Parameter Estimation and Error Reduction for OFDM-Based WLANs

Jianhua Liu, *Student Member*, *IEEE*, and Jian Li, *Senior Member*, *IEEE*

Abstract—We consider parameter estimation and error reduction for orthogonal frequency-division multiplexing (OFDM) based high-speed wireless local area networks (WLANs). We devise or select algorithms that can provide benefit to the *overall* system performance and can be efficiently implemented in real-time. In particular, first, we give a channel model which is especially useful for assessing the channel parameter estimation methods devised for OFDM-based WLANs. Second, we provide a sequential method for the estimation of carrier frequency offset (CFO), symbol timing, and channel response by exploiting the structure of the packet preamble specified by the IEEE 802.11a standard. Finally, to correct the residue CFO induced phase error using the pilot tones, we consider maximum-likelihood phase tracking and least-squares phase fitting approaches; to improve the channel estimation accuracy using the decoded data, we present a semiblind channel estimation method; to mitigate the sampling clock induced time delay error, we provide a sampling clock synchronization approach that obviates the need of an automatic frequency control clock recovery circuit. The overall system performance of using our algorithms is demonstrated via several numerical examples.

Index Terms—OFDM, WLAN, channel model, symbol timing, carrier synchronization, channel estimation, pilot tones, sampling clock synchronization, WLAN standards.

1 Introduction

Orthogonal frequency-division multiplexing (OFDM) has been selected as the basis for several new high-speed wireless local area network (WLAN) standards [1], including IEEE 802.11a [2], IEEE 802.11g, and HIPERLAN/2. IEEE 802.11g and HIPERLAN/2 are very similar to IEEE 802.11a in terms of signal generation and detection/decoding. We use IEEE 802.11a to exemplify our presentation in this paper.

The OFDM-based WLAN system, as specified by the IEEE 802.11a standard, uses packet-based transmission. Each packet, as shown in Fig. 1, consists of an OFDM packet preamble, a SIGNAL field, and an OFDM DATA field. The packet preamble is used to determine the channel parameters, including the carrier frequency offset (CFO), symbol timing, as well as the channel response between the transmitter and the receiver. These parameters are needed to detect the data bits contained in the SIGNAL field and the OFDM DATA field.

We consider channel parameter estimation and error reduction for the OFDM-based WLANs. We devise or select algorithms that can help improve the *overall* system performance and can be efficiently implemented in real-time. The overall system performance of using our algorithms is demonstrated via numerical examples. The important issues that we address are stated as follows.

First, we consider the channel modeling problem. The channels for the OFDM signaling is often modeled as an FIR (finite impulse response) filter [3], [4]. We will clarify the fact, first revealed in [5], but largely overlooked ever since, that the FIR channel model is only an approximate model for the OFDM signaling. We consider a channel model useful for characterizing the true channels for the OFDM signaling. This channel model is especially useful for assessing the performance of the channel parameter estimation methods devised for OFDM-based WLANs.

Second, we consider the channel parameter estimation problem. Many methods have been proposed to estimate *one or two* of the aforementioned channel parameters. (See, for example, [6], [7], [8], [9], [10], [11], [12], [13], [14], and the references therein.) Among them, those discussed in [12], [13], [14] are particularly tailored to the preamble structure of the IEEE 802.11a standard. However, the methods given in [13] work well only for FIR channels but not for the true channels, especially at high signal-to-noise ratios (SNRs). The symbol timing method of [14] was proposed for Rician fading channels and can also encounter problems for the true channels. We give herein a sequential parameter estimation method that can be used to estimate *all* of the aforementioned channel parameters for the true channels. The sequential method fully exploits the structure of the packet preamble as specified by the IEEE 802.11a standard.

Finally, we consider the error reduction problem. First, we give a maximum-likelihood (ML) phase tracking approach using pilot tones to estimate (and correct) the residue *C*FO induced *p*hase *e*rror (CPE) for each received data OFDM symbol. Second, we supply a least-squares (LS) phase fitting approach using the estimated CPEs from each of the data OFDM symbols to improve the accuracy of the error estimates. Third, we present a semiblind method, which is a modified version of the one in [15], to improve the channel estimation accuracy using the decoded data iteratively. Fourth, we provide a sampling clock synchronization approach to mitigate the sampling clock induced

• *The authors are with the Department of Electrical and Computer Engineering, PO Box 116130, University of Florida, Gainesville, FL 32611. E-mail: {jhliu, li}@dsp.ufl.edu.*

Manuscript received 22 May 2003; accepted 21 Aug. 2003.
For information on obtaining reprints of this article, please send e-mail to: tmc@computer.org, and reference IEEECS Log Number TMC-0069-0503.

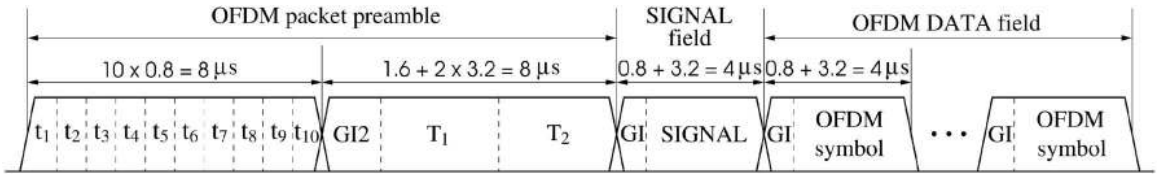

Fig. 1. The packet structure of the IEEE 802.11a standard.

time delay error. This synchronization approach is especially useful for the packet-based transmission and eliminates the need of using the AFC (automatic frequency control) clock recovery circuit.

The remainder of this paper is organized as follows: Section 2 gives a channel model for characterizing the true channels for the OFDM signaling and describes the data model for the OFDM-based WLANs. Section 3 presents the sequential parameter estimation method. The error reduction schemes are provided in Section 4. Numerical examples are presented in Section 5. Finally, we conclude in Section 6.

2 Data Model

In this section, we first give a brief overview of the IEEE 802.11a-based system. Then, we describe the data OFDM symbol generation. Next, we present a channel model for characterizing the true channels for the OFDM signaling. Finally, we give the data model for the OFDM-based WLANs.

2.1 IEEE 802.11a Standard

Fig. 1 shows the packet structure as specified by the IEEE 802.11a standard. The nominal bandwidth of the OFDM signal is 20 MHz and the in-phase/quadrature (I/Q) sampling interval t_S is 50 ns. The OFDM packet preamble consists of 10 identical short OFDM training symbols $\mathrm{t}_i, i = 1, 2, \ldots, 10$, each of which contains $N_C = 16$ samples and two identical long OFDM training symbols $\mathrm{T}_i, i = 1, 2$, each of which contains $N_S = 64$ samples. Between the short and long OFDM training symbols, there is a long guard interval (GI2) consisting of $2N_C = 32$ data samples. GI2 is the cyclic prefix (CP) for the long OFDM training symbol T_1, i.e., it is the exact replica of the last $2N_C$ samples of T_1.

The information carrying data are encoded in the OFDM DATA field. The binary source data sequence is first scrambled and then Convolutionally enCoded (CC) by an industrial standard constraint length 7, rate 1/2 encoder. The CC encoded output is then punctured according to the data rate requirement and is segmented into blocks of length N_{CBPS} (the number of coded bits per data OFDM symbol, which is determined by the transmission data rate), each of which corresponds to a data OFDM symbol. The binary data in each block is first interleaved among the subcarriers and then mapped (in groups of $\log_2 A$ bits) into A-QAM symbols, which are used to modulate the different data carrying subcarriers. Each data OFDM symbol in the OFDM DATA field employs $N_S = 64$ subcarriers, 48 of which are used for data symbols and four for pilot tones. There are also 12 null subcarriers with one in the center and the other 11 on the two ends of the frequency band. The data OFDM symbols, each of which consists of $N_S = 64$ samples, are obtained via taking IFFT (inverse fast Fourier transform) of the data symbols, pilot tones, and nulls on these N_S subcarriers. To eliminate the intersymbol interference (ISI), each data OFDM symbol is preceded by a CP or guard interval (GI), which contains the last N_C samples of the data OFDM symbol.

The SIGNAL field contains the information including the transmission data rate and data length of the packet. The information is encoded in 16 binary bits. There is also a reserved bit (for future use) and a parity check bit. These 18 bits, padded with six zeros, are then CC encoded (by the same encoder as for the OFDM DATA field) to obtain a 48 bit binary sequence. The CC encoded sequence is then interleaved and used to modulate the 48 data carrying subcarriers via BPSK. The SIGNAL field consists of 64 samples and is obtained via taking IFFT of these 48 BPSK symbols, four pilot tones, and 12 nulls. Also, there is a CP of length N_C to separate the preamble from the SIGNAL field.

2.2 Data OFDM Symbol Generation

Consider the generation of the kth data OFDM symbol in the OFDM DATA field shown in Fig. 1. Let

$$\mathbf{x}_k = [x_{k,1}\ x_{k,2}\ \ldots\ x_{k,N_S}]^T,$$

$k = 1, 2, \ldots, K$ be a vector of N_S symbols, where $(\cdot)^T$ denotes the transpose and x_{k,n_S}, $n_S = 1, 2, \ldots, N_S$ is the symbol modulating the n_Sth subcarrier and is equal to 0 for the null subcarriers, 1 or -1 for the pilot tones, or a member in a constellation $\mathcal{C}$ for data carrying subcarriers. Here, K is the number of data OFDM symbols in a packet and $\mathcal{C}$ is a finite constellation, such as BPSK, QPSK, 16-QAM, or 64-QAM. Let $\mathbf{W}_{N_S} \in \mathbb{C}^{N_S \times N_S}$ be the fast Fourier transform (FFT) matrix. Then, the kth data OFDM symbol $\mathbf{s}_k$ corresponding to $\mathbf{x}_k$ is obtained by taking the IFFT of $\mathbf{x}_k$. That is,

$$\mathbf{s}_k = \mathbf{W}_{N_S}^H \mathbf{x}_k / N_S, \tag{1}$$

where $(\cdot)^H$ denotes the conjugate transpose. To eliminate the intersymbol interference (ISI), each data OFDM symbol is preceded by a CP or guard interval (GI) formed using $\mathbf{s}_k$. The short and long OFDM training symbols in the packet preamble and the OFDM symbol in the SIGNAL field can be generated similarly. By stacking the packet preamble and the OFDM symbols (together with there corresponding CPs) in the SIGNAL and OFDM DATA fields, we obtain the entire packet $\mathbf{s} \in \mathbb{C}^{(5+K)(N_S+N_C)\times 1}$.

Passing $\mathbf{s}$ through a pair of D/A converters (for both the real and imaginary parts of $\mathbf{s}$) and the corresponding lower-pass filters, we obtain the (complex) analog waveform $s(t)$, as shown in Fig. 2.

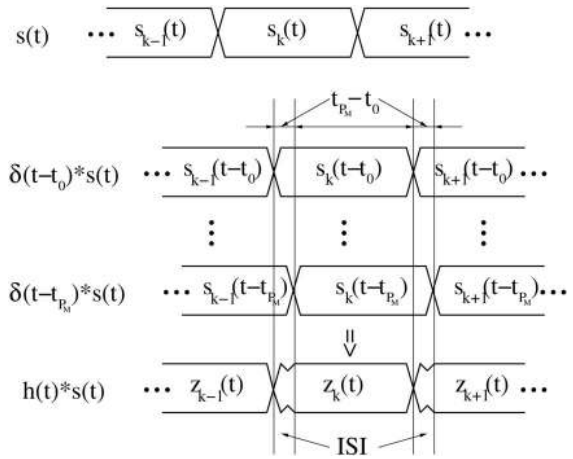

Fig. 2. Analog waveforms for data OFDM symbols and the multipath channel output.

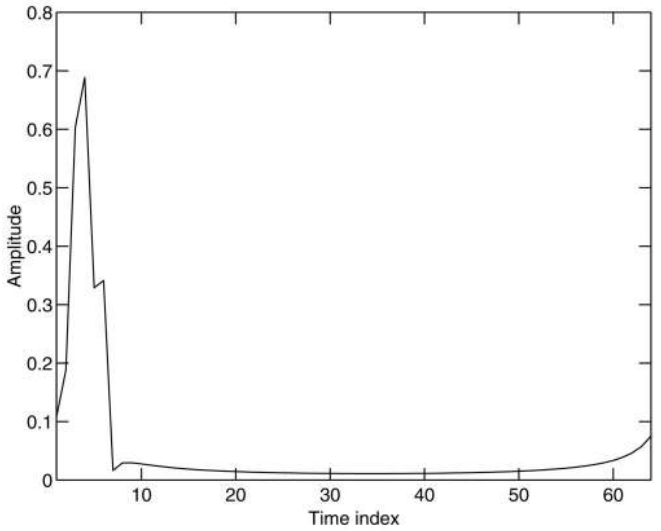

Fig. 3. Illustration of the equivalent discrete channel response for $h(t) = \delta(t - 2.5t_S) - 0.5\delta(t - 4.8t_S)$.

2.3 Channel Model

Let

$$h(t) = \sum_{p=0}^{P_M} \alpha_p \delta(t - t_p) \tag{2}$$

denote the (baseband) time-domain *analog* channel impulse response of a frequency-selective (multipath) time-invariant fading channel (called the *true channel* for short), where α_p and $t_p = \tau_p t_S$ ($\tau_0 \le \tau_1 \le \cdots \le \tau_{P_M}$, $t_S = 50$ ns) are the complex gain and time delay of the pth path, respectively. Then, the (baseband) input to the receiver is $z(t) = h(t) * s(t)$ (see Fig. 2). To detect the data symbols in each data OFDM symbol, we need to sample $z(t)$ (with sampling interval t_S) and process the sampled data in blocks of length N_S. Let

$$\mathbf{h}^{(\mathrm{t})} = [h_0^{(\mathrm{t})}\ h_1^{(\mathrm{t})}\ \ldots\ h_{N_S-1}^{(\mathrm{t})}]^T \tag{3}$$

be the corresponding time-domain *discrete* channel response (referred to as the *equivalent* discrete channel response of $h(t)$) for the sampled data blocks. Then, the lth element of $\mathbf{h}^{(\mathrm{t})}$, $l = 0, 1, \ldots, N_S - 1$ can be written as

$$h_l^{(\mathrm{t})} = \sum_p \alpha_p e^{-j\pi(l+(N_S-1)\tau_p)/N_S} \frac{\sin(\pi\tau_p)}{\sin(\pi(\tau_p - l)/N_S)}, \tag{4}$$

which includes the leakage effect due to the frequency-domain sampling [5]. Note that, by *equivalence*, we mean the frequency-domain response of (4) is the same as that of (2) on the N_S subcarriers, i.e., if

$$\mathbf{h} = \mathbf{W}_{N_S}\mathbf{h}^{(\mathrm{t})} \triangleq [h_1\ h_2\ \ldots\ h_{N_S}]^T$$

is the sampled frequency-domain channel response, then for $n_S = 1, 2, \ldots, N_S$,

$$h_{n_S} = \sum_p \alpha_p e^{-j\tau_p t_S \omega}\big|_{\omega=\frac{2\pi(n_S-1)}{N_S t_S}}. \tag{5}$$

We remark that the length of the equivalent discrete channel response is usually N_S (this is true even when we have only one path, but t_0 is not a multiple of t_S), as shown in Fig. 3. This length is much longer than N_C, the length of CP. However, this will not cause the ISI problem as long as $\tau_{P_M} - \tau_0 \le N_C - 1$ and we have the correct symbol timing. The reason is that the (sampled) receiver output due to the multipath channel is *not* the convolution of the equivalent discrete channel response in (4) with s; rather, it is due to the convolution between $h(t)$ and $s(t)$, as shown in Fig. 2.

The equivalent discrete channel response $\mathbf{h}^{(\mathrm{t})}$ is often approximated by an exponentially decaying FIR filter with length $L_F < N_S$ [4], denoted as

$$h_e(t) = \sum_{l_F=0}^{L_F-1} h_{l_F}^{(\mathrm{t})} \delta(t - l_F t_S), \tag{6}$$

where

$$h_{l_F}^{(\mathrm{t})} \sim \mathcal{N}\Big(0, \Big(1 - e^{-1/t_n}\Big)e^{-l_F/t_n}\Big), \tag{7}$$

with

$$\Big\{h_{l_F}^{(\mathrm{t})}\Big\}_{l_F=0}^{L_F-1}$$

being independent of each other, $t_n = t_r/t_S$, t_r being the root-mean-square (rms) delay spread of the frequency-selective fading channel, $L_F = \lceil 10t_n \rceil + 1$, and $\lceil x \rceil$ denoting the smallest integer not less than x. This channel model is sometimes referred to as the exponential channel model [3].

The exponential channel model is efficient for numerical simulations; however, we should not devise parameter estimation methods based on its limited length property, since, as was shown in (4), the FIR filter is only an approximation of the true channel for the OFDM signaling.

The direct simulation of (2) is difficult due to a potentially large P_M and the distributions of α_p. Instead, we modify (6) slightly to better characterize the true channel as follows:

$$h_m(t) = \sum_{l_F=0}^{L_F-1} h_{l_F}^{(\mathrm{t})} \delta(t - l_F t_S - t_{l_F}), \tag{8}$$

where $h_{l_F}^{(t)}$ is as in (7) and $t_{l_F}, l_F = 0, 1, \ldots, L_F - 1$, is uniformly distributed over $[0, t_S]$. We also assume that $\left\{h_{l_F}^{(t)}\right\}_{l_F=0}^{L_F-1}$ and $\{t_{l_F}\}_{l_F=0}^{L_F-1}$ are independent of each other. This modified exponential channel model is more realistic than the one in (6) due to the time delays $\{t_{l_F}\}_{l_F=0}^{L_F-1}$ introduced in (8).

2.4 Data Model

Consider now the received data model for the OFDM-based WLANs. Let $\mathbf{z}_k^{\text{ne}} \in \mathbb{C}^{N_S \times 1}$ be the noise-free and CFO-free received signal vector, obtained from sampling the baseband signal due to receiving $s_k(t)$, as shown in Fig. 2, with a correct symbol timing. (Note that N_C samples, which include the ISI, have been discarded when forming $\mathbf{z}_k^{\text{ne}}$.) Let $\mathbf{h} = [h_1\ h_2\ \ldots\ h_{N_S}]^T$ be the corresponding frequency-domain channel response (on the N_S subcarriers). The FFT output of the received data vector $\mathbf{z}_k = \mathbf{z}_k^{\text{ne}} + \mathbf{w}_k$, where $\mathbf{w}_k \sim \mathcal{N}(0, (\sigma^2/N_S)\mathbf{I}_{N_S})$ is the additive zero-mean white circularly symmetric complex Gaussian noise with variance σ^2/N_S, can be written as [4]

$$\begin{aligned}\mathbf{y}_k &= \mathbf{W}_{N_S}\mathbf{z}_k = \text{diag}\{\mathbf{h}\}\mathbf{x}_k + \mathbf{W}_{N_S}\mathbf{w}_k \\ &\triangleq \text{diag}\{\mathbf{h}\}\mathbf{x}_k + \mathbf{e}_k \in \mathbb{C}^{N_S \times 1},\end{aligned} \tag{9}$$

where $\text{diag}\{\mathbf{h}\}$ denotes the diagonal matrix formed from $\mathbf{h}$ and $\mathbf{e}_k \sim \mathcal{N}(0, \sigma^2\mathbf{I}_{N_S})$. The data model in (9) can also represent the OFDM symbols in the SIGNAL field and the packet preamble.

Equation (9) can also be written as

$$\mathbf{y}_k = \text{diag}\{\mathbf{x}_k\}\mathbf{h} + \mathbf{e}_k. \tag{10}$$

Note that (9) is useful for the data symbol (or data bit) detection whereas (10) can be used for the channel response estimation (with $\mathbf{x}_k$ being the known training data symbol vector, denoted as $\mathbf{x}_B$ later in the paper, for the case of channel estimation).

3 Sequential Estimation of CFO, Symbol Timing, and Channel Response

We now present the sequential CFO, symbol timing, and channel response estimation method which exploits the packet preamble structure for the OFDM-based WLAN system. The CFO can be estimated from the samples of two consecutive data blocks due to periodic inputs. Because of the fact that the CFO can be outside the unambiguous range measurable by the long OFDM training symbols, we have to estimate the CFO in two steps: a coarse CFO estimation using the short OFDM training symbols and then a fine CFO estimation, to determine the residue of the coarse CFO correction, using the long OFDM training symbols. After estimating and accounting for the CFO, we can obtain the symbol timing. We estimate the symbol timing also in two steps: the coarse symbol timing and fine symbol timing. The former is obtained by using the later portion of the short OFDM training symbols, which is suggested by the IEEE 802.11a standard for the symbol timing estimation and, hence, the coarse symbol timing can also be considered as the *standard symbol timing*. The fine symbol timing is obtained using the long OFDM training symbols. Finally, we obtain the channel estimate. The parameter estimates are obtained in the order presented below.

3.1 Coarse CFO Estimation

Let $z(l) = z^{\text{ne}}(l) + w(l)$ denote the lth time sample of the signal received from the receive antenna, starting from the time sample that the receiver AGC (automatic gain control) becomes stationary (the receiver AGC is assumed to become stationary at least before receiving the last two short OFDM training symbols and will remain stationary while receiving the remainder of the packet). The noise-free counterpart of $z(l)$ is $z^{\text{ne}}(l)$ and $w(l)$ is the additive zero-mean white circularly symmetric complex Gaussian noise. In the presence of CFO ϵ, we have [7]:

$$z^{\text{ne}}(l + N_C) = z^{\text{ne}}(l)e^{j2N_C\pi\epsilon}. \tag{11}$$

The correlation between two consecutive noise-free received data blocks, each with length N_C, can be written as

$$\begin{aligned}\sum_{l=k}^{k+N_C-1} z^{\text{ne}}(l)(z^{\text{ne}}(l + N_C))^* &= e^{-j2N_C\pi\epsilon}\sum_{l=0}^{N_C-1}|z^{\text{ne}}(l)|^2 \\ &\triangleq Pe^{-j2N_C\pi\epsilon},\end{aligned} \tag{12}$$

where $(\cdot)^*$ denotes the complex conjugate and k is any nonnegative integer such that $z^{\text{ne}}(k + 2N_C - 1)$ is a sample of the receive antenna output due to the input (transmitted signal) being a sample of the short OFDM training symbols of the packet preamble. Then, similarly, in the presence of noise, we have

$$\begin{aligned}P_S &= \sum_{l=0}^{N_C-1} z(l)z^*(l + N_C) \\ &= Pe^{-j2N_C\pi\epsilon} + e_P,\end{aligned} \tag{13}$$

where e_P is due to the noise. We calculate the coarse CFO as [12]

$$\hat{\epsilon}_C = -\frac{1}{2N_C\pi}\angle P_S, \tag{14}$$

where $\angle\, x$ denotes taking the argument of x.

We next correct the CFO using $\hat{\epsilon}_C$ to get the data samples $z^{(C)}(l)$ as follows:

$$z^{(C)}(l) = z(l)e^{-j2l\pi\hat{\epsilon}_C}. \tag{15}$$

In the sequel, we only consider the CFO corrected data as given above. For notational convenience, we drop the superscript of $z^{(C)}(l)$.

3.2 Coarse Symbol Timing Estimation

Now, we can use a correlation method, modified based on the one in [7], to estimate the coarse symbol timing. The symbol timing is referred to as the starting time sample due to the input being the long OFDM training symbol T_1. Once the starting time sample, due to the long OFDM training symbol T_1, is determined, we can determine the starting time sample due to every OFDM symbol thereafter. (According to the specification of the IEEE 802.11a standard and the sampling rate of 20 MHz, the true symbol timing T_0

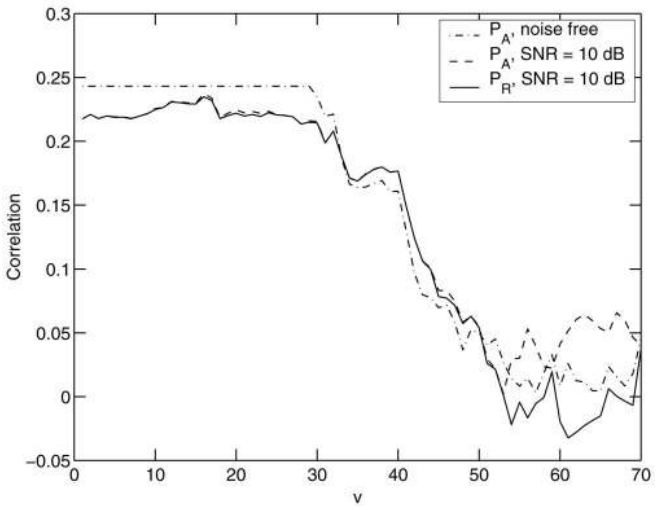

Fig. 4. Demonstration of different correlation sequences.

is 193, which comes from adding up the lengths of the 10 short OFDM training symbols and the GI2.)

From (13), we note that the correlation after the CFO correction, i.e., $P_S e^{j2N_C\pi\hat{\epsilon}_C}$, is approximately the real-valued scalar P plus the complex-valued noise. Hence, we propose herein to use the following *real-valued* correlation sequence for coarse symbol timing determination. We calculate the correlation sequence iteratively similar to the *complex-valued* approach in [7] as follows:

$$\begin{aligned} P_R(v+1) &= P_R(v) + \mathrm{Re}\{z(v+N_C)z^*(v+2N_C) \\ &\quad -z(v)z^*(v+N_C)\} \\ &= P_R(v) + \bar{z}(v+N_C)[\bar{z}(v+2N_C) - \bar{z}(v)] \\ &\quad + \tilde{z}(v+N_C)[\tilde{z}(v+2N_C) - \tilde{z}(v)]\}, \end{aligned} \tag{16}$$

where both $\mathrm{Re}(\cdot)$ and $\bar{(\cdot)}$ denote the real part of a complex entity and $\tilde{(\cdot)}$ stands for the imaginary part. We start the iteration by using $P_R(0) = \mathrm{Re}(P_S)$. Note that the new real-valued correlation approach given in (16) is superior to the absolute-valued one given in [7] since the former:

1. uses fewer computations,
2. lowers the noise level (noise variance reduced in half) in the correlation sequence, and
3. drops closer to zero as the sliding data blocks starting to slide out of the region due to the input being the short OFDM training symbols in the preamble.

The advantages of items 1 and 2 can be seen readily in (13) and (16) and the advantage of item 3 can be observed from Fig. 4, where $P_A(v)$ denotes the absolute-valued correlation sequence of [7].

When some of the data samples of the sliding data blocks are taken from the received data due to the input being GI2 or the long OFDM training symbols following the short OFDM training symbols, $P_R(v)$ will drop since (11) no longer holds. This property is used to obtain the coarse symbol timing. Let T_P denote the first time sample when $P_R(v)$ drops to below half of its peak value. We can have the coarse symbol timing estimate as:

$$T_C = T_P + \frac{3}{2}N_C + N_C. \tag{17}$$

The second term at the right-hand side of the above equation is due to the fact that $P_R(v)$ will drop to approximately one half of its maximum value when the data samples of the second half of the second of the two sliding blocks are due to the GI2 in the preamble; the third term is due to one half of the length of GI2. When $\tau_{P_M} - \tau_0 \leq N_C - 1$, only the first half of GI2 can suffer from ISI. Hence, our goal of coarse timing determination is to place the coarse timing estimate between the true timing $T_0 = 193$ and $T_0 - N_C = 177$ to make accurate fine CFO estimation possible. This explains why we use N_C instead of $2N_C$ for the third term in (17).

We have noticed that the coarse or standard symbol timing is not sufficiently accurate. Hence, a fine symbol timing estimation method is needed. Before we present our fine-timing estimation approach, let us consider the fine CFO estimation first, since a better corrected CFO can lead to a better fine symbol timing estimation.

3.3 Fine CFO Estimation

Similar to the coarse CFO estimation case, we calculate the correlation between the received signals due to the two long OFDM training symbols as follows:

$$P_L = \sum_{l=0}^{N_S-1} z(l+T_C)z^*(l+T_C+N_S). \tag{18}$$

Then, the fine CFO estimate can be computed as

$$\hat{\epsilon}_F = -\frac{1}{2N_S\pi}\angle P_L. \tag{19}$$

We can use $\hat{\epsilon}_F$ in the same way as $\hat{\epsilon}_C$ to correct the CFO. We assume that the data we use below have $\hat{\epsilon}_F$ corrected already.

3.4 Fine Symbol Timing Estimation

We now move on to obtain the fine symbol timing by using the long OFDM training symbols. The fine symbol timing is estimated by using a data block of length N_S, starting from the time sample $T_C + 3N_C$. With this choice, due to the fact that T_1 is identical to T_2, the data block is most likely due to the input being the second half of T_1 and the first half of T_2, even when the coarse symbol timing has a large error.

Let $\mathbf{y}_T$ denote the N_S-point FFT of the data block and $\mathbf{h}^{(\mathrm{t})}$ be the equivalent discrete channel response in the time-domain between the transmit and receive antennas. Then, by neglecting the existence of the residue CFO, $\mathbf{y}_T$ can be written as [cf. (10)]

$$\mathbf{y}_T = \mathbf{X}_B\mathbf{W}_{N_S}\mathbf{h}^{(\mathrm{t})} + \mathbf{e}_T, \tag{20}$$

where $\mathbf{X}_B$ is a diagonal matrix formed from $\mathbf{x}_B$, which is the training data symbol vector consisting of the 52 known BPSK symbols and 12 zeros, used to form the T_1 in Fig. 1. Since the Moore-Penrose pseudoinverse of $\mathbf{X}_B$ is $\mathbf{X}_B$ itself and $\mathbf{W}_{N_S}/N_S^{1/2}$ is unitary, we get an estimate of $\mathbf{h}^{(\mathrm{t})}$ as

$$\hat{\mathbf{h}}^{(\mathrm{t})} = \frac{1}{N_S}\mathbf{W}_{N_S}^H\mathbf{X}_B\mathbf{y}_T. \tag{21}$$

Let T_I denote the first index of the elements of $|\hat{\mathbf{h}}^{(\mathrm{t})}|$ that are above $1/3$ of the maximum value of the elements of $|\hat{\mathbf{h}}^{(\mathrm{t})}|$. (Our empirical experience suggests that selecting the threshold to be $1/3$ gives the best result.) Then, the fine symbol timing T_F is obtained as

$$T_F = T_C - N_C + T_I - 3. \tag{22}$$

The second term above is used to compensate for the aforementioned $3N_C$ shift due to the fact that $N_S - 3N_C = N_C$ and the last term above is chosen to be 3 to ensure that $T_F > T_0$ with negligible probability.

3.5 Channel Response Estimation

After we obtain T_F, we can now estimate the channel response. Let $\mathbf{y}_L$ denote the N_S-point FFT of the average of the two consecutive received data blocks, $\mathbf{z}_{\mathrm{T}_1}$ and $\mathbf{z}_{\mathrm{T}_2}$, each of which is of length N_S, due to the input being the two long OFDM training symbols. Let h denote the channel response from the transmit antenna to the receive antenna for the n_Sth subcarrier, $n_S = 1, 2, \ldots, N_S$ (we drop the dependence on n_S for notational convenience). Then, for the n_Sth subcarrier, we have

$$y_L = x_B h + e_L, \tag{23}$$

where x_B denotes the n_Sth diagonal element of $\mathbf{X}_B$, y_L denotes the n_Sth element of $\mathbf{y}_L$, and $e_L \sim \mathcal{N}(0, \sigma^2/2)$. (Here, the variance $\sigma^2/2$ is due to averaging the two consecutive data blocks.) We can readily have, from (23), that:

$$\hat{h} = x_B y_L. \tag{24}$$

We can readily see that the variance of the estimation error for h is $\sigma^2/2$. Moreover, $\hat{\sigma}^2$, the estimate of σ^2, can be easily obtained from the difference $\mathbf{z}_\Delta \triangleq \mathbf{z}_{\mathrm{T}_1} - \mathbf{z}_{\mathrm{T}_2}$. Let $\mathbf{y}_\Delta$ denote the FFT of $\mathbf{z}_\Delta$. Then, $\mathbf{y}_\Delta \sim \mathcal{N}(0, 2\sigma^2\mathbf{I}_{N_S})$. Let $\tilde{\mathbf{y}}_\Delta$ be a subvector of $\mathbf{y}_\Delta$ containing the elements on the 52 subcarriers other than the null subcarriers. We compute $\hat{\sigma}^2$ as

$$\hat{\sigma}^2 = \frac{1}{104}\|\tilde{\mathbf{y}}_\Delta\|^2, \tag{25}$$

where $\|\cdot\|$ denotes the Euclidean norm. This $\hat{\sigma}^2$ is needed later by the soft-detector.

4 Error Reduction

With the CFO estimated and accounted for as well as the symbol timing and channel response determined, we can proceed to detect the data bits contained in the OFDM DATA field. To improve the detection performance (which can lead to improved decoding performance), we consider several error reduction issues.

4.1 ML Phase Tracking

The pilot tones in the data OFDM symbols can be used to track the CPE. This should be performed before data bit detection. For the kth received data block $\mathbf{z}_k$ [cf. (9)], $k = 1, 2, \ldots, K$, let ϕ_k be the CPE, $\mathbf{p}_k$ denote the 4×1 vector containing the BPSK pilot bits in $\mathbf{x}_k$, and $\mathbf{y}_k^{(\mathrm{p})}$ denote the 4×1 data vector corresponding to the pilot carrying subcarriers in $\mathbf{y}_k$. Let $\hat{\mathbf{h}}^{(\mathrm{p})}$ be the 4×1 estimated channel response vector corresponding to the pilot carrying subcarriers. Then,

$$\mathbf{y}_k^{(\mathrm{p})} = e^{j\phi_k}\mathbf{P}_k\hat{\mathbf{h}}^{(\mathrm{p})} + \mathbf{e}_k^{(\mathrm{p})}, \tag{26}$$

where $\mathbf{P}_k = \mathrm{diag}\{\mathbf{p}_k\}$ and $\mathbf{e}_k^{(\mathrm{p})}$ denotes the corresponding zero-mean white circularly symmetric complex Gaussian noise vector. The ML estimate of ϕ_k is then computed as

$$\begin{aligned}\hat{\phi}_k &= \arg\min_{\phi_k}\left\|\mathbf{y}_k^{(\mathrm{p})} - e^{j\phi_k}\mathbf{P}_k\hat{\mathbf{h}}^{(\mathrm{p})}\right\|^2 \\ &= \angle\left\{\left(\hat{\mathbf{h}}^{(\mathrm{p})}\right)^H\mathbf{P}_k^H\mathbf{y}_k^{(\mathrm{p})}\right\}.\end{aligned} \tag{27}$$

This approach computes $\hat{\phi}_k$ from $\mathbf{z}_k$. We can compensate out $\hat{\phi}_k$ either in the time domain via $\mathbf{z}_k e^{-j\hat{\phi}_k}$ or in the frequency domain via $\mathbf{y}_k e^{-j\hat{\phi}_k}$.

After receiving the entire OFDM DATA field, we can improve the phase estimation accuracy by using the LS approach to fit a linear model to the phase estimates since ϕ_k is caused by the residue CFO and, hence, is a linear function of k in the absence of the oscillator phase noise. This LS approach still works well even in the presence of small to moderate oscillator phase noise, which is often the case in practice. However, the LS approach should be avoided when the oscillator phase noise is large due to the large error in the linear model.

4.2 LS Phase Fitting

First, we consider the case where all K data OFDM symbols in the OFDM DATA field are used for fitting the phase errors into a line. The phase error due to the residue CFO can be modeled as

$$\phi_k = \psi_1 + k\psi_2, \quad k = 0, 1, \ldots, K, \tag{28}$$

where $k = 0$ corresponds to the OFDM symbol in the SIGNAL field. We could have used an ML approach to obtain ψ_1 and ψ_2. However, the ML approach is computationally expensive and does not provide detection improvement over the following simple LS phase error fitting approach, which we prefer over the former. Let

$$\hat{\boldsymbol{\phi}} \triangleq \begin{bmatrix}\hat{\phi}_1 \\ \hat{\phi}_2 \\ \vdots \\ \hat{\phi}_K\end{bmatrix} \approx \begin{bmatrix}1 & 0 \\ 1 & 1 \\ \vdots & \vdots \\ 1 & K\end{bmatrix}\begin{bmatrix}\psi_1 \\ \psi_2\end{bmatrix} \triangleq \mathbf{D}\boldsymbol{\psi}, \tag{29}$$

where $\{\hat{\phi}_k\}_{k=0}^K$ are first obtained with the ML approach in (27) and then phase unwrapped before used in (29). The LS estimate of $\boldsymbol{\psi}$ is then given by:

$$\hat{\boldsymbol{\psi}} = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T\hat{\boldsymbol{\phi}}. \tag{30}$$

Once we obtain $\hat{\boldsymbol{\psi}}$, we remove the linear phase error $\hat{\boldsymbol{\phi}}_F = \mathbf{D}\hat{\boldsymbol{\psi}}$ from the data sequence in the OFDM DATA field.

Second, we consider the case where the kth data OFDM symbol and $K_S < K$ data OFDM symbols ahead of it are used to form $\hat{\boldsymbol{\phi}}$ to obtain an improved estimate of the CPE for the kth OFDM symbol. Here, we still estimate $\hat{\boldsymbol{\psi}}$ as shown in (30) (but with $\mathbf{D} \in \mathbb{R}^{(K_S+1)\times 2}$); however, we only use

$$\hat{\phi}_F = [1\ K_S]\hat{\boldsymbol{\psi}} \tag{31}$$

as the estimate of the CPE for the kth data OFDM symbol now. We can precompute $(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T$ to save computations. When $k < K_S$, we can pad zeros at the beginning of $\hat{\boldsymbol{\phi}}$ so as to use the same CPE estimation formula of (31).

While the first case can give better accuracy, the second case is more suitable for real-time implementations and is referred to as the real-time LS phase fitting approach.

4.3 Semiblind Channel Estimation

After decoding the entire packet, we can use the decoded data to improve the accuracy of the channel estimation in a semi-blind way, which, in turn, can lead to a better detection/decoding performance. The semiblind channel estimation can be performed in the following three steps:

First, we generate the estimated data symbols by using the same procedures of scrambling, convolutional encoding, interleaving, and symbol mapping as for the original data symbols, using the decoded data sequence (obtained via the Viterbi decoder).

Second, we determine whether or not the 48 estimated data symbols associated with $\mathbf{s}_k$, $k = 1, 2, \ldots, K$ will be used for the semiblind channel estimation. We add this selection step on the data OFDM symbol level for two reasons: 1) the decoding error for the convolutional codes tends to occur in bursts and 2) the interleaving is performed within a data OFDM symbol. Let x_k, $k = 1, 2, \ldots, K$ denote the data symbol on the n_Sth subcarrier, which is used to form the data OFDM symbol $\mathbf{s}_k$ in the OFDM DATA field. (For notational convenience, we again drop the dependence of our notation on n_S.) Let $\check{x}_k$ denote the estimate of x_k obtained from the decoded data sequence. We calculate the distance

$$d_k = |y_k - \hat{h}\check{x}_k|^2 \tag{32}$$

for the n_Sth subcarrier, where y_k is the n_Sth element of $\mathbf{y}_k$ in (9) and $\hat{h}$ is the channel estimate on the n_Sth subcarrier. We add up such distances for all 48 data carrying subcarriers and compare the sum with a threshold, which we choose to be $96\hat{\sigma}^2$ (i.e., twice the noise variance, $48\hat{\sigma}^2$) herein. If the sum is below the threshold, we use the estimated data symbol $\check{x}_k$ associated with $\mathbf{s}_k$ for the semiblind channel estimation; otherwise, we drop it from further considerations.

Finally, we estimate the channel response for each subcarrier in a semiblind way. Consider the n_Sth subcarrier. Let $\mathbf{y}_B$ be the column vector formed by stacking all the y_ks associated with the $\mathbf{s}_k$s that survive the aforementioned selection process. Let $\check{\mathbf{x}}_B$ be the vector formed from the corresponding $\check{\mathbf{x}}_k$s similarly to $\mathbf{y}_B$. We can also form vectors $\mathbf{y}_S$ and $\mathbf{x}_S$ similarly to $\mathbf{y}_B$ and $\check{\mathbf{x}}_B$ for the long OFDM training symbols. Then,

$$\mathbf{y}_{SB} \triangleq \begin{bmatrix} \mathbf{y}_S \\ \mathbf{y}_B \end{bmatrix} \approx \begin{bmatrix} \mathbf{x}_S \\ \check{\mathbf{x}}_B \end{bmatrix} h \triangleq \check{\mathbf{x}}_{SB} h, \tag{33}$$

from which we can obtain the semiblind estimate of h via the LS approach:

$$\hat{h}_{SB} = \left(\check{\mathbf{x}}_{SB}^H \check{\mathbf{x}}_{SB}\right)^{-1} \check{\mathbf{x}}_{SB}^H \mathbf{y}_{SB}. \tag{34}$$

Due to the structure of the preamble, it is easy to verify that

$$\mathbf{x}_S^H \mathbf{x}_S = 2. \tag{35}$$

Let

$$\beta \triangleq \check{\mathbf{x}}_B^H \check{\mathbf{x}}_B. \tag{36}$$

Then, we have

$$\begin{aligned} \hat{h}_{SB} &= \frac{1}{2+\beta}\left(\mathbf{x}_S^H \mathbf{y}_S + \check{\mathbf{x}}_B^H \mathbf{y}_B\right) \\ &= \frac{2}{2+\beta}\hat{h} + \frac{\beta}{2+\beta}\check{\mathbf{x}}_B^H \mathbf{y}_B, \end{aligned} \tag{37}$$

where $\hat{h}$ is the one estimated in (24).

Since it is more complicated to determine the accuracy of $\hat{h}_{SB}$ due to the statistical dependence of $\check{\mathbf{x}}_B$ and $\mathbf{y}_B$, we simply ignore the error in $\hat{h}_{SB}$ when using the soft-detector [cf (43) of the Appendix] and use $\hat{\sigma}^2/|\hat{h}|^2$ instead in the soft-detector. This approximation will not have any noticeable effect on the decoding performance since what is important for the soft-information is not the absolute SNR, $\hat{\sigma}^2/|\hat{h}|^2$, on each subcarrier, but the relative SNR among the subcarriers. We also remark that this semiblind approach can be used iteratively. However, our numerical examples indicate that the iteration is not needed due to negligible performance improvement.

Note that we did not use the SIGNAL field to improve the accuracy of the semiblind channel estimation since the improvement of using the SIGNAL field is negligible for large K. However, if one does not wish to experience the latency caused by using the OFDM DATA field, one can use the SIGNAL field instead. This is referred to as the real-time semiblind channel estimation method. For this approach, the data selection step is not needed due to the high detection accuracy as a result of the simple BPSK used in the SIGNAL field, as well as the high decoding accuracy due to the unpunctured channel coding.

4.4 Sampling Clock Synchronization

So far, we have assumed that the receiver sampling clock is synchronized with the transmitter symbol clock frequency. This assumption is valid if the AFC (automatic frequency control) clock recovery circuit suggested in the IEEE 802.11a standard can work well. Unfortunately, this circuit, which usually uses the delay-lock loop (DLL) [10], does not work well for packet-based short data sequences. We will address this problem by using a signal processing algorithm instead.

The sampling clock synchronization is different from the symbol timing in that the latter is concerned with determining the correct position to discard the GI, while the former focuses on aligning the receiver sampling clock to the transmitter symbol clock. The sampling clock synchronization is important since, without it, the WLAN performance will degrade severely.

The IEEE 802.11a standard suggests that, at the transmitter, both of the transmit center frequency and the symbol clock frequency be derived from the same reference oscillator. We suggest the same at the receiver, i.e., the sampling clock frequency and the receive center frequency be derived from the same reference oscillator. Then, each sample of the received signal will suffer from a delay of $f_S\epsilon/f_R$, where ϵ is the true CFO, f_S is the sampling frequency, and f_R is the receive center or carrier frequency. Both f_S and f_R are known to the receiver. Although this

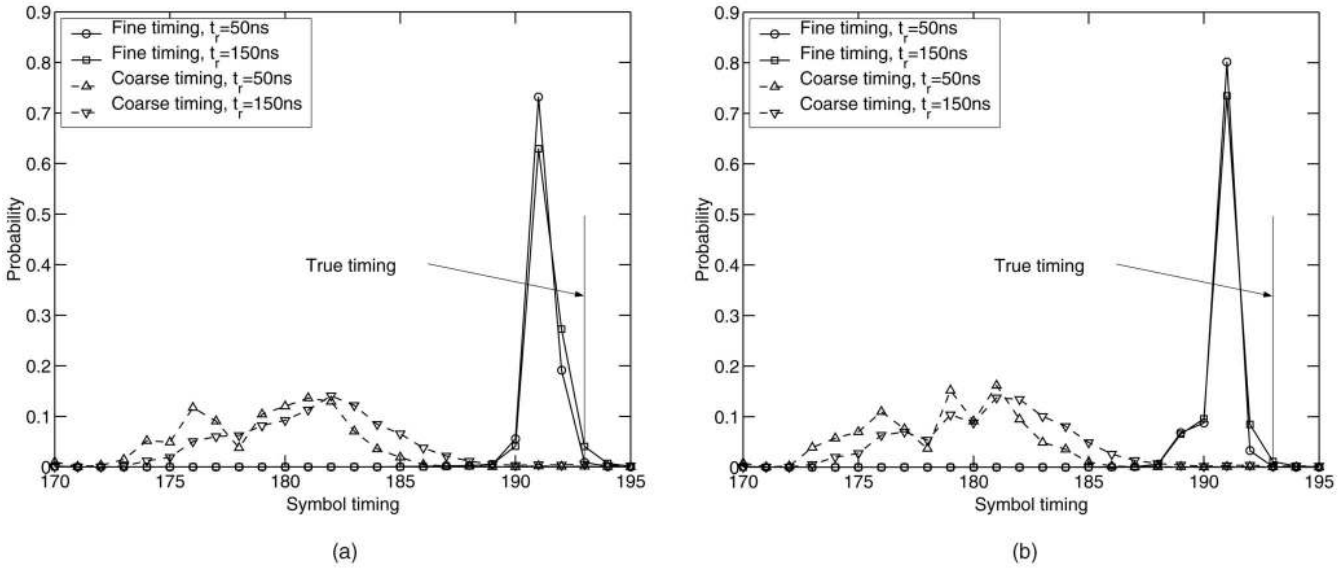

Fig. 5. Probability distribution (or histogram) of coarse and fine symbol timing estimates for different t_r when $\mathrm{SNR} = 10$dB for channels generated according to: (a) the modified exponential channel model and (b) the exponential channel model.

delay is too small to cause the intercarrier interference (ICI), it will accumulate over time. Let $N_T = N_S + N_C = 80$. The accumulated time delays for the samples of the kth data OFDM symbol are different but are approximately equal to

$$T_k = kN_T \frac{f_S \epsilon}{f_R}. \tag{38}$$

The time delays suffered by the samples of a data OFDM symbol will result in an approximately linear phase change across the subcarriers, which can drastically decrease the detection/decoding performance, as we will demonstrate using the numerical examples in the next section.

To address the above problem, we consider the following strategy: Let $\hat{T}_k$ denote the estimate of T_k with $\hat{T}_0 = 0$. For the kth data OFDM symbol, we first compensate out the time delay using $\hat{T}_{k-1}$ to achieve sampling clock synchronization. This approach should work well since T_{k-1} and T_k should be quite close to each other. Next, we compute $\hat{T}_k$ as follows: First, calculate the phase error $\hat{\phi}_k$ using the pilot tones using the method suggested in (27), second, unwrap the phase error $\hat{\phi}_k$, third, obtain an updated CFO estimate as

$$\hat{\epsilon}_I + \frac{\hat{\phi}_k}{2\pi N_T k},$$

where $\hat{\epsilon}_I$ denote the initial CFO estimate (which includes both the coarse and fine CFO estimates obtained from the packet preamble), and, finally, replace the ϵ in (38) using the updated CFO estimate to get

$$\hat{T}_k = \frac{f_S}{f_R}\left[kN_T\hat{\epsilon}_I + \frac{\hat{\phi}_k}{2\pi}\right]. \tag{39}$$

Note that, when we use the semiblind method to improve the channel estimation accuracy, we must first account for the phase change caused by T_k for the kth data OFDM symbol, $k = 1, 2, \ldots, K$, before doing the average in (37).

5 Numerical Examples

We provide several numerical examples to demonstrate the overall system performance of using our algorithms. In some of our simulation examples, we consider two transmission data rates: 12 Mbps (where the QPSK constellation is used and the channel coding rate is $R = 1/2$) and 54 Mbps (where the 64-QAM constellation is used and the channel coding rate is $R = 3/4$, which comes from puncturing the $R_C = 1/2$ convolutionally encoded sequence with the puncturing rate $R_P = 2/3$). Due to the fact that 52 out of 64 subcarriers are used in the OFDM-based WLAN system, the SNR used in this paper is defined as $52/(64\sigma^2)$ for the constellations whose average energies are normalized to 1. All simulation results are obtained with 10^4 Monte-Carlo trials.

Example 1. *Probability distribution (or histogram) of the coarse and fine symbol timing estimates for different t_rs.* The probability of the symbol timing estimation over channels generated according to the modified exponential and exponential models are shown in Figs. 5a and 5b, respectively. The two dashed curves in each figure show the probability distribution of the coarse symbol timing estimates for channels with $t_r =$ 50 and 150 ns at a low SNR = 10 dB. The two solid curves in the figures show the probability distribution of the fine symbol timing estimates. We can see that the simple fine symbol timing approach gives highly accurate timing estimates, compared to the coarse one, even at the low SNR of 10 dB. We can also see that the fine symbol timing estimation method, like the coarse symbol timing one, works equally well for both the channels.

Example 2. *Probability distribution of the coarse and fine symbol timing estimation for different SNRs.* Fig. 6a shows the probability distribution of the coarse symbol timing estimates when $t_r = 100$ ns, for $\mathrm{SNR} = 5$ and 10 dB.

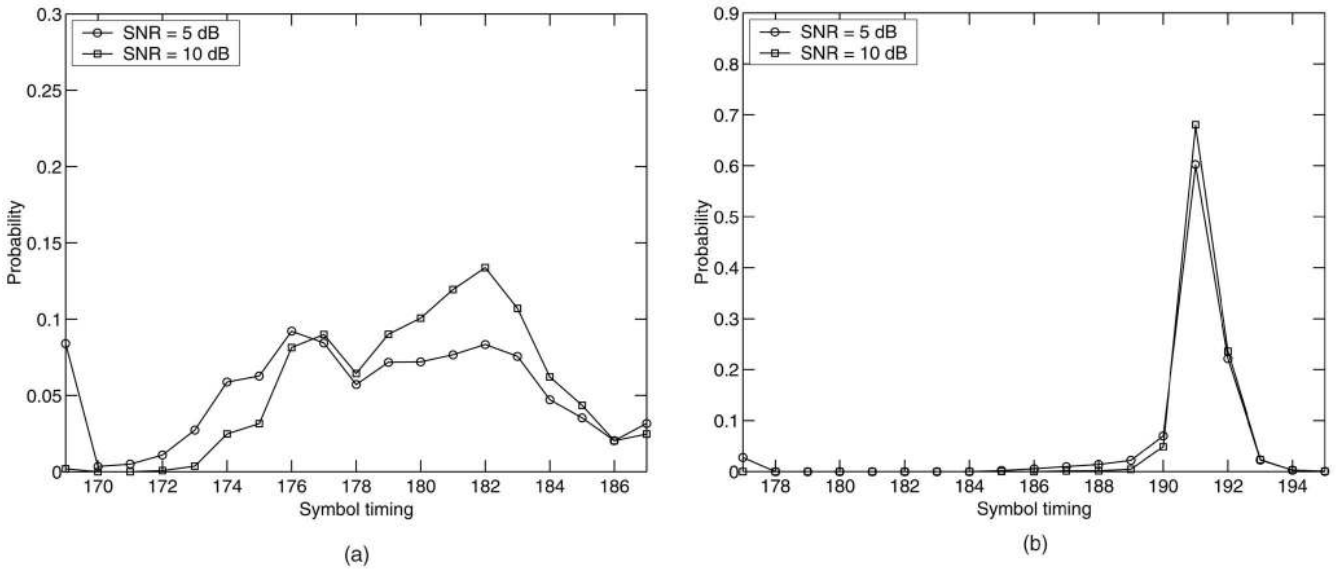

Fig. 6. Symbol timing estimates for different SNRs when $t_r = 100$ ns: (a) coarse symbol timing and (b) fine symbol timing.

Fig. 6b shows the probability distribution of the fine symbol timing estimates. (The probabilities at the two ends of the axes are actually the sum of all of the probabilities of the symbol timing estimates being equal to or beyond the ends.) We observe that:

- The accuracy of the coarse symbol timing estimates does not increase significantly with the increase of the SNR.
- Sufficient fine symbol timing accuracy can be achieved even when the SNR is as low as 5 dB.
- The fine symbol timing approach can tolerate a large coarse symbol timing error.

Example 3. *Overall system performance as a function of SNR with perfectly synchronized sampling clock.* This simulation example shows the overall performance, in terms of PER (packet error rate), of a few combinations of our algorithms. In our simulations, one packet consists of 1,000 bytes. According to the IEEE 802.11a standard, even if only one error occurs in a packet, the entire packet is considered to be wrong and is discarded. We consider the following five cases:

- Case 1. Estimated channel parameters (CFO, symbol timing, and channel response are all estimated) together with ML phase tracking using pilot tones;
- Case 2. Estimated channel parameters (the same as Case 1) with LS phase fitting using all data OFDM symbols in the OFDM DATA field;
- Case 3. Estimated channel parameters (the same as Case 1) with LS phase fitting (the same as Case 2) plus one iteration of the semiblind channel estimation method;
- Case 4. Estimated channel parameters (the same as Case 1) with LS phase fitting (the same as Case 2) plus two iterations of the semiblind channel estimation method;
- Case 5. Perfect channel knowledge (with CFO, symbol timing, and channel response all assumed known).

In Figs. 7a and 7b, we show the PER performance as a function of the SNR for $t_r = 100$ ns when the transmission data rates are 12 and 54 Mbps, respectively. We observe from the PER curves that:

- The sequential parameter estimation method is effective in that the PER curve with estimated channel parameters is reasonably close to that with perfect channel knowledge.
- LS phase fitting can outperform the ML phase tracking by as much as 1 dB for low SNRs; the reason is that the ML phase tracking is not very accurate at low SNRs.
- Semiblind channel estimation offers significant additional gain (around 1.5 dB) compared to the case of channel estimation by using the preamble only; however, more than one iteration is not needed.

Example 4. *PER as a function of SNR with unsynchronized sampling clock.* The simulation conditions are as follows:

1. the same as for Example 3 except for the sampling clock error,
2. $f_R = 5.25$ GHz, and
3. offsets of the reference oscillators at the transmitter and the receiver are identically, independently, and uniformly distributed over $[-20, 20]$ ppm (parts per million).

(The maximum reference oscillator tolerance suggested by the standard is ± 20 ppm for both the transmitter and the receiver.) We consider three cases here:

- Case i. Standard processing, i.e., the processing based on estimated channel parameters and tracked CPE, as in Case 1 of Example 3.

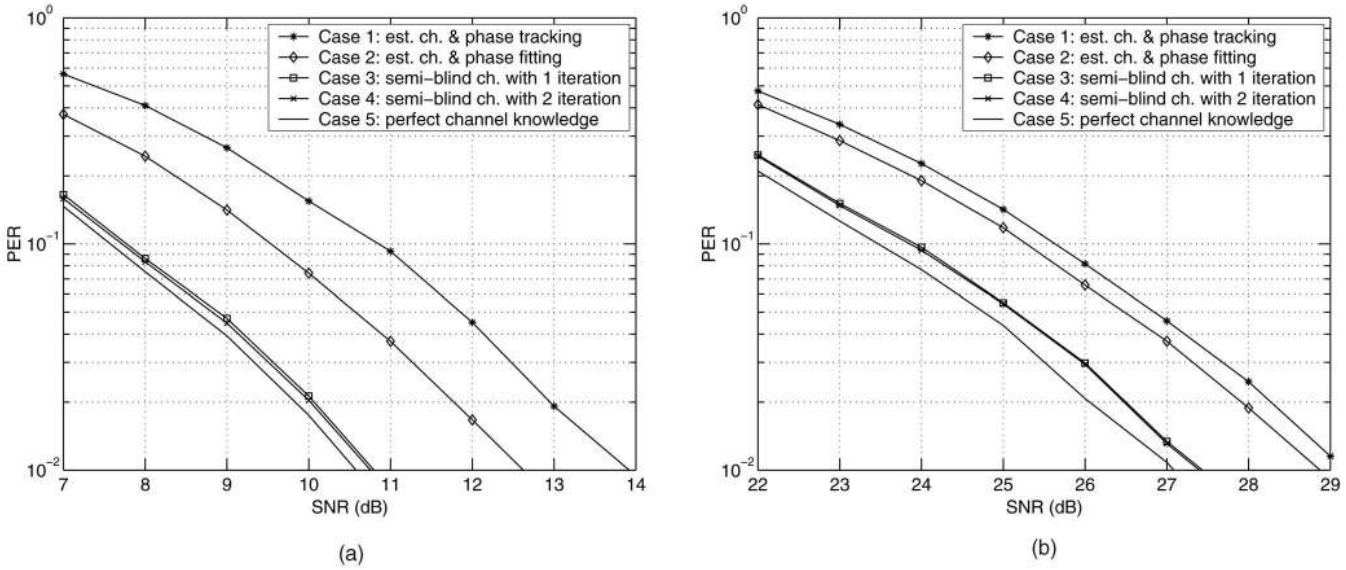

Fig. 7. PER versus SNR for t_r = 100 ns: (a) data rate = 12 Mbps and (b) data rate = 54 Mbps.

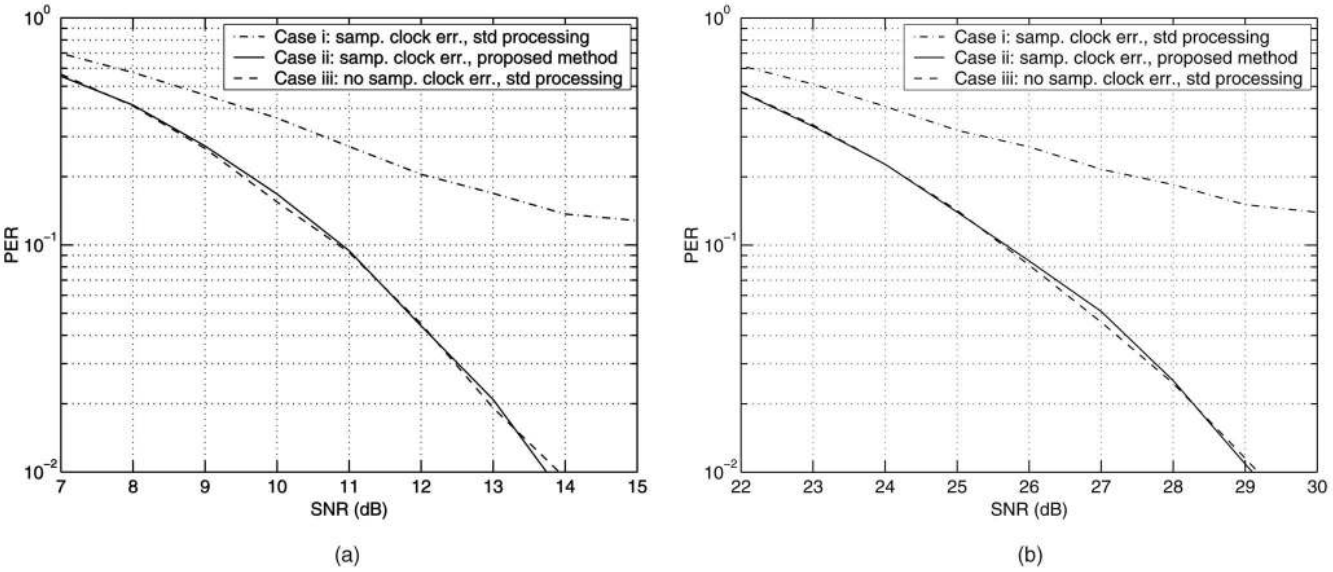

Fig. 8. PER versus SNR for $t_r = 100$ ns in the presence of sampling clock error: (a) data rate = 12 Mbps and (b) data rate = 54 Mbps.

- Case ii. Standard processing plus sampling clock synchronization.
- Case iii. No sampling clock error, i.e., the same as Case 1 of Example 3, included as a reference.

Figs. 8a and 8b show the PER curves for the above three cases as a function of the SNR for $t_r = 100$ ns when the transmission data rates are 12 and 54 Mbps, respectively. The following observations are immediate:

- The sampling clock error degrades the PER performance severely.
- The sampling clock synchronization approach works effectively in that it improves the PER significantly by pushing the PER curve toward the performance bound represented by the reference curve.

Example 5. *PER as a function of SNR for the real-time LS phase fitting approach and the real-time semiblind channel estimation method.* With the same simulation conditions as Example 4, we demonstrate the overall system performance using our algorithms without iterative processing. We consider four cases here:

- Case a. The same as Case ii of Example 4.
- Case b. Processing of Case a plus real-time LS phase fitting with $K_S = 9$.
- Case c. Processing of Case b plus real-time semiblind channel estimation.
- Case d. The same as Case 5 of Example 3.

Figs. 9a and 9b show the PER curves for the above four cases as a function of the SNR for $t_r = 100$ ns when the transmission data rates are 12 and 54 Mbps, respectively. Significant PER improvements can be observed from these figures, especially for the case of

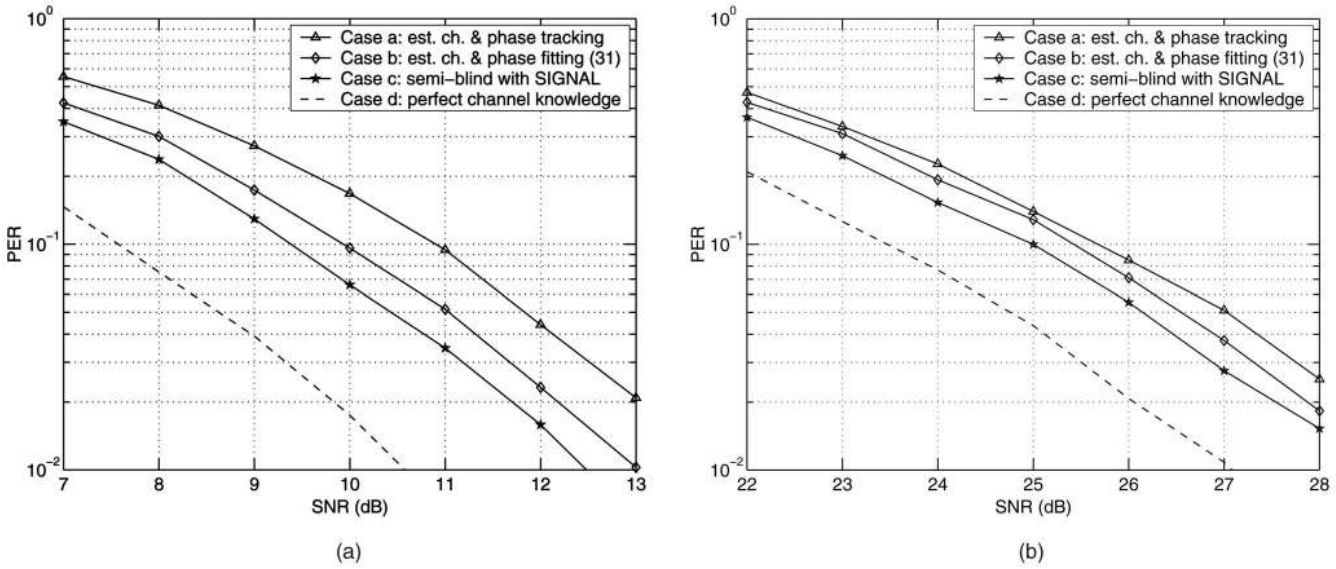

Fig. 9. PER versus SNR for t_r = 100 ns for real-time processing in the presence of sampling clock error: (a) data rate = 12 Mbps and (b) data rate = 54 Mbps.

lower data rate. The larger gaps between Cases c and d in Fig. 9 as compared to the gaps between Cases 3 and 5 in Fig. 7 are due to the poorer performance of the real-time semiblind method, which uses the SIGNAL field only.

Example 6. *Comparison of computational complexities.* As a final example, we compare the average MATLAB flops needed by an existing channel parameter estimation method in [13] and our sequential parameter estimation method to demonstrate the computational efficiency of our new method. Numerical results, as shown in Table 1, manifest that our new sequential method is about 70 times more efficient than the existing one.

6 Concluding Remarks

We have devised or selected channel parameter estimation and error reduction algorithms to improve the overall system performance for the OFDM-based WLANs and to be readily implemented in real-time. In particular, first, we have given a channel model which can be used to characterize the true channels for the OFDM signaling. Second, by exploiting the structure of the packet preamble specified by the IEEE 802.11a standard, we have provided a sequential method for the estimation of CFO, symbol timing, and channel response. Unlike some of the existing channel parameter estimation methods, our method works well for the true channels and is 70 times more efficient than an existing method. Finally, to correct the CPE using the pilot tones, we have considered ML phase tracking and LS phase fitting approaches. The proper phase fitting can yield a gain of up to 1.2 dB. Moreover, a semiblind channel estimation method has been presented to improve the channel estimation accuracy, which results in an additional gain of up to 1.5 dB. Furthermore, to correct the sampling clock error, we have provided a sampling clock synchronization approach that avoids the use of an AFC circuit. The algorithms given herein can be readily extended to the case of multiple receive antennas. We are currently implementing our algorithms on the DSP hardware.

TABLE 1
Comparison of the Average Numbers of Flops

	Existing method	Proposed method
Coarse timing	4,247	1,171
Fine timing	712,694	5,783
Channel est.	9,443	3,278
Overall	726,384	10,232

Appendix

A Simple Soft-Detector

Since we use the estimated channel for detection, a soft-detector should take into account the channel estimation error. Consider the n_Sth subcarrier after taking the FFT of the kth received data OFDM symbol. (For notational convenience, we again drop the dependence on n_S.) Then, the received data on the n_Sth subcarrier can be written as

$$y_k = hx_k + e_k. \tag{40}$$

Let

$$\hat{h} \triangleq h - h_\Delta. \tag{41}$$

Since only the estimated channel response is available to the detector, we have

$$\begin{aligned}\hat{x}_k &= \frac{1}{|\hat{h}|^2}\hat{h}^* y_k \\ &= x_k + \frac{1}{|\hat{h}|^2}\hat{h}^* h_\Delta x_k + \frac{1}{|\hat{h}|^2}\hat{h}^* e_k \\ &\triangleq x_k + n_k.\end{aligned} \tag{42}$$

Note that the error term n_k includes both the error due to the estimated channel and the perturbation due to the additive white Gaussian noise. By assuming statistical independence among x_k, h_Δ, and e_k and using the facts that $\mathrm{E}[x_k x_k^*] = 1$, $\mathrm{E}[e_k e_k^*] = \sigma^2$, and $\mathrm{E}[h_\Delta x_k x_k^* h_\Delta^*] = \sigma^2/2$, we have

$$\begin{aligned}\mathrm{E}[n_k n_k^*] &= \frac{1}{|\hat{h}|^4}\hat{h}^*\mathrm{E}[h_\Delta x_k x_k^* h_\Delta^*]\hat{h} + \frac{1}{|\hat{h}|^4}\hat{h}^*\mathrm{E}[e_k e_k^*]\hat{h} \\ &= \frac{\sigma^2}{2|\hat{h}|^4}\hat{h}^*\hat{h} + \frac{\sigma^2}{|\hat{h}|^4}\hat{h}^*\hat{h} \\ &= \frac{3\sigma^2}{2|\hat{h}|^2}.\end{aligned} \tag{43}$$

Keeping the values of $\hat{x}_k$ and using it together with $3\hat{\sigma}^2/(2|\hat{h}|^2)$ (see (25) on obtaining $\hat{\sigma}^2$) can provide the soft-detection output for the data symbol x_k. This output can be converted to the soft-information for each data bit in x_k [16], which is needed by the Viterbi decoder.

ACKNOWLEDGMENTS

This work was supported by the US National Science Foundation Grant CCR-0097114 and the Intersil Corporation Contract 2001056.

REFERENCES

[1] R. van Nee, G. Awater, M. Morikura, H. Takanashi, M. Webster, and K. Halford, "New High-Rate Wireless LAN Standards," *IEEE Comm. Magazine,* vol. 37, pp. 82-88, Dec. 1999.

[2] IEEE Standard 802.11a-1999, "Supplement to Information Technology—Telecomm. and Information Exchange between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: High Speed Physical Layer (PHY) in the 5 GHz Band," 1999.

[3] N. Chayat, "Tentative Criteria for Comparison of Modulation Methods," *doc: IEEE P802.11-97/96,* Sept. 1997.

[4] Z. Wang and G. Giannakis, "Wireless Multicarrier Communications," *IEEE Signal Processing Magazine,* vol. 17, pp. 29-48, May 2000.

[5] J.J. van de Beek, O. Edfors, M. Sandell, S.K. Wilson, and P.O. Borjesson, "On Channel Estimation in OFDM Systems," *Proc. IEEE Vehicular Technology Conf.,* vol. 2, pp. 815-819, July 1995.

[6] P.H. Moose, "A Technique for Orthogonal Frequency Division Multiplexing Frequency Offset Correction," *IEEE Trans. Comm.,* vol. 42, pp. 2908-2914, Oct. 1994.

[7] T.M. Schmidl and D.C. Cox, "Robust Frequency and Timing Synchronization for OFDM," *IEEE Trans. Comm.,* vol. 45, pp. 1613-1621, Dec. 1997.

[8] R. Negi and J. Coiffi, "Pilot Tone Selection for Channel Estimation in a Mobile OFDM System," *IEEE Trans. Consumer Electronics,* vol. 44, pp. 1122-1128, Aug. 1998.

[9] M. Morelli and U. Mengali, "An Improved Frequency Offset Estimator for OFDM Applications," *IEEE Comm. Letters,* vol. 3, pp. 75-77, Mar. 1999.

[10] B. Yang, K.B. Letaief, R.S. Chen, and Z. Cao, "Timing Recovery for OFDM Transmission," *IEEE J. Selected Areas in Comm.,* vol. 18, pp. 2278-2291, Nov. 2000.

[11] Y.H. Kim, I. Song, S. Yoon, and S.R. Park, "An Efficient Frequency Offset Estimator for OFDM Systems and Its Performance Characteristics," *IEEE Trans. Vehicular Technology,* vol. 50, pp. 1307-1312, Sept. 2001.

[12] J. Li, G. Liu, and G.B. Giannakis, "Carrier Frequency Offset Estimation for OFDM Based WLANs," *IEEE Signal Processing Letters,* vol. 8, pp. 80-82, Mar. 2001.

[13] E.G. Larsson, G. Liu, and J. Li, G.B. Giannakis, "Joint Symbol Timing and Channel Estimation for OFDM Based WLANs," *IEEE Comm. Letters,* vol. 5, pp. 325-327, Aug. 2001.

[14] K.-W. Yip, T.-S. Ng, and Y.-C. Wu, "Impacts of Multipath Fading on the Timing Synchronization of IEEE 802.11a Wireless LANs," *Proc. IEEE Int'l Conf. Comm., (ICC 2002),* vol. 1, pp. 517-521, 2002.

[15] E.G. Larsson, P. Stoica, and J. Li, "Orthogonal Space-Time Block Codes: ML Detection for Unknown Channels and Unstructured Interferences," *IEEE Trans. Signal Processing,* vol. 51, pp. 362-372, Feb. 2003.

[16] G. Caire, G. Taricco, and E. Biglieri, "Bit-Interleaved Coded Modulation," *IEEE Trans. Information Theory,* vol. 44, pp. 927-946, May 1998.

Jianhua Liu (S'01) received the BS degree in electrical engineering from Dalian Maritime University, Dalian, China, in 1984, the MS degree in electrical engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 1987, and the PhD degree in electronic engineering from Tsinghua University, Beijing, China, in 1998. From February 1987 to February 1999, he worked at the Communications, Telemetry, and Telecontrol Research Institute, Shijiazhuang, China, where he was an assistant engineer, engineer, senior engineer, and fellow engineer. From March 1995 to August 1998, he was also a research assistant at Tsinghua University. From February 1999 to June 2000, he worked at Nanyang Technonlgical University, Singapore, as a research fellow. Since June 2000, he has been a research assistant in the Department of Electrical and Computer Engineering at the University of Florida, Gainesville, working toward a PhD degree, majoring in electrical engineering and minoring in statistics. His research interests include wireless communications, statistical signal processing, and sensor array processing. He is a student member of the IEEE.

Jian Li (S'87-M'91-SM'97) received the MSc and PhD degrees in electrical engineering from The Ohio State University, Columbus, in 1987 and 1991, respectively. From April 1991 to June 1991, she was an adjunct assistant professor with the Department of Electrical Engineering, The Ohio State University, Columbus. From July 1991 to June 1993, she was an assistant professor with the Department of Electrical Engineering, University of Kentucky, Lexington. Since August 1993, she has been with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, where she is currently a professor. Her current research interests include spectral estimation, array signal processing, and their applications. Dr. Li is a member of Sigma Xi and Phi Kappa Phi. She received the 1994 US National Science Foundation Young Investigator Award and the 1996 US Office of Naval Research Young Investigator Award. She was an executive committee member of the 2002 International Conference on Acoustics, Speech, and Signal Processing. She has been an associate editor of the *IEEE Transactions on Signal Processing* since 1999 and an associate editor of the *IEEE Signal Processing Magazine* since 2003. She is presently a member of the Signal Processing Theory and Methods (SPTM) Technical Committee of the IEEE Signal Processing Society. She is a senior member of the IEEE.