Efficient Training of Kalman Algorithm for MIMO Channel Tracking

Emna Eitel and Joachim Speidel
Institute of Telecommunications, University of Stuttgart
Stuttgart, Germany

***Abstract*—In this paper, a Kalman algorithm is applied to track a time-varying flat fading MIMO channel. The importance of training and appropriate initialization in combination with the Kalman tracking algorithm is shown. Adopting a periodical training scheme with a given bandwidth efficiency, a trade-off between investing pilots for good initialization and training the algorithm exclusively leads to the lowest BER. We also introduce a training on request scheme, in order to overcome the error propagation encountered by the Kalman filter after a series of detection errors. For this purpose, two metrics to detect the Kalman filter divergence are developed. We show the effectiveness of the new aperiodical training scheme in reducing the channel estimation error and saving bandwidth at the same time.**

I. Introduction

MIMO systems with coherent detection can deliver high bit rates provided that an accurate knowledge of the channel is available at the receiver. The performance can even be enhanced if the channel state information (CSI) is also available at the transmitter. Algorithms to precisely estimate the CSI are therefore of paramount importance. Often periodical pilot-assisted channel estimation (PACE) is employed. However, in fast-varying channels, PACE does not only decrease the bandwidth efficiency but is also incapable of detecting fast variations of the channel. Therefore, additional tracking techniques have to be applied. A method that does not require pilots is decision-directed channel estimation. It uses previously detected symbols and can therefore feed the channel estimation module with new measurements that permanently reflect the current channel state. Exploiting detected data, adaptive filtering techniques such as Kalman filter (KF), least mean squares (LMS) or recursive least squares (RLS) filter can also be used for channel tracking. In combination with a high-order autoregressive (AR) channel model, the KF shows the best performance among them but at the expense of higher complexity. However, low-order AR models can capture most of the channel dynamics for small estimation lags, as it is the case for symbolwise tracking, and lead to effective tracking performance [1]. A drawback of the KF is its lack of robustness with respect to wrongly detected data. To cope with this problem, periodical pilot patterns are often inserted to stop the filter divergence. An alternative solution exploits reliability information about detected data [2]. But this approach requires iterative receiver structures which introduce a significant delay and a high complexity [2], [3].

In this paper, we improve the tracking performance of the KF by two means. First, we show that in case of periodical training, investing a fraction of the available training data to provide the KF with an appropriate initialization fastens the filter convergence and improves the tracking performance. Second, we introduce a novel aperiodical training scheme that applies training when needed, i.e. on request. A request for pilots is initiated in case the filter diverges as a consequence of many outliers causing error propagation. Two detection methods for the error propagation are proposed and evaluated. It is clear that dropping the periodical training scheme, transmitter and receiver have the burden of a more complicated signalling task. Nevertheless, our approach is applicable independently of the detection scheme. Besides, we show that the novel aperiodical training leads to a significant tracking performance improvement and more than 50% reduction of required training data.

II. System Model

We consider an $M \times N$ MIMO system. The $N \times 1$ receive signal vector at time instant n is given by:

$$\mathbf{y}(n) = \mathbf{H}(n)\mathbf{s}(n) + \mathbf{w}(n) \tag{1}$$

where $\mathbf{s}(n)$ denotes the $M \times 1$ sent signal vector, $\mathbf{H}(n)$ the $N \times M$ MIMO flat fading channel matrix and $\mathbf{w}(n)$ the $N \times 1$ additive white Gaussian noise (AWGN) vector whose complex elements are i.i.d and $CN(0, 2\sigma_0^2)$. Without loss of generality, we assume a spatially uncorrelated MIMO Rayleigh fading channel. An element $h_{ij}(n)$ of $\mathbf{H}(n)$ represents the channel coefficient between the *j*th transmit and *i*th receive antenna and is $CN(0, 1)$ distributed. The temporal autocorrelation function of $h_{ij}(n)$ satisfies:

$$E\left\{h_{ij}(n)h_{ij}(n')^*\right\} = J_0(2\pi f_d(n-n')) \tag{2}$$

where f_d stands for the normalized Doppler frequency and J_0 is the Bessel function of first kind and order zero. In order to estimate the channel at the receiver, orthogonal pilot symbol vectors $\mathbf{s}_p$ are periodically sent during the training period that takes L_p symbol intervals T_s. At the end of the training phase, a channel estimate $\hat{\mathbf{H}}_\mathbf{p}$ is computed by means of the received pilots according to the maximum likelihood or the minimum mean squared error principle. The training phase is followed by a data transmission phase where L_d symbol vectors are sent. In the absence of tracking, the PACE estimate is used for the coherent detection of data symbols during the subsequent L_d symbol periods. We introduce the discrete time index τ to denote the time elapsed between the end of the training

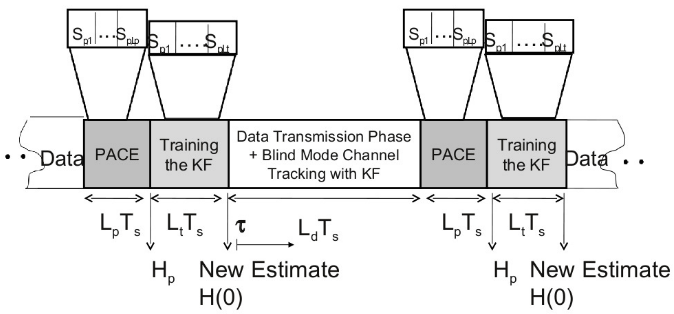


Fig. 1. Alternating training and data phases in the hybrid training scheme

phase and the end of the data transmission phase, i.e. $0 \leq \tau \leq L_d$, as shown in Fig. 1. In case of channel tracking, the PACE estimate can be used as good initial value for the tracking algorithm at the start of every training interval. A basic prerequisite is a recursive tracking algorithm, as it is the case for the KF.

Doing so, the question that arises is how good the initialization has to be. In case of slow fading, increasing the number of pilots improves the PACE estimate and we can expect a faster KF convergence. However, if the channel is varying fast, the PACE estimate has to be built upon a few pilots. Therefore, we first give a deeper insight into PACE in order to determine the optimal training length. On the other hand and to the best of our knowledge, KF training in the literature is only applied to the filter itself [4], [5], [2]. In other words, the measurements from periodically sent pilots are used as input to train the statistical variables of the algorithm. Instead, we adopt a hybrid training scheme. This approach was first introduced in [6] and has proved to significantly improve the tracking performance of the RLS algorithm. Motivated by these results and the existing correspondences between RLS and KF [7], we applied the hybrid training scheme in [6] to the KF. The simulations results in Section VII confirm the expected performance improvement.

Additionally, we show that if the mean squared error (MSE) of the KF tracking at the end of the frame is smaller than the PACE MSE, the initialization with PACE is disadvantageous. In this case, pilots are not needed anymore and the KF can operate in a quasi-blind decision-directed mode. Due to the KF sensitivity to wrongly detected data, pilots have still to be requested in case of error propagation, which we introduce as a new aperiodical, pilot-on-request training scheme.

III. Pilot-Assisted Channel Estimation

During the training phase, pilot symbol vectors $\mathbf{s}_{p,i}$ with $1 \leq i \leq L_p$ are transmitted. The corresponding received $\mathbf{y}_{p,i}$ are impaired by AWGN vectors $\mathbf{w}_{p,i}$. From (1) follows:

$$\mathbf{y}_{p,i} = \mathbf{H}\mathbf{s}_{p,i} + \mathbf{w}_{p,i} \qquad \text{for} \quad 1 \leq i \leq L_p. \tag{3}$$

Assembling all pilot symbol vectors $\mathbf{s}_{p,i}$, all corresponding receive symbol vectors $\mathbf{y}_{p,i}$ as well as $\mathbf{w}_i$ in matrices and assuming that the channel does not change during the PACE training phase, we get

$$\mathbf{Y}_p = \mathbf{H}\mathbf{S}_p + \mathbf{W}_p. \tag{4}$$

with $\mathbf{S}_p = \left[\mathbf{s}_{p,1} \ \cdots \ \mathbf{s}_{p,L_p}\right]$, $\mathbf{Y}_p = \left[\mathbf{y}_{p,1} \ \cdots \ \mathbf{y}_{p,L_p}\right]$ and $\mathbf{W}_p = \left[\mathbf{w}_{p,1} \ \cdots \ \mathbf{w}_{p,L_p}\right]$. The PACE estimate is computed upon the received pilots according to

$$\hat{\mathbf{H}}_p^{ML} = \mathbf{Y}_p \cdot \mathbf{S}_p^H \cdot \left(\mathbf{S}_p\mathbf{S}_p^H\right)^{-1} \tag{5}$$

for the ML estimate, where $(.)^H$ refers to the Hermitian of a matrix. If knowledge about the SNR and the channel spatial correlation properties is available at the receiver, a better PACE estimate can be computed according to the MMSE principle. Therefore, we rewrite (4) in vector form to apply standard results from estimation theory:

$$\underbrace{vec\left(\mathbf{Y}_p\right)}_{\hat{\mathbf{y}}_p} = \underbrace{\left(\mathbf{S}_p^T \otimes \mathbf{I}\right)}_{\mathbf{X}_p} \cdot \underbrace{vec\left(\mathbf{H}\right)}_{\mathbf{h}} + \underbrace{vec\left(\mathbf{W}_p\right)}_{\tilde{\mathbf{w}}} \tag{6}$$

where $\otimes$ is the Kronecker product. The MMSE estimate $\hat{\mathbf{h}}_p^{MMSE} = vec(\hat{\mathbf{H}}_p^{MMSE})$ is given by:

$$\hat{\mathbf{h}}_p^{MMSE} = \mathbf{R_{hh}}\mathbf{X}_p^H \left(\mathbf{X}_p\mathbf{R_{hh}}\mathbf{X}_p^H + \mathbf{R}_{\tilde{\mathbf{w}}\tilde{\mathbf{w}}}\right)^{-1} \tilde{\mathbf{y}}_\mathbf{p} \tag{7}$$

where $\mathbf{R_{hh}} = E\left\{\mathbf{h}\mathbf{h}^H\right\}$ and $\mathbf{R}_{\tilde{\mathbf{w}}\tilde{\mathbf{w}}} = E\left\{\tilde{\mathbf{w}}\tilde{\mathbf{w}}^H\right\}$. When using PACE for the initialization of the tracking algorithm, we have to get a deeper insight into its estimation quality. An appropriate means to do so is to consider the channel estimation MSE $\zeta(\tau)$, which we define throughout this paper by:

$$\zeta(\tau) = \frac{1}{MN} E\left\{\|\mathbf{H}(\tau) - \hat{\mathbf{H}}\|_F^2\right\} \tag{8}$$

where $\|\cdot\|_F$ is the Frobenius norm and $\hat{\mathbf{H}} = \hat{\mathbf{H}}_\mathbf{p}$ in case of PACE. We now take account of the channel time variations during the training phase. With orthogonal training data , i.e. $\mathbf{S}_p\mathbf{S}_p^H = L_p\mathbf{I}_M$ and the channel according to (2), we derive the ML mean squared estimation error as in (9) [1]. A similar expression was derived in [8] but only for one tx antenna. Our expression holds for an arbitrary number M of tx antennas.

$$\zeta(\tau) = \underbrace{\frac{M}{L_p^2}\sum_{i=1}^{L_p}\xi(i,\tau)}_{\zeta_1(\tau)} + \underbrace{\frac{2\sigma_0^2}{L_p}}_{\zeta_2} \tag{9}$$

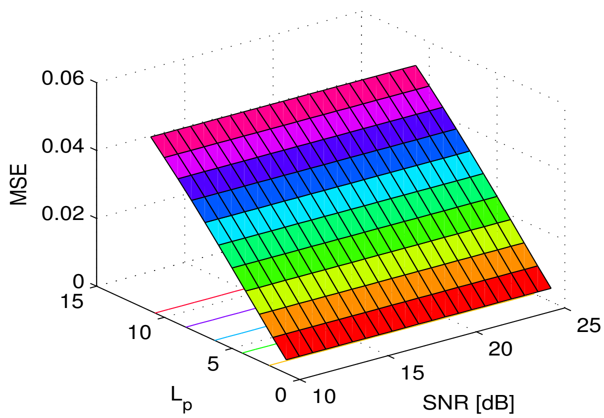
where $\xi(i,\tau) = 2(1 - J_0(2\pi f_d(\tau + L_p - i)))$. (9) shows that the PACE MSE is composed of two quantities: $\zeta_1(\tau)$ depending on the channel time variance and ζ_2 which is related to the AWGN. We see that increasing the number of pilots L_p decreases ζ_2 but may increase $\zeta_1(\tau)$. Especially for high f_d and depending on the SNR an optimal L_p that minimizes (9) exists. This is illustrated in Fig. 2, where $\zeta(0)$ is plotted as a function of the SNR and L_p for $f_d = 0.02$. $\tau = 0$ is considered since $\zeta(0)$ is of interest when using PACE for the initialization of the tracking algorithm.

[1] For space reasons, we only give $\zeta(\tau)$ for ML estimation. $\zeta(\tau)$ for the MMSE estimate in (7) can be derived analogously

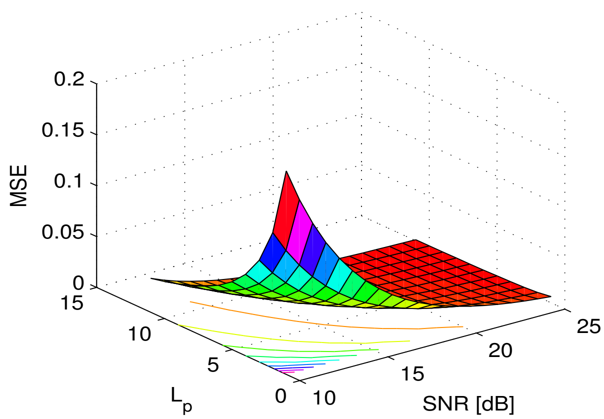
If we use the approximation $J_0(x) \approx 1 - x^2/4$ for $x \ll 1$, and set the first derivative of (9) with respect to L_p to zero, the optimal training length which minimizes (9) can be derived to:

$$L_{p,opt} = \begin{cases} \left\lfloor \sqrt{\frac{1}{2} + \frac{3\sigma_0^2}{\pi^2 f_d^2 M}} \right\rfloor & \text{if } \sqrt{\frac{1}{2} + \frac{3\sigma_0^2}{\pi^2 f_d^2 M}} > M \\ M & \text{otherwise} \end{cases} \tag{10}$$

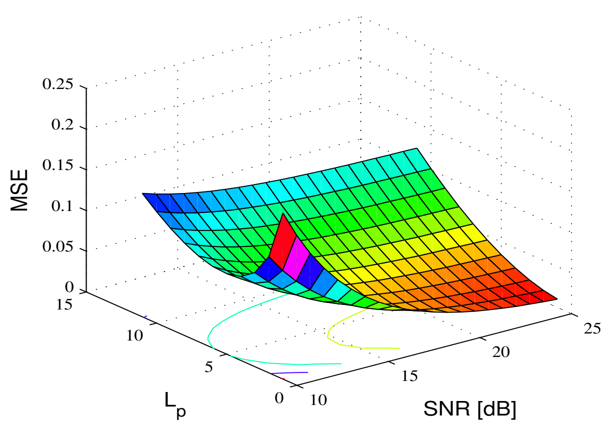
where $\lfloor \cdot \rfloor$ refers to the floor operation. (10) shows that $L_{p,opt}$ increases with increasing σ_0^2 and decreases with increasing f_d or M. We should keep in mind that $L_p \geq M$ must be satisfied which is a necessary condition for the inversion in (5).



(a) MSE $\zeta_1(0)$



(b) MSE ζ_2



(c) MSE $\zeta(0) = \zeta_1(0) + \zeta_2$

Fig. 2. PACE MSE as function of SNR and L_p for $f_d = 0.02$

IV. The Kalman Algorithm

If the fading channel can be modeled as an autoregressive process of order p (AR(p)), then the KF is the optimal MMSE estimator. However, since the first few channel correlation terms in (2) are basically important for symbolwise tracking, AR(2) modeling is adopted in this work as in [1]. The Kalman algorithm relies on a state-space formulation composed of the observation equation (11) and the process equation(12).

$$\mathbf{y}(n) = \mathbf{X}(n) \cdot \mathbf{z}(n) + \mathbf{w}(n) \tag{11}$$

$$\mathbf{z}(n) = \mathbf{F}\mathbf{z}(n-1) + \mathbf{B}\mathbf{u}(n) \tag{12}$$

where $\mathbf{z}(n) = [\mathbf{h}^T(n) \quad \mathbf{h}^T(n-1) \quad \cdots \quad \mathbf{h}^T(n-p+1)]^T$ with $\mathbf{h}(n) = vec(\mathbf{H}(n))$ and $\mathbf{F}$ is the state transition matrix. $\mathbf{X}(n)$ contains the detected symbol vector $\hat{\mathbf{s}}(n)$ according to $\mathbf{X}(n) = [\hat{\mathbf{s}}(n) \otimes \mathbf{I}_N \quad \mathbf{O}_{N \times NM(p-1)}]$. $\mathbf{u}$ is the driving noise with $E[\mathbf{u}(n)\mathbf{u}(n)^H] = \mathbf{I}_{MN}$ [2]. We briefly list the key equations of the Kalman tracking algorithm:

- Predicted channel state

$$\hat{\mathbf{z}}(n|n-1) = \mathbf{F}\hat{\mathbf{z}}(n-1|n-1) \tag{13}$$

- Predicted MSE

$$\mathbf{P}(n|n-1) = \mathbf{F}\mathbf{P}(n-1|n-1)\mathbf{F}^H + \mathbf{B}\mathbf{B}^H \tag{14}$$

- Kalman Gain

$$\mathbf{K}(n) = \mathbf{P}(n|n-1)\mathbf{X}^H(n) \cdot \left(\mathbf{X}(n)\mathbf{P}(n|n-1)\mathbf{X}^H(n) + \mathbf{R}_{\mathbf{ww}}\right)^{-1} \tag{15}$$

- Corrected channel state

$$\hat{\mathbf{z}}(n|n) = \hat{\mathbf{z}}(n|n-1) + \mathbf{K}(n)(\mathbf{y}(n) - \mathbf{X}(n)\hat{\mathbf{z}}(n|n-1)) \tag{16}$$

- Corrected MSE

$$\mathbf{P}(n|n) = (\mathbf{I} - \mathbf{K}(n)\mathbf{X}(n))\mathbf{P}(n|n-1) \tag{17}$$

Some initial values for $\hat{\mathbf{z}}(0|0)$ and $\mathbf{P}(0|0)$ must be chosen to launch the algorithm, the so-called starting conditions. So far in the literature the starting conditions are set to arbitrary values or to the mean value of the corresponding variable if known [5]. By means of replacing actual data by training symbols (*full training*), we can find the amount of training needed for convergence of the filter. The full training analysis reveals that the convergence can be dramatically accelerated by choosing more appropriate starting conditions.

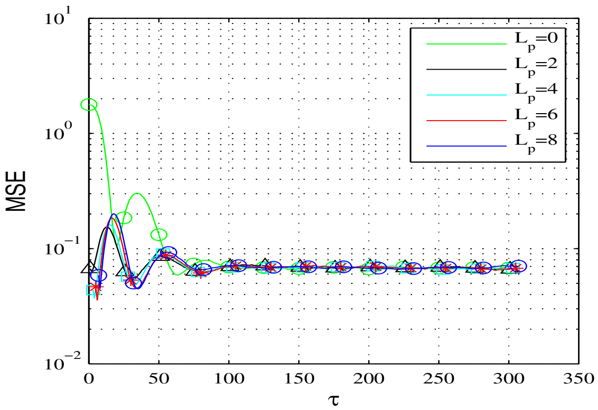
According to the initialization of the algorithm, we differentiate between two periodical training schemes: A scheme where only $\hat{\mathbf{z}}(0|0)$ is trained by means of PACE, called "conventional periodical training" (CPT), and a "hybrid periodical training" (HPT), where both variables $\hat{\mathbf{z}}(0|0)$ and $\mathbf{P}(0|0)$ are trained. Both schemes will be discussed in the next section.

[2] $\mathbf{F}$ and $\mathbf{B}$ have to be computed depending on the AR process order p such that (2) is fulfilled. In the following they are assumed to be known at the receiver. Please refer to [1] for explicit definition.

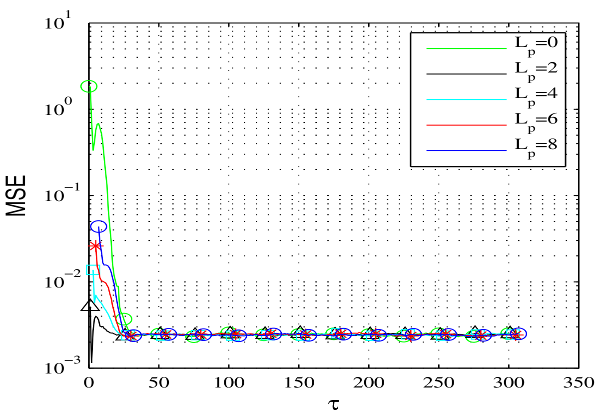
V. Periodical Training of the Kalman Tracking Algorithm

As can be seen in Fig. 1, the periodically sent pilots are divided into two sequences. One sequence of length L_p that is attributed to the PACE block provides the tracking algorithm with a good initial estimate. The second sequence trains the algorithm and takes $L_t T_s$ time. For a fair comparison of the new training scheme with the previously established ones, the optimal L_p and L_t are chosen such that $(L_p + L_t)/L_d$ is kept constant.

In order to study the convergence of the KF, Fig. 3 and Fig. 4 plot the MSE $\zeta(\tau)$ in case of full training. The performance of the tracking algorithm depends highly on the quality of the initialization. In case of initialization with zero, the filter might even not converge within a frame. The convergence is however drastically accelerated if an amount of the training data is spent on more appropriate initialization with PACE.



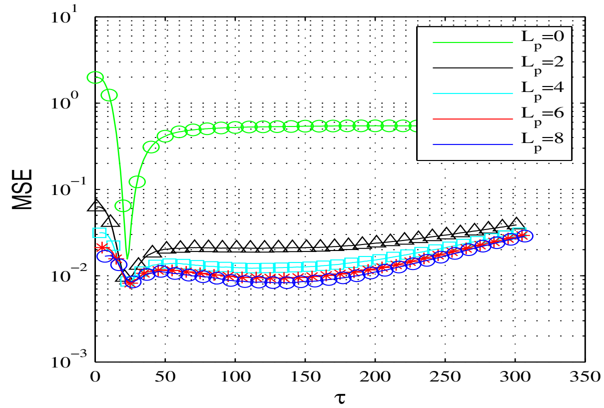
(a) $SNR = 15dB$



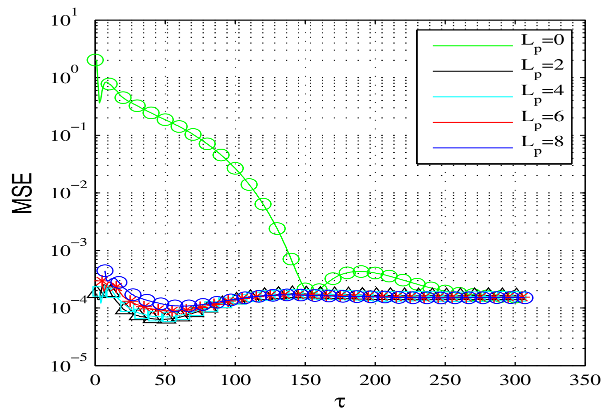
(b) $SNR = 42dB$

Fig. 3. Channel estimation MSE $\zeta(\tau)$ for full training with different L_p at $f_d = 0.01$

The full training analysis leads to a further result. In the steady state at the end of a frame, the MSE can be smaller than the PACE MSE at the beginning of a frame, i.e. $\zeta(L_d) < \zeta(o)$. This happens for example in the desicion-directed mode at high SNR when the detected data is mostly correct. In this case, reinitializing the algorithm is not advantageous. Therefore, we have to think about a mechanism to decide whether to reinitialize with PACE or not. The theoretical PACE MSE (9)



(a) $SNR = 15dB$



(b) $SNR = 42dB$

Fig. 4. Channel estimation MSE $\zeta(\tau)$ for full training with different L_p at $f_d = 0.001$

and the error covariance $\mathbf{P}$ provide us with an efficient tool to do so. If the theoretical MSE is much smaller than the trace of $\mathbf{P}$, reinitialization makes sense. Otherwise, the received pilots are not needed. Because of the periodical training, this means that pilots are transmitted at the beginning of each frame but are not used which is a real waste of bandwidth. This leads us to the aperiodical training scheme where pilots are only sent when needed, i.e on request. The pilot on request training scheme (PRQT) is discussed in the next section.

VI. Aperiodical Training: Pilots on Request

In this training scheme, pilots are transmitted on request [3]. The necessity for pilots arises when the Kalman filter diverges as a consequence of a series of detection errors. KF works robustly as long as the detected symbols are almost correct. In case of misdetections the model in (11) is not matched anymore and the channel estimation quality deteriorates which may result in more misdetections in the following steps and to error propagation. Accurate detection of error propagation is a key issue for the novel PRQT in order not to impair the spectral efficiency.

[3]We assume that the transmission of the pilots is delay-free. Consideration of a stochastically delayed time of arrival of the receive pilot signal is subject to future work.

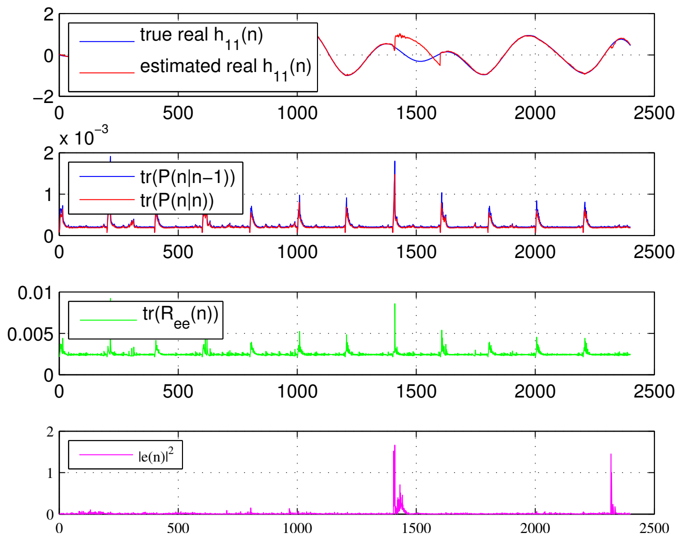


Fig. 5. Analysis of different variables in the KF as function of the discrete time n

In [2], [9] reliability information about the detected data is used to detect wrongly detected symbols and exclude them from channel tracking. This approach is feasible as long as the channel is almost invariant on a received block. Besides, it requires an iterative receiver that can provide statistical reliability information. Instead, if symbolwise tracking with a non-iterative receiver is required due to fast fading, this scheme is not applicable anymore and we have to think of other indicators for the error propagation.

Closer analysis of different statistical quantities involved in the KF algorithm suggests that a filter divergence occurs in most of the cases just after a steep peak has appeared in their progress. This is for example the case for the magnitude of the innovation process $\mathbf{e}(n) = \mathbf{y}(n) - \hat{\mathbf{X}}(n)\hat{\mathbf{z}}(n|n-1)$. However, other variables such as $\mathbf{P}(n|n-1)$, $\mathbf{P}(n|n)$, and $\mathbf{R}_{ee}(n)$ remain unchanged. $\mathbf{R}_{ee}(n)$ is the covariance of the innovation defined by $\mathbf{R}_{ee}(n) = E[\mathbf{e}(n)\mathbf{e}(n)^H]$. Further mathematical manipulations on $\mathbf{R}_{ee}$ yield (18). Computation of (18) is performed within the Kalman gain in (15) at each iteration and therefore does not require any further computational resources.

$$\mathbf{R}_{ee}(n) = \mathbf{X}(n)\mathbf{P}(n|n-1)\mathbf{X}^H(n) + \mathbf{R}_{ww} \qquad (18)$$

Fig. 5 shows some variables involved in the KF tracking process for $f_d = 0.004$, $L_t = 2$ and $L_d = 200$. We can see that a large $|\mathbf{e}(n)|$ due to an instantaneous high noise value gives birth to a filter divergence. The error propagates until the beginning of the next frame where the divergence is interrupted by setting the estimate to the PACE value.

Armed with these observations, we develop a first metric m_1 to detect a filter divergence. If $m_1 = |\mathbf{e}(n)|$ exceeds a threshold Q_{th} which is related to the expectation in (18), an error propagation is occuring and pilots are requested to stop it. Intuitively, Q_{th} is expected to depend on the SNR γ_{dB}. If Q_{th} is small, we would be requesting and transmitting pilots all the time instead of data, reducing the spectral efficiency. On the other hand, a large Q_{th} can fail in detecting many

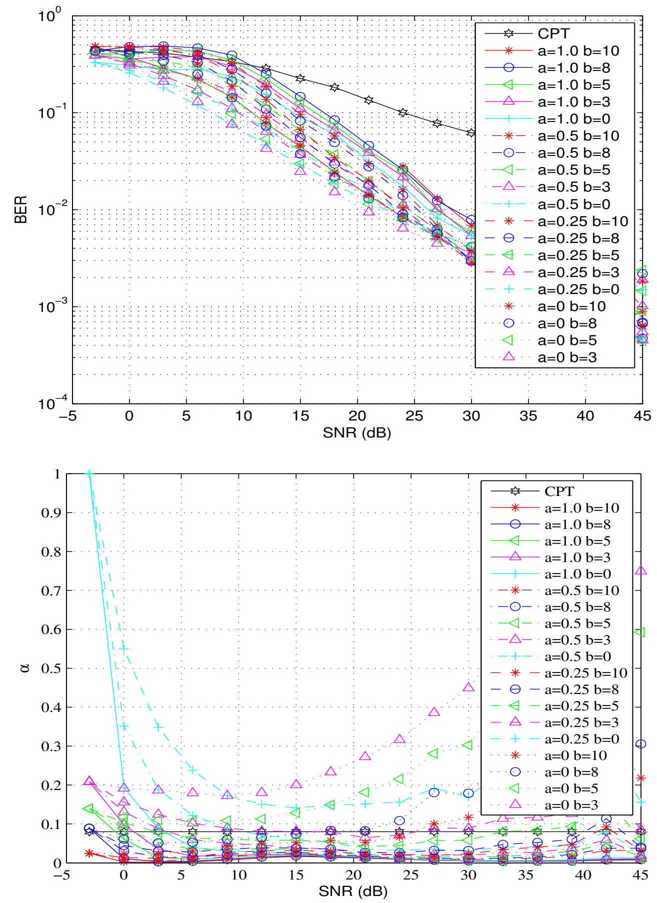


Fig. 6. BER as function of the SNR for various (a, b) (top). Ratio pilots over data α as function of the SNR for various (a, b) (bottom)

error propagations. Thus, finding the optimal threshold is a constrained optimization problem. We have to search for the threshold that minimizes the BER under the constraint that the spectral efficiency remains beyond a certain value. Out of lack of mathematical tractability for this problem and for the sake of simplicity, Q_{th} is defined as an affine function of γ_{dB} by means of the coefficients a and b as follows:

$$Q_{th} = (a \cdot \gamma_{db} + b)\sqrt{tr\left\{\mathbf{R}_{ee}\right\}} \qquad (19)$$

By intensive simulations, we determine the coefficients a and b which minimize the BER keeping the spectral efficiency beyond a certain value. To evaluate the quality of spectral efficiency, we introduce the parameter α which denotes the ratio of number of pilots over data. Some results of this optimization process are illustrated in Fig. 6. Therein, the trade-off between small BER and large spectral efficiency is plain to see. For instance, $(a, b) = (0, 3)$ leads to the lowest BER but the required number of pilots is very high. On the other hand, $(a, b) = (1, 10)$ requires the smallest number of pilots but at the expense of large BER.

As a second approach, we suggest to consider the *normalized innovation squared* (NIS), in order to provide a metric m_2 independent of the SNR. The NIS m_2 is defined as:

$$m_2 = \mathbf{e}(n)^H \mathbf{R}_{ee}^{-1}(n)\mathbf{e}(n) \qquad (20)$$

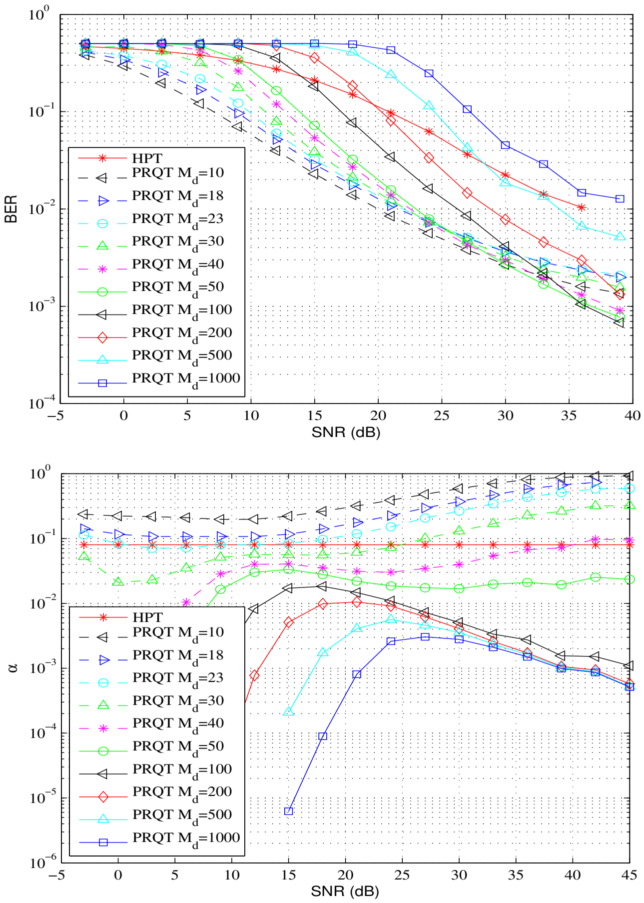


Fig. 7. BER as function of the SNR for various M_d (top). Ratio pilots over data α as function of the SNR for various M_d (bottom)

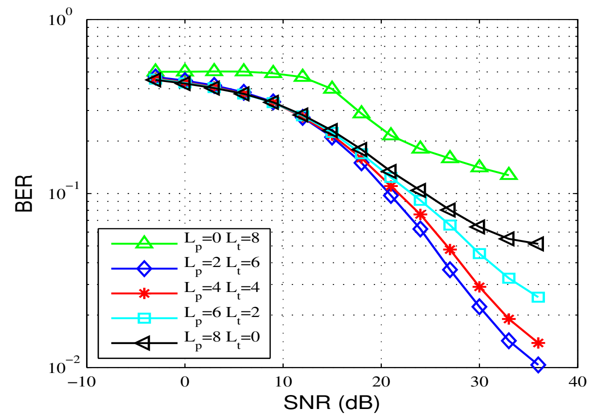
An error propagation occurs if m_2 exceeds a threshold M_d. This approach is known as "validation gating" and is widely known in the field of target tracking to exclude very unlikely measurement-to-track associations [10]. The NIS follows a chi-square probability density function. Thus $\mathbf{e}(n)^H \mathbf{R}_{ee}^{-1} \mathbf{e}(n) < M_d$ means that for a probability that p% of true associations are accepted, M_d can be computed from

$$\frac{p}{100} = P(\frac{N}{2}, \frac{M_d}{2}) = \frac{1}{\Gamma(N/2)} \int_0^{M_d/2} e^{-t} t^{N/2-1} \, dt \quad (21)$$

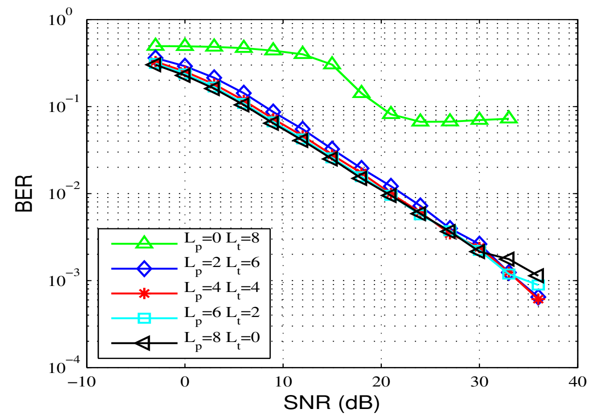
where Γ is the Gamma function. For instance, for $p = 99,99\%$ and $N = 2$, $M_d = 18.42$. The BER for different M_d values and the corresponding α are illustrated in Fig. 7. The trade-off between low BER and high spectral efficiency is again plain to see.

VII. Simulation Results

A 2×2 MIMO system with BPSK modulation and zero-forcing receiver is considered. We assume that a constant spectral efficiency is given. This means that for periodical training, we keep the ratio of training and data transmission phase lengths $\alpha = (L_p + L_t)/L_d$ constant. For our simulations, we take $L_d = 100$ and $L_p + L_t = 8$. The corresponding BER results for HPT with different f_d are illustrated in Fig. 8. These results suggest that training the algorithm exclusively performs worse than allocating an amount of the training to supply the algorithm with a PACE initialization for both f_d. Furthermore, we notice that HPT with $L_p = 2, L_t = 6$ performs best for $f_d = 0.01$. Indeed, at $f_d = 0.01$, $L_p = 2$ leads to the smallest PACE MSE. At smaller f_d, investing all pilots for PACE initialization leads to the lowest BER on the whole considered SNR range.



(a) $f_d = 0.01$



(b) $f_d = 0.001$

Fig. 8. BER as function of the SNR for hybrid training scheme with $L_p + L_t = 8$

The BER results for PRQT, optimized under the constraint that $\alpha \leq 8\%$ for comparison fairness, are plotted in Fig. 9 for $f_d = 0.01$. PRQT with both suggested metrics outperforms CPT and HPT significantly. The ratio of required pilots is even reduced to less than 2% as can be seen in Fig. 10. For PRQT with m_2, $M_d = 30$ is chosen for an $SNR < 24dB$ and $M_d = 50$ beyond since this yields to a good trade-off between BER and spectral efficiency, respecting the constraint $\alpha \leq 8\%$ on the whole considered SNR range. PRQT with m_2 outperforms PRQT with m_1 for $SNR < 12dB$. For low SNR however, it leads to similar BER as for CPT and HPT. The discontinuities in Fig. 10 arise from adopting different parameters ((a, b) for m_1, and M_d for m_2) depending on the SNR. Considering that it is less complicated to optimize the threshold M_d in comparison to Q_{th} (2 degrees of freedom with the coefficients a and b), the NIS metric m_2 is preferred over m_1 if the operating SNR is large enough.

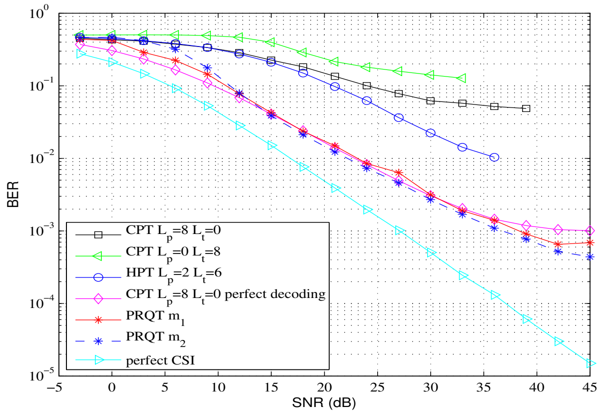


Fig. 9. BER as function of the SNR for CPT, HPT, PRQT and perfect CSI

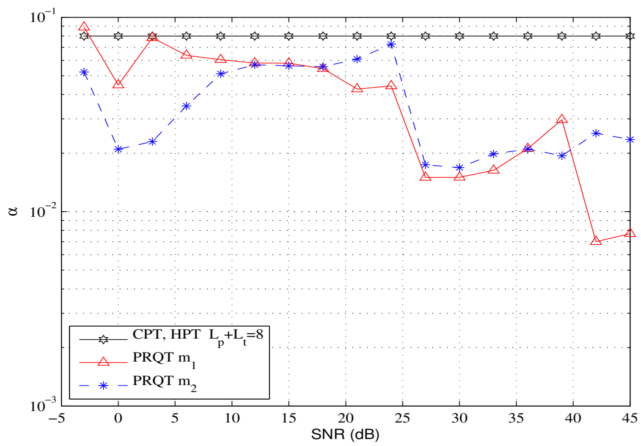


Fig. 10. Ratio pilots over data α for PRQT and CPT with 8% training

A further very remarkable result is that the BER curves for KF with PRQT and KF operating with CPT and perfect detection are overlapping between 12dB and 36dB. For SNR<12dB, PRQT generally performs worse due to the constraint of spectral efficiency. In the high SNR however, it even outperforms the tracking with perfect detection. The performance gap between PRQT and perfect CSI is basically due to the model mismatch with AR(2). We expect this gap to be smaller with higher AR order.

VIII. Conclusion

In this paper, we deal with tracking fast-varying MIMO channels by applying the Kalman algorithm. Different training schemes for this algorithm are suggested such as the hybrid periodical scheme and the pilot on request scheme. They are compared to the conventional periodical training. The hybrid scheme can decrease the BER floor by an order of magnitude in comparison to the conventional periodical training while maintaining the same spectral efficiency. For the aperiodical pilot on request training, we develop two different metrics for detecting the error propagation. Finally, we show that with the novel training on request scheme the BER can be significantly decreased while the number of required pilots is remarkably reduced.

References

[1] C. Komninakis, C. Fragouli, A. Sayed, and R. Wesel, "Multi-input multi-output fading channel tracking and equalization using Kalman estimation," *IEEE Transactions on Signal Processing*, vol. 50, no. 5, pp. 1065 –1076, May. 2002.

[2] I. Nevat and J. Yuan, "Joint channel tracking and decoding for BICM-OFDM systems using consistency tests and adaptive detection selection," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 8, pp. 4316 –4328, Oct. 2009.

[3] J. Choi, M. Bouchard, and T. H. Yeap, "Adaptive filtering-based iterative channel estimation for MIMO wireless communications," *IEEE International Symposium on Circuits and Systems*, pp. 4951 – 4954 Vol. 5, May. 2005.

[4] E. Karami and M. Shiva, "Decision-directed Recursive Least Squares MIMO Channel Tracking," *EURASIP Journal on Wireless Communications and Networking*, Dec. 2005.

[5] S. Haykin, *Adaptive Filter Theory*. Prentice-Hall, Inc., 1996, ISBN 0-13-322760-X.

[6] E. Eitel, R. A. Salem, and J. Speidel, "Improved decision-directed recursive least squares MIMO channel tracking," *IEEE International Conference on Communications*, pp. 1 –5, Jun. 2009.

[7] A. Sayed and T. Kailath, "A state-space approach to adaptive RLS filtering," *IEEE Signal Processing Magazine*, vol. 11, no. 3, pp. 18 –60, Jul. 1994.

[8] Q. Sun, D. Cox, H. Huang, and A. Lozano, "Estimation of continuous flat fading MIMO channels," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 549–553, Oct. 2002.

[9] I. Nevat and J. Yuan, "Channel tracking using pruning for MIMO-OFDM systems over Gauss-Markov channels," *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 3, pp. III–193 –III–196, Apr. 2007.

[10] T. Bailey, B. Upcroft, and H. Durrant-Whyte, "Validation gating for non-linear non-Gaussian target tracking," *9th International Conference on Information Fusion*, pp. 1 –6, Jul. 2006.