Improved Bayesian MIMO Channel Tracking for Wireless Communications: Incorporating a Dynamical Model

Kris Huber and Simon Haykin, *Fellow, IEEE*

***Abstract*— This paper investigates the improved decoder performance offered by incorporating dynamic linear modelling techniques when applied to particle filters for use in tracking the MIMO wireless channel. Conventional Bayesian-based receivers that perform channel tracking necessarily require a wireless channel model, typified by the use of a low order auto-regressive (AR) model. Normally, the model parameters are static in nature and are estimated *a priori* of any transmission; thus if the channel conditions change, a model mismatch occurs and system performance is degraded. Our method allows for time-varying channel statistics by modelling the channel fading rate as a Markov random walk. This new procedure allows the channel model to assume a time-varying behavior. As will be shown through simulations, the incorporation of dynamic modelling of time-dispersive channels not only offers superior performance, but at high SNR eliminates the error-rate floor commonly seen in systems using the static AR models.**

***Index Terms*— MIMO, wireless channel tracking, dynamic linear modelling, particle filtering.**

I. Introduction

In this paper, we consider the problem of providing reliable channel estimates to coherent decoders for use in next-generation wireless networks. Enticed by the promise of increased spectral efficiency and higher data rates, next-generation wireless systems will necessarily operate under a multiple-input, multiple-output (MIMO) antenna network $\{N_t, N_r\}$, where N_t and N_r denote the number of transmit and receive antenna elements, respectively. Central to the ability of obtaining these high data rates is the type of space-time code (STC) being used [1]. Coherent space-time decoders require not only the received signal to perform demodulation, but also a reliable estimate of the channel-state information (CSI). Should the estimation mechanism fail to yield accurate estimates of the fading process, the channel decoder will necessarily also perform poorly. Current wireless systems obtain the CSI through a process known as Pilot-Assisted Transmission (PAT) [2], [3]. PAT multiplexes a periodic known sequence of symbols with the information-bearing symbols in each frame of transmitted data. Using the training data, the receiver is then enabled to obtain an estimate of the CSI. However, this estimate remains static between successive training sequences, during which time the channel itself will be continually changing. Additionally, due to channel noise and the time-varying nature of the wireless channel, the CSI estimate itself may not be accurate [4], [5].

Initial work for improving the channel estimate by applying tracking techniques were performed using Kalman filtering [6] and Mixture-Kalman filtering [7]. Kalman-based systems necessarily require a state model of the system of interest, in this case the wireless channel. These systems modelled the channel with a static AR process, and the additive channel noise with a Gaussian distribution. Recent work by Komninakis [8] has shown that using low-order AR models to represent the channel produces an error-floor in performance at high signal-to-noise ratios. The error-floor arises from the fact that low-order AR processes cannot truly model the autocorrelation sequence of the wireless channel [9]. In effect a model mismatch between the AR model and the channel itself occurs. In order to overcome this deficiency, it becomes necessary to increase the order (and necessarily the overall complexity) of the state equation. Work by Blackard, Rappaport and Bostian [10] focused on the more realistic modelling of the channel noise as non-Gaussian. This was evinced by channel sounding measurements of typical wireless environments. Under these assumptions, it became pertinent to replace the Kalman filter with a particle filter [11]. MIMO channel trackers implemented by Haykin, Huber and Chen [12] and Chin, Ward, and Constantinides [13] revealed the receiver-performance improvements of particle based systems over their Kalman counterparts. It should be noted that in the above mentioned work, all of which incorporate the static AR model, the error-floor phenomenon is present (the degree of which is dependent upon the details of each experimental setup) and is thus a systematic error which needs to be addressed.

In order to develop a dynamic state model for the time-varying wireless channel, concepts from the field of Bayesian forecasting are used. Bayesian forecasting is concerned with the optimal learning and prediction of different classes of dynamic models [14]; in this paper, we are concerned with the extension of the static AR model of order one to its time-varying counterpart. Work by Djurić and Kotecha [15] has shown how a particle filter might be implemented to adapt to unknown static or piecewise changing AR parame-

Manuscript received May, 2004; revised December, 2005; accepted February 6, 2006. The associate editor coordinating the review of this paper and approving it for publication was A. Molisch. This work is supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada.

K. Huber is with Array Systems Computing Inc., Ontario, M3J 3H7 Canada (e-mail: huber@soma.mcmaster.ca).

S. Haykin is with the Faculty of Engineering at McMaster University, Hamilton, Ontario L8S 4K1, Canada (e-mail: haykin@mcmaster.ca).

Digital Object Identifier 10.1109/TWC.2006.04313.

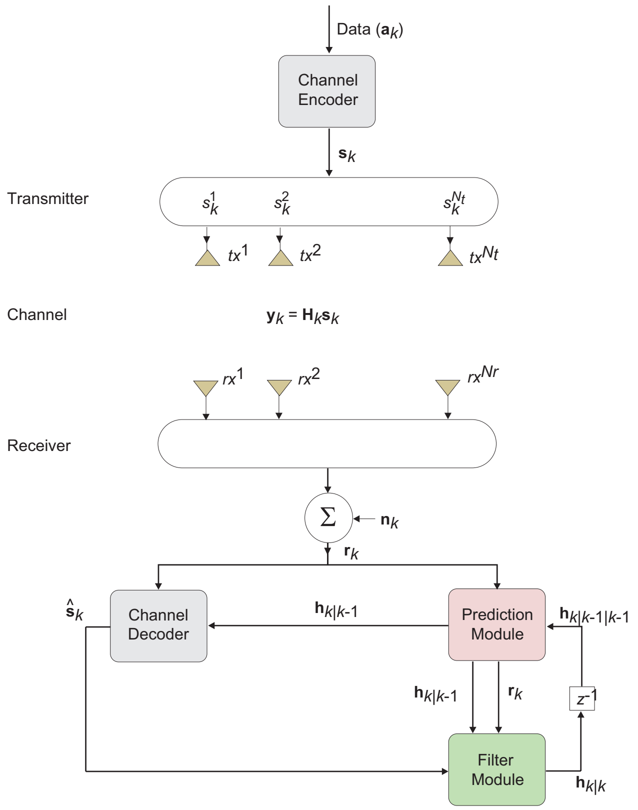


Fig. 1. Block receiver design structure incorporating the channel tracking modules.

ters; however, the emphasis in [15] is on adaptation to the unknown parameters and not on the use of these parameters for prediction, which is a necessary requirement in the MIMO receiver case. The rest of this paper is organized as follows. Section II presents the physical system setup. Section III reviews particle filtering, and details the state and measurement equations needed for particle filtering. The static state model is then improved upon by the incorporation of a dynamical channel model. Section IV presents experimental evaluations of the proposed channel tracking algorithm. Finally, section V concludes the paper.

II. Problem Formulation

Consider a wireless transceiver operating in a frequency flat, time-selective fading channel with N_t transmit and N_r receive antennas as shown in Fig. 1. The incoming data stream is encoded, multiplexed and transmitted across the wireless channel. The received signal is then iteratively processed by the channel decoder and the prediction and filter modules that collectively form the channel-tracking algorithm using simulations. The goal of the receiver is to produce an estimate of the transmitted symbols $\mathbf{s}(k)$.

We model the input-output relationships as [16]:

$$\mathbf{r}_k = \mathbf{H}_k\mathbf{s}_k + \mathbf{n}_k \tag{1}$$

where $\mathbf{r}_k \in \mathbb{C}^{N_r}$ is the complex baseband received signal at transmit time k, $\mathbf{H}_k \in \mathbb{C}^{N_r,N_t}$ is the complex channel matrix at time k, and $\mathbf{s}_k \in \mathbb{C}^{N_t}$ is the transmitted symbol vector at time k. The additive measurement noise $\mathbf{n}_k \in \mathbb{C}^{N_r}$ can be modelled as either a complex-Gaussian distribution $p_{\mathbf{n}}(z) = \mathcal{N}_{\mathbb{C}}(z;0,\sigma^2)$ with argument z, mean zero, and variance σ^2, or as the *Middleton Class-A noise model*. This latter model has been used to model the impulsive noise commonly generated in an indoor/urban wireless environment [10], [17]. The probability density function of the noise model is given by

$$p(z) = (1-\epsilon)\mathcal{N}_{\mathbb{C}}(z;0,\varsigma^2) + \epsilon\mathcal{N}_{\mathbb{C}}(z;0,\kappa\varsigma^2), \tag{2}$$

where $0 \le \epsilon \le 1$ and $\kappa >> 1$. The first component $(1-\epsilon)\mathcal{N}_{\mathbb{C}}(z;0,\varsigma^2)$ represents the ambient background noise with probability $1-\epsilon$, while $\epsilon\mathcal{N}_{\mathbb{C}}(z;0,\kappa\varsigma^2)$ denotes the presence of an impulsive component occurring with probability ϵ. In order to maintain a constant noise variance σ^2 for a particular SNR, we may vary the parameters ϵ and κ such that

$$\sigma^2 = (1-\epsilon)\varsigma^2 + \epsilon\kappa\varsigma^2. \tag{3}$$

Finally it should be noted that by setting $\epsilon = 0$, the mixture model reverts to the Gaussian distribution.

III. Receiver Structure

A. Particle Filter

In using particle filters for the tracking mechanism, the goal is to estimate the posterior distribution of the state, given all the available information. In the context of tracking, where at each observation new information becomes available, we would naturally wish to update our estimate of the state. Thus it is desirable to obtain a recursive form for the solution. The Bayesian solution is implemented in a closed-loop recursive process involving two steps: prediction, and updating. To proceed, define the state vector of dimension $\{N_t \times N_r, 1\}$ given by $\mathbf{h}_k = \text{Vec}(\mathbf{H}_k)$. Then, if we assume that an initial estimate of the posterior density $p(\mathbf{h}_{k-1}|\mathbf{r}_{k-1})$ at time $k-1$ is available, where $\mathbf{h}_k$ is the state variable and $\mathbf{r}_k$ is the observation, we can predict how the state will evolve over time k via the Chapman-Kolmogorov equation:

$$p(\mathbf{h}_k|\mathbf{r}_{k-1}) = \int p(\mathbf{h}_k|\mathbf{h}_{k-1})p(\mathbf{h}_{k-1}|\mathbf{r}_{k-1})d\mathbf{h}_{k-1} \tag{4}$$

where $p(\mathbf{h}_k|\mathbf{h}_{k-1})$ describes how the state density evolves with time k, and is defined by the state equation. When the current observation $\mathbf{r}_k$ becomes available, we may update the prior (4) via Bayes' rule, obtaining

$$p(\mathbf{h}_k|\mathbf{r}_k) = \frac{p(\mathbf{r}_k|\mathbf{h}_k)p(\mathbf{h}_k|\mathbf{r}_{k-1})}{\int p(\mathbf{r}_k|\mathbf{h}_k)p(\mathbf{h}_k|\mathbf{r}_{k-1})d\mathbf{h}_k} \tag{5}$$

where $p(\mathbf{r}_k|\mathbf{h}_k)$ is the *likelihood* of receiving the observation $\mathbf{r}_k$, given the state $\mathbf{h}_k$. In the wireless MIMO case, the likelihood is determined by the observation equation (1). The denominator term in (5) is necessary in order to keep the new estimate of the posterior properly normalized such that $\int p(\mathbf{h}_k|\mathbf{r}_k)d\mathbf{h}_k = 1$ for all k. From the distribution $p(\mathbf{h}_k|\mathbf{r}_k)$ we may obtain our channel estimate $\hat{\mathbf{h}}_k$. This value is needed by the channel decoder to compute an estimate $\hat{\mathbf{s}}_k$ of the transmitted data symbols. In this work we define $\hat{\mathbf{h}}_k$ as the expectation over $p(\mathbf{h}_k|\mathbf{r}_k)$, i.e., $\hat{\mathbf{h}}_k = \mathsf{E}[p(\mathbf{h}_k|\mathbf{r}_k)]$, where E is the expectation operator.

While the above equations outline the tracking procedure in principle, in reality obtaining analytical solutions can be difficult if not impossible. In order to recursively evaluate (4) and (5), we utilize the method of *Importance Sampling*[1], which is a common Monte Carlo (MC) method for sequential MC filters [18], [19]. The idea is to represent the required posterior density by a set of weighted particles:

$$p(\mathbf{h}_k|\mathbf{r}_k) \simeq \sum_{\ell=1}^{L} w_k^\ell \delta(\mathbf{h}_k - \mathbf{h}_k^\ell) \tag{6}$$

where L is the number of particles, $\delta(\cdot)$ is the Dirac delta function, and $\mathbf{h}_k^\ell$ is the state of particle $\ell = 1, ..., L$ at time k. The weights themselves are normalized such that at each time k, $\sum_{\ell=1}^{L} w_k^\ell = 1$. As the number of particles becomes large, the approximation in (6) converges to the true posterior pdf.

New particles are drawn from a known distribution referred to as the *proposal distribution*[2]

$$\mathbf{h}_k \sim q(\mathbf{h}_k|\mathbf{h}_{k-1}^\ell, \mathbf{r}_k). \tag{7}$$

The proposal distribution contains all the *a priori* knowledge one has about the nature of system under study. In general, choice of the proposal distribution is problem-dependent. While, in principle, the optimal importance distribution defined as the density which minimizes the variance of the weights (w_k^ℓ) conditioned on $(\mathbf{h}_{k-1}^\ell)$ and $(\mathbf{r}_k)$ does exist, it is generally difficult to define for the problem at hand; moreover, determination of the optimal distribution is computationally intensive. Consequently, much work has gone into finding suboptimal proposal distributions [12], [20].

For the work reported in this paper, in order to increase the sampling efficiency of the proposal distribution, we adapt the *gradient particle filter* (GPF) discussed in [12]. Implementation of the standard particle filter generally sets the proposal distribution equal to the state evolution density (commonly referred to as the bootstrap filter). However, this form of implementation can be inefficient for sampling. For example, states occurring in the tails of the distribution will require many particles because of each particles' low probability of being sampled in that region. By contrast, the GPF incorporates the current observation $\mathbf{r}_k$ in its predictive estimate. The GPF operates by applying a correction factor based on the gradient of the likelihood for each resampled particle. This correction tends to guide the particles towards the high probability regions of the density. Using the GPF serves a dual purpose: Allowing the filter to operate with a reduced number of particles, and providing a more reliable predictive density needed for the next time step.

Following the selection of the particles from (7), the weights for $\ell = 1, ..., L$ at time 'k' are sequentially updated as follows [11]:

$$w_k^\ell = w_{k-1}^\ell \frac{p(\mathbf{r}_k|\mathbf{h}_k^\ell)p(\mathbf{h}_k^\ell|\mathbf{h}_{k-1}^\ell)}{q(\mathbf{h}_k^\ell|\mathbf{h}_{k-1}^\ell, \mathbf{r}_k)}. \tag{8}$$

Note that the likelihood function $p(\mathbf{r}_k|\mathbf{h}_k^\ell)$ implicitly assumes that the estimates of the transmitted symbols $(\mathbf{s}_k)$ obtained by the receiver are available. The key point to realize is that we have used a weighted set of particles drawn from a known importance density in order to approximate an unknown target distribution of interest. In practice, however, it has been shown [21] that the distribution of the importance weights becomes more and more skewed as time increases. This phenomenon is called *weight degeneracy* or *sample impoverishment*. To monitor the degeneracy, a suggested measure called the *effective sample size*,

$$\hat{N}_{\text{eff}} = \frac{1}{\sum_{\ell=1}^{L}(w_k^\ell)^2}, \tag{9}$$

is usually introduced [18]. Whenever $\hat{N}_{\text{eff}}$ is below a predefined threshold N_T (typically $N_T = \frac{2}{3}L$), a resampling procedure is performed. Specifically, particles with low weights are discarded, forming a subset of particles $\{\mathbf{h}_k^p\}$. New particles $\mathbf{h}_k^\ell$ are generated by respampling with replacement (to keep L constant) particles from the subset $\{\mathbf{h}_k^p\}$ with probability $\Pr(\mathbf{h}_k^\ell = \mathbf{h}_k^p) = w_k^p$. The weights must now be normalized by resetting them to $w_k^\ell = 1/L$. In a sequential filtering framework, the resampling step is almost inevitable; however, it also introduces increased random variation into the estimation procedure.

B. Modelling the Wireless Channel

In formulating a channel model suitable for use in the channel tracker, the goal is to accurately capture the dynamics of the wireless channel yet remaining mathematically tractable for implementation in a discrete-time state-space context. We first note that according to the Bello model [22], the fading process from transmit antenna i to receive antenna j is modelled as a complex Gaussian process. A suitable model is thus an *auto-regressive* (AR) model. Information-theoretic results have shown that a first-order AR model is sufficient to accurately represent the local behavior of the time-varying wireless channel [23]. A higher order model while providing more accurate long-term channel estimates, necessarily requires an AR order of 100 - 200 coefficients [9] and is thus highly intractable for the state model. Using the first-order assumption, we finally realize the state evolution at time k by using a first-order autoregressive model of the form

$$\mathbf{h}_k = \beta \mathbf{h}_{k-1} + \mathbf{v}_k \tag{10}$$

where β is the static AR coefficient, $\mathbf{h}_k$ is a vector of length $N_t N_r$ where each element is the channel gain at time k for path from the i^{th} transmit antenna to the j^{th} receive antenna, and $\mathbf{v}_k \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \sigma_v^2 \mathbf{I})$ is the complex driving noise of the model. No statistical dependence is assumed between the noise terms across the indices i and j for all k.

Additional advantages of using the AR model for describing the evolution of the channel state include:

1) The model is simple and mathematically tractable.
2) The true channel impulse response tends to revert to zero; the behavior of (10) also tends to revert to zero.
3) Like the wireless channel, the AR model is a Markov process. This implies that the pdf for the current estimate, denoted by $p(\mathbf{h}_k|\mathbf{h}_{1:k-1})$ is not dependent upon all previous estimates but only on the most recent estimate $p(\mathbf{h}_k|\mathbf{h}_{k-1})$. Owing to the Markovian property,

[1]For a complete description of Importance Sampling including the pseudocode, we refer the reader to [18].

[2]It is important that we can both draw from this distribution and evaluate the likelihood of those particles using this distribution.

the AR model greatly simplifies the complexity of the recurrence relations used in the particle filter.

In order to parameterize (10), we note from [22] that for a time lag (τ) the autocorrelation of the channel fading process is:

$$\mathsf{E}[\mathbf{h}_k \mathbf{h}^\dagger_{(k-\tau)}] = \mathrm{J}_0(2\pi f_D \tau)\mathbf{I}, \quad (11)$$

where $\mathbf{I}$ is the identity matrix, $\mathrm{J}_0(\cdot)$ is the zeroth-order Bessel function, τ is the time lag, and f_D denotes the Doppler frequency resulting from relative motion between the transmitter and receiver. The Doppler shift itself is given by

$$f_D = \frac{v}{c} f_c \quad (12)$$

where v is the mobile speed, c is the speed of light, and f_c is the carrier frequency. Equating (10) to the autocorrelation of (11) for time lag $\tau = \{0, T_s\}$, we respectively have

$$\begin{aligned} \beta^2 + \sigma_v^2 &= 1, & \tau &= 0 & \quad (13a) \\ \beta &= \mathrm{J}_0(2\pi f_d T_s), & \tau &= T_s & \quad (13b) \end{aligned}$$

where, in the second equation, $1/T_s$ is the sampling rate. For example, if the normalized desired fading rate is $f_D T_s = 0.01$ (a typical fast fading rate), then $\beta = 0.999$, and $\sigma_v^2 = 1.972 \times 10^{-3}$.

A final comment that illustrates the suitability of the model is in order. By projecting (10) τ time steps into the future, the expected value of a future channel state conditioned on the current value is given by

$$\mathsf{E}[\mathbf{h}_{(k+\tau)} | \mathbf{h}_k] = \beta^\tau \mathbf{h}_k. \quad (14)$$

For a β value near one, then $\mathbf{h}_{(k+\tau)} \approx \mathbf{h}_k$, i.e., the best guess about a future estimate is the current estimate. Note that this is precisely what is assumed by sending periodic training codes for the wireless channel; once the channel has been estimated, it is assumed to remain approximately constant until the next set of training data is sent. Significant changes over longer periods of time are expected but since the emphasis is on short-term prediction, we are not interested in the longer-term variation.

C. Incorporation of Dynamic Modelling in Particle Filtering

Examination of the state equation (10) assumes that the autoregressive coefficient β and the driving noise variance σ_ν^2 are static for all time t_k. This implies that the model is only accurate so long as the long-term statistics of the channel (for example, the fading rate) remain constant for all time. However, the short-term statistics of the channel may vary dramatically. To account for these short-term statistics we allow the parameters of the state equation itself to become time-varying. To justify this modification, consider Fig. 2, which shows the short-time evolution of a typical fast-fading wireless channel. This figure includes two deep fades which occurred at 0.14s and 0.6s respectively. Now, if we were to use the first-order AR model of (10) to account for the short-term evolution depicted in Fig. 2, then by virtue of the zero-mean assumption implicit in (10) we would find that the effect of the fading phenomena is practically ignored, i.e., the state equation's predictive capability is ill-suited to the channel during deep fades. Such an end result is clearly unacceptable, given the requirement to track the channel state as accurately as possible, which is an essential ingredient in the semi-blind strategy used for channel-state estimation.

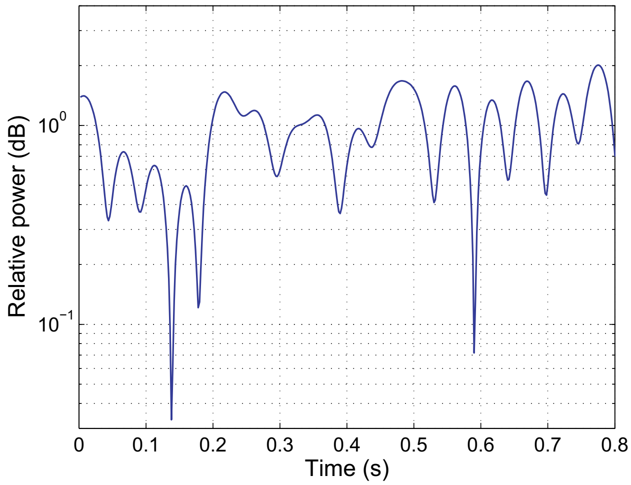


Fig. 2. A typical Doppler faded channel realization. For this fading rate ($f_D T_s = 0.01$) there are two deep fades at times 0.14s and 0.6s respectively. During these instances the static AR model of (10) is ill-suited for providing predictive channel estimates.

To incorporate a dynamic channel model in the channel-tracking strategy, we first rewrite the state-equation (10) as

$$\mathbf{h}_k = \mathbf{h}_{k-1} + (\beta - 1)\mathbf{h}_{k-1} + \mathbf{v}_k \quad (15)$$

and now define the new term

$$\boldsymbol{\mu}_k = (\beta - 1)\mathbf{h}_{k-1} \quad (16)$$

Accordingly we may account for the time-varying characterization of the AR coefficient β and noise variance σ_v^2 by introducing a *primitive equation*, which is modeled as the first-order Markov process

$$\boldsymbol{\mu}_k = \boldsymbol{\mu}_{k-1} + \mathbf{w}_k \quad (17)$$

where the new noise term $\mathbf{w}_k \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \sigma_w^2 \mathbf{I})$.

Our goal is now to track the primitive density function $p(\boldsymbol{\mu}_k)$. Given that both the state equation (15) and primitive equation (17) are linear and Gaussian, we can model the distribution of $p(\boldsymbol{\mu}_k)$, as a Gaussian distribution $\mathcal{N}_{\mathbb{C}}(\mathbf{m}_k, \mathbf{P}_k)$, and thus utilize a Kalman filter formulation to perform the tracking recursions (see Table I). Using this primitive density, we can now draw our estimate $\boldsymbol{\mu}_k \sim \mathcal{N}_{\mathbb{C}}(\mathbf{m}_k, \mathbf{P}_k)$ corresponding to the change in the wireless channel estimate. In effect, we have taken the static state parameters in equation (10) and shown how they may be reformulated as dynamic parameters (15) which are distributed as complex-Gaussian distribution.

A state-flow interpretation of the iterative estimation process is shown in Fig. 3. At time k, the primitive distribution $\mathcal{N}_{\mathbb{C}}(\mathbf{m}_{k-1}, \mathbf{P}_{k-1})$ is first propagated forward from time $k-1$ to k (step 1). In step 2, the predicted primitive distribution $\mathcal{N}_{\mathbb{C}}(\bar{\mathbf{m}}_k, \mathbf{Q}_k)$ is then used by the channel state prediction module to get the predictive samples $\mathbf{h}_{k|k-1}$ (these samples are used by the channel decoder to get $\hat{\mathbf{s}}_k$). Referring to Table I, using the estimated symbols obtained by the decoder (step 3) and the predicted samples obtained in step 2, the posterior distribution is now estimated via (6) and (8). Finally,

TABLE I

PROCEDURE FOR DYNAMIC STATE EQUATION UPDATES.

For time steps $k, k+1, k+2, \cdots$

1: Starting from posterior estimate for time $k-1$:

$$\mathcal{N}_{\mathbb{C}}(\mathbf{m}_{k-1}, \mathbf{P}_{k-1})$$

For some mean $\mathbf{m}_{k-1}$ and variance $\mathbf{P}_{k-1}$.

2: Update the prior distribution and perform prediction.

$$\mathcal{N}_{\mathbb{C}}(\mathbf{m}_{k-1}, \mathbf{P}_{k-1}) \rightarrow \mathcal{N}_{\mathbb{C}}(\bar{\mathbf{m}}_k, \mathbf{Q}_k)$$

where,

$$\bar{\mathbf{m}}_k = \mathbf{m}_{k-1} \quad (18)$$

$$\mathbf{R}_k = \mathbf{P}_{k-1} + \sigma_w^2 \mathbf{I} \quad (19)$$

$$\mathbf{Q}_k = \mathbf{R}_k + \sigma_v^2 \mathbf{I} \quad (20)$$

3: Posterior estimate for time k:

$$\mathcal{N}_{\mathbb{C}}(\bar{\mathbf{m}}_k, \mathbf{Q}_k) \rightarrow \mathcal{N}_{\mathbb{C}}(\mathbf{m}_k, \mathbf{P}_k)$$

where,

$$\mathbf{m}_k = \bar{\mathbf{m}}_k + \mathbf{R}_k \mathbf{Q}_k^{-1} \left[\hat{\mathbf{h}}_k - (\hat{\mathbf{h}}_{k-1} + \bar{\mathbf{m}}_k) \right] \quad (21)$$

$$\mathbf{P}_k = \mathbf{R}_k \mathbf{Q}_k^{-1} \sigma_v^2 \quad (22)$$

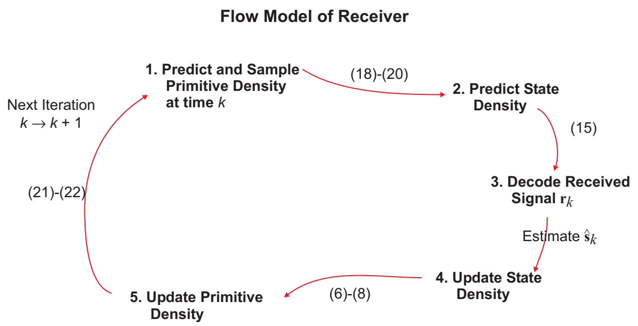


Fig. 3. Flow-graph illustrating how to incorporate a dynamical AR module. The goal is to model the changes in the channel as a Gaussian distribution whose mean and variance correspond to the parameters of the parametric AR(1) model given by (21) and (22).

the primitive density is updated according to (21, 22) in Table I, to reflect the new channel estimate. Since the primitive distribution is continually being updated according to the current channel estimates, it is 'aware' of changes in the underlying dynamics of the channel (i.e., it takes into account recency). It is this adaptation that allows the state prediction module (step 2) to produce a more accurate predictive channel estimate at time $k+1$ (For one iteration of the complete tracking algorithm, see Appendix I at the end of the paper).

Using convergence results arising from the limiting behavior of the recurrence relations (i.e., 18 through 22 in Table I), it can be shown that:

$$\text{As } k \rightarrow \infty, \quad \begin{cases} \mathbf{K}_k \triangleq \mathbf{R}_k \mathbf{Q}_k^{-1} \rightarrow \mathbf{K}, \\ \mathbf{P}_k \rightarrow \mathbf{P} = \mathbf{K}\sigma_v^2 \end{cases} \quad (23)$$

where,

$$\mathbf{K} = \frac{\varrho(\sqrt{1 + 4/\varrho} - 1)}{2} \mathbf{I} = \xi \mathbf{I}. \quad (24)$$

Here ϱ is the ratio $\frac{\sigma_w^2}{\sigma_v^2}$, which plays a role similar to the traditional signal-to-noise ratio of the transmitted signal to the channel noise. The value ξ in (24) is bounded by $0 \leq \xi \leq 1$ and is referred to as the *rate of adaption*. In the closed model given in (17) (closed in the sense that no additional information is obtained outside of the primitive equation), the adaption coefficient rapidly converges to a constant value. Using this convergence result provides insight into the nature of the adaptive gain. Using (18) and the first line of (23), we can rewrite (21) as

$$\mathbf{m}_k \approx \mathbf{K}(\hat{\mathbf{h}}_k - \hat{\mathbf{h}}_{k-1}) + (\mathbf{I} - \mathbf{K})\mathbf{m}_{k-1}, \quad (25)$$

where the approximation is good for large k. If ξ is close to unity ≈ 1, then only the current observation influences the model. Conversely, if $\xi \approx 0$, then current observations are almost completely rejected and the model degenerates into a pure random walk model. Therefore, there is a trade-off between the sensitivity of the predictor ($\xi \approx 1$) and robustness of the model ($\xi \approx 0$).

D. Determining σ_w^2 via discount factors

Clearly, it is important for the adaptive gain to converge to a 'suitable' value for the problem at hand. From (24), it is seen that the value of ξ reflects the relative variation between σ_w^2 and σ_v^2 through the parameter ϱ. The question is how to choose the proper ϱ, or more aptly, how to choose σ_w^2?

Using the convergence results just presented, (19) becomes $\mathbf{R} = \mathbf{P} + \sigma_w^2 \mathbf{I} = \mathbf{P}(\mathbf{I} - \mathbf{K})^{-1}$. Combining (19) and (22), the limiting behavior of the primitive variance is thus given by

$$\sigma_w^2 \mathbf{I} = \mathbf{K}\mathbf{P}(\mathbf{I} - \mathbf{K})^{-1}. \quad (26)$$

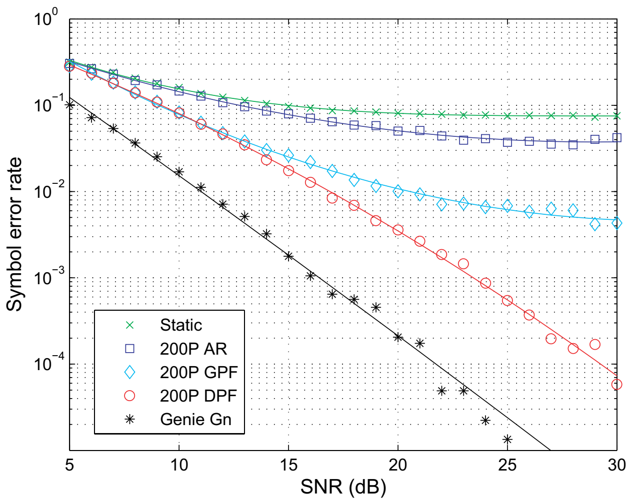
Thus between observations, the addition of the error $\mathbf{w}_k$ leads to an increase of $\mathbf{K}(\mathbf{I} - \mathbf{K})^{-1}$ of the initial variance $\mathbf{P}$; recall that the role of the system variance is to provide a diffusive effect. By defining the *discount factor* $\delta = 1 - \xi$, the choice of δ directly affects the adaption rate. For example, setting $\delta = 0.9$ and $\xi = 0.1$, we find that $\sigma_w^2 \mathbf{I}$ becomes roughly 11% of $\mathbf{P}$, that is, we are putting more weight on our model than on the observations. Since the limiting behavior is rapidly achieved, the discount factor essentially implies a constant rate of increase in uncertainty, for all k, not just in the limit. Thus for a given discount factor we can define the time-dependent primitive variance as

$$\sigma_{w,k}^2 \mathbf{I} = \frac{(1 - \delta)}{\delta} \mathbf{P}_{k-1}. \quad (27)$$

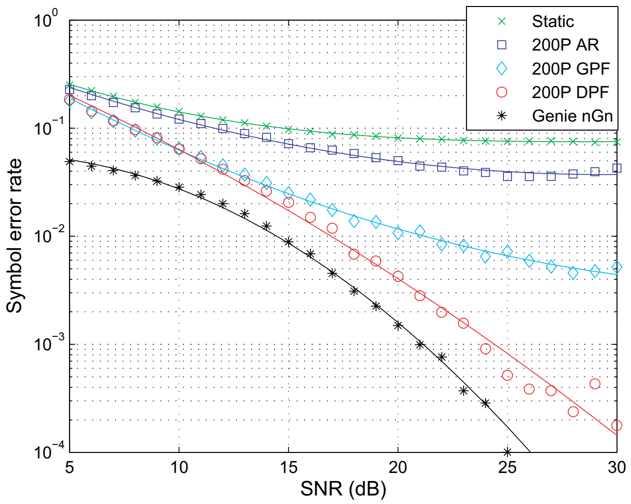
Using this method allows the model to adopt a time-varying structure. Specifically, if the dynamics of the wireless channel were to change, for example the Doppler fading rate were to change, the dynamic AR model is more able to cope with such a change than a static AR model. As mentioned previously, the procedure for a complete iteration of the tracking filter algorithm is shown in Appendix I.

IV. SIMULATIONS

We now present simulation experiments incorporating the dynamic linear modelling for wireless channel tracking. The simulations are carried out on a $N_t = 2$, $N_r = 3$ receiver system using a 4-PSK symbol set. The number of particles to use is a balance between choosing enough particles to reliably sample the state-space, and using too many which will not



(a) Gaussian Noise



(b) Non-Guassian Noise

Fig. 4. Symbol error rate vs. SNR for the different receiver types for both Gaussian (a), and non-Gaussian noise. In the highly dispersive channel the particle filter using the dynamic AR model provides the best results.

increase performance but increase complexity. In this work, the particle filter uses 200 particles. A zero-forcing space-time decoder is used for symbol estimation. The receiver uses a PAT training scheme where one training symbol is inserted for every 10 transmitted symbols. During a training sequence, the minimum-mean squared estimate (MMSE) of the channel is calculated from the training symbols. The MMSE estimate is then used to guide the particles by averaging the MMSE estimate with each particles' estimate of the channel. During a non-training interval, the particle filter will use the symbol estimate from the channel decoder in order to produce a corresponding channel estimate. Since the tracking algorithm is expected to accurately estimate the channel even when no training data are present, the receiver operates in a semi-blind mode. For this work, the channel transfer function is modelled as a frequency flat time-correlated function, and it is simulated using an improved version of Jakes fading model [24]. The particles were initially set equal to the first PAT estimate, and then resampled according to (10). The initial distribution for the primitive equation was set to $\mathbf{m}_0 = \mathbf{0}$ and $\mathbf{P}_0 = 10^{-4}\mathbf{I}$.

In an attempt to model the noise statistics of the urban wireless channel, two different distributions of receiver noise are used. The first is the widely accepted Gaussian distribution, and the second is distributed according to the previously discussed Middleton class-A model (2), with parameters $\epsilon = 0.1$ and $\kappa = 100$.

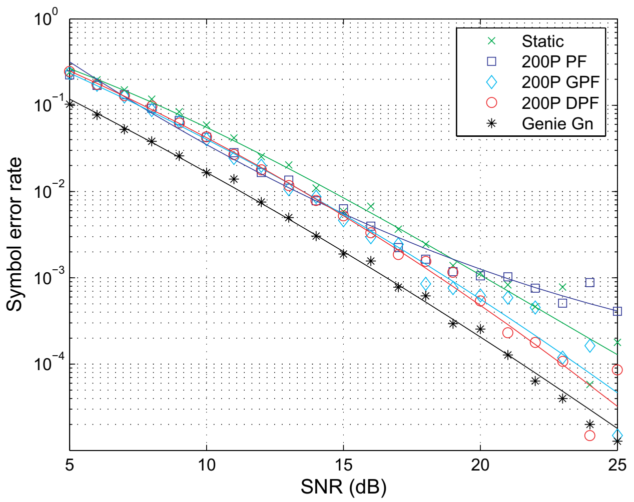
Figures 4(a) and 4(b) show the symbol error rate under a normalized Doppler rate (product of Doppler rate and sampling period) of $f_D t_s = 0.009$, in both Gaussian and non-Gaussian noise, respectively. This represents a highly time-dispersive channel. Immediately apparent is the failure of the decoder using the PAT method, i.e. static channel estimates, especially at high SNR where the performance rapidly reaches a lower bound. This is true for both the Gaussian and non-Gaussian distributions. The primary reason for the poor performance is simply that the channel is changing too rapidly. The only solution is to increase the number of training symbols; however, this is undesirable since it is wasteful of radio spectrum.

The bootstrap particle filter also performs poorly, rapidly achieving an asymptotic error floor. Here the primary cause of poor performance is that the simple static AR model of order one is insufficient to capture the dynamics of the rapidly fading channel. Since the bootstrap filter updates the weights based solely on the likelihood function in (8), the poorly resampled particles provide little support for obtaining a good posterior estimate.

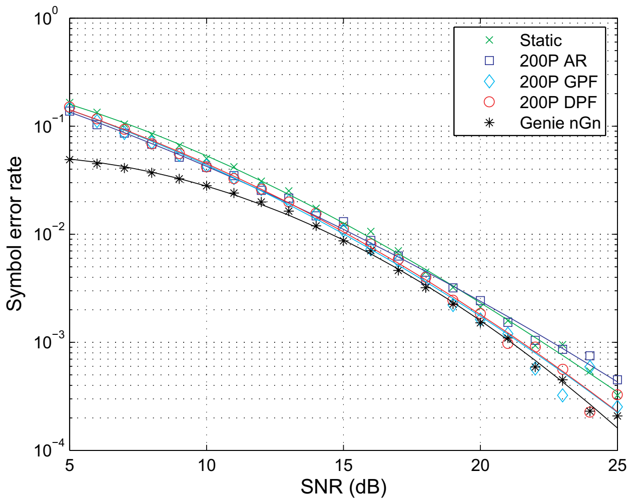
Looking at the gradient particle filter (GPF) [12], utilizing the static AR model, while the symbol error rate is much improved over the PAT method and bootstrap filter, performance (relative to the perfect known channel case) does tend to worsen as SNR increases. The improved performance of the GPF over the bootstrap filter is due to the incorporation of the gradient method which resamples the particles with higher accuracy. However, even with the improved sampling, owing to the static state model, we find that the performance does approach an error floor at high SNR.

In contrast, the particle filter incorporating the dynamic AR model is seen not only to outperform the static GPF receiver, but it also tracks the performance of the receiver using perfectly known channel estimates. There is no apparent divergence at high SNR as in the previous two methods (note: at very high SNR > 30dB for the non-Gaussian case a small divergence is observed; however, 30dB is very high for mobile wireless communications and the receiver is unlikely to ever see such a strong signal). Thus, the dynamic AR receiver has overcome much of the modelling limitations of the static AR model.

In Figs. 5(a) and 5(b), the normalized Doppler rate has been reduced to $f_D t_s = 0.001$. If we assume our cellular system is using a carrier of 2.1GHz and transmitting under a GSM/EDGE symbol rate [25], then this rate of fading would be expected for vehicular users operating on a freeway. Relative to the performance of the optimal known channel receiver, overall performance of all methods is close to the optimal performance. The channel has become less dispersive, and thus it presents an easier tracking task. The static method as well as the bootstrap PF, previously shown to be unsuited to track a highly dispersive channel, now perform well. However,



(a) Gaussian Noise



(b) Non-Guassian Noise

Fig. 5. Symbol error rate vs. SNR for the different receiver types for both Gaussian (a), and non-Gaussian noise. In the slowly fading channel, all receivers perform much better. Note however, the particle filter using the dynamic AR model still provides the best results.

even at this slower fading rate the bootstrap PF in Gaussian noise is starting to show signs of approaching an error floor. Despite the improved performance of all receiver methods, the dynamic AR model still offers a performance increase relative to the static AR model.

Returning now to the simulations, in Fig. 6, we show the tracking performance of the different receiver types as the dispersive nature of the channel increases. The signal-to-noise ratio of the channel was set at 20 dB and the noise distribution was Gaussian. Initially, when the normalized Doppler rate is small (< 0.001), the channel remains approximately constant between successive training symbols. Little adaptation is necessary, and consequently all receiver tracking schemes perform satisfactorily. However, as the Doppler rate increases, the channel becomes more dispersive and tracking the channel becomes increasingly difficult. Eventually, at very high fading rates (> 0.02), even the dynamic AR model will begin to show poor performance. However, for current cellular systems, the upper Doppler limit we might expect to encounter would be 0.01. Accordingly, we are justified by saying that the dynamic AR model is robust for realistic wireless situations, which, in the final analysis, is what matters in practice.

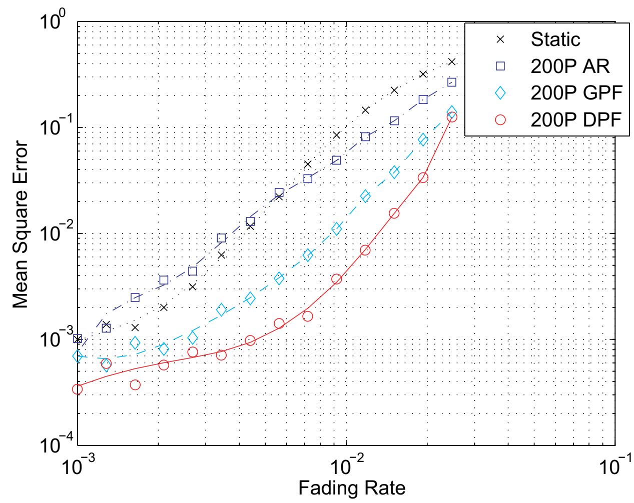


Fig. 6. Mean squared error tracking performance (averaged over all channels) vs. normalized Doppler rate.

A. Comments on algorithmic performance

1) A proper choice of δ: Recall that the role of the discount factor δ introduced in (27), is to provide a measure of the rate of increase of uncertainty (or loss of information). Picking the 'optimal' δ is problem-specific and it generally requires an experimental approach. However, when the goal is short-term prediction, the discount factor is typically in the range ($0.8 \leq \delta \leq 1$), where the main benefit is derived from data smoothing. Returning to the wireless case, at low SNR where, because of the high noise variance, we have little belief in each channel estimate, we adopt $\delta \sim 1$ corresponding to a small rate of loss of information. Equivalently, in a high-noise condition, we choose to do more smoothing and put less weight on each filtered estimate. Similarly, in high SNR, we decrease δ; thus we are allowing the model to readily adapt to each new channel estimate, and put less emphasis on 'older' measurements. In all the experiments performed in this paper, at a SNR of 5 dB, the discount factor δ was set to 0.999. Then as the SNR increases, δ is steadily decreased to 0.89 at 25dB. Experimentally, the discount factor in this range was found to produce the best performance results. These observations suggest the need for devising a practical scheme that selects δ on-the-fly in response to changes in signal-to-noise ratio.

2) Quality of the channel estimate: Information-theoretic work by Lapidoth and Shami [26] examined how the effect of estimation errors in the side information, or the fading process, might affect the robustness of a communication scheme. The authors conclude that in order to avoid performance degradation, the second moment of the estimation error should be small compared to the reciprocal of the SNR of the channel. Specifically, they show the generalized mutual information

I_{GMI} for any single channel h_{ij} is given by

$$I_{GMI} = \mathsf{E}\left[\log\left(1+\frac{\mathcal{E}_s|\hat{h}_{ij}|^2}{\sigma^2+\mathcal{E}_s\mathsf{E}\left[|h_{ij}-\hat{h}_{ij}|^2\right]}\right)\right] \quad (28)$$

where $\mathcal{E}_s$ is the symbol energy, σ^2 is the noise power, $\bar{h}_{ij}$ is the true value of the channel from transmit antenna i to receive antenna j, and $\hat{h}_{ij}$ is the corresponding estimate. Of interest is the term $e = \mathsf{E}\left[|h_{ij}-\hat{h}_{ij}|^2\right]$ in the denominator of (28), which denotes the error in the estimate. If we now assume that the channel tracker is providing perfect estimates, then $e \to 0$ and (28) simplifies to the well known capacity result where the receiver has perfect knowledge of the fading information [26]. In our paper we experimentally confirm this intuitively-satisfying theoretical result. If we assume that 'a small difference' between the channel estimate and the inverse of the SNR is given by at least an order of magnitude, that is, $\frac{e}{SNR^{-1}} \leq 0.1$, then from Fig. 6 we would expect that at 20dB, the particle filter incorporating the dynamic model performs almost at the known channel bound for fading rates of 0.005 or less. Noticeable degradations will occur for fading rates above this value. Returning now to Figs. 4 and 5, we see that the performance is nearly equivalent to the known channel performance for a fading rate of 0.001, while for the fast fading rate of 0.009 a noticeable deviation is perceived.

Note that (28) also serves to help explain the error floor phenomena for static models. As the SNR becomes large, the value of σ^2 in the denominator of (28) becomes negligible, leaving only the error in the channel estimate e. Since, by definition, a static model will not change, the diffusive effect caused by the driving noise in the predictive equation (10) will eventually come to be the dominant source of error (the modelling error). Thus even if the SNR tends to infinity, the estimation error will remain fixed, in which case the achievable rates will be bounded in power and will therefore not grow to infinity. By contrast, the dynamic method allows the driving noise variance of the predictor model to vary as channel conditions require; thus it is able to decrease the driving noise variance (and thereby provide better performance) as the SNR becomes large.

V. Conclusion

This paper provides convincing evidence for the benefit of incorporating dynamic linear modelling techniques for use in tracking a rapidly changing MIMO wireless channel. The AR model of the channel, conventionally assumed to be static, is now recast to allow for a time-varying behavior. The time-varying parameter is modelled by a new equation, referred to as the primitive equation, which assumes a Markovian nature. The unknown dynamic noise variance in the primitive equation is obtained using the method of discount factoring, which is well-known in the Bayesian forecasting literature. Modelling the wireless channel as an ordered couple of equations allows the receiver to dynamically adapt to the time-varying behavior of the channel in a successful manner. In effect, incorporating the primitive equation in the AR state space model of the channel allows a Kalman filter to track statistical variations of the channel. In this way an improved estimate of the channel is provided to the particle filter for tracking the channel.

The superiority of the dynamic AR model was particularly emphasized in the difficult to track, highly time-dispersive channels. In these conditions, at high signal-to-noise ratios traditional static models tend to suffer an error-rate floor in performance. Information-theoretic results from [26] serve to justify and explain why tracking algorithms which incorporate a static auto-regressive model tend to produce an error floor. By contrast, the use of a dynamic auto-regressive model is seen to overcome the error floor phenomenon. It does so by being able to adapt the variance of the driving noise in the state equation according to current channel conditions.

In addition to the results presented in this paper using simulated channel data, experimental results based on real-life channel data have also shown that the dynamic auto-regressive model offers a performance increase relative to the traditional static auto-regressive model [27].

Appendix I
Pseudo-code for Iterative Procedure for Wireless Channel Tracking Algorithm.

For time steps $k, k+1, k+2, \cdots$

1: Starting from primitive and state posterior estimate for time $k-1$, respectively:

$$\mathcal{N}_{\mathbb{C}}(\mathbf{m}_{k-1}, \mathbf{P}_{k-1}) \text{ and } p(\mathbf{h}_{k-1}|\mathbf{r}_{k-1})$$

For some mean $\mathbf{m}_{k-1}$ variance $\mathbf{P}_{k-1}$, particles $\mathbf{h}_k^i$ and associated weights w_k^i.

2: Update the prior distribution and perform prediction using (18)-(20) and (27).

$$\mathcal{N}_{\mathbb{C}}(\mathbf{m}_{k-1}, \mathbf{P}_{k-1}) \to \mathcal{N}_{\mathbb{C}}(\bar{\mathbf{m}}_k, \mathbf{Q}_k)$$

where,

$$\bar{\mathbf{m}}_k = \mathbf{m}_{k-1}$$
$$\mathbf{R}_k = \mathbf{P}_{k-1} + \sigma_w^2\mathbf{I} = \mathbf{P}_{k-1} + \frac{(1-\delta)}{\delta}\mathbf{P}_{k-1}$$
$$\mathbf{Q}_k = \mathbf{R}_k + \sigma_v^2\mathbf{I}$$

3: Using the method of importance sampling, predict the state density by propagating particles $\ell = 1, ..., L$, from time $k-1$ to k using (15),

$$\mathbf{h}_k^\ell = \mathbf{h}_{k-1}^\ell + \boldsymbol{\mu}_k^\ell + \mathbf{v}_k^\ell$$

where, $\boldsymbol{\mu}_k^\ell \sim \mathcal{N}_{\mathbb{C}}(\bar{\mathbf{m}}_k, \mathbf{Q}_k)$

4: Using the channel decoder, decode the received signal $\mathbf{r}_k$, using predicted channel estimate $\hat{\mathbf{h}}_k$. The output of the detector are the estimated transmit symbols $\hat{\mathbf{s}}_k$.

5: Evaluate the weights (8), of the particles obtained in Step 3:

$$w_k^\ell = w_{k-1}^\ell \frac{p(\mathbf{r}_k|\mathbf{h}_k^\ell)p(\mathbf{h}_k^\ell|\mathbf{h}_{k-1}^\ell)}{q(\mathbf{h}_k^\ell|\mathbf{h}_{k-1}^\ell, \mathbf{r}_k)}.$$

Calculate N_{eff} and resample if necessary. The channel posterior estimate is given by (6).

$$p(\mathbf{h}_k|\mathbf{r}_k) \simeq \sum_{\ell=1}^{L} w_k^\ell \delta(\mathbf{h}_k - \mathbf{h}_k^\ell)$$

6: Calculate primitive posterior estimate $\mathcal{N}_{\mathbb{C}}(\mathbf{m}_k, \mathbf{P}_k)$ for time k using (21)-(22):

$$\mathcal{N}_{\mathbb{C}}(\bar{\mathbf{m}}_k, \mathbf{Q}_k) \rightarrow \mathcal{N}_{\mathbb{C}}(\mathbf{m}_k, \mathbf{P}_k)$$

where,

$$\mathbf{m}_k = \bar{\mathbf{m}}_k + \mathbf{R}_k \mathbf{Q}_k^{-1} \left[\hat{\mathbf{h}}_k - (\hat{\mathbf{h}}_{k-1} + \bar{\mathbf{m}}_k) \right]$$
$$\mathbf{P}_k = \sigma_v^2 \mathbf{R}_k \mathbf{Q}_k^{-1}$$

7: Increase time increment $k \rightarrow k + 1$, return to step 2.

ACKNOWLEDGEMENTS

The authors would like to thank the reviewers for their helpful comments. The financial support provided by the Natural Sciences and Engineering Research Council of Canada is deeply appreciated.

REFERENCES

[1] L. Zheng and D. Tse, "Diversity and multiplexing: A fundamental tradeoff in multiple-antenna channels," *IEEE Trans. Inform. Theory*, vol. 49, no. 5, pp. 1073–1096, May 2003.

[2] J. K. Cavers, "An analysis of pilot symbol assisted modulation for Rayleigh fading channels," in *Proc. IEEE Vehicular Technology Conference*, vol. 40, no. 4, pp. 686–693, Dec. 1991.

[3] L. Tong, B. M. Sadler, and M. Dong, "Pilot-assisted wireless transmissions," *IEEE Signal Processing Mag.*, vol. 21, no. 6, pp. 12–25, Nov. 2004.

[4] D. Samardzija and N. Mandayam, "Pilot-assisted estimation of MIMO fading channel responses and achievable data rates," *IEEE Trans. Signal Processing*, vol. 51, no. 11, pp. 2882–2890, Nov. 2003.

[5] K. Huber and S. Haykin, "Application of particle filters to MIMO wireless communications," in *Proceedings IEEE International Conference on Communications*, pp. 2311–2315, May 2003.

[6] Z. Liu, X. Ma, and G. Giannakis, "Space-time coding and Kalman filtering for time-selective fading channels," *IEEE Trans. Commun.*, vol. 50, no. 2, pp. 183–186, February 2002.

[7] X. Wang, R. Chen, and D. Guo, "Delayed-pilot sampling for mixture Kalman filter with application in fading channels," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 241–254, Feb. 2002.

[8] C. Komninakis *et al.*, "Multi-input multi-output fading channel tracking and equalization using Kalman estimation," *IEEE Trans. Signal Processing*, vol. 50, no. 5, pp. 1065–1076, May 2002.

[9] K. E. Baddour and N. C. Beaulieu, "Autoregressive models for fading channel simulation," in *Proceedings of the IEEE Global Telecommunications Conference*, pp. 1187–1192, November 2001.

[10] K. Blackard, T. Rappaport, and C. Bostian, "Measurements and models of radio frequency impulsive noise for indoor wireless communications," *IEEE J. Select. Areas Commun.*, vol. 11, no. 7, pp. 991–1001, Sept. 1993.

[11] M. S. Arulampalam *et al.*, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[12] S. Haykin, K. Huber, and Z. Chen, "Bayesian sequential state estimation for MIMO wireless communications," in *Proceedings of the IEEE*, vol. 92, no. 3, pp. 439–455, Ma. 2004.

[13] W. H. Chin, D. B. Ward, and A. G. Constantinides, "Semi-blind MIMO channel tracking using auxiliary particle filtering," in *Proceedings of the IEEE Global Telecommunications Conference*, vol. 1, pp. 322–325, Nov. 2002.

[14] M. West and J. Harrison, *Bayesian Forecasting and Dynamic Models, 2nd edition*. Springer-Verlag, New York, 1989.

[15] P. M. Djurić and J. Kotecha, "Estimation of time-varying non-Gaussian autoregressive processes by particle filters with forgetting factors," *Department of Electrical and Computer Engineering, State university of New York at Stony Brook, Stony Brook, NY 11794*, USA, 2002.

[16] G. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Commun.*, vol. 6, no. 3, pp. 311–315, Mar. 1998.

[17] X. Wang and V. Poor, "Robust multiuser detection in non-Gaussian channels," *IEEE Trans. Signal Processing*, vol. 47, no. 2, pp. 289–305, Feb. 1999.

[18] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statististical Computing*, vol. 10, pp. 197–208, 2000.

[19] P. M. Djurić *et al.*, "Particle filtering," *IEEE Signal Processing Mag.*, pp. 19–38.

[20] Y. Rui and Y. Chen, "Better proposal distributions: Object tracking using unscented particle filter," *IEEE International Conference on Computer Vision*, 2001.

[21] A. Kong, J. S. Liu, and W. H. Wong, "Sequential imputations and Bayesian missing data problems," *Journal of the American Statistical Association*, vol. 89, no. 425, pp. 278–288, Mar. 1994.

[22] P. A. Bello, "Characterization of randomly time-variant linear channels," *IEEE Trans. Commun. Systems*, vol. CS–11, pp. 360–393, Dec. 1963.

[23] H. Wang and P. Chang, "On verifying the first order Markovian assumption for a Rayleigh fading channel model," *IEEE Trans. Vehic. Technol.*, vol. 45, no. 2, pp. 353–357, May 1996.

[24] P. Dent, G. E. Bottomley, and T. Croft, "Jakes fading model revisited," *IEEE Electronics Letters*, vol. 29, no. 13, pp. 1162–1163, 24th June 1993.

[25] A. Furuskar, S. Mazur, F. Muller, and H. Olofsson, "EDGE: Enhanced data rates for GSM and TDMA/136 evolution," *Personal Communications*, vol. 6, no. 3, pp. 56–66, June 1999.

[26] A. Lapidoth and S. Shamai, "Fading channels: How perfect need 'perfect side information' be?" *IEEE Trans. Inform. Theory*, vol. 48, no. 5, pp. 1118–1134, May 2002.

[27] K. Huber, "Bayesian Sequential State Estimation for MIMO Wirless Communications," Ph.D. dissertation, Department of Electrical and Computer Engineering, McMaster University, Hamilton, Apr. 2005, Available: http://soma.mcmaster.ca/huber

Kris Huber received his B.Sc. degree in 1997 from the Department of Physics, the M.Eng. in 2000 and his Ph.D degree in 2005 from the Department of Computer and Electrical Engineering at McMaster University, Hamilton, Ontario, Canada. In 2000, he spent a year at Bell Labs, Lucent Technologies, in New Jersey where he helped develop one of the first MIMO test-beds designed to measure the complex channel coefficients between a 16-element transmit array and a 16-element receive array.

Dr. Huber currently works at Array Systems Computing in Toronto, Ontario, where he manages the Software Defined Radio (SDR) research and development team. This work is focused on developing and bringing to market wireless communications protocols (such as DVB-S2 and IEEE 802.16) adapted to operate with any radio platform made compliant to the Software Communications Architecture Core Framework (SCA-CF) standard.

Simon Haykin Simon Haykin received his B.Sc. (First-class Honours), Ph.D., and D.Sc., all in Electrical Engineering from the University of Birmingham, England. He is the author of numerous books, including the most widely used books: Communication Systems (4th edition, Wiley), Adaptive Filter Theory (4th edition, Prentice-Hall), Neural Networks: A Comprehensive Foundation (2nd edition, Prentice-Hall) and the about to be published new book on Adaptive Radar: Toward the Development of Cognitive Radar (Wiley), as well as numerous refereed journal papers. He is a Fellow of the Royal Society of Canada, recipient of the Honourary Degree of Doctor of Technical Sciences from ETH, Zurich, Switzerland, and the Henry Booker Gold Medal from URSI, as well as other prizes and awards. Currently, he holds the title Distinguished University Professor in the ECE Department at McMaster University, Canada.