms adjust the parameters of the membership functions which characterize the fuzzy concepts in the IF-THEN rules, by minimizing some criterion function. In this paper, we introduce a new type of fuzzy adaptive equalizer using extended Kalman filter (EKF) algorithm for powerline channel equalization. The performance for this type of fuzzy adaptive equalizer is compared with two other types of fuzzy adaptive equalizers using recursive least squares (RLS) and least mean squares (LMS) adaption algorithm. The simulation results show that the extended Kalman filter based fuzzy adaptive equalizer has faster convergent speed compared to the other two fuzzy adaptive equalizers. The bit error rate of extended Kalman filter based fuzzy adaptive equalizer is close to that of the optimal equalizer.

I. INTRODUCTION
Powerline Communication (PLC) is a wire line method of communication using the existing electric power transmission and electricity distribution lines. Powerline channel is a very harsh and noisy transmission medium, hence for high-speed data communication, channel equalization is necessary [1]. Fuzzy adaptive equalizer (FAE) is a device that performs digital signal processing, and is able to adapt its performance to optimally suit the input signal. The most important advantage of using fuzzy adaptive equalizers in powerline channel equalization rather than polynomials, radial basis functions or other nonlinear equalizers is that beside numerical information (input-output pairs), linguistic information (in the form of fuzzy IF-THEN rules) from human expert can be embedded into the equalizers[2].

In [3], two types of fuzzy adaptive equalizers, the RLS-based FAE and LMS-based FAE were applied to nonlinear channel equalization in which the simulation results show that these fuzzy adaptive equalizers worked quite well without using any linguistic information. By incorporating some linguistic rules into the fuzzy adaptive equalizers, the adaptation speed was greatly improved. Bit error rate of those fuzzy equalizer were close to that of the maximum-a-posteriori probability (MAP) equalizer.

This paper presents a fuzzy adaptive equalizer that uses an extended Kalman filter (EKF) adaptation algorithm [4] for equalization of powerline channel. Simulation results show that this type of equalizer has faster convergent speed compared to both RLS-based FAE and LMS-based FAE. Section II briefly comments on the powerline data transmission system. Section III presents the proposed equalizer. Section IV reports some performance comparison between the proposed equalizer, RLS-based FAE and LMS-based FAE. Section V summarizes the work.

II. POWERLINE DATA TRANSMISSION SYSTEM
The baseband powerline data transmission system considered in this paper is shown in Fig. 1 where the channel includes the effects of the transmission filter, PLC channel and reception filter. The transmitted data sequences s(k) are binary sequences, taking values from { -1,1 } with equal probability. The inputs to the equalizer x(k), x(k-1),…, x(k-n+1) are the channel outputs corrupted by noise. The task of the equalizer at the sampling instant k is to produce an estimate of the transmitted symbol $\hat{s}(k-d)$, using the information contained in x(k), x(k-1),…, x(k-n+1) where the integer n and d are the order and lag of the equalizer respectively.

The PLC channel model suggested in [5] is given by:

$$H(f) = \sum_{i=1}^{N} g_i A(f, d_i) e^{-j2\pi \tau_i}$$  \hspace{1cm} (1)

where $g_i$ is the weighting factor that summarizes the reflection and transmission factors along the i-th path, $A(f, d_i)$ is the cable attenuation that increases with length and frequency, and $\tau_i$ is the propagation delay of the i-th multipath. The parameters $g_i$ and $A(f, d_i)$ can be derived from measured transfer function [5].

As stated in [1],[6],[7], there exist various types of noise on power lines such as: (i) Colored background noise (ii) Narrowband noise (iii) Periodic impulsive noise (iv) Random/Asynchronous impulsive noise. The properties of noise types (i) and (ii) usually remain stationary over periods of seconds and minutes or sometimes even for hours, and
can be summarized as background noise [8]. On the other hand, the noise types (iii) and (iv) are time invariant in terms of microseconds to milliseconds. During the occurrence of such impulses, the power spectral density of the noise rises abruptly and this causes bit or burst errors in data transmission on power line.

Background noise and periodic impulsive noise can be modeled as Gaussian type with zero mean and variance $\sigma^2$. But for random/asynchronous impulsive noise, a stochastic model is required. A special form of the Markov chain, called partitioned Markov chain approach [8] turns out to be appropriate for modeling such event. Since the optimal decision boundary is normally nonlinear for channel equalization problem, linear equalization methods are no longer adequate for the task. In this case, nonlinear equalizers that have the ability to perform nonlinear input-output mapping can be applied to minimize the error probability. In this paper, we propose a fuzzy adaptive equalizer with extended Kalman filter adaptation algorithm for effective and efficient powerline channel equalization.

III. EXTENDED KALMAN FILTER BASED FUZZY ADAPTIVE EQUALIZER

The extended Kalman filter based fuzzy adaptive equalizer (EKFAE) is constructed with three steps:

1.) Define $M$ fuzzy sets $F_i^l$ for each interval $[C_i^-; C_i^+]$ of the input space, $U$, with Gaussian membership functions [4]:

$$
\mu_{F_i^l} = \exp \left[ -\frac{1}{2} \left( \frac{x_i - \bar{x}_i^l}{\sigma_i^l} \right)^2 \right] \tag{2}
$$

where $l = 1, 2, \ldots, M$, $i = 1, 2, \ldots, n$, $x_i = x(k-i+l)$ is the input to the equalizer, $\bar{x}_i^l$ is center of $i$-th membership function in the $l$-th rule and $\sigma_i^l$ represents the width of the $i$-th membership function in the $l$-th rule. $\bar{x}_i^1$ and $\sigma_i^1$ are free parameters which will be optimized using the EKF algorithm. The reason we choose Gaussian membership functions rather than triangular, trapezoidal, etc. is because, it can be shown by using Stone Weierstrass theorem, the Gaussian network is a universal approximator that can be used to uniformly approximate continuous functions on a compact set [9],[10]. However, if other types of membership functions are employed in such network, the universal approximation capability may not be easily verified, and at the meantime, large number of rules may be needed to carry out a function approximation.

2.) Construct a set of changeable fuzzy IF-THEN rules either by linguistic information or numerical information from the matching input-output data pairs:

$$
R^l : \text{IF } x_i \text{ is } F_i^l \text{ and } \ldots \text{ and } x_n \text{ is } F_n^l, \text{ THEN } d \text{ is } G^l. \tag{3}
$$

where $d \in R$ is the desired output, $F_i^l$’s are defined in step 1, $G^l$’s are fuzzy sets defined in $R$ which are determined as follows: if there are linguistic rules in the form of (3), set $F_i$’s and $G^l$ to be the labels of these linguistic rules, otherwise, choose $\mu_{G^l}$ and the parameters $\bar{x}_i^1$ and $\sigma_i^1$ arbitrarily. These parameters will change during the adaptation process. The rules constructed in this step are initial rules of the fuzzy adaptive equalizer. We may incorporate the linguistic rules into the EKFAE by constructing the initial equalizer based on these rules.

3.) Construct the equalizer $f(x)$ based on the set of $M$ rules by using product inference and centroid defuzzification [9]:

$$
f(x) = \frac{\sum_{i=1}^{M} \prod_{l=1}^{n} \mu_{F_i^l}(x_i)}{\sum_{i=1}^{M} \prod_{l=1}^{n} \mu_{F_i^l}(x_i)} \tag{4}
$$

where $x = [x_1, \ldots, x_n]^T$, $\mu_{F_i^l}$’s are the Gaussian membership functions defined in (2), and $\theta^l \in R$ is the
value which $\mu_G$ achieves its maximum. Since we pick our membership function, $\mu_{F_i}(x_i)$ to be Gaussian function which is nonzero for any $x_i \in [C_{i-}, C_i^+]$, the denominator of (4) is nonzero for any $x \in U$. Therefore (4) is well defined. Since $\theta'$ as well as $\bar{x}_i'$ and $\sigma_i'$ are free parameters, the equalizer (4) is nonlinear in the parameters.

The parameters of the filter, $\theta'$, $\bar{x}_i'$ and $\sigma_i'$ are updated using the EKF algorithm [4]:

Correction

$$\Omega(k) = \Omega(k-1) + G(k) [d(k) - f(x(k))]$$  

(5)

Kalman Gain Matrix

$$G(k) = E(k-1)H^T(k) [H(k)E(k-1)H^T(k) + N]^{-1}$$  

(6)

(Posterior) error covariance matrix

$$E(k) = E(k-1) - G(k)H(k)E(k-1)$$  

(7)

where $k$ is the discrete time index.

$\Omega(k)$ is the equalizer parameter (i.e. $\theta'$, $\bar{x}_i'$ or $\sigma_i'$).

$x(k)$ is the input vector at time $k$.  

$d(k)$ is the desired output at time $k$.  

$f(x(k))$ is the equalizer output at time $k$.  

$N$ is the measurement noise covariance.  

This also involves the computation of the Jacobian $\Omega(k)$, which is obtained as the linearization about the current value of those nonlinear parameters ($\theta'$, $\bar{x}_i'$ and $\sigma_i'$).

Let  

$$a'(k-1) = \prod_{i=1}^{n} \exp \left[ -\frac{1}{2} \left( \frac{x_i(k) - \bar{x}_i'(k-1)}{\sigma_i'(k-1)} \right)^2 \right]$$  

(8)

$$b(k-1) = \sum_{i=1}^{n} a'(k-1)$$  

(9)

$$c'(k-1) = \left( \frac{x_i - \bar{x}_i'}{\sigma_i'} \right)^2$$  

(10)

$$e'(k-1) = \left( \frac{x_i - \bar{x}_i'}{\sigma_i'} \right)^2$$  

(11)

Therefore, the equalizer output as from equation (4) becomes:

$$f_d(x(k)) = \frac{\sum_{i=1}^{M} \theta_i \prod_{j=1}^{n} \mu_{F_i,j}(x_i)}{\sum_{i=1}^{M} \prod_{j=1}^{n} \mu_{F_i,j}(x_i)}$$  

The computation of the Jacobian $\Omega(k)$ requires the evolution of the partial derivatives of $f_d(x(k))$ vs. the parameters of the fuzzy network:

$$\Omega_{\theta} = \frac{\partial f_d(x(k))}{\partial \theta'} = \frac{\partial}{\partial \theta'} \left( \frac{\sum_{i=1}^{M} \theta' a'(k-1)}{b(k-1)} \right) = \frac{a'(k-1)}{b(k-1)}$$  

(13)

$$\Omega_{\sigma} = \frac{\partial f_d(x(k))}{\partial \sigma_i'} = \frac{\partial}{\partial \sigma_i'} \left( \frac{\sum_{i=1}^{M} \theta' a'(k-1)}{b(k-1)} \right)$$  

(14)

Therefore,

$$\Omega_{\theta} = \frac{\sum_{i=1}^{M} \prod_{j=1}^{n} \mu_{F_i,j}(x_i)}{\sum_{i=1}^{M} \prod_{j=1}^{n} \mu_{F_i,j}} \times \left[ \theta' - f_k(x(k)) \right] \times \frac{\left( \frac{x_i - \bar{x}_i'}{\sigma_i'} \right)^2}{\left( \frac{x_i - \bar{x}_i'}{\sigma_i'} \right)^2}$$  

(15)

and  

$$\frac{\partial \mu_{F_i'}}{\partial \sigma_i'} = \frac{\partial}{\partial \sigma_i'} \left( \exp \left[ -\frac{1}{2} \left( \frac{x_i - \bar{x}_i'}{\sigma_i'} \right)^2 \right] \right)$$  

(16)

Therefore,

$$\Omega_{\sigma} = \frac{\sum_{i=1}^{M} \prod_{j=1}^{n} \mu_{F_i,j}(x_i)}{\sum_{i=1}^{M} \prod_{j=1}^{n} \mu_{F_i,j}} \times \left[ \theta' - f_k(x(k)) \right] \times \frac{\left( \frac{x_i - \bar{x}_i'}{\sigma_i'} \right)^2}{\left( \frac{x_i - \bar{x}_i'}{\sigma_i'} \right)^2}$$  

(17)

Therefore,

$$\Omega_{\theta} = \frac{\sum_{i=1}^{M} \prod_{j=1}^{n} \mu_{F_i,j}(x_i)}{\sum_{i=1}^{M} \prod_{j=1}^{n} \mu_{F_i,j}} \times \left[ \theta' - f_k(x(k)) \right] \times \frac{\left( \frac{x_i - \bar{x}_i'}{\sigma_i'} \right)^2}{\left( \frac{x_i - \bar{x}_i'}{\sigma_i'} \right)^2}$$  

(18)
\[ \Omega_{i} = \frac{\partial f_{i}(x(k))}{\partial x_{i}} = \frac{\partial f_{i}(x(k))}{\partial \mu_{\theta_{i}}} \times \frac{\partial \mu_{\theta_{i}}}{\partial x_{i}} \]

since \[ \frac{\partial f_{i}(x(k))}{\partial \mu_{\theta_{i}}} = \prod_{j=1}^{n} \mu_{\theta_{j}} \times \left[ \theta_{j} - f_{k}(x(k)) \right] \]

and \[ \frac{\partial}{\partial x_{i}} \left\{ \exp \left[ -\frac{1}{2} \left( \frac{x_{i} - \tilde{x}_{i}}{\sigma_{i}} \right)^{2} \right] \right\} = \mu_{\theta_{i}} \left( x_{i} - \tilde{x}_{i} \right) \left( \sigma_{i} \right)^{-1} \]

Therefore,

\[ \Omega_{i} = \frac{\prod_{j=1}^{n} \mu_{\theta_{j}}}{\sum_{i=1}^{M} \prod_{j=1}^{n} \mu_{\theta_{j}}} \times \left[ \theta_{j} - f_{k}(x(k)) \right] \times \mu_{\theta_{i}} \times \left( \frac{x_{i} - \tilde{x}_{i}}{\sigma_{i}} \right)^{-2} \]

\[ = \frac{\left( \theta_{j} - f_{k}(x(k)) \right) \mu_{\theta_{i}}(k-1)c_{\theta_{i}}(k-1)}{b(k-1)} \] \hspace{1cm} (15)

**IV. NUMERICAL RESULTS**

In this section, we investigate the performance of the proposed equalizer in terms of convergence speed and steady-state bit error rate (BER), and compare the results to other related algorithms. We consider an arbitrary channel with transfer function \( h(z) = 0.5 + z^{-1} \). The order and lag of the equalizer are \( n=2 \) and \( d=0 \) respectively. We have compared the performance of EKFAE with two types of fuzzy adaptive equalizers, namely RLS-based FAE and LMS-based FAE as proposed in [3]. The background noise is modeled as zero mean Gaussian with variance \( \sigma_{l}^{2} \). The time behavior of the impulsive noise is based on a 7-state partitioned Markov chain model with transition probabilities given in [8]. The EKFAE and LMS-based FAE have \( M=25 \) rules while the fuzzy RLS adaptive equalizer has \( m_{1} \times m_{2} = 5 \times 5 \) rules.

**Example 1** We first consider the situation where there is no linguistic information. We randomly set \( \theta(0) \) in \([-0.5, 0.5], \tilde{x}_{i}(0) \) in \([-2.0, 2.0] \) and \( \sigma_{i}(0) \) in \([0.1, 0.3] \). For EKFAE, we set \( N=0.999 \) and \( E(0)=I \), where \( I \) is an identity matrix of size \( M \times M \). For LMS-based FAE, we set the step size for adaptation, \( \alpha = 0.05 \). For RLS-based FAE, we set the forgetting factor, \( \lambda = 0.999 \) and \( \sigma = 0.01 \). When both signal to background noise ratio and signal to impulsive noise ratio are equal to 10dB, we plot the convergence curves for the three equalizers in Fig. 2. The simulation results show that without using any linguistic information, the EKFAE is a well performing nonlinear adaptive equalizer. The EKFAE achieves the fastest convergent speed. The EKFAE also outperforms the RLS-based FAE and LMS-based FAE in term of steady-state BER.

**Example 2** Next, we incorporate some linguistic information about the decision region into the fuzzy adaptive equalizers. For EKFAE and LMS-based FAE, the first \( M \) training data pairs are used to construct the initial fuzzy rules. In particular, the centers of the fuzzy antecedents \( \tilde{x} = [\tilde{x}_{1}, \tilde{x}_{2}] \) are set to the received signals \( \{x(l) x(l-1)\} \) where \( l=1,2,\ldots,M \). Similarly, the centers of the fuzzy consequences \( \theta \) are set to the corresponding scaled desired data, \( \beta d(l) \), where \( l=1,2,\ldots,M \). In the simulation, we use \( \beta = 0.5 \). For RLS-based FAE, we use 25 training data pairs to construct the initial fuzzy rules. The centers for the fuzzy antecedents, \( \tilde{x}_{i}^{h} = -1.6, -0.8, 0, 0.8, 1.6 \) are fixed and will not change during the adaptation, where \( j=1,2,\ldots,5 \) and \( i=1,2 \). The decision region of the RLS-based FAE can be partitioned as shown in Fig 5. Every section corresponds to a certain fuzzy rule. The initial fuzzy rules can be constructed as follow: if the equalizer inputs \( \{x(l) x(l-1)\} \) fall in the region that corresponds to the rule \( R^{(j,l)} \), then set the center of the fuzzy consequence \( \theta^{(j,l)} \) to the corresponding scaled desired data, \( \beta d(l) \). For the region that the received signal does not fall in, we choose their \( \theta^{(j,l)} \) arbitrary in \([-0.3, 0.3] \). Other parameters remain the same as in Example 1. When both signal to background noise ratio and signal to impulsive noise ratio are equal to 10 dB, we plot the convergence curves for the three equalizers in Fig. 3. The simulation results show that by incorporating some noisy linguistic information about the decision region into the equalizers, the convergence rates of the three fuzzy adaptive equalizers are greatly improved. We observed that EKFAE achieves the fastest convergence and lowest steady-state BER compared to the LMS-based FAE and RLS-based FAE.

**Example 3** Finally, the bit error rate of EKFAE is compared to the maximum-a-posteriori probability (MAP) equalizer that has the minimum probability of error. Fig. 4 shows the bit error rates of the optimum MAP equalizer and the EKFAE for different signal to noises (background and impulsive) ratios. We see that the bit error rate of the EKFAE is very close to the optimal one.
V. CONCLUSION

In this paper, we have derived a fuzzy adaptive equalizer with extended Kalman filter adaptation (EKFAE) for powerline channel equalization. The merits of EKFAE as compared to two other fuzzy adaptive equalizers (RLS-based FAE and LMS-based FAE) are faster convergent speed and lower steady state BER. Simulation results show that EKFAE converges fastest without using any linguistic information. By incorporating some linguistic rules about the equalizer inputs into EKFAE, the adaptation speed is greatly improved. The steady state BER of EKFAE is very close to that of the optimal equalizer.

REFERENCE


