Model-based Adaptive Algorithms for Time-Varying Communication Channels with Application to Adaptive Multiuser Detection

Zarko B. Krusevac  
National ICT Australia  
Australian National University  
Canberra, ACT 2052 Australia  
zarko.krusevac@rsise.anu.edu.au

Rodney A. Kennedy  
National ICT Australia  
Australian National University  
Canberra, ACT 2052 Australia  
rodney.kennedy@anu.edu.au

Predrag B. Rapajic  
University of Greenwich at Med. Pemb.  
Chatham Maritime KENT ME4 4TB  
United Kingdom  
P.Rapajic@greenwich.ac.uk

Abstract—This paper shows through theory and simulation the superiority of model-based adaptive algorithms relative to observation-only-based adaptive algorithms, such as LMS and RLS, when applied to tracking time-varying channels. The model-based formulation reveals RLS as a degenerate algorithm which does not explicitly recognize the time-varying nature of the channel and consequently is ill-suited to tracking in non-stationary environments. Simulation results for MSE performance of the various adaptive algorithms applied to adaptive MMSE multiuser receiver corroborate the theoretical analysis.

I. INTRODUCTION

1) Background and Motivation: Adaptive algorithms are often used in non-stationary environments where they are required to track time variations in an unknown time-varying system or channel. Despite considerable attention in the literature over the last two decades regarding the ability of adaptive algorithms to track such variations [1]–[7], the theoretical limit of performance of adaptive algorithms in non stationary environments in the presence of the multiuser interference has not been studied in depth. Since observation-only-based adaptive algorithms, such as LMS and RLS, use only the channel output and a priori knowledge of training sequence to estimate system coefficients [1]–[3], [6], [7], their overall performance in non-stationary environments, where the dynamics of the channel are unmodeled, is not so promising [2]. However, a model-based approach exploits additional knowledge of the propagation conditions to model time variations of the “true” system coefficients [2]. In addition to the observation equation, model-based adaptive algorithms use this model, generally in state equation form, to improve tracking performance over time-varying channels [1]–[3], [6], [7].

If the model is correct, the model-based adaptive algorithms can provide very good performance in non-stationary environments. However, modeling radio channels is a complicated task and the complexity of the solution to Maxwell’s equations needs to be reduced to an appropriately reduced set of parameters to make it mathematically amenable (and statistically parsimonious). A stochastic description helps to overcome the complexity of the real propagation environment, provides a focus on the critical broad attributes and tends to average out nuisance parameters. A system design based on a stochastic channel model will only achieve average performance — but it will achieve this performance at a wide variety of sites. In contrast, a system designed with full knowledge of the propagation conditions at a certain site would be able to exploit these conditions, resulting in superior performance but at the cost of reduced robustness.

2) Contributions: This paper provides a comparative analysis of model-based and observation-only-based approach to the adaptive algorithm design for adaptive communication systems over time-varying communication channels in the presence of multiuser interference.

- Our theoretical analysis shows that model-based adaptive (Kalman) algorithms offers superior tracking performance to observation-only-based LMS and RLS algorithms.
- The model based approach reveals the RLS algorithm to be a degenerate form of model-based adaptive (Kalman) algorithm where there is no explicit recognition of the time-varying nature of the “true” weight vector. Consequently, the RLS algorithm cannot offer good (tracking) performance in non-stationary environments.
- We simulate MSE performance of the adaptive algorithms for the adaptive MMSE multi-user receiver over time-varying Rayleigh fading channels to corroborate our theoretical analysis.

II. ADAPTIVE MULTIUSER DETECTION

A. Adaptive MMSE Receiver

The adaptive MMSE receiver structure for multipath fading channels is presented in [8]. It consists of a bank of adaptive fractionally spaced MMSE FIR filters along with the (ML) detector part of the receiver for data detection, Fig.1. The number of intercell interferers $K_I$ is unknown, and only $K$ (the number of users in the cell of interest) input MMSE filters are used in the proposed receiver. In the following we follow the treatment given in [8] unless otherwise stated.
The fractionally-spaced, discrete-time received sample vector at time \( n \), \( y(n) \), over a running window of length \( 2P + 1 \) can be expressed in a matrix form
\[
y(n) = Fx(n) + n(n)
\] (1)
where \( F \) is the matrix of fractionally spaced sampled signatures and \( n(n) \) is AWGN. The vector \( x(n) = x_D(n) + x_U(n) \) contains the transmitted symbols during that window period, where \( x_D(n) \) contains known symbols with unknown symbols set to zero and \( x_U(n) \) contains unknown symbols with known symbols set to zero.

The output of the MMSE filter at the \( n \)th symbol interval for the \( k \)th user is
\[
a_k(n) = \sum_{m=-P}^{P} v_k(m) y_k(nT - m T_f) \] (2)
where \( y_k(n) \) is discrete time received sample of \( k \)th user at time \( n \), \( \{v_k(m)\} \) are the adaptive filter coefficients of \( k \)th user and \( T_f = (T_c/p) \) with \( p > 1 \) and \( T_c \) being the chip interval. The total number of adaptive filter coefficients (running window length) \( 2P + 1 \) is chosen to be \( (2P + 1) > pN \), where \( N \) is the spreading gain. The symbol estimate \( \tilde{x}_k(n) \) is obtained as
\[
\tilde{x}_k(n) = a_k(n) - d_k^H x_k(n)
\] (3)
where \((\cdot)^H\) denotes Hermitian transpose and \( d_k \) is a vector of tentative decision aided coefficient sequences, defined as \( d_k = (d_k^{(0)}, d_k^{(2)}, \ldots, d_k^{(2K)})^T \) with \( d_k^{(m)} = (d_{km}(0), \ldots, d_{km}(M - 1))^T \). The elements of \( d_{km} \) represent the weighting coefficients, which multiply the respective known symbols (tentative decisions) coming from the \( m \)th user. The vector \( x_k \) contains symbols known to the receiver with unknown symbols set to zero. It is defined as \( x_k = (x_k^T, x_k^{(2)}, \ldots, x_k^{(2K)})^T \) with \( x_k^{(m)} = (x_k^{(m)}(0), \ldots, x_k^{(m)}(M - 1))^T \).

**B. Training of the Adaptive MMSE Receiver**

The coefficients \( \omega_k = (\psi_k^T, d_k^T)^T \) are obtained adaptively during the training period, by minimizing the MSE, \( E[|e_k(n)|^2] \), where
\[
e_k(n) = x_k(n) - \tilde{x}_k(n);
\] (4)
\( x_k(n) \) is the \( k \)th user training sequence, \( a \) priori known to the receiver. By combining (2) in (3), the \( k \)th user symbol estimate at the \( n \)th symbol interval, \( \tilde{x}_k(n) \) can be calculated as
\[
\tilde{x}_k(n) = \omega_k^H(n-1)u(n)
\] (5)
where \( u(n) = (y^T(n) x_k^T(n))^T \). By using (5), (4) becomes
\[
e_k(n) = x_k(n) - \omega_k^H(n-1)u(n)
\] (6)
used by the observation equation, see Fig. 1.

By denoting \( \omega_0 = (\psi_0^T, d_0^T)^T \) to be the optimal coefficients which minimize the \( E[|e_k(n)|^2] \) criterion, the minimum error \( e_k \) can be expressed
\[
e_k(n) = x_k(n) - \omega_0^H(n)u(n)
\] (7)
We use \( \sigma_k^2 \) to denote the variance of the “true” error \( e_k \).

While in time-invariant environments the optimal coefficients \( \omega_0 \) are constant in time, if the channel parameters vary with time, then \( \omega_0 \) becomes time-varying.

**III. Observation-only Based Adaptive Algorithms**

**A. Algorithm Design**

Observation-only based adaptive algorithms use the observation equation (1) and observation error (6) to estimate the coefficients \( \omega_k \) throughout the standard iterative procedure
\[
\omega_k(\text{new}) = \omega_k(\text{old}) + \text{gain} \times \text{gradient} \times e_k(\omega_k)
\] (8)
where \( \text{gradient} = dE[|e_k(\omega_k)|^2]/d\omega_k \). In a real-time setting statistical information \( E[|e_k(\omega_k)|^2] \) is not available. Adaptive algorithms replace \( E[|e_k(\omega_k)|^2] \) with a real-time computable approximation and optimizes that directly in real-time. Most common iterative methods are steepest descent (and its variations) or the Newton Method. The nature of the gain in (8) is crucial because it controls speed of adaptation.

**B. Least Mean Square (LMS) Adaptive Algorithm**

The Least Mean Square (LMS) is an observation based, steepest descent adaptive algorithm, which uses instantaneous MSE criterion — \( E[|e_k(n)|^2] \) in (8) is replaced with \( |e_k(n)|^2 \). The following steps correspond to the LMS algorithm when applied to the adaptive MMSE-ML receiver structure from Fig. 1, [8]
\[
v_k(n+1) = v_k(n) + \alpha_1 e_k^2(n) y(n)
\] (9a)
\[
d_k(n+1) = d_k(n) - \alpha_2 e_k(n) b_P(n)
\] (9b)
for \( n = 0, 1, 2, \ldots \) and \( \alpha_1 \) and \( \alpha_2 \) are the step sizes of the algorithm. Thus, the LMS is model free, computationally very simple but may be slow.

**C. Recursive Least Square (RLS) Adaptive Algorithm**

Recursive Least Square (RLS) is an observation based, Newton method adaptive algorithm. The Newton method iterative procedure uses second derivative of MSE to speed-up the iterations.

Newton method (10) fails when applied to the instantaneous MSE since the second derivative of \( |e_k(\omega_k)|^2 \) has rank 1 and
is not invertible. Instead, the RLS uses exponentially weighted least square (EWLS) criterion, as follows

$$J_k(\omega_k(n)) = \sum_{r=1}^{n} \lambda^{n-r} |e_k(\omega_k(r))|^2$$  \hspace{1cm} (10)

where \(\lambda\) is the forgetting factor and \(0 < \lambda < 1\). Only short memory RLS \((0 < \lambda < 1)\) is capable of tracking time varying parameters [2].

When applied to the adaptive MMSE-ML receiver structure from Fig. 1, the RLS algorithm has the following steps:

$$g_k(n+1) = \frac{\lambda^{-1} P_k(n)u(n+1)}{1 + \lambda^{-1} u^H(n+1)P_k(n)u(n+1)}$$

$$e_k(n+1) = x_k(n+1) - \omega_k(n) u(n+1)$$

$$\omega_k(n+1) = \omega_k(n) + g_k(n+1)e_k(n+1)$$

$$P_k(n+1) = \lambda^{-1} P_k(n) - \lambda^{-1} g_k(n+1) u^H(n+1) P_k(n)$$  \hspace{1cm} (11)

The RLS algorithm is almost model free (only one tuning parameter \(\lambda\) is used), is generally faster than the LMS algorithm but can be badly biased.

IV. MODEL-BASED ADAPTIVE ALGORITHMS

In a non-stationary environment (time-varying channel), the “true” coefficients \(\omega_{k0} = [v_k^T, d_k^T]^T\) vary with time and an adaptive algorithm has to be able to track these time variations in order to provide good performance. The main idea behind the model-based adaptive algorithm design is to use the knowledge of the propagation conditions (channel model) to describe how “true” filter coefficients vary with time. In addition to the observation equation (1) and observation error equation (6), the model based adaptive algorithms use a model for the time-varying “true” filter coefficients to improve algorithm (tracking) performance in non-stationary environments, at the price of additional computational complexity.

Our analysis confirms that if the model is correct, the model-based adaptive approach outperforms observation-only based adaptive algorithms in non-stationary environments. However, modeling the time-varying communication channels to capture the actual filter coefficient time variations is a complicated task. The tradeoff here is between the optimal utilization of site-specific propagation features and system robustness. A system designed with full knowledge of the propagation conditions at a certain site would be able to exploit these conditions, resulting in superior performance, whereas a system design based on a stochastic channel model will only achieve average performance — but it will be achieve this performance at a wide variety of sites whereas the former will not.

A. Algorithm Design

Here we assume first order Markov or vector AR stochastic model for the time “true” coefficients \(\omega_{k0}\)

$$\omega_{k0}(n+1) = F_k(n)\omega_{k0}(n) + q_k(n)$$  \hspace{1cm} (12)

where \(F_k(n)\) is the system matrix, and \(q_k(n)\) is a white noise sequence with \(\text{var}(q_k) = \sigma_{q_k}^2 I\).

Based on the state equation (12) and the observation equation (1), which make a state space model, using the observation error equation (6), the model-based adaptive (Kalman) algorithm, for the adaptive MMSE-ML receiver structure from Fig. 1, can be derived in the following form [2]

$$\omega_k(n) = F_k(\omega_k(n-1) + G_k(n)\nu_k(n)$$

$$\nu_k(n) = x_k(n) - \omega_k^H(n) u(n)$$

$$G_k(n) = \frac{F_k(n)P_k(n-1)u(n)}{\sigma_{q_k}^2 + u^H(n)P_k(n-1)u(n)}$$

$$P_k(n) = F_k(n)P_k(n-1)F_k^T(n) + \sigma_{q_k}^2 I$$

$$- G_k(n)(\sigma_{q_k}^2 + u^H(n)P_k(n-1)u(n))G_k^T(n)$$  \hspace{1cm} (13)

V. RLS AS A DEGENERATE MODEL-BASED ADAPTIVE ALGORITHM

By assuming \(F_k = I\) and \(q_k = 0\), the stochastic model for time-varying “true” weights (12) becomes the trivial, deterministic time-invariant model

$$\omega_{k0}(n+1) = \omega_{k0}(n)$$  \hspace{1cm} (14)

However, by using model (14) and assuming that “true” noise from (7) decays as time proceeds in the sense of \(\sigma_{e_k}^2(n) = \sigma_{e_k}^2 e^{-\gamma t}\), then the model-based adaptive algorithm (13) is reduced back to the RLS adaptive algorithm (11), [2].

Thus, we have the insight that the RLS algorithm does not explicitly recognize the time-varying nature of the optimal weight vector \(\omega_{k0}\), and consequently cannot offer good (tracking) performance in non-stationary environments.

VI. ADAPTIVE ALGORITHMS PERFORMANCE ANALYSIS

Although the centralized ML detection improves significantly the performance of the adaptive MMSE multiuser receiver [8], it does not affect tracking performance of the adaptive filters during the training period since training sequence is already known at the receiver. Since we analyze the performance of adaptive algorithms during the training period, without loss of generality, from this point forward, we assume \(\omega_k = v_k\) and \(u_k = y_k\). Consequently, the LMS system (9) is reduced to (9a) and the RLS and model based adaptive (Kalman) algorithm keep the same form (11) and (13), respectively, assuming above simplifications. Additionally, we assume that “true” coefficients \(\omega_{k0} = v_{k0}\) are time-varying, apart from the algorithm design. We denote \(\sigma_{q_k}^2 = \sigma_{q_k}^2 e^{-\gamma t}\) where \(\sigma_{q_k}^2\) is the variance of the “true” weight process and \(\gamma\) is “true” speed of change of the time varying “true” weights.

A. LMS Performance Analysis

1) Convergence: The first order LMS convergence criterion is given by [2]

$$[1 - \alpha \lambda_u] < 1, \ u = 1, 2, \ldots, p, \ i.e., \lambda_{\text{min}} > 0 \ 	ext{and} \ \alpha \lambda_{\text{max}} < 2$$  \hspace{1cm} (15)

where \(\lambda_u, u = 1, 2, \ldots, p\) are eigenvalues of \(R_y = E[y(n)y^H(n)]\).
The time constant for convergence is given by

\[ \tau = \frac{-1}{\log(\frac{\kappa - 1}{\kappa})} \]  

where \( \kappa = \lambda_{\text{max}}/\lambda_{\text{min}} \) is condition number of \( R_y \) [2].

Convergence criterion (15) depends on the \( R_y \) eigenvalue spread and, consequently, on the time-varying channel conditions. Since near-far effect usually causes a large \( R_y \) eigenvalue spread which significantly increases the time constant (16) and reduces the speed of convergence, then LMS is very near-far sensitive.

2) Second Order Stability and Tracking: The LMS algorithm relative steady state excess MSE or misadjustment due to noisy adaptation to time-varying “true” weights [2]

\[
M_{\text{tot}} = \frac{E_{\infty} - \sigma^2_k}{\sigma^2_k} = \frac{\alpha_0 p}{2} + \gamma \frac{\sigma^2_{\text{ne}} \sigma^2_k p}{\sigma^2_k} + O(\alpha_0^2)
\]

\[
= M_{\text{noise}} + M_{\text{lag}} + O(\alpha_0^2)
\]  

(17)

where \( E_{\infty} = \lim_{n \to \infty} E[|e(n)|^2] \); \( \alpha_0 = \sigma^2_0 \alpha; \sigma^2_y \) is the variance of \( y(n); p = tr(R_y)/\sigma^2_{\text{ne}}; tr(R_y) = \text{trace}(R_y) \) is the sum of the eigenvalues of \( R_y \); and \( c = O(\alpha^2) \) means \( c(\alpha)/\alpha^2 \) tends to a non-zero constant as \( \alpha \to 0 \). The expression \( M_{\text{noise}} \) in (17) is the excess MSE due to noise adaptation and \( M_{\text{lag}} \) is the excess “lag” terms due to the time-varying optimal weights.

Clearly \( M_{\text{tot}} \) (17) has an optimum wrt \( \alpha_0 \), namely

\[
M_{\text{opt}} = \frac{\gamma \sigma_{\text{ne}} \sigma_y}{\sigma_k} \quad \text{for} \quad \alpha_0,_{\text{opt}} = \frac{\gamma \sigma_{\text{ne}} \sigma_y}{\sigma_k}
\]

(18)

The second order stability requires [2]

\[
M_{\text{tot}} = \frac{\alpha_0 p}{2} + \gamma \frac{\sigma^2_{\text{ne}} \sigma^2_k p}{\sigma^2_k} + O(\alpha_0^2) < \frac{1}{3}
\]

(19)

In general, there is a conflict between speed of convergence, the steady state fluctuation and the excess lag fluctuation (tracking). The normalized step size \( \alpha_0 = \alpha \sigma^2_0 \) should be chosen to provide an appropriate tradeoff. However, since \( \alpha_0 \) depends on signal statistics \( \sigma^2_y \), channel time variations may cause local instabilities and the algorithm may blow up.

B. RLS Performance Analysis

1) Convergence: The first order RLS convergence criterion is given by

\[
0 < \mu_0 < 2
\]

(20)

where \( \mu_0 = 1 - \lambda \) is the forgetting factor [2]. The time constant for convergence is given by

\[
\tau = \frac{-1}{\log(1 - \mu_0)}.
\]

(21)

Convergence criterion (20) and the time constant for convergence (21) do not depend on the \( R_y \) eigenvalue spread. Consequently the RLS is near-far resistant. As a Newton method algorithm, the RLS provides a fast initial convergence. However, in non-stationary environments, this fast initial convergence usually has to be traded-off to improve stability and tracking.

2) Second Order Stability and Tracking Analysis: The relative steady state excess MSE or misadjustment for the RLS algorithm with time-varying “true” weights is [2]:

\[
M_{\text{tot}} = \frac{E_{\infty} - \sigma^2_k}{\sigma^2_k} = \frac{p \mu_0}{2} + \gamma \frac{\sigma^2_{\text{ne}} \sigma^2_k p}{\sigma^2_k} + O(\mu_0^2)
\]

\[
= M_{\text{noise}} + M_{\text{lag}} + O(\mu_0^2)
\]  

(22)

The second order stability requires [2]

\[
M_{\text{tot}} = M_{\text{noise}} + M_{\text{lag}} < \frac{1}{3}
\]

(23)

Furthermore, \( \mu_0,_{\text{opt}} \) and \( \mu_{\text{opt}} \) can be found to be

\[
M_{\text{opt}} = \frac{\gamma \sigma_{\text{ne}} \sigma_y}{\sigma_k} \quad \text{for} \quad \mu_{\text{opt}} = \frac{\gamma \sigma_{\text{ne}} \sigma_y}{\sigma_k}
\]

(24)

The model based approach in Section V reveals that the RLS explicitly assumes time-invariant optimal (“true”) weights, even in non-stationary environments. It means that, by choosing an appropriate \( \mu_0 = 1 - \lambda \) to provide a good tradeoff between speed of convergence, the steady state fluctuation and tracking, too much memory is usually involved through EWLS criterion (10). Thus, the RLS may provide very limited tracking performance.

C. Model-based Algorithm Performance Analysis

1) Convergence: Following [2], the first order convergence criterion is given by

\[
0 < \rho \frac{1}{\lambda_{\text{max}}} < 2
\]

(25)

where \( \rho = \sigma^2_{\text{ne}}/\sigma^2_k \) and \( \lambda_{\text{max}} = \lambda_{\text{max}}(R_y) \).

The time constant for convergence for given by

\[
\tau = \frac{-1}{\log\left(\frac{\kappa - 1}{\kappa}\right)}
\]

(26)

where \( \kappa = \lambda_{\text{max}}/\lambda_{\text{min}} \) is condition number of \( R_y \). Time constant for convergence (26) depends on eigenvalue spread but much more weakly than for LMS. For example, for for conditional number \( \kappa = \lambda_{\text{max}}/\lambda_{\text{min}} = 16 \), the LMS time constant (16) is equal to 8 and the model-based algorithm time constant (26) is 3. Thus, the model-based adaptive algorithm is much less near-far sensitive than LMS.

2) Second Order Stability and Tracking Analysis: Again following [2], the relative steady state excess MSE or misadjustment for the model-based adaptive algorithm with time-varying “true” weights is

\[
M_{\text{tot}} = \frac{E_{\infty} - \sigma^2_k}{\sigma^2_k} = \frac{pc_y}{2} + \frac{p}{2} \gamma \frac{\sigma_{\text{ne}} \sigma_y p}{\sigma_k} + O(\rho^2)
\]

\[
= M_{\text{noise}} + M_{\text{lag}} + O(\rho^2)
\]

(27)

where \( c_y = tr(R_y^2)/p \).

Furthermore, \( \rho_{\text{opt}} \) and \( \mu_{\text{opt}} \) can be found

\[
M_{\text{opt}} = \frac{pc_y \gamma \sigma_{\text{ne}} \sigma_y}{\sigma_k} \quad \text{for} \quad \rho_{\text{opt}} = \frac{\gamma \sigma_{\text{ne}} \sigma_y}{\sigma_k}
\]

(28)
The second order stability requires $M_{tot} = M_{noise} + M_{log} < \frac{1}{3}$ [2]. In addition
\begin{equation}
ct = \frac{1}{p}\text{tr}(R^T) = \frac{1}{p} \sum_{i=1}^p \lambda_i^\frac{1}{2}
\end{equation}
leads to $M_{opt}(\text{Kalman}) \leq M_{opt}(\text{LMS}), M_{opt}(\text{RLS})$, which means that model-based adaptive algorithm offers superior tracking to LMS and RLS.

VII. SIMULATION STUDY

We use the MATLAB multiple path fading channel rayleighchan.m, 8-users, spreading gain of 8. User no.1 is the user of interest. Simulations are performed for a range of normalized fading rates $f_D T_S$.

Fig. 2 shows the MSE performance of the adaptive algorithms for $f_D T_S = 10^{-3}$ (medium-speed fading conditions). Superior tracking performance of the model-based adaptive algorithm and very poor tracking performance of the RLS are demonstrated. It is a mistake often found in the literature to assert that tracking is optimized when convergence speed is maximal [7]. However, the fastest speed of convergence is not equivalent to the best tracking [7]. Fig. 2 corroborates that the LMS algorithm provides much better tracking than RLS, for the given channel conditions, even though the LMS exhibits slower initial convergence than RLS [3].

Fig. 3 depicts the results for very slow fading conditions ($f_D T_S = 10^{-7}$). Since the channel is almost time-invariant, the fastest initial convergence speed and reduced tracking requirements significantly improve the overall performance of RLS. However, by adjusting the model parameters according to the fading conditions the model-based algorithm still performs (slightly) better than the RLS and better than the LMS.

Fig. 4 based on the average square error performance (experimental MSE) corroborates the theoretical performance analysis provided in Section VI. The model-based adaptive algorithms provide superior tracking performance to LMS and RLS, especially for the medium speed fading conditions and the LMS algorithm outperforms the RLS which provides very limited tracking performance. On the other hand, if the channel is very slow (quasi time-invariant) with $f_D T_S \to 0$, then the superior initial convergence speed of RLS provides a performance advantage comparing to the LMS, given the reduced tracking requirements. Even more, overall performance of the RLS approaches to that of the model-based algorithm, since the model-based algorithm structure converges to the RLS algorithm structure [14].

REFERENCES