# ADAPTIVE MEMORY CONTROL FOR POLYNOMIAL RLS ALGORITHM IN FAST FADING CHANNELS

*P.C.Vijay* *K.Giridhar*

Telecommunication and Networking (TeNeT) Group
Department of Electrical Engineering
Indian Institute of Technology, Madras
vijay,giri@tenet.res.in

## ABSTRACT

The Polynomial RLS algorithm (PRLS) based receiver estimates and tracks a fast frequency selective fading channel by modeling it as a polynomial in time and then estimating the polynomial coefficients using a rectangular windowed RLS algorithm. In this paper, we describe a method of adaptively varying the size of the rectangular window that results in a much reduced chance of tracking failure without any significant increase in computational effort. Simulation results, comparing the frame error rate performance of the above method with a Least Mean Squares based tracking algorithm are presented.The frame error rate performance of the Joint Viterbi Decoder for two co-channel signals using PRLS tracking is typically one order of magnitude superior to LMS based tracking.

## 1. INTRODUCTION

Most receivers for fading channels require an accurate channel estimate to get good error rate performance. Examples include the Adaptive Maximum Likelihood Sequence Detector, the Per-Survivor Processing (PSP) based receiver [1] and Minimum Mean Square Error (MMSE) based equalizers [2]. Usually, the Least Mean Square (LMS) or Recursive Least Squares (RLS) algorithms are used to estimate and track the channel [3],[4]. However, these methods do not perform satisfactorily under fast fading. Similarly, the Kalman filter based techniques and the Linear Predictor based techniques [5] can be used only when the channel statistics are known.

We propose a technique that models the channel as a polynomial in time with unknown, random coefficients. It is based on the estimator given in [6]. However, we have extended the approach to include co-channel interference conditions (described in our previous paper [7]). This algorithm can also be used in spatial multiplexing schemes since the receiver is seen to perform well even at high signal to interference ratios. These coefficients are determined using an RLS based approach. The PRLS algorithm uses rectangular windowed RLS to evaluate the coefficients, where the effect of the oldest data is removed at each instant after utilizing the effect of the new data, it is possible that for some data sequences, the estimator fails. This could lead to faulty decisions which result in even higher channel estimation error, finally causing a total failure of the Viterbi Decoder. To solve this problem, the estimator algorithm has been modified to change the RLS window size adaptively.

The paper is organized as follow. In Section 2, the transmitter and channel models are considered. In Section 3, a brief overview of the PRLS algorithm for multiple channels is given. In section 4, the method of adaptive window control is discussed. In Section 5, the

receiver structure based on the conventional Joint Maximum Likelihood Sequence Detector (JMLSD), and using the above channel estimator, is given. In Section 6, simulation results are presented before concluding in Section 7.

## 2. SYSTEM MODEL

The system model is identical to the one in [7]. Finite ray model with symbol spaced taps is assumed for the multipath channel. Then, the received signal samples, which form a sufficient statistic, are given by,

$$y(lT) = \sum_{k=1}^{2} \sum_{i=0}^{M-1} a_{k,i} c_k(lT, lT - iT) + w(lT) \tag{1}$$

where $a_{k,i}$ is the complex symbol transmitted by the $k^{th}$ source at time $i$, $c_k(lT, lT - iT)$ is the $i^{th}$ tap of the multipath Rayleigh fading channel corresponding to that source at time $lT$ and $w(lT)$ are AWGN samples. Equation (1) can also be written in simpler notation as

$$y_l = \sum_{k=1}^{2} \sum_{i=l-L_k+1}^{l} a_{k,i} c_k(l, i) + w_l \tag{2}$$

The above assumptions have been made for the sake of simplicity. However the proposed method can be extended to include arbitrary multipath channels also. The Maximum Likelihood criterion states that the most probable pair of transmitted sequences, $\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2$ is that for which the log-likelihood metric $\Lambda(\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2)$ given by

$$\Lambda(\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2) = \sum_{l=0}^{M-1} \left[ \frac{y_l - \hat{y}_l(\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2)}{\hat{\sigma}_l^2(\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2)} + \hat{\sigma}_l^2(\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2) \right] \tag{3}$$

is a minimum. Here, $\hat{y}_l(\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2)$ is the estimate of $y_l$ based on the previous received samples and $\hat{\sigma}_l^2(\hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2)$ is the variance of the estimate. This can be implemented using the Viterbi Algorithm, provided reliable channel estimates are available.

## 3. POLYNOMIAL PREDICTOR BASED CHANNEL ESTIMATOR

Conventional Least squares approach will not work since the number of variables for a window size of $N$ will be $2LN$, whereas the number of equations would be just $N$. However, if we can model the channel using a small number of parameters, say $K$, that can be assumed to be constant over the window, we could use the least squares approach provided $N > 2KL$. It has been observed that the time variation of a smoothly varying channel, such as a band-limited mobile wireless channel, can be closely approximated over a short interval by a small set of polynomial basis functions. That is,

$$c(t, s) = \sum_{i=0}^{K} \alpha_n^{(i)}(s) \left( \frac{t - nT}{T} \right)^i + R_K(t, s) \tag{4}$$

for some tap $s$ of the channel.Hence, if we use a $K^{th}$ order polynomial to represent the $L_k$ taps of each channel over a window of length $N$, we can write the channel tap $c_k(l, i)$ as,

$$c_k(l, i) = \sum_{p=0}^{K} b_{k,i}(p, l) q_l^p \tag{5}$$

where $q_l \in \mathcal{Z}$, $-\lceil N/2 \rceil < q_l \leq N - \lceil N/2 \rceil$ and $b_{k,i}(p, l)$, $p = 0 \ldots K$ are the unknown coefficients. Assuming the coefficients to remain constant over the window, we can write a set of $N$ equations. In matrix form,

$$\mathbf{c}_l = \mathbf{Q}' \mathbf{b}'_l \tag{6}$$

where,

$$\begin{aligned} \mathbf{c}_l &= \left[ \mathbf{c}_{1,l-N+1}^T \, \mathbf{c}_{2,l-N+1}^T \cdots \mathbf{c}_{1,l}^T \, \mathbf{c}_{2,l}^T \right]^T \\ \mathbf{c}_{k,l-j} &= \left[ c_k(l-j, 1) \cdots c_k(l-j, L_k) \right]^T \end{aligned}$$

$\mathbf{b}'_l$ is the coefficient vector with

$$\begin{aligned} \mathbf{b}'_l &= \left[ \mathbf{b}'^T_{1,0} \, \mathbf{b}'^T_{2,0} \cdots \mathbf{b}'^T_{1,K} \, \mathbf{b}'^T_{2,K} \right]^T \\ \mathbf{b}'_{k,j} &= \left[ b'_{k,1}(j, l) \cdots b'_{k,L_k}(j, l) \right]^T \end{aligned}$$

and $\mathbf{Q}'$ is the $LN \times L(K+1)$ matrix containing the basis vectors, where we have set $L = L_1 + L_2$. Let $\mathbf{y}_l = [y(l-N+1) \cdots y(l)]$ be the vector of received values in the window. Then, we can write,

$$\mathbf{y}_l = \mathbf{A}_l \mathbf{c}_l + \mathbf{w}_l \tag{7}$$

where $\mathbf{A}_l$ is a block diagonal matrix containing the data sequence given by,

$$\mathbf{A}_l = \mathrm{diag}\{(\mathbf{a}_{1,l-N+1}\, \mathbf{a}_{2,l-N+1}) \cdots (\mathbf{a}_{1,l}\, \mathbf{a}_{2,l})\}$$

with

$$\mathbf{a}_{k,l-j} = [a_{k,l-j} \cdots a_{k,l-j-L_k+1}]$$

and $\mathbf{w}_l = [w_{l-N+1} \cdots w_l]$ is the AWGN vector. Using (6) and (7), we have

$$\mathbf{y}_l = \mathbf{A}_l \mathbf{Q}' \mathbf{b}'_l + \mathbf{w}_l \tag{8}$$

If $\Gamma' = \mathbf{A}_l \mathbf{Q}'$, the least squares solution to (8) is given by,

$$\hat{\mathbf{b}}'_l = \Gamma'^{\dagger} \mathbf{y}_l \tag{9}$$

where $\Gamma'^{\dagger}$ is the pseudo-inverse of $\Gamma'$. A necessary condition for a unique solution to exist is that $N > L(K+1)$. This equation can also be solved recursively using the RLS algorithm which results in less computational effort. However to ensure that $\Gamma'^{\dagger}$ exists, QR decomposition of $\mathbf{Q}'$ has to be performed. In that case the final expression becomes

$$\hat{\mathbf{b}}_l = \Gamma^{\dagger} \mathbf{y}_l \tag{10}$$

where $\Gamma = \mathbf{A}_l \mathbf{Q}$ and $\mathbf{Q}' = \mathbf{QR}$.

## 4. ADAPTIVE WINDOW SIZE CONTROL

It is observed that in spite of QR decomposition, the data matrix $\mathbf{A}_l$ may be such that $\Gamma'$ is not invertible. One way to solve this would be to increase the window size to such an extent that the probability of such a data matrix appearing is very small. However this would lead to worsening accuracy of the polynomial approximation. Another solution would be to do transmit coding to prevent such a data matrix from occurring. Though this method may be useful during training, in a decision-directed tracking mode, for example in a Maximum Likelihood Sequence Detector, the hypothesized sequence may be different from the actual sequence and hence cause the same problem. To handle this difficulty we propose a adaptive window sizing technique which we describe below.

The rectangular windowed RLS algorithm which is used to update the coefficients of the channel modelling polynomial consists of three steps, *u*pdation, *d*owndation and *r*e-normalization. Updation refers to the inclusion of the new data at time $n$ in the computation of the pseudo-inverse. Downdation results in the removal of the data at time $n-N$ from the computation and re-normalization is applied so that the polynomial basis matrix stays orthonormal. During updation the computation of the RLS gain vector involves only additions in the denominator and hence the gain vector remains finite. However, during downdation it is possible for the gain vector to become very large, depending on the data vector at time $n-N$. In such a case, downdation is skipped and only renormalization performed. This means that the window size has been temporarily incremented by one.

As more and more new symbols are used in computing the pseudo-inverse, the data matrix changes so that downdation can be performed with low probabilty of the gain vector becoming large again. In such a case, one can repeat the downdation step as many times as one wants, the only constraint being that the gain vector remains finite. However, if one is unable to perform downdation even after increasing the window size to a certain limit, then one can simply remove the old data vectors from the data matrix without updating the pseudo-inverse or the coefficients.

Since the window size before the renormalization step can vary, the corresponding normalization matrices will also be different.

However, for a fixed upper limit on the window size, these matrices can be pre-computed so that there is no increase in the complexity of the algorithm.

## 5. RECEIVER STRUCTURE

During the decision directed phase, the matrix $\mathbf{A}_l$ in (7) containing the actual symbols transmitted, would be replaced by $\hat{\mathbf{A}}_l$, containing low delay decisions on the transmitted symbols. These decisions are obtained using Per-Survivor Processing [1], where a separate channel estimate is maintained for each survivor sequence in the Viterbi Algorithm. Essentially, it involves the following steps repeated for every symbol.

I *Final decoder decision* The survivor with the minimum metric is chosen and a decision is made on the $(n-N_d)^{th}$ symbol at time $n$, $N_d$ being the decoding delay.

II *Survivor extension*

As in the standard Viterbi Algorithm, the survivors are extended based on the cumulative metric at time $l+1$ as,

$$CM_{l+1}^n = \min_m \; CM_l^m + (y_{l+1} - \hat{y}_{l+1}^m)^2 \quad m, n = 1 \ldots N_s \tag{11}$$

where the superscript refers to the survivor path that we are extending and $N_s$ is the number of states in the trellis. The predictions of the received value $\hat{y}_{l+1}^m$ are made using the channel estimates corresponding to that particular survivor.

III *Channel estimation updation* The polynomial coefficients are computed using the method given in the previous section. Using the set of polynomial coefficients $\mathbf{b}_l$ corresponding to each survivor, the channel estimate at time $l+1$ is computed as $\mathbf{c}_{l+1} = \mathbf{Q}''\mathbf{b}_l$, where $\mathbf{Q}''$ is the matrix that extrapolates the polynomial to time $n+1$.

## 6. SIMULATION RESULTS

The above receiver was simulated for two fast, frequency selective Rayleigh fading channels with different fade rates. Jakes model with a flat power delay profile was used to simulate the Rayleigh fading channel A single interfering channel was considered with the Signal to Interference ratio (SIR) being 0dB. . QPSK modulation was used for the source and a framing scheme with $N_p = 8$ slots per frame, each slot containing $N_s = 100$ symbols was assumed. Each slot contained a preamble sequence of length $N_t = 15$ symbols. The simulation was run over a total of $N_f = 100$ frames and the FER (frame error rate), given as the number of frames having more than 5% of the symbols in error, was plotted.

The PRLS algorithm was run using two different starting window sizes and in each case, the excess window size, defined as the maximum increase in the RLS window allowed, was taken to be half the starting window size. A first order polynomial was found to be sufficient to model the channel for the fade rates under consideration.

The multi-channel variant of the LMS algorithm, where the same prediction error is applied to the update equations of both channels, was used for the simultaneous tracking of the primary and the co-channel. Also, the LMS algorithm was simulated for different step sizes $\mu$. Further, the effect of delayed but error-free, decisions on the LMS algorithm was also observed.

## 7. CONCLUSIONS

An improved polynomial predictor based estimator having adaptive memory control was presented. A Joint Viterbi Decoder based receiver using the above estimator was also discussed. Simulation results comparing the performance of the above receiver with an LMS based one were given. It is seen that the proposed receiver outperforms the LMS based receiver for very fast fading channels. It is

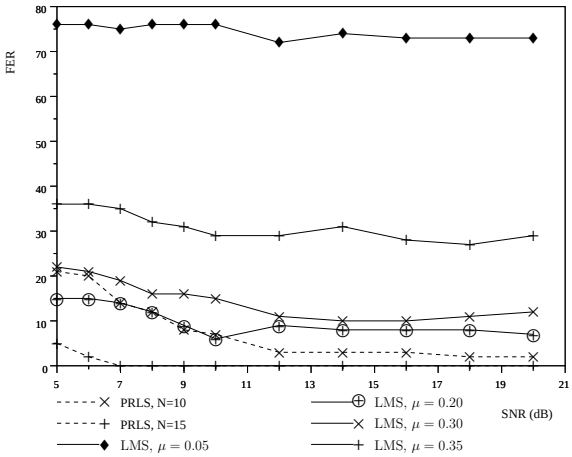


Figure 1: FER performance for $f_d = 200Hz, Ts = 8\mu s$, Proposed method Vs LMS

also expected that the adaptive window control would help the PRLS algorithm perform better in non-stationary environments.

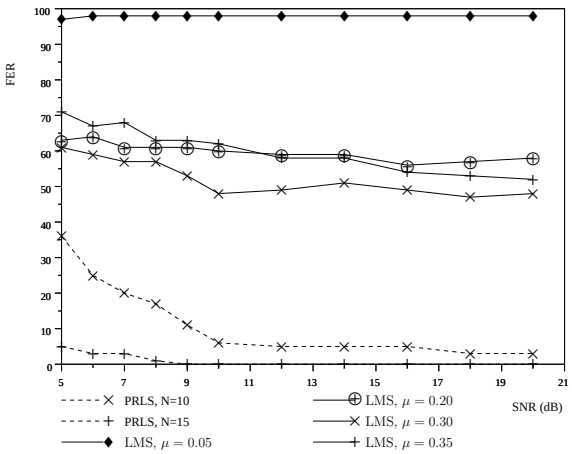


Figure 2: FER performance for $f_d = 50Hz, Ts = 100\mu s$, Proposed method Vs LMS

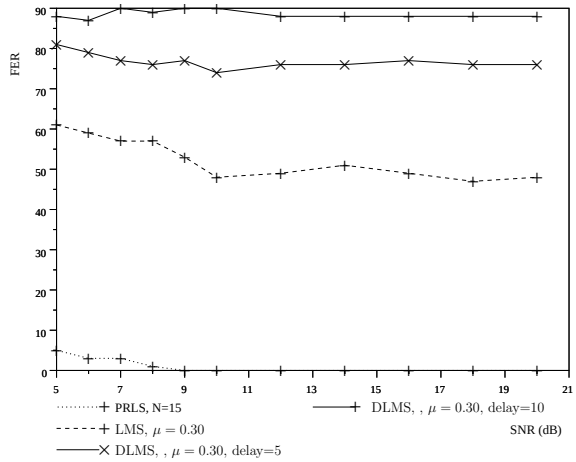


Figure 3: FER performance for $f_d = 50Hz, Ts = 100\mu s$, LMS Vs Delayed decision LMS

## REFERENCES


[1] A. R.Raheli and C.K.Tzou, "Per-survivor processing: A general approach to MLSE in uncertain environments," *IEEE Trans. Commun.*, vol. 43, pp. 354–364, Feb/Mar/Apr 1995.

[2] J. G. Proakis, *Digital Communications.* McGraw Hill, 1995.

[3] S. Haykin, *Adaptive Filter Theory.* Prentice-Hall, third ed., 1995.

[4] J.G.Proakis, "Adaptive equalization for TDMA digital mobile radio," *IEEE Trans. Veh. Technol.*, vol. 40, pp. 333–341, May 1991.

[5] X.Yu and S.Pasupathy, "Innovations-based MLSE for Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 43, pp. 1534–1544, Feb/Mar/Apr 1995.

[6] D. K. Borah and B. D. Hart, "Frequency-Selective Fading Channel Estimation with a Polynomial Time-Varying Channel Model," *IEEE Trans. Commun.*, vol. 47, pp. 862–873, June 1999.

[7] P.C.Vijay and K.Giridhar, "Estimation and Tracking of Fast Frequency Selective Fading Channels with Co-channel Interference," *National Conference on Communications (NCC)*, 2002.