

Available online at www.sciencedirect.com



Digital Signal Processing 14 (2004) 312–331 www.elsevier.com/locate/dsp

Blind equalization for time-varying channels and multiple samples processing using particle filtering $\stackrel{\text{$\approx$}}{=}$

Tadesse Ghirmai,^{a,*} Jayesh H. Kotecha,^b and Petar M. Djurić^a

 ^a Department of Electrical and Computer Engineering at Stony Brook University, Stony Brook, NY 11794, USA
 ^b Department of Electrical and Computer Engineering at University of Wisconsin–Madison, Madison, WI 53705, USA

Available online 23 December 2003

Abstract

In this paper we address the problem of equalization of time-varying frequency-selective channels. We formulate the problem by modeling the frequency-selective channel by an FIR filter with *time-varying tap weights* whose variation is characterized by an AR process. Our approach to the problem is based on Bayesian estimation using sequential Monte Carlo filtering commonly referred to as particle filtering. This estimation method represents the target posterior distribution by a set of random discrete samples and their associated weights. In this paper, we also extend the technique of equalization using particle filtering for cases where we have multiple samples per symbol and demonstrate that significant performance improvement can be achieved by processing multiple samples. The proposed algorithm is recursive and blind for it requires no training symbols for channel estimation. However, it assumes knowledge of the variance of the additive noise and the coefficients of the AR process used to model the variation of the fading channel tap weights. The proposed scheme is highly parallelizable and hence is suitable for VLSI (very large scale integration) implementation. © 2003 Elsevier Inc. All rights reserved.

Keywords: Equalization; Frequency-selective channels; Particle filtering

^{*} Corresponding author.

 $^{^{*}}$ This work is supported by the NSF award CCR-0082607.

E-mail addresses: tadesse@ece.sunysb.edu (T. Ghirmai), jkotecha@ece.wisc.edu (J.H. Kotecha), djuric@ece.sunysb.edu (P.M. Djurić).

^{1051-2004/\$ –} see front matter $\hfill \ensuremath{\mathbb{C}}$ 2003 Elsevier Inc. All rights reserved. doi:10.1016/j.dsp.2003.12.001

1. Introduction

Wide band mobile communication channels are generally considered as time-varying frequency-selective channels. Data transmitted over these channels undergo time and frequency spread causing serious impairments to the received signal. While time dispersion produces intersymbol interference (ISI) between the symbols, large frequency spread results in fast channel variation. Detection of data in such environment requires adaptive receivers with fast convergence. The design of this type of receivers is a challenging task and as a result, a considerable research effort has been directed towards it.

Most of the structures proposed in the literature for detection of data transmitted over frequency-selective channels employ adaptive maximum likelihood sequence detectors (MLSD) [1,2] based on the Viterbi algorithm (VA) [3]. In conventional MLSD, since the VA has an inherent decision delay, the metrics of the branches of its trellis have to be evaluated on delayed estimates of the channel parameters which are later updated according to the detected data. Such methods are not, however, suitable for fast fading channels because the detection of the data is based on outdated estimates of the channel impulse response. A more appropriate method for fast fading channels is a per-survival-processing MLSD (PSP-MLSD) [4,5], which avoids using delayed estimates of the channel for data detection by allowing each surviving branch of the trellis to update its own channel impulse response based on its hypothesized symbols. In this method, the channel estimates of the states of the trellis are updated using the least mean square (LMS) [6] method, the recursive least square (RLS) [7,8] method, or Kalman filters [9,10]. Alternative receivers for data detection in fading channels are the maximum a posteriori (MAP) receivers [11,12]. These methods use a single Kalman filter and the a posteriori probabilities (APPs) of the state of the ISI to estimate the channel. For fast fading channels, following a similar approach, a blind MAP equalizer with a bank of Kalman filters is proposed in [13].

Our approach to the problem is based on a Bayesian formulation in which we use a simulation-based recursive algorithm from the family of sequential Monte Carlo (SMC) methods also referred to as particle filtering. Sequential Monte Carlo filtering has been successfully applied in the past to flat fading channels [14–17]. In [18] and [19], the equalization problem has been tackled using a similar approach but addressed the problem for time-invariant channels and orthogonal frequency division multiplexing (OFDM) systems, respectively. Moreover, SMC methods are also applied to problems of data detection in other communications systems such as synchronous and asynchronous code division multiple access (CDMA) systems [17,20,21].

We model the frequency-selective channel using a finite impulse response (FIR) filter with time-varying tap weights, which are considered as randomly varying complex values whose magnitude is Rayleigh distributed. This type of variation of the tap weights can be characterized by an autoregressive (AR) process driven by a complex zero mean Gaussian noise. Such modeling of the channel allows to formulate the problem as a dynamic state space (DSS) system. The channel impulse response and the transmitted data are the unknown (hidden) states of the DSS and the received signal is the observation of the system. With this formulation, particle filtering can be applied to jointly estimate the channel and the transmitted data. The underlying idea of particle filtering consists of representing a probability distribution by a collection of properly weighted samples drawn from a proposal density. In our problem, we use particle filtering to represent approximately the joint posterior distribution of the channel vector and the transmitted data given all the available observations. MAP or minimum mean square error (MMSE) are used then to obtain estimates of the channel and the transmitted symbols.

Our algorithm is sequential and blind as it requires no training symbols. However, we assume the knowledge of the coefficients of the AR process used to model the time variation of the tap weights of the channel FIR filter and the variances of the complex additive noises. As described in [8], the assumption of knowing the AR coefficients, however, is not a serious limitation since they depend only on the fading rate (Doppler frequency shift and symbol rate) and the estimation of the AR coefficients, refer to [22,23] or [8] where a computer program is provided.

The main contributions of the paper are: the development of algorithms for detection of signals for time-varying frequency-selective fading channels and the extension of these methods for processing multiple samples per symbol for data detection using particle filtering. Parts of this work has been presented in [24].

The remaining of the paper is organized as follows. Section 2 describes the signal model. The state space formulation of the problem is presented in Section 3. A brief overview of particle filtering is provided in Section 4. The proposed algorithms are developed in Section 5. Simulations and results are presented in Section 6 and finally, conclusions are provided in Section 7.

2. System model

Figure 1 shows a block diagram of a baseband communication system. The input data sequence, consisting of a complex data symbols, b_i , that take values from a symbol set $\mathcal{B} = \{b_1, b_2, \dots, b_{|\mathcal{B}|}\}$, is applied to a pulse shaping filter, g(t). The output of the filter, s(t), is given by

$$s(t) = \sum_{m=1}^{M} b_m g(t - mT),$$
(1)

where *T* is the symbol period and *M* is the total number of symbols. This signal, s(t), is transmitted over a frequency-selective Rayleigh fading channel whose impulse response is denoted by $c(t, \tau)$ representing the response of the channel at time *t* for an impulse input applied at $t - \tau$. Frequency-selective Rayleigh fading channels disperse the transmitted signal both in time and frequency. The characteristics of such channels are usually modeled



Fig. 1. System model.



Fig. 2. An FIR representation of the channel model.

by an FIR filter with time-varying tap weights as shown in Fig. 2. Mathematically, we can express the impulse response of the channel as

$$c(t,\tau) = \sum_{l=0}^{d} c_l(t)\delta(\tau - lT_s),$$
(2)

where $c_l(t)$ are complex-valued tap weights whose magnitudes are randomly varying Rayleigh processes, $d + 1 = \lceil \tau_d / T_s \rceil + 1$ is the length of the FIR filter and τ_d is the maximum delay spread of the channel.

The received signal, r(t), is written as

$$r(t) = s(t) * c(t, \tau) + u(t),$$
(3)

where * represents the convolution operation and the additive term u(t) is a zero mean complex Gaussian noise with power spectrum density of N'_0 . We can also write r(t) as

$$r(t) = \sum_{m=1}^{M} b_m h(t, t - mT) + u(t) = z(t) + u(t),$$
(4)

where $h(t, t - mT) = c(t, \tau) * g(t - mT)$.

The transmitted signal, s(t), can be, generally, assumed to be bandlimited. Similarly, the signal component of the received signal, z(t), can be considered bandlimited having the same bandwidth as s(t) except for the slight expansion caused by the Doppler spread. We select the ideal low-pass filter (ILPF) to have equal bandwidth as z(t) which is *B* Hz. The output of the ILPF, y(t), can be written as

$$y(t) = z(t) + n(t), \tag{5}$$

where n(t) is a low-pass filtered additive Gaussian noise with a power spectrum density $N_0 = 2BN'_0$. The output of the low-pass filter is sampled at a rate of $1/T_s = 2B$ which is conveniently chosen to be an integer multiple of the symbol rate, $T = \alpha T_s$, where α is a positive integer. Depending on the length of the delay spread of the channel, the period of the data symbols, and the length of the truncated impulse response of the pulse shaping filter, the received sample at time *k* is correlated with only a few past samples. If *L* denotes



Fig. 3. Generation of the tap weights of a Rayleigh fading channel and the received signal.

the number of past symbols correlated with the *k*th sample, then the sampled signal, y_k , is given by

$$y_k = \sum_{i=0}^{L} b_{\lceil k/\alpha \rceil - i} h_{k,i} + n_k, \tag{6}$$

where n_k is a complex uncorrelated zero mean Gaussian sequence with variance $\sigma_n^2 = 2BN'_0$ and the operator $\lceil \cdot \rceil$ represents the smallest integer greater than or equal to (·).

Fading channels are usually considered as wide stationary unscattering processes [25] and their theoretical power spectrum of the complex envelope of a received signal over a Rayleigh fading channel is given by [26]

$$S(f) = \begin{cases} \frac{\sigma^2}{2\pi f_d \sqrt{1 - (f/f_d)^2}}, & |f| < f_d, \\ 0, & \text{otherwise,} \end{cases}$$
(7)

where σ^2 is the root mean square (rms) value of the signal envelope and f_d is the maximum Doppler shift corresponding to the speed of the receiver. The value of f_d is calculated as v/λ , where v is the speed of the vehicle (receiver) and λ is the wavelength of the carrier frequency.

The simplest method to simulate a frequency-selective fading channel is to amplitude modulate the transmitted signal by low-pass filtered complex Gaussian noise process as shown in Fig. 3. The response of the low-pass fading filter characterizes the fading process of the channel. The spectral density of the simulated received signal is determined by the transfer function of the fading filter. To obtain a received signal with spectral density as in (7), the fading filter has to be designed so that its transfer function is proportional to the square root of S(f). Following [8], we approximate the fading filter as an infinite impulse response (IIR) filter whose coefficients are a function of the f_d and the symbol period T. Such approximation allows to model the time-variation of the tap weights of the channel

filter by an AR process driven by a zero mean complex Gaussian noise. In our formulation, we model the time variation of the channel by a second order AR process,

$$h_{k,i} = \gamma_1 h_{k-1,i} + \gamma_2 h_{k-2,i} + \nu_{k,i}, \tag{8}$$

where $h_{k,i}$ represents the coefficient of the *i*th tap of the filter at time instant k and $v_{k,i}$ is a zero mean Gaussian noise. As described earlier, the coefficients γ_1 , γ_2 and the variance of the noise depend only on f_d and the symbol period, T [8], and, thus their estimation can be incorporated as part of the receiver.

3. The state-space model

The signal model described in the previous section can be formulated by a DSS model. In this section, we restrict our discussion to a single sample per symbol case for which the model is

$$\mathbf{x}_{k} = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{D}\mathbf{v}_{k},$$

$$y_{k} = \mathbf{b}_{k}^{\top}\mathbf{x}_{k} + n_{k},$$
(9)

where \mathbf{x}_k and y_k denote the channel state vector and the received signal at time *k*, respectively. It is to be noted that the measurement equation of (9) is identical to (6) with $\mathbf{b}_k = (b_k, b_{k-1}, \dots, b_{k-L}, 0, 0, \dots, 0)^{\top}$. The channel state vector, \mathbf{x}_k has a dimension of $2(L+1) \times 1$ and is defined as

$$\mathbf{x}_{k}^{\top} = \left(\mathbf{h}_{k}^{\top} \mathbf{h}_{k-1}^{\top}\right),\tag{10}$$

where \mathbf{h}_k is an $(L+1) \times 1$ vector given by $\mathbf{h}_k = (h_{k,0}, h_{k,1}, \dots, h_{k,L})^{\top}$.

Both matrices **A** and **D** have the size of $2(L + 1) \times 2(L + 1)$ and $2(L + 1) \times (L + 1)$, respectively, and are defined by

$$\mathbf{A} = \begin{bmatrix} \gamma_1 \mathbf{I} & \gamma_2 \mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \quad \text{and} \quad \mathbf{D} = \begin{pmatrix} \varepsilon \mathbf{I} \\ \mathbf{0} \end{pmatrix},$$

where **I** and **0** are identity and zero matrices with sizes $(L + 1) \times (L + 1)$. Note that v_k is an $(L + 1) \times 1$ zero mean white Gaussian noise vector whose covariance matrix is equal to **I**. The coefficient ε in the matrix **D** represents a fractional power of the received signal for each lag, which in this case is assumed to be identical for all the lags.

Our objective is to jointly estimate the state of the channel \mathbf{x}_k and the transmitted symbols \mathbf{b}_k given the received signal y_k .

4. A brief overview of sequential Monte Carlo filtering

Our approach to the solution of the problem is based on a recursive algorithm known as sequential Monte Carlo (SMC) or particle filtering. Consider the following general DSS model:

$$\mathbf{x}_{k} = f(\mathbf{x}_{k-1}, \mathbf{u}_{k}),$$

$$\mathbf{y}_{k} = g(\mathbf{x}_{k}, \mathbf{v}_{k}),$$
 (11)

where $f(\cdot)$ and $g(\cdot)$ are the state and observation functions, respectively, \mathbf{x}_k is a hidden state vector at time k, \mathbf{y}_k denotes the observed value of the model, \mathbf{u}_k and \mathbf{v}_k represent the state and observed noises, respectively. The objective is to estimate $\mathbf{x}_{1:k}$ sequentially based on the observation $\mathbf{y}_{1:k}$. Note that the notation $\mathbf{y}_{1:k}$ represents the set of vectors $\{\mathbf{y}_1, \dots, \mathbf{y}_k\}$. From Bayesian perspective, all the information about the state $\mathbf{x}_{1:k}$ is contained in the posterior density $p(\mathbf{x}_{1:k} | \mathbf{y}_{1:k})$. Thus, our interest is to evaluate the posterior density and its corresponding expectations with respect to the posterior $p(\mathbf{x}_{1:k} | \mathbf{y}_{1:k})$ such as $E_p(h(\mathbf{x}_{1:k}) |$ $\mathbf{y}_{1:k})$, where $h(\mathbf{x}_{1:k})$ is arbitrary function. If the state and observation functions, $f(\cdot)$ and $g(\cdot)$, of the DSS model are linear and the state and observation noises are Gaussian, then the posterior distribution can be exactly determined using the Kalman filter. If the system is, however, nonlinear and/or non-Gaussian, analytical evaluation is, in most cases, not possible. In such cases we resort to simulation-based methods such as particle filtering.

The basic concept of Monte Carlo filtering is to approximately represent the posterior density with properly weighted samples (particles) drawn from a proposal distribution, $\pi(\mathbf{x}_{1:k} | \mathbf{y}_{1:k})$, also called importance function. If *N* trajectories, $\mathcal{X}_k = {\{\mathbf{x}_{1:k}^{(1)}, \dots, \mathbf{x}_{1:k}^{(N)}\}}$, are drawn from this proposal distribution, the posterior density can be approximated as

$$p(\mathbf{x}_{1:k} | \mathbf{y}_{1:k}) \approx \sum_{i=1}^{N} \tilde{w}_{k}^{(i)} \delta(\mathbf{x}_{1:k} - \mathbf{x}_{1:k}^{(i)}),$$
(12)

where $\delta(\cdot)$ is the Dirac delta function and, $\tilde{w}_k^{(i)}$, are weights obtained from

$$w_k^{(i)} = \frac{p(\mathbf{x}_{1:k}^{(i)} \mid \mathbf{y}_{1:k})}{\pi(\mathbf{x}_{1:k}^{(i)} \mid \mathbf{y}_{1:k})},$$
(13)

where the weights $w_k^{(i)}$ are normalized to $\tilde{w}_k^{(i)}$ such that $\sum_{i=1}^N \tilde{w}_k^{(i)} = 1$. Expectations can then be estimated by

$$E_p(h(\mathbf{x}_{1:k}) | \mathbf{y}_{1:k}) \approx \sum_{i=1}^N h(\mathbf{x}_{1:k}^{(i)}) \tilde{w}_k^{(i)}.$$
(14)

It has been shown that the estimate in (14) is unbiased and, according to the strong law of large numbers, as the number of particles approaches infinity it converges almost surely towards the true expectation, $E(h(\mathbf{x}_{1:k}))$ [27]. Sequential importance sampling (SIS) is a popular recursive Monte Carlo filtering algorithm which allows a sequential approximation of the posterior densities by propagating particles and updating their corresponding weights as new data become available. Suppose at time k we represent the density $p(\mathbf{x}_{1:k} | \mathbf{y}_{1:k})$ by N trajectories of particles $\{\mathbf{x}_{1:k}^{(i)}\}_{i=1}^N$ and their associated weights $\{\tilde{w}_k^{(i)}\}_{i=1}^N$. When a new observation \mathbf{y}_{k+1} arrives, a new set of particles, $\mathbf{x}_{k+1}^{(i)}$, are generated and the weights are updated, $\tilde{w}_{k+1}^{(i)}$, so that $\{\mathbf{x}_{1:k+1}^{(i)}, \tilde{w}_{k+1}\}_{i=1}^N$ represents the density $p(\mathbf{x}_{1:k+1} | \mathbf{y}_{1:k+1})$. The procedure has the following three steps:

(1) Generating particles from $\mathbf{x}_{k+1}^{(i)} \sim \pi(\mathbf{x}_{k+1} | \mathbf{x}_{1:k}^{(i)}, \mathbf{y}_{1:k+1})$ for i = 1, 2, ..., N.

(2) Updating the weights for i = 1, 2, ..., N as

$$w_{k+1}^{(i)} \propto w_k^{(i)} rac{p(\mathbf{y}_k \mid \mathbf{x}_k^{(i)}) p(\mathbf{x}_{k+1}^{(i)} \mid \mathbf{x}_k^{(i)})}{\pi(\mathbf{x}_{k+1}^{(i)} \mid \mathbf{x}_{1:k}^{(i)}, \mathbf{y}_{1:k+1})}.$$

(3) Normalizing the weights

$$\tilde{w}_{k+1}^{(i)} = \frac{w_{k+1}^{(i)}}{\sum_{j=1}^{N} w_{k+1}^{(j)}}.$$

Once N particles are collected, estimates of unknowns can readily be obtained.

For efficient estimation of the state vector or its function, the choice of the proposal distribution $\pi(\mathbf{x}_{k+1} | \mathbf{x}_{1:k}^{(i)}, \mathbf{y}_{1:k+1})$ is critical. The proposal distribution $\pi(\mathbf{x}_{k+1} | \mathbf{x}_{1:k}^{(i)}, \mathbf{y}_{1:k+1}) = p(\mathbf{x}_{k+1} | \mathbf{x}_{1:k}^{(i)}, \mathbf{y}_{1:k+1})$ is optimal in the sense that it minimizes the relative variation of the weights resulting in minimum variance of the estimates. Although this proposal density is the best choice, in most cases drawing samples from it is difficult. In practice, the most important factor dictating the choice of the proposal distribution depends on the ease of drawing samples. Several forms of proposal distributions have been used in the literature.

In the SIS algorithm, as the particles propagate in time, the variance of the weights can only increase [28,29]. In practice, after a short time run, only a few of the particles have a significant weight while most of the other particles have weights which are approximately equal to zero. Unless such degeneracy is addressed, a lot of computational power is wasted in updating the trajectories of those particles which have insignificant contribution to the final estimate. One method of dealing with the problem of degeneracy is to apply a procedure called resampling. Resampling simply eliminates samples with small weights and replicates the ones with larger weights. Several resampling schemes have been proposed in the literature [28,30,31]. Resampling is applied periodically or at systematically determined instants. Liu and Chen [30] introduced a measure known as *effective sample size*, which is computed as

$$N_{\rm eff} = \frac{1}{\sum_{i=1}^{N} (\tilde{w}_k^{(i)})^2}$$
(15)

and proposed to apply a resampling procedure whenever the effective sample size drops below a certain predefined threshold value.

For our problem, we are interested in determining the joint posterior distribution, $p(\mathbf{b}_{1:k}, \mathbf{x}_{1:k} | y_{1:k})$, which we use to estimate the transmitted symbols and the channel vector. We apply the SIS algorithm recursively and approximate the joint posterior distribution by appropriately weighted particles, $\{(\mathbf{b}_{1:k}^{(i)}, \mathbf{x}_{1:k}^{(i)}), \tilde{w}_k^{(i)}\}_{i=1}^N$. In the following section, we develop a particle filtering algorithm for this purpose.

5. Algorithms for data detection and channel estimation over frequency-selective fading channels

Our objective is to sequentially determine the vectors \mathbf{b}_k and \mathbf{x}_k given the observations $y_{1:k}$. To do so, we employ the strategy of Rao–Blackwellization [20,28,32] or, as described in [33], mixture Kalman filtering (MKF). Note that if we are able to determine the transmitted symbol vector $\mathbf{b}_{1:k}$, which has a discrete support set, then it is possible to obtain optimal estimates of the channel vector, $\mathbf{x}_{1:k}$, using the Kalman filter. Hence, for the problem of joint estimation of the channel and detection of transmitted symbols, we can apply particle filtering to approximate the posterior distribution of the transmitted symbols, and employ a bank of Kalman filters to determine the distribution of the channel vector. We start by expressing the joint posterior of $\mathbf{b}_{1:k}$ and $\mathbf{x}_{1:k}$ by

$$p(\mathbf{b}_{1:k}, \mathbf{x}_{1:k} \mid y_{1:k}) = p(\mathbf{x}_{1:k} \mid \mathbf{b}_{1:k}, y_{1:k}) p(\mathbf{b}_{1:k} \mid y_{1:k}).$$
(16)

The first factor on the right hand side (r.h.s.) of the above expression can be computed by Kalman filters, while the second factor can be determined using a particle filter. This results in an approximation of the joint posterior distribution by a mixture of Gaussians. Once the joint posterior distribution is obtained, the MAP or the MMSE estimator can be applied to compute the estimates of the transmitted symbols and the channel.

In the sequel, we derive algorithms that approximate the posterior distribution of the transmitted symbols using particle filtering for two different proposal densities—prior and optimal proposal densities. At the end of the section, we extend the same algorithms for multiple samples per symbol.

5.1. Prior proposal density

Consider the posterior distribution $p(b_{1:k}^{(i)} | y_{1:k})$. This posterior distribution can be rewritten as

$$p(b_{1:k}^{(i)} | y_{1:k}) = \frac{p(y_k | y_{1:k-1}, b_{1:k}^{(i)}) p(b_k^{(i)} | b_{1:k-1}^{(i)}, y_{1:k-1}) p(b_{1:k-1}^{(i)} | y_{1:k-1})}{p(y_k | y_{1:k-1})} \\ \propto p(b_k^{(i)} | b_{k-1}^{(i)}) p(y_k | y_{1:k-1}, b_{1:k}^{(i)}) p(b_{1:k-1}^{(i)} | y_{1:k-1}),$$
(17)

where the first factor on the r.h.s. of (17) is the prior distribution of the transmitted symbol b_k . If no coding is applied, the prior probability of b_k is simply $p(b_k)$. We consider the proposal distribution

$$\pi\left(b_k \mid b_{1:k-1}^{(i)}, y_{1:k}\right) = p\left(b_k \mid b_{1:k-1}^{(i)}\right) = p(b_k).$$
(18)

Since the support set of the transmitted symbols is discrete, we can easily draw samples from this distribution. The associated weight of the particles can then be updated by the second factor on the r.h.s. of (17), $p(y_k | y_{1:k-1}, b_{1:k}^{(i)})$, which is readily obtained by using (17) and (13) as follows:

~~

$$w_{k}^{(i)} = w_{k-1}^{(i)} \frac{p(b_{1:k}^{(i)} | y_{1:k})}{\pi(b_{k}^{(i)} | b_{1:k-1}^{(i)}, y_{1:k})p(b_{1:k-1}^{(i)} | y_{1:k-1})}$$

= $w_{k-1}^{(i)} p(y_{k} | y_{1:k-1}, b_{1:k}^{(i)}),$ (19)

where the density $p(y_k | y_{1:k-1}, b_{1:k}^{(i)})$ is the predictive density of the observed signal y_k . It can be shown that this density is a complex Gaussian distribution given by

$$p(y_k | y_{1:k-1}, b_{1:k}^{(i)}) = \mathcal{N}_c(y_k; \bar{y}_{k|k-1}^{(i)}, \sigma_{k,y}^{2(i)}),$$
(20)

where $\bar{y}_{k|k-1}$ and $\sigma_{k,y}^2$ are the predictive mean and variance of the observed signal, y_k , respectively, which can be computed according to

$$\bar{y}_{k|k-1}^{(i)} = \mathbf{b}_{k}^{(i)\top} \boldsymbol{\mu}_{k|k-1}^{(i)},$$

$$\sigma_{k,y}^{2(i)} = \mathbf{b}_{k}^{(i)\top} \boldsymbol{\Sigma}_{k|k-1}^{(i)} \mathbf{b}_{k}^{(i)} + \sigma_{n}^{2},$$
(21)

where $\boldsymbol{\mu}_{k|k-1}^{(i)}$ and $\boldsymbol{\Sigma}_{k|k-1}^{(i)}$ are the predictive mean and covariance of the channel vector, $\mathbf{x}_{k}^{(i)}$, respectively. The mean and covariance of the channel vector are tracked by the time and measurement updates of the bank of Kalman filters as follows:

• Time update:

$$\boldsymbol{\mu}_{k|k-1}^{(i)} = \mathbf{A}\boldsymbol{\mu}_{k-1}^{(i)},$$
$$\boldsymbol{\Sigma}_{k|k-1}^{(i)} = \mathbf{A}\boldsymbol{\Sigma}_{k-1}^{(i)}\mathbf{A}^{\top} + \mathbf{D}^{\top}\mathbf{D}$$

• Measurement update:

$$\boldsymbol{\mu}_{k}^{(i)} = \boldsymbol{\mu}_{k|k-1}^{(i)} + \boldsymbol{\Sigma}_{k|k-1}^{(i)} \mathbf{b}_{k}^{(i)} \sigma_{k,y}^{2(i)^{-1}} (y_{k} - \bar{y}_{k|k-1}^{(i)}),$$

$$\boldsymbol{\Sigma}_{k}^{(i)} = \boldsymbol{\Sigma}_{k|k-1}^{(i)} - \boldsymbol{\Sigma}_{k|k-1}^{(i)} \mathbf{b}_{k}^{(i)} \sigma_{k,y}^{2(i)^{-1}} \mathbf{b}_{k}^{(i)\top} \boldsymbol{\Sigma}_{k|k-1}^{(i)}.$$

5.2. Optimal proposal density

Unlike the prior distribution, the optimal proposal distribution uses all information available at time *k* in order to propose a new sample [20,24]. Consider the optimal proposal distribution $p(b_k | b_{1:k-1}^{(i)}, y_{1:k})$. We obtain samples $b_k^{(i)}$ from $\mathcal{B} = \{b_1, b_2, \dots, b_{|\mathcal{B}|}\}$ with probabilities $\rho_{k,j}^{(i)}$ where

$$\rho_{k,j}^{(i)} = p(b_k^{(i)} = b_j \mid b_{1:k-1}^{(i)}, y_{1:k}) \quad \text{(where } b_j \in \mathcal{B}\text{)}$$

$$\propto p(y_k \mid b_k^{(i)} = b_j, b_{1:k-1}^{(i)}, y_{1:k-1}) p(b_k^{(i)} = b_j \mid b_{1:k-1}^{(i)}, y_{1:k-1})$$

$$\propto p(y_k \mid b_k^{(i)} = b_j, b_{1:k-1}^{(i)}, y_{1:k-1}). \quad (22)$$

In the last step, we assumed that the symbols are random and the occurrence of $b_k^{(i)} = b_j$ is independent of the previous samples. Further, we can show that the last density of (22) can be written as

$$p(y_{k} | b_{k}^{(i)} = b_{j}, b_{1:k-1}^{(i)}, y_{1:k-1}) = \int p(y_{k}, \mathbf{x}_{k} | b_{k}^{(i)} = b_{j}, b_{1:k-1}^{(i)}, y_{1:k-1}) d\mathbf{x}_{k}$$
$$= \int p(y_{k} | \mathbf{x}_{k}, b_{k}^{(i)} = b_{j}, b_{1:k-1}^{(i)}, y_{1:k-1})$$
$$\times p(\mathbf{x}_{k} | b_{k}^{(i)} = b_{j}, b_{1:k-1}^{(i)}, y_{1:k-1}) d\mathbf{x}_{k}.$$
(23)

The second factor in the last integration of (23) is the predictive density of the channel vector and, as explained, in the previous section, it can be tracked using Kalman filters. Therefore, the expression for the proposal density can be rewritten as

$$\rho_{k,j}^{(i)} \propto \int \mathcal{N}_c(y_k; \mathbf{b}_{k,j}^{(i)\top} \mathbf{x}_k, \sigma_n^2) \mathcal{N}_c(\mathbf{x}_k; \boldsymbol{\mu}_{k|k-1}^{(i)}, \boldsymbol{\Sigma}_{k|k-1}^{(i)}) d\mathbf{x}_k$$

= $\mathcal{N}_c(y_k; \mathbf{b}_{k,j}^{(i)\top} \boldsymbol{\mu}_{k|k-1}^{(i)}, \sigma_n^2 + \mathbf{b}_{k,j}^{(i)\top} \boldsymbol{\Sigma}_{k|k-1}^{(i)} \mathbf{b}_{k,j}^{(i)}),$ (24)

where $\boldsymbol{\mu}_{k|k-1}^{(i)}$ and $\boldsymbol{\Sigma}_{k|k-1}^{(i)}$ are the predictive mean and covariance of \mathbf{x}_k , respectively, and $\mathbf{b}_{k,j}^{(i)} = (b_k^{(i)} = b_j, b_{k-1}^{(i)}, \dots, b_{k-L}^{(i)}, 0, \dots, 0)^\top$. The corresponding weights can be evaluated as

$$w_{k}^{(i)} = w_{k-1}^{(i)} \frac{p(b_{1:k-1}^{(i)} \mid y_{1:k-1}) \ p(b_{k}^{(i)} \mid b_{1:k-1}^{(i)}, y_{1:k})}{p(b_{1:k-1}^{(i)} \mid y_{1:k-1}) \ p(b_{k}^{(i)} \mid b_{1:k-1}^{(i)}, y_{1:k})}$$

$$= w_{k-1}^{(i)} \frac{p(b_{1:k-1}^{(i)} \mid y_{1:k-1})}{p(b_{1:k-1}^{(i)} \mid y_{1:k-1})}$$

$$\propto w_{k-1}^{(i)} \sum_{b_{j} \in \mathcal{B}} p(y_{k} \mid b_{1:k-1}^{(i)}, b_{k}^{(i)} = b_{j}, y_{1:k-1}) p(b_{k}^{(i)} = b_{j} \mid b_{1:k-1}^{(i)})$$

$$\propto w_{k-1}^{(i)} \sum_{b_{j} \in \mathcal{B}} \mathcal{N}_{c}(y_{k}; \mathbf{b}_{k,j}^{(i)\top} \boldsymbol{\mu}_{k|k-1}^{(i)}, \sigma_{n}^{2} + \mathbf{b}_{k,j}^{(i)\top} \boldsymbol{\Sigma}_{k|k-1}^{(i)} \mathbf{b}_{k,j}^{(i)}).$$
(25)

From the weight update equations obtained for both proposal distributions, we can see that a bank of Kalman filters, equal to the numbers of particles, are required to compute the predictive mean and covariance of the channel vector. Since the prior proposal distribution does not depend on the parameters of the channel vector, the symbol imputation of an algorithm based on such distribution can be performed at the outset. For the algorithm based on the optimal proposal distribution, however, since the probability of the symbols, $\rho_{k,i}^{(i)}$, is a function of the predictive mean and covariance of the channel vectors, we need to update these channel parameters for each possible symbol in the alphabet set before a symbol sample is drawn. The predictive mean and covariance of the channel vector are obtained by computing the time update equations of the Kalman filter. Therefore, the algorithm which is based on the optimal proposal distribution is slightly more complex than the algorithm based on the prior distribution. However, it should be noted that, for achieving the same performance, the algorithm with optimal proposal distribution should, at least in theory, require less number of particles (N) than the one implemented using the prior proposal distribution. This is because the proposal distribution generates particles using all the information available at that time. After a symbol sample is drawn, the weights can be updated using (19) or (25). Then, the parameters of the Kalman filters are updated using the measurement equations.

As described earlier, after short run of the used algorithm, only a few particles will have significant weight. Estimation based on such particles renders the method ineffective

323

as the particles with insignificant weights have almost no contribution. To reduce such impoverishment of particles, we apply resampling whenever the effective sample size falls below a certain threshold value. After N particles and their corresponding weights and channel values, $\{(b_k^{(i)}, \boldsymbol{\mu}_k^{(i)}), \tilde{\boldsymbol{w}}_k^{(i)}\}_{i=1}^N$, are generated and computed, then MAP or MMSE estimator are applied to determine the estimates of the transmitted symbol and channel vector. It is to be noted that $\{\boldsymbol{\mu}_k^{(i)}\}_{i=1}^N$ are the channel estimates of each trajectory. The channel vector can be estimated using MMSE as

$$\hat{\mathbf{x}}_k = \sum_{i=1}^N \boldsymbol{\mu}_k^{(i)} \, \tilde{\boldsymbol{w}}_k^{(i)} \tag{26}$$

and similarly, the MAP estimate of the transmitted symbol is obtained by

$$\hat{\mathbf{b}}_{1:k} = \arg \max_{\mathbf{b}_{1:k} \in \{\mathbf{b}_{1:k}^{(i)}\}_{i=1}^{N}} \left\{ \sum_{i=1}^{N} \delta(\mathbf{b}_{1:k}^{(i)} - \mathbf{b}_{1:k}) \tilde{w}_{k}^{(i)} \right\}.$$
(27)

The complete algorithms using the prior and optimal proposal distributions are summarized in Tables 1 and 2, respectively.

5.3. Multiple samples per symbol case

Processing multiple samples per symbol provides an implicit time diversity [34]. It is found that, in fast fading channels, improvement in bit error rate (BER) of a receiver and substantial lowering of error floor can be obtained if more than one sample per symbol is processed [35]. Obviously, if more samples per symbol are available, the channel can be tracked better and the knowledge that some samples belong to same symbols can be exploited.

In this section we extend the algorithms developed in the previous section to the multiple samples per symbol case. We develop algorithms for both prior and optimal proposal densities just like for the one-sample case. It is noted that at a given instant k, where k is an integer multiple of α (the oversampling factor), all the received samples $y_{k-\alpha+1:k}$ are a function of the symbols $b_{k/\alpha:k/\alpha-L}$, where L + 1 is the length of the ISI. Therefore, we can write the posterior density as

$$p(b_{1:k/\alpha} \mid y_{1:k}) \propto p(b_{k/\alpha} \mid b_{1:k/\alpha-1}) \prod_{j=0}^{\alpha-1} p(y_{k-j} \mid b_{1:k/\alpha}, y_{1:k-1-j}) \times p(b_{1:k/\alpha-1} \mid y_{1:k-\alpha}).$$
(28)

If we draw samples from the first factor on the r.h.s. of (28) (which is a prior distribution of $b_{k/\alpha}$), it can be shown, similarly to the one-sample per symbol case, that the weights of the particles can be updated using the second factor on the r.h.s. of (28). This second factor in (28) is a product of the predictive values of the received samples which can be obtained using a Kalman filter. Therefore, in the algorithm for each proposed symbol, a Kalman filter is run α times and the corresponding channel vectors are updated. Note that the weight equation can be sequentially updated as a new sample arrives even though

Table 1

Particle filtering algorithm for equalization using the prior proposal density

Initialize $\Sigma_{0}^{(i)} = \mathbf{I}, \boldsymbol{\mu}_{0}^{(i)} = \mathbf{0}$, and $w_{0}^{(i)} = 1/N$ for i = 1 to NFor k = 1 to M (total number of symbols) For i = 1 to N (total number of particles) • Generate samples from $p(b_{k}) \sim \mathcal{U}(|\mathcal{B}|)$ • Time-update the channel vector $\boldsymbol{\mu}_{k|k-1}^{(i)} = \mathbf{A}\boldsymbol{\mu}_{k-1}^{(i)}$ $\boldsymbol{\Sigma}_{k|k-1}^{(i)} = \mathbf{A}\boldsymbol{\Sigma}_{k-1}^{(i)}\mathbf{A}^{\top} + \mathbf{D}^{\top}\mathbf{D}$ • Evaluate the predictive mean and variance of the observed signal y_{k} $\bar{y}_{k|k-1}^{(i)} = \mathbf{b}_{k}^{(i)^{\top}}\boldsymbol{\mu}_{k|k-1}^{(i)}$ $\sigma_{k,y}^{2(i)} = \mathbf{b}_{k}^{(i)^{\top}}\boldsymbol{\Sigma}_{k|k-1}^{(i)}\mathbf{b}_{k}^{(i)} + \sigma_{n}^{2}$ • Update the weights $\boldsymbol{w}_{k}^{(i)} = \boldsymbol{w}_{k-1}^{(i)}\mathcal{N}(y_{k}; \bar{y}_{k|k-1}^{(i)}, \sigma_{k,y}^{2(i)})$ • Measurement-update the channel vector $\boldsymbol{\mu}_{k}^{(i)} = \boldsymbol{\mu}_{k|k-1}^{(i)} + \boldsymbol{\Sigma}_{k|k-1}^{(i)}\mathbf{b}_{k}^{(i)}\sigma_{k,y}^{2(i)-1}(y_{k} - \bar{y}_{k|k-1}^{(i)})$ $\boldsymbol{\Sigma}_{k}^{(i)} = \boldsymbol{\Sigma}_{k|k-1}^{(i)} - \boldsymbol{\Sigma}_{k|k-1}^{(i)}\mathbf{b}_{k}^{(i)}\sigma_{k,y}^{2(i)-1}\mathbf{b}_{k}^{(i)} \boldsymbol{\Sigma}_{k|k-1}^{(i)}$ • Normalize the weights $\tilde{\boldsymbol{w}}_{k}^{(i)} = \boldsymbol{w}_{k}^{(i)} / \left(\sum_{i=1}^{N} \boldsymbol{w}_{k}^{(i)}\right)$ end • Evaluate the effective sample size

• Evaluate the effective sample size

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^{N} (\tilde{w}_k^{(i)})^2}$$

If $(N_{\text{eff}} \leq 0.5N)$
Apply resampling

end

• Compute the MMSE estimate of the channel and MAP estimate of the symbols

$$\hat{\mathbf{x}}_{k} = \sum_{i=1}^{N} \boldsymbol{\mu}_{k}^{(i)} \ \tilde{w}_{k}^{(i)}$$
$$\hat{\mathbf{b}}_{1:k} = \arg \max_{\mathbf{b}_{1:k}} \left\{ \sum_{i=1}^{N} \delta(\mathbf{b}_{1:k} - \mathbf{b}_{1:k}^{(i)}) \tilde{w}_{k}^{(i)} \right\}$$

the estimates and resampling operations are performed after all the samples of the same symbols are received. We can rewrite the weight equation as follows:

$$w_{k}^{(i)} \propto w_{k-\alpha}^{(i)} \prod_{j=0}^{\alpha-1} p\left(y_{k-j} \mid b_{1:k/\alpha}^{(i)}, y_{1:k-1-j}\right)$$
$$= w_{k-\alpha}^{(i)} \prod_{j=0}^{\alpha-1} \mathcal{N}_{c}\left(y_{k-j}; \bar{y}_{(k-j)|(k-j)-1}^{(i)}, \sigma_{k-j,y}^{2(i)}\right),$$
(29)

where $\bar{y}_{(k-j)|(k-j)-1}^{(i)}$ and $\sigma_{k-j,y}^{2(i)}$ are the predictive mean and predictive variance of y_{k-j} . Similar equations can be developed for the optimal proposal density by rewriting the pos-

Table 2

Particle filtering algorithm for equalization using the optimal proposal density

Initialize $\Sigma_0^{(i)} = \mathbf{I}, \boldsymbol{\mu}_0^{(i)} = \mathbf{0}$, and $w_0^{(i)} = 1/N$ for i = 1 to N For k = 1 to M (total number of symbols) For i = 1 to N (total number of particles) • Time-update of the channel vector $\boldsymbol{\mu}_{k|k-1}^{(i)} = \mathbf{A}\boldsymbol{\mu}_{k-1}^{(i)}$ $\boldsymbol{\Sigma}_{k|k-1}^{(i)} = \mathbf{A}\boldsymbol{\Sigma}_{k-1}^{(i)} \mathbf{A}^{\top} + \mathbf{D}^{\top}\mathbf{D}$ For j = 1 to $|\mathcal{B}|$ • Evaluate the predictive mean and variance of the observed signal y_k for each symbol
$$\begin{split} \bar{\mathbf{y}}_{(k)|k-1),j}^{(i)} &= \mathbf{b}_{k,j}^{(i)\top} \boldsymbol{\mu}_{k|k-1}^{(i)} \\ \sigma_{(k,y),j}^{(2i)} &= \mathbf{b}_{k,j}^{(i)\top} \boldsymbol{\Sigma}_{k|k-1}^{(i)} \mathbf{b}_{k,j}^{(i)} + \sigma_n^2 \\ \bullet \text{ Compute the proposal density for each possible symbol} \\ \rho_{k,j}^{(i)} &\propto \mathcal{N}_c(\mathbf{y}_k; \bar{\mathbf{y}}_{(k|k-1),j}^{(i)}, \sigma_{(k,y),j}^{2(i)}) \end{split}$$
end • Draw a symbol from \mathcal{B} using $P(b_k^{(i)} = b_j) \propto \rho_{k,j}^{(i)}$ • Update the weights $w_k^{(i)} = w_{k-1}^{(i)} \sum_{b_j \in \mathcal{B}} \mathcal{N}_c(y_k; \mathbf{b}_{k,j}^{(i)} \boldsymbol{\mu}_{k|k-1}^{(i)}, \sigma_n^2 + \mathbf{b}_{k,j}^{(i)\top} \boldsymbol{\Sigma}_{k|k-1}^{(i)} \mathbf{b}_{k,j}^{(i)})$ • Measurement-update the channel vector using the drawn symbol $\boldsymbol{\mu}_{k}^{(i)} = \boldsymbol{\mu}_{k|k-1}^{(i)} + \boldsymbol{\Sigma}_{k|k-1}^{(i)} \mathbf{b}_{k}^{(i)} \sigma_{k,y}^{2(i)^{-1}} \left(y_{k} - \bar{y}_{k|k-1}^{(i)} \right)$ $\boldsymbol{\Sigma}_{k}^{(i)} = \boldsymbol{\Sigma}_{k|k-1}^{(i)} - \boldsymbol{\Sigma}_{k|k-1}^{(i)} \mathbf{b}_{k}^{(i)} \sigma_{k,y}^{2(i)^{-1}} \mathbf{b}_{k}^{(i)\top} \boldsymbol{\Sigma}_{k|k-1}^{(i)}$ • Normalize the weights $\tilde{w}_k^{(i)} = w_k^{(i)} \middle/ \left(\sum_{i=1}^N w_k^{(i)}\right)$ end • Evaluate the effective sample size 1

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^{N} (\tilde{w}_k^{(i)})^2}$$

• If $(N_{\text{eff}} \leq 0.5N)$

Apply resampling

• Compute the MMSE estimate of the channel and MAP estimate of the symbols $\sum_{i=1}^{N} (i) = (i)$

$$\mathbf{x}_{k} = \sum_{i=1}^{N} \boldsymbol{\mu}_{k}^{(i)} \boldsymbol{w}_{k}^{(i)}$$
$$\hat{\mathbf{b}}_{1:k} = \arg\max_{\mathbf{b}_{1:k}} \left\{ \sum_{i=1}^{N} \delta(\mathbf{b}_{1:k} - \mathbf{b}_{1:k}^{(i)}) \tilde{\boldsymbol{w}}_{k}^{(i)} \right\}$$

terior density as

end

$$p(b_{1:k/\alpha} | y_{1:k}) \propto p(b_{k/\alpha} | b_{1:k/\alpha-1}, y_{1:k}) p(b_{1:k/\alpha-1} | y_{1:k-\alpha}) \\ \times \prod_{j=0}^{\alpha-1} p(y_{k-j} | b_{1:k/\alpha-1}, y_{1:k-1-j}).$$
(30)

Note that we can draw samples from the first factor on the r.h.s. of (30), which is the optimal proposal density of $b_{k/\alpha}$. Then, the third factor is used in the weight update equation, i.e.,

$$w_{k}^{(i)} \propto w_{k-\alpha}^{(i)} \prod_{j=0}^{\alpha-1} p(y_{k-j} | b_{1:k/\alpha-1}^{(i)}, y_{1:k-1-j})$$

$$= w_{k-\alpha}^{(i)} \prod_{j=0}^{\alpha-1} \sum_{b_{l} \in \mathcal{B}} p(y_{k-j} | b_{k/\alpha}^{(i)} = b_{l}, b_{1:k/\alpha-1}^{(i)}, y_{1:k-1-j})$$

$$= w_{k-\alpha}^{(i)} \prod_{j=0}^{\alpha-1} \sum_{b_{l} \in \mathcal{B}} \mathcal{N}(y_{k-j}; \bar{y}_{((k-j)|(k-j)-1),l}^{(i)}, \sigma_{(k-j,y),l}^{2(i)}).$$
(31)

The symbol imputation is deferred until all the samples of a symbol are received. This helps in exploiting the prior information, that multiple samples come from the same symbols. However, updating the channel vector of a given trajectory, requires the prior knowledge of the proposed symbol. We circumvent this problem by updating the channel vector for all possible symbols in the symbol alphabet. Once a symbol proposal is made, the corresponding channel vector is selected and carried through. Following similar steps of the single sample per symbol case, we can rewrite the optimal proposal density as

$$\rho_{k/\alpha,l}^{(i)} = p(b_{k/\alpha} = b_l \mid b_{1:k/\alpha-1}^{(i)}, y_{1:k})
\propto \prod_{j=0}^{\alpha-1} p(y_{k-j} \mid b_{k/\alpha} = b_l, b_{1:k/\alpha-1}^{(i)}, y_{1:k-j-1})
\propto \prod_{j=0}^{\alpha-1} \int \mathcal{N}_c(y_{k-j}; \mathbf{b}_{k/\alpha,l}^{(i)\top} \mathbf{x}_{k-j}, \sigma_n^2)
\times \mathcal{N}_c(\mathbf{x}_{k-j}; \boldsymbol{\mu}_{(k-j)|(k-j)-1}^{(i)}, \boldsymbol{\Sigma}_{(k-j)|(k-j)-1}^{(i)}) d\mathbf{x}_{k-j}
= \prod_{j=0}^{\alpha-1} \mathcal{N}_c(y_{k-j}; \mathbf{b}_{k/\alpha,l}^{(i)\top} \boldsymbol{\mu}_{(k-j)|(k-j)-1}^{(i)}, \sigma_n^2 + \mathbf{b}_{k/\alpha,l}^{(i)\top} \boldsymbol{\Sigma}_{(k-j)|(k-j)-1}^{(i)} \mathbf{b}_{k/\alpha,l}^{(i)}), \quad (32)$$

where $b_l \in \mathcal{B}$ and $\mathbf{b}_{k/\alpha,l}^{(i)\top} = (b_{k/\alpha}^{(i)} = b_l, b_{k/\alpha-1}, \dots, b_{k/\alpha-L}, 0, \dots, 0)$. As described in the one-sample case, the algorithms for the multiple samples per symbol case also require resampling to reduce the impoverishment of particles of significant weights. Once the particles are obtained, MAP and MMSE estimators may be applied to find estimates of the transmitted symbols and channel vectors.

6. Simulations and results

Computer simulation experiments have been conducted to determine the performance of the developed algorithms. Experiments for a two-ray and a three-ray multi-path channels have been carried out although, the presented algorithm works for more general multipath channels. The channels were modeled for a fading rate of $f_d T = 0.022$, which corresponds



Fig. 4. Simulation result-BER vs. SNR (dB) for the two-ray channel.

to a carrier frequency of $f_c = 2$ GHz, a vehicle speed of v = 75 miles/h and data rate of 10,000 symbols/s. Second order AR processes, with coefficients $\gamma_1 = 1.9602$ and $\gamma_2 = -0.9701$, were chosen to model the time variation of the tap weights of the FIR channels. All the tap weights of the channels had equal power.

A differential BPSK encoding, where $\mathcal{B} = \{+1, -1\}$ was used in order to avoid phase ambiguity that arises in blind equalization problems. The developed algorithms can also be used without modification for this encoding because the probability mass function of the identically and independently distributed (i.i.d.) binary symbols remains unaltered after encoding.

The algorithms were run for 100 particles for a signal-to-noise ratio (SNR) range of 5–40 dB. A resampling method known as systematic resampling [28] was used and was applied whenever the effective sample size droped below half of the total number of particles. Finally, the symbol estimate was performed using the MAP estimator.

Figures 4 and 5 show the performance of the developed algorithms depicting the BER achieved for different SNRs. The results in these two figures refer to the case of single sample per symbol. Figure 4 shows the performance achieved for the two-ray channel and Fig. 5 depicts that of the three-ray channel. Two of the curves represent the performance achieved using the developed algorithms for both the prior and optimal proposal distributions. As can be seen the achieved performance using the optimal proposal distribution is slightly better than the one obtained using the prior proposal distribution. It was observed that, especially for the algorithms based on the optimal proposal density, the increase in number of particles did not result in significant improvement of performance.

For comparison purposes, for both channels, we have simulated a PSP–MLSD receiver based on the LMS algorithm for channel tracking. The step size of the LMS algorithm is set to be 0.25. The PSP–MLSD algorithm was simulated by organizing the symbols into frames of length of 300 symbols and five training symbols were used for the first frame. It



Fig. 5. Simulation result-BER vs. SNR (dB) for the three-ray channel.

is observed that our algorithms achieve better performance than the PSP–MLSD receiver, in particular at high SNRs. The PSP–LMS algorithm exhibits an error floor at about 25 dB.

MLSD receivers with known and with genie-aided channels were also simulated. These methods provide lower bounds for the proposed schemes. For the genie-aided simulation an LMS algorithm was used for channel tracking. It has been observed that the PSP–LMS genie-aided algorithm also exhibits error floor at about SNR = 25 dB.

Figures 6 and 7 depict the performance achieved for the multiple samples per symbol case for the two channels described above. An oversampling factor of two, $\alpha = 2$, was utilized. As seen in the figures, we have also simulated the per-branch-processing-LMS (PBP–LMS) [8] algorithm and the MLSD with known channel for comparison purposes. From the figures, we can see that the proposed SIS algorithms outperform the PBP–LMS method, in particular at high SNRs for both channels. As expected, it is also observed that, at high SNRs, a significant gain in performance is achieved when two samples per symbol are processed in comparison with processing a single sample per symbol. This is because at high SNR the performance is limited by channel fading and multiple samples per symbol processing can track the channel better. However, it should be noted that processing multiple samples per symbol requires more computational power.

7. Conclusions

In this paper we proposed blind equalization methods for frequency-selective fading channels. The methods estimate the channel and detect the transmitted symbols jointly. We have modeled the frequency-selective channel using a multi-tap FIR filter whose tap weights vary as AR processes.



Fig. 6. A BER vs. SNR (dB) (two sample per symbol processing) for the two-ray channel.



Fig. 7. A BER vs. SNR (dB) (two sample per symbol processing) for the three-ray channel.

The developed algorithms are based on SMC filtering. We have used the strategy of Rao–Blackwellization where particle filtering is used to determine the posterior distribution of the transmitted symbols while a bank of Kalman filters is employed for estimating the posterior density of the channel vector. Computer simulations show that the proposed algorithms outperform the PSP–MLSD receivers based on LMS channel tracking.

T. Ghirmai et al. / Digital Signal Processing 14 (2004) 312-331

The algorithms are also extended to the case of multiple samples per symbol. Multiple samples per symbol provide implicit diversity which results in performance improvement. The computer simulations demonstrate that at medium and high SNR values a significant gain can be achieved compared to the single sample per symbol case.

Sequential Monte Carlo methods are inherently computationally intensive and the same is true about our algorithms. However, these algorithms are highly parallelizable and suitable for VLSI implementation.

References

- G.D. Forney, Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference, IEEE Trans. Inform. Theory 18 (1971) 363–378.
- [2] H. Meyr, M. Moeneclaey, S.A. Fechtel, Digital Communication Receivers: Synchronization, Channel Estimation and Signal Processing, Wiley, New York, 1998.
- [3] F.R. Magee, J.G. Proakis, Adaptive maximum-likelihood sequence estimation for digital signaling in the presence of intersymbol interference, IEEE Trans. Inform. Theory 19 (1973) 120–124.
- [4] R. Raheli, A. Polydoros, C. Tzou, Per-survivor-processing: A general approach to MLSE in uncertain environments, IEEE Trans. Commun. 43 (1995) 354–364.
- [5] N. Seshadri, Joint data and channel estimation using blind trellis search techniques, IEEE Trans. Commun. 42 (1994) 1000–1011.
- [6] H. Kubo, K. Murakami, T. Fujino, An adaptive Viterbi detector for fast-fading time-varying intersymbol interference channels, IEEE Trans. Commun. 42 (1994) 1972–1980.
- [7] D.K. Borah, B.D. Hart, Receiver structures for time-varying frequency-selective channels, IEEE J. Select. Areas Commun. 17 (1999) 1863–1875.
- [8] M. Omidi, S. Pasupathy, P. Gulak, Joint data and Kalman estimation for Rayleigh fading channels, Wireless Personal Commun. 10 (1999) 319–339.
- [9] Q. Dai, E. Shewdyk, Detection of bandlimited signals over frequency-selective Rayleigh fading channels, IEEE Trans. Commun. 42 (2–4) (1994) 941–959.
- [10] M.E. Rollins, S.J. Simmons, Simplified per-survivor Kalman processing in fast frequency-selective fading channels, IEEE Trans. Commun. 45 (1997) 544–553.
- [11] R. Cusani, J. Mattila, Equalization of digital radio channels with large multipath delay for cellular land mobile applications, IEEE Trans. Commun. 47 (1999) 348–351.
- [12] E. Baccarelli, R. Cusani, S. Galli, A novel adaptive receiver with enhanced channel tracking capability for TDMA-based mobile radio communications, IEEE J. Select. Areas Commun. 16 (1998) 1630–1639.
- [13] R.A. Iltis, J. Shynk, K. Giridhar, Bayesian algorithms for blind equalization using parallel adaptive filtering, IEEE Trans. Commun. 42 (1994) 1017–1032.
- [14] R. Chen, X. Wang, J. Liu, Adaptive joint detection and decoding in flat-fading channels via mixture Kalman filtering, Trans. Inform. Process. 46 (6) (2000) 2079–2094.
- [15] E. Punskaya, C. Andrieu, A. Doucet, Particle filtering for optimal detection in fading channels, Technical report, University of Cambridge, Trumpington St., CB2 1PZ, Cambridge, UK, 2001.
- [16] J.H. Kotecha, P.M. Djurić, Sequential Monte Carlo sampling detector for Rayleigh fast-fading channels, in: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Istanbul, Turkey, 2000.
- [17] P.M. Djurić, J.H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M.F. Bugallo, J. Miguez, Particle filtering, IEEE Signal Process. Mag. 20 (5) (2003) 19–38.
- [18] J. Miguez, P.M. Djurić, Blind equalization by sequential importance sampling, in: Proceedings of IEEE International Symposium on Circuits and Systems, vol. 1, 2002, pp. 845–848.
- [19] Z. Yang, X. Wang, A sequential Monte Carlo blind receiver for OFDM systems in frequency-selective fading channels, IEEE Trans. Signal Process. 50 (2) (2002) 271–280.
- [20] E. Punskaya, A. Doucet, W.J. Fitzgerald, On the use and misuse of particle filtering, in: Proceedings of EUSIPCO, vol. II, Toulouse, 2002, pp. 173–176.

- [21] R.A. Iltis, A sequential Monte Carlo filter for joint linear/nonlinear state estimation with applications to DS-CDMA, IEEE Trans. Signal Process. 51 (2) (2003) 417–426.
- [22] Y. Huang, P.M. Djurić, A blind particle filtering detector for joint channel estimation, and data detection over flat fading channels, in: Proceedings of EUSIPCO, vol. III, 2002, pp. 331–334.
- [23] P.H.-Y. Wu, A. Duel-Hallen, Multiuser detectors with disjoint Kalman channel estimators for synchronous CDMA mobile radio channels, IEEE Trans. Commun. 48 (5) (2000) 752–756.
- [24] T. Ghirmai, J. Kotecha, P.M. Djurić, Joint channel estimation and data detection over frequency-selective fading channel using sequential Monte Carlo filtering, in: Proceedings of Conf. in Information Sci. and Syst., Princeton University, 2002.
- [25] K. Yip, T. Ng, Efficient simulation of digital transmission over WSSUS channels, IEEE Trans. Commun. 43 (12) (1995) 2907–2913.
- [26] W. Jake, Microwave Mobile Communications, Wiley, 1974.
- [27] A. Doucet, N. de Freitas, N. Gordon (Eds.), Sequential Monte Carlo Methods in Practice, Springer, New York, 2001.
- [28] A. Doucet, S.J. Godsill, C. Andrieu, On sequential Monte Carlo sampling methods for Bayesian filtering, Statist. Comput. (2000) 197–208.
- [29] A. Kong, J. Liu, W. Wong, Sequential imputation and Bayesian missing data problems, J. Amer. Statist. Assoc. 9 (278–288).
- [30] J.S. Liu, R. Chen, Sequential Monte Carlo methods for dynamic systems, J. Amer. Stastist. Assoc. 93 (443) (1998) 1032–1044.
- [31] J. Carpenter, P. Clifford, P. Fearnhead, An improved particle filter for non-linear problems, IEE Proc. F Radar Sonar Navigation 146 (1999) 2–7.
- [32] G. Casella, C.P. Robert, Rao-Blackwellization of sampling schemes, Biometrika 84 (1996) 81-94.
- [33] R. Chen, J.S. Liu, Mixture Kalman filters, J. Roy. Statist. Soc. B 62 (2000) 493–508.
- [34] R.S. Kennedy, Fading in Dispersive Communication Channels, Wiley–Interscience, New York, 1969.
- [35] G.M. Vitetta, D. Taylor, Multisampling receivers for uncoded and coded PSK signal sequences transmitted over Rayleigh frequency-flat fading channels, IEEE Trans. Commun. 44 (2) (1996) 130–133.