Low-Complexity Adaptive High-Resolution Channel Prediction for OFDM Systems

Ian C. Wong and Brian L. Evans

Wireless Networking and Communications Group
The University of Texas at Austin, Austin, Texas 78712
Email: {iwong, bevans}@ece.utexas.edu

***Abstract*— We propose a low-complexity adaptive high-resolution channel prediction algorithm for pilot symbol assisted orthogonal frequency division multiplexing (OFDM) systems. The algorithm is derived assuming a general time- and frequency-selective ray-based physical channel model, wherein each ray is parameterized by a complex amplitude, time-delay, and Doppler frequency. The algorithm is based on an improved rank and subspace adaptive Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT). The adaptive ESPRIT is used to efficiently extract the slowly varying time-delays and Doppler frequencies of each ray, followed by a simple rotational update to compute the complex amplitudes. Our algorithm has a principal computational complexity that is linear in the number of pilot subcarriers used for prediction, in contrast to cubic complexity required for a non-adaptive block processing based algorithm. We compare our approach with a previously proposed adaptive OFDM channel prediction algorithm based on standard least mean square (LMS) and recursive least squares (RLS) adaptive filters, and show that our algorithm achieves lower mean square error at a comparable computational complexity. We provide simulation results based on the IEEE 802.16e standard.**

I. Introduction

Next-generation OFDM-based wireless networks benefit greatly from the availability of channel state information (CSI) at the transmitter, allowing higher capacity and link reliability by using adaptive transmission techniques, e.g. adaptive power control, adaptive modulation and coding, and adaptive beamforming [1]. Unfortunately, in high mobility environments where the Doppler frequency is high and the channel changes rapidly, the CSI used by the transmitter would be outdated due to the processing and feedback delays. An effective means of overcoming the feedback delay is *channel prediction*.

Channel prediction algorithms for flat-fading channels have been investigated extensively in the past several years (see e.g. [2] and references therein). More recently, channel prediction algorithms for time- and frequency-selective fading channels in the context of OFDM have also been studied. In [3], each subcarrier in the OFDM symbol is assumed to be a flat-fading autoregressive wide-sense stationary (WSS) stochastic process, and a linear prediction filter based on previous downsampled channel estimates similar to that in [2] was used. In [4], a Kalman-filtering based channel estimation and unbiased channel power prediction was used on the pilot-subcarriers. In [5], adaptive short-term channel prediction on the time domain (TD) channel taps was proposed. Their approach uses an IFFT/FFT pair to derive the TD channel taps, perform the prediction, and then return to the frequency domain (FD). In [6], we proposed a high-resolution OFDM channel prediction based on 2-step, 1-dimensional frequency estimation using ESPRIT. This algorithm was shown to outperform previous methods, but has cubic computational complexity because of the required eigenvalue decomposition (EVD).

In this paper, we propose an adaptive version of the 2-step 1-dimensional approach that avoids the EVD by using efficient subspace tracking methods and a linear complexity adaptive ESPRIT formulation. The rank and subspace adaptive ESPRIT is used to efficiently extract the slowly varying time-delays and Doppler frequencies of each ray, followed by a simple rotational update to compute the complex amplitudes. We compare our approach with [5], which is similarly an adaptive OFDM channel prediction algorithm based on standard LMS and RLS adaptive filters. We show that our algorithm achieves lower mean square error (MSE) at a comparable computational complexity using simulation results based on the IEEE 802.16e [7] mobile broadband wireless access standard.

II. System Model

We consider an OFDM system with symbol period T_{sym} and subcarrier spacing Δf having proper cyclic prefix (CP) extension and sample timing, such that the sampled channel at the kth subcarrier of the nth OFDM symbol is

$$H(n,k) = \sum_{i=0}^{M-1} \alpha_i e^{j\nu_i} e^{j2\pi(f_i n - \tau_i k)} \tag{1}$$

where α_i, ν_i, f_i and τ_i are the amplitude, phase, discrete-time Doppler frequency, and discrete-time delay for the ith ray, respectively (see e.g. [8] for a similar model).

The received signal for the kth subcarrier of the nth OFDM symbol is given by $Y(n,k) = H(n,k)X(n,k)G_T(k)G_R(k) + W(n,k)$ where $G_T(k)$ and $G_R(k)$ are the pulse-shaping transmitter and receiver filter frequency response values at the kth subcarrier frequency, and $W(n,k)$ is a zero-mean, circular-symmetric, complex additive white Gaussian noise (AWGN) with variance σ^2. It is reasonable to assume that the used subcarriers are in the flat region of the transmitter and receiver filter frequency responses, such that the received signal is

$$Y(n,k) = H(n,k)X(n,k) + W(n,k). \tag{2}$$

We further assume that there are N_f equally spaced pilot subcarriers in every OFDM symbol inserted within the N_u used subcarriers. Let $D_f = \lfloor N_u/N_f \rfloor$ denote the frequency spacing in terms of the number of subcarriers between two adjacent pilot subcarriers. In order to avoid aliasing in the frequency domain, we require $D_f \leq 1/(\Delta f \bar{\tau})$ where $\bar{\tau}$ is the maximum delay spread. We denote the set of pilot subcarrier indices as $k_q = \left(\frac{N_f}{2} - q\right) D_f$. We also assume that a block of N_t current and previous OFDM symbols (which we call pilot symbols) equally spaced in time are available for the channel prediction algorithm. Let D_t denote the time spacing in terms of the number of OFDM symbols between pilot symbols. In order to avoid aliasing in the time domain, we require $D_t \leq 1/(2T_{sym}\bar{f})$ where $\bar{f}$ is the maximum Doppler frequency. We denote the set of pilot symbol indices as $n_l = lD_t$.

Using these $N_t \times N_f$ pilot subcarriers, we can perform a least-squares estimate of the channel at the pilot locations using the received signal $Y(n_l, k_q)$ and the known pilot symbols $X(n_l, k_q)$, given as

$$\hat{H}_{LS}(l,q) = \sum_{i=0}^{M-1} \alpha_i e^{j\phi_i} e^{j(\omega_i l + \varphi_i q)} + \widetilde{W}(l,q) \qquad (3)$$

where $\omega_i = 2\pi f_i D_t$, $\varphi_i = 2\pi \tau_i D_f$, and $\phi_i = \nu_i - \varphi_i \frac{N_f}{2}$.

For notational convenience, let

$$\hat{\mathbf{H}}_{LS} = \begin{bmatrix} \hat{\mathbf{h}}_0 & \dots & \hat{\mathbf{h}}_{N_t-1} \end{bmatrix} \qquad (4)$$

be the $N_f \times N_t$ matrix of the least-squares estimates, where

$$\hat{\mathbf{h}}_l = [\hat{H}_{LS}(l,0) \quad \dots \quad \hat{H}_{LS}(l,N_f-1)]^T \qquad (5)$$

III. OFDM Channel Prediction Algorithm

The main idea of our channel prediction algorithm is to first estimate the parameters in our deterministic channel model using the least squares estimates $\hat{\mathbf{H}}_{LS}$ in (4), and then simply extrapolate this model to predict the future channel.

We decompose (3) as

$$\begin{aligned} \hat{H}_{LS}(l,q) &\approx \sum_{p=0}^{L-1} \sum_{r=0}^{M_p-1} \alpha_{r,p} e^{j\phi_{r,p}} e^{j(\omega_{r,p} l + \varphi_p q)} + \widetilde{W}(l,q) \\ &= \sum_{p=0}^{L-1} g_p(l) e^{j\varphi_p q} + \widetilde{W}(l,q) \end{aligned}$$

where M_p is the number of sinusoidal rays and $g_p(l) = \sum_{r=0}^{M_p-1} c_{r,p} e^{j\omega_{r,p} l}$ is the complex gain for the pth propagation path, with $c_{r,p} = \alpha_{r,p} e^{j\phi_{r,p}}$ as the complex amplitude. In this decomposition, we exploit the fact that several (M_p) sinusoidal rays actually share the same time-delay value φ_p, and the total number of rays is $M = \sum_{p=0}^{L-1} M_p$. This is based on the observation that typical outdoor mobile wireless propagation environments involve clusters of scatters, e.g. a group of large buildings, and far away hills. Thus, the numerous propagating rays typically share a common time delay, which is essentially the assumption used in the classical tapped-delay line wide-band channel models [9]. This decomposition allows us to divide the estimation task into a *time delay (φ_p) estimation step*, and then a *Doppler frequency ($\omega_{r,p}$) estimation step*. We call this the 2-Step 1-Dimensional approach to channel parameter estimation. In [6], we proposed an ESPRIT-based block processing channel prediction algorithm based on this concept. Unfortunately, this algorithm has cubic computational complexity, which is high for an efficient implementation.

In this paper, we derive an adaptive version of the 2-step 1-dimensional ESPRIT-based algorithm that has linear complexity and is able to efficiently track the slowly time-varying parameters, and also adapt to the time-varying model order (see Table I). This can be used as a completely adaptive stand-alone algorithm, or it can be used for tracking the channel parameters after being initialized with [6]. Since both time delay and Doppler frequency estimation entail essentially the same operations, we shall limit our discussion to the time-delay estimation step, and explain briefly the modifications necessary for the Doppler frequency estimation step.

A. Autocorrelation estimation and subspace tracking

We update the matrix of least-squares estimates using an exponential window, given as

$$\hat{\mathbf{H}}_{LS,n}^H = \begin{bmatrix} (1-\beta)^{1/2} \hat{\mathbf{h}}_n^H \\ \beta^{1/2} \hat{\mathbf{H}}_{LS,n-1}^H \end{bmatrix} \qquad (6)$$

where $\hat{\mathbf{H}}_{LS,n-1}^H$ is the "old" observation matrix, $\hat{\mathbf{h}}_n$ is the "new" LS-estimated N_f-length vector, and $0 < \beta < 1$ is the exponential forgetting factor. This leads to the development of efficient singular value decomposition (SVD) updating algorithms for computing the $\mathcal{L}$[1] dominant singular values and right singular vectors of the growing data matrix at each time step [10], where typically, $\mathcal{L} \ll N_f$. Our algorithm is based on the Bi-iterative SVD 3 subspace tracker from [10, Table III], with the stabilizing modifications proposed in [11, p. 2995] (see Table II). We chose this algorithm over other algorithms primarily due to its ability to track the eigenvalues (for model-order tracking), its guarantee of orthonormality of the singular vectors (for efficient adaptive ESPRIT implementation, which will be discussed later), and its low computational complexity.

B. Model-order estimation

Note that by squaring the estimated $\mathcal{L}$ singular values, we have an estimate of the dominant eigenvalues. However, in order to employ the minimum description length (MDL) criterion [12], we also require an estimate of the rest of the $N_f - \mathcal{L}$ non-dominant eigenvalues. Theoretically, the noise eigenvalues should all be equal to the noise power σ^2. Thus, we estimate these non-dominant eigenvalues by simply equating them to an estimate of the noise power. Note that $\hat{\mathbf{h}}^{\perp}(n)$ in Step 2 of Table II is the component of $\hat{\mathbf{h}}(n)$ that lies in the noise subspace. The noise power estimate is then

$$\hat{\sigma}^2(n) = \beta \hat{\sigma}^2(n-1) + (1-\beta) \frac{\hat{\mathbf{h}}^{\perp}(n)^H \hat{\mathbf{h}}^{\perp}(n)}{N_f - \mathcal{L}} \qquad (7)$$

[1]We assume that the maximum number of paths $\mathcal{L}$ is known *à priori*. This value is typically determined by the propagation environment, and the desired accuracy of the channel characterization.

TABLE I

PROPOSED ALGORITHM STEP 1: TIME-DELAY ESTIMATION

Input: $\hat{\mathbf{h}}$

Output: $\hat{L}, \{\hat{\varphi}_p\}_{p=1}^{\hat{L}}, \hat{\mathbf{g}}$

Operation	**Comp.**
Autocorrelation estimation and subspace tracking	
Run subspace tracker $\xrightarrow{\text{ModBi-SVD3}}$ $\hat{\mathbf{V}}(n), \hat{\boldsymbol{\sigma}}, \mathbf{G}_A, \hat{\mathbf{h}}^{\perp}$	$10N_f\mathcal{L}$
Model-order estimation	
$\hat{\sigma}^2(n) = \beta\hat{\sigma}^2(n-1) + (1-\beta)\frac{\hat{\mathbf{h}}^{\perp H}\hat{\mathbf{h}}^{\perp}}{N_f-\mathcal{L}}$	N_f
$\hat{\lambda_i}^f = \begin{cases} \hat{\sigma_i}^2, & i \in [1,\mathcal{L}] \\ \hat{\sigma}^2(n), & i \in [\mathcal{L}+1, N_f] \end{cases}$	$\mathcal{L}$
$\tilde{N}_t = \frac{1-\beta^n}{1-\beta}$	
$\hat{L} = \arg\min_{\mu} -\log\left(\frac{\left(\prod_{k=\mu+1}^{N_f}\hat{\lambda}_k^f\right)^{\frac{1}{N_f-\mu}}}{\frac{1}{N_f-\mu}\sum_{k=\mu+1}^{N_f}\hat{\lambda}_k^f}\right)^{\tilde{N}_t(N_f-\mu)}$ s.t. $\mu \in [1,\mathcal{L}]$ $+\frac{1}{2}\mu(2N_f-\mu)\log\tilde{N}_t$	$N_f\mathcal{L}$
Time-delay estimation using adaptive ESPRIT	
$\overline{\hat{\mathbf{h}}^{\perp}} = \hat{\mathbf{h}}^{\perp}/\parallel\hat{\mathbf{h}}^{\perp}\parallel$	
$\overline{\hat{\mathbf{h}}_1^{\perp}} = [\mathbf{I}_{N_f-1} \quad \mathbf{0}_{(N_f-1)\times 1}]\overline{\hat{\mathbf{h}}^{\perp}}$	
$\overline{\hat{\mathbf{h}}_2^{\perp}} = [\mathbf{0}_{(N_f-1)\times 1} \quad \mathbf{I}_{N_f-1}]\overline{\hat{\mathbf{h}}^{\perp}}$	
$\hat{\mathbf{V}}_1(n-1) = [\mathbf{I}_{N_f-1} \quad \mathbf{0}_{(N_f-1)\times 1}]\hat{\mathbf{V}}(n-1)$	
$\hat{\mathbf{V}}_2(n-1) = [\mathbf{0}_{(N_f-1)\times 1} \quad \mathbf{I}_{N_f-1}]\hat{\mathbf{V}}(n-1)$	
$\begin{bmatrix} \boldsymbol{\Upsilon}_{\mathcal{L}}(n) & * \\ * & * \end{bmatrix} = \mathbf{G}(n)\times$ $\begin{bmatrix} \boldsymbol{\Upsilon}(n-1) & \hat{\mathbf{V}}_1^H(n-1)\overline{\hat{\mathbf{h}}_2^{\perp}} \\ \overline{\hat{\mathbf{h}}_1^{\perp H}}\hat{\mathbf{V}}_2(n-1) & \overline{\hat{\mathbf{h}}_1^{\perp H}}\hat{\mathbf{h}}_2^{\perp}(n) \end{bmatrix}\mathbf{G}^H(n)$	$2N_f\mathcal{L}$
$\boldsymbol{\Upsilon}_{\mathcal{L}}(n) \xrightarrow{\text{Extract } \hat{L}\times\hat{L} \text{ top-left submatrix}} \boldsymbol{\Upsilon}_{\hat{L}}(n)$	
$\hat{\mathbf{V}}(n) \xrightarrow{\text{Extract } \hat{L} \text{ bottom-left-most row vector}} \hat{\mathbf{v}}_{\hat{L},1}^H(n)$	
$\hat{\boldsymbol{\Phi}}_{\hat{L}}(n) = \boldsymbol{\Upsilon}_{\hat{L}}(n) + \frac{\hat{\mathbf{v}}_{\hat{L},1}(n)\left(\boldsymbol{\Upsilon}_{\hat{L}}(n)^H\hat{\mathbf{v}}_{\hat{L},1}(n)\right)^H}{1-\parallel\hat{\mathbf{v}}_{\hat{L},1}(n)\parallel^2}$	$\hat{L}^2$
$\hat{\boldsymbol{\Phi}}_{\hat{L}}(n) \xrightarrow{\text{EVD}} \sum_{p=1}^{\hat{L}}\epsilon_{\hat{p}}\mathbf{u}_p\mathbf{u}_p^H$	$O(\hat{L}^3)$
$\hat{\varphi}_p = \arg(\hat{\epsilon}_p), \quad p = 0,\ldots,\hat{L}-1$	$\hat{L}$
Complex amplitude estimation	
$\Delta\hat{\varphi}_p(n) = \hat{\varphi}_p(n) - \hat{\varphi}_p(n-1), \quad p = 1,\ldots,\hat{L}$	$\hat{L}$
$\boldsymbol{\Delta}_{\hat{\varphi}}(n) = \text{diag}\{e^{j\Delta\hat{\varphi}_0(n)},\ldots,e^{j\Delta\hat{\varphi}_{\hat{L}}(n)}\}$	$\hat{L}$
$\mathbf{Q}(n) = \mathbf{Q}(n-1)$	
$\mathbf{R}(n) = \mathbf{R}(n-1)\boldsymbol{\Delta}_{\hat{\varphi}}(n)$	$\hat{L}^2/2$
$\mathbf{R}(n)\hat{\mathbf{g}}(n) = \mathbf{Q}^H(n)\hat{\mathbf{h}}(n) \xrightarrow{\text{Backsubstitution}} \hat{\mathbf{g}}(n)$	$N_f\hat{L} + \frac{\hat{L}^2}{2}$

Therefore, our eigenvalue estimates can be written as

$$\hat{\lambda_i}^f(n) = \begin{cases} \hat{\sigma_i}^2(n), & i \in [1,\mathcal{L}] \\ \hat{\sigma}^2(n), & i \in [\mathcal{L}+1, N_f] \end{cases} \tag{8}$$

where $\hat{\sigma}_i(n)$ is the ith element of the singular value estimates given in Step 11 of Table II. Finally, by using the effective window length at time n, given as $\tilde{N}_t(n) = \frac{1-\beta^n}{1-\beta}$, we derive the time-varying model order estimate as

$$\hat{L} = \arg\min_{\mu} -\log\left(\frac{\left(\prod_{k=\mu+1}^{N_f}\hat{\lambda}_k^f(n)\right)^{\frac{1}{N_f-\mu}}}{\frac{1}{N_f-\mu}\sum_{k=\mu+1}^{N_f}\hat{\lambda}_k^f(n)}\right)^{\tilde{N}_t(n)(N_f-\mu)} \quad \text{s.t. } \mu \in [1,\mathcal{L}] \quad +\frac{1}{2}\mu(2N_f-\mu)\log\tilde{N}_t(n) \tag{9}$$

TABLE II

SUBSPACE TRACKER BI-SVD 3 [10] MODIFIED BY [11, P. 2995]. $\mathbf{G}_A(n)$ IS A SEQUENCE OF $(2\mathcal{L}-1)$ GIVENS ROTATIONS.

Input: $\hat{\mathbf{h}}(n)$

Output: $\hat{\mathbf{V}}(n), \hat{\boldsymbol{\sigma}}(n), \mathbf{G}_A(n), \hat{\mathbf{h}}^{\perp}(n)$

	Operation	**Comp.**
1	$\mathbf{d}(n) = \hat{\mathbf{V}}^H(n-1)\hat{\mathbf{h}}(n)$	$N_f\mathcal{L}$
2	$\hat{\mathbf{h}}^{\perp}(n) = \hat{\mathbf{h}}(n) - \hat{\mathbf{V}}(n-1)\mathbf{d}(n)$	$N_f\mathcal{L}$
3	$\mathbf{D}(n) = \mathbf{R}_B(n-1)\boldsymbol{\Theta}_A(n-1)$	$\mathcal{L}^3/3$
4	$\begin{bmatrix} (1-\beta)^{1/2}\mathbf{d}^H(n) \\ \beta^{1/2}\mathbf{D}(n) \end{bmatrix} \xrightarrow{\text{Givens QR}} \begin{bmatrix} \mathbf{R}_B(n) \\ \mathbf{0}_{1\times\mathcal{L}} \end{bmatrix}$	$2\mathcal{L}^3$
5	$\mathbf{d}_R^H(n)\mathbf{R}_B(n) = \mathbf{d}^H(n) \xrightarrow{\text{Back substitution}} \mathbf{d}_R^H(n)$	$\mathcal{L}^2/2$
6	$\mathbf{D}_R(n)\mathbf{R}_B(n) = \mathbf{D}(n) \xrightarrow{\text{Back substitution}} \mathbf{D}_R(n)$	$\mathcal{L}^3/3$
7	$\mathbf{R}_A(n-1)\mathbf{D}_R(n) \xrightarrow{\text{Extract upper triangular portion}} \mathbf{T}$	
8	$\begin{bmatrix} \mathbf{R}_A(n) \\ \mathbf{0}_{1\times\mathcal{L}} \end{bmatrix} = \mathbf{G}_A(n)\begin{bmatrix} \beta\mathbf{T} + (1-\beta)\mathbf{d}(n)\mathbf{d}_R^H(n) \\ (1-\beta)\parallel\hat{\mathbf{h}}^{\perp}(n)\parallel\mathbf{d}_R^H(n) \end{bmatrix}$	$4\mathcal{L}^2$
9	$\begin{bmatrix} \hat{\mathbf{V}}(n) & \mathbf{q}(n) \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{V}}(n-1) & \frac{\hat{\mathbf{h}}^{\perp}(n)}{\parallel\hat{\mathbf{h}}^{\perp}(n)\parallel} \end{bmatrix}\mathbf{G}_A^H(n)$	$8N_f\mathcal{L}$
10	$\mathbf{G}_A^H(n) \xrightarrow{\text{Extract upper left } \mathcal{L} \text{ square matrix}} \boldsymbol{\Theta}_A(n)$	
11	$\hat{\boldsymbol{\sigma}}(n) = \text{diag}(\mathbf{R}_B(n)\boldsymbol{\Theta}_A(n))$	$\mathcal{L}^3/3$

C. Time-delay estimation using adaptive ESPRIT

Fast adaptive ESPRIT algorithms have already been investigated in the past [13] [14] [15]. However, they are either unstable for closely spaced sinusoids (e.g. [13] as observed in [11] and [15]); or are stable but not model-order adaptive [14] [15]. Thus, we develop our own fast adaptive ESPRIT algorithm for time-delay estimation with $O(N_f\mathcal{L})$ complexity that is both stable and model-order adaptive.

Let $\hat{\mathbf{V}}_{\hat{L}}(n)$ be the matrix composed of the dominant (leftmost) $\hat{L}$ eigenvector columns of $\hat{\mathbf{V}}(n)$, the output of the subspace tracker from Table II. Perform the partitions

$$\hat{\mathbf{V}}_{\hat{L}}(n) = \begin{bmatrix} \hat{\mathbf{V}}_{\hat{L},1}(n) \\ \hat{\mathbf{v}}_{\hat{L},1}^H(n) \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{v}}_{\hat{L},2}^H(n) \\ \hat{\mathbf{V}}_{\hat{L},2}(n) \end{bmatrix} \tag{10}$$

where $\hat{\mathbf{V}}_{\hat{L},1}(n)$ and $\hat{\mathbf{V}}_{\hat{L},2}(n)$ are the $(N_f-1)\times\hat{L}$ upper and lower submatrices of $\hat{\mathbf{V}}_{\hat{L}}(n)$; $\hat{\mathbf{v}}_{\hat{L},1}^H(n)$ and $\hat{\mathbf{v}}_{\hat{L},2}^H(n)$ are the bottommost and topmost row vectors of the same matrix. The adaptive equivalent of the ESPRIT spectral matrix [6] is

$$\hat{\boldsymbol{\Phi}}_{\hat{L}}(n) = \left(\hat{\mathbf{V}}_{\hat{L},1}^H(n)\hat{\mathbf{V}}_{\hat{L},1}(n)\right)^{-1}\hat{\mathbf{V}}_{\hat{L},1}^H(n)\hat{\mathbf{V}}_{\hat{L},2}(n) \tag{11}$$

Our initial objective is to derive an $O(N_f\mathcal{L})$ updating scheme for $\hat{\boldsymbol{\Phi}}_{\hat{L}}(n)$, where we proceed similarly as in [15, Sec. 3.1]. Since $\hat{\mathbf{V}}_{\hat{L}}(n)$ has orthonormal columns, we have $\hat{\mathbf{V}}_{\hat{L},1}^H(n)\hat{\mathbf{V}}_{\hat{L},1}(n) = \mathbf{I}_{\hat{L}} - \hat{\mathbf{v}}_{\hat{L},1}(n)\hat{\mathbf{v}}_{\hat{L},1}^H(n)$, which is simply a rank-one modification of the $\hat{L}\times\hat{L}$ identity matrix. Using the matrix inversion lemma [16], we can write $\left(\hat{\mathbf{V}}_{\hat{L},1}^H(n)\hat{\mathbf{V}}_{\hat{L},1}(n)\right)^{-1} = \mathbf{I}_{\hat{L}} + \frac{\hat{\mathbf{v}}_{\hat{L},1}(n)\hat{\mathbf{v}}_{\hat{L},1}^H(n)}{1-\parallel\hat{\mathbf{v}}_{\hat{L},1}(n)\parallel^2}$. Using this in (11), we have

$$\hat{\boldsymbol{\Phi}}_{\hat{L}}(n) = \boldsymbol{\Upsilon}_{\hat{L}}(n) + \frac{\hat{\mathbf{v}}_{\hat{L},1}(n)\left(\boldsymbol{\Upsilon}_{\hat{L}}(n)^H\hat{\mathbf{v}}_{\hat{L},1}(n)\right)^H}{1-\parallel\hat{\mathbf{v}}_{\hat{L},1}(n)\parallel^2} \tag{12}$$

$$\boldsymbol{\Upsilon}_{\hat{L}}(n) = \hat{\mathbf{V}}_{\hat{L},1}^H(n)\hat{\mathbf{V}}_{\hat{L},2}(n) \tag{13}$$

Note that the spectral matrix (12) is simply an $O(\hat{L}^2)$ rank-one update to (13), but computing (13) directly requires $O(N_f\hat{L}^2)$. In [15, Sec. 3.1], they proceed to use subspace trackers that employ rank-one updates to derive an $O(N_f\mathcal{L})$ update for (13). Unfortunately, these subspace trackers do not track the actual eigenvalues, and are therefore unable to estimate and track the time-varying model order $\hat{L}$. Thus, we proceed to derive a novel $O(N_f\mathcal{L})$ recursion for (13).

The main subspace update step of the modified Bi-SVD 3 is given by (cf. Step 9 of Table II)

$$\begin{bmatrix}\hat{\mathbf{V}}(n) & \mathbf{q}(n)\end{bmatrix} = \begin{bmatrix}\hat{\mathbf{V}}(n-1) & \overline{\hat{\mathbf{h}}^{\perp}(n)}\end{bmatrix}\mathbf{G}^H(n) \tag{14}$$

where $\overline{\hat{\mathbf{h}}^{\perp}(n)} = \frac{\hat{\mathbf{h}}^{\perp}(n)}{\|\hat{\mathbf{h}}^{\perp}(n)\|}$. By pre-multiplying both sides of (14) by $[\mathbf{I}_{N_f-1} \quad \mathbf{0}_{(N_f-1)\times 1}]$ and $[\mathbf{0}_{(N_f-1)\times 1} \quad \mathbf{I}_{N_f-1}]$, we get the $(N_f-1)\times(\mathcal{L}+1)$ top submatrix

$$\begin{bmatrix}\hat{\mathbf{V}}_1(n) & \mathbf{q}_1(n)\end{bmatrix} = \begin{bmatrix}\hat{\mathbf{V}}_1(n-1) & \overline{\hat{\mathbf{h}}_1^{\perp}(n)}\end{bmatrix}\mathbf{G}^H(n) \tag{15}$$

and the $(N_f-1)\times(\mathcal{L}+1)$ bottom submatrix

$$\begin{bmatrix}\hat{\mathbf{V}}_2(n) & \mathbf{q}_2(n)\end{bmatrix} = \begin{bmatrix}\hat{\mathbf{V}}_2(n-1) & \overline{\hat{\mathbf{h}}_2^{\perp}(n)}\end{bmatrix}\mathbf{G}^H(n) \tag{16}$$

respectively. Taking the Hermitian transpose of both sides of (15) and pre-multiplying it to (16), we have

$$\begin{bmatrix}\boldsymbol{\Upsilon}_{\mathcal{L}}(n) & * \\ * & *\end{bmatrix} = \mathbf{G}(n)\times \begin{bmatrix}\boldsymbol{\Upsilon}_{\mathcal{L}}(n-1) & \hat{\mathbf{V}}_1^H(n-1)\overline{\hat{\mathbf{h}}_2^{\perp}(n)} \\ \overline{\hat{\mathbf{h}}_1^{\perp H}(n)}\hat{\mathbf{V}}_2(n-1) & \overline{\hat{\mathbf{h}}_1^{\perp H}(n)\hat{\mathbf{h}}_2^{\perp}(n)}\end{bmatrix}\mathbf{G}^H(n) \tag{17}$$

$$\boldsymbol{\Upsilon}_{\mathcal{L}}(n) = \hat{\mathbf{V}}_1^H(n)\hat{\mathbf{V}}_2(n) \tag{18}$$

and '$*$' represents unused quantities. Observe that once we initialize (18), we no longer need to perform this $O(N_f\mathcal{L}^2)$ matrix multiplication, since we can simply update it as a single $\mathcal{L}\times\mathcal{L}$ matrix. Furthermore, since $\mathbf{G}(n)$ represents a sequence of $2\mathcal{L}-1$ Givens plane rotations [17], we can update $\boldsymbol{\Upsilon}_{\mathcal{L}}(n)$ in just $8\mathcal{L}$ operations. Also notice that by partitioning $\hat{\mathbf{V}}_1(n) = [\hat{\mathbf{V}}_{\hat{L},1}(n) \quad *]$ and $\hat{\mathbf{V}}_2(n) = [\hat{\mathbf{V}}_{\hat{L},2}(n) \quad *]$,

$$\boldsymbol{\Upsilon}_{\mathcal{L}}(n) = \begin{bmatrix}\boldsymbol{\Upsilon}_{\hat{L}}(n) & * \\ * & *\end{bmatrix} \tag{19}$$

where $\boldsymbol{\Upsilon}_{\hat{L}}(n)$ is given by (13). Thus, by extracting the $\hat{L}\times\hat{L}$ upper-left submatrix of the left hand side of (17) and using it in (12), we arrive at an efficient adaptive ESPRIT spectral matrix updating algorithm with principal complexity $O(N_f\mathcal{L})$. Finally, we extract the radian phase of each of the complex eigenvalues of the small $\hat{L}\times\hat{L}$ matrix $\hat{\boldsymbol{\Phi}}_{\hat{L}}(n)$ (12), giving an estimate of the time-delays $\{\hat{\varphi}_p\}_{p=1}^{\hat{L}}$.

D. Complex amplitude estimation

Given the time delay estimates $\{\hat{\varphi}_p\}_{p=1}^{\hat{L}}$, the maximum-likelihood estimate for the $\hat{L}\times N_t$ matrix $\hat{\mathbf{G}}_{ML}$ of complex amplitudes can be computed using back-substitution of

$$\mathbf{R}\hat{\mathbf{G}}_{ML} = \mathbf{Q}^H\hat{\mathbf{H}}_{LS} \tag{20}$$

where $\hat{\mathbf{G}}_{ML} = [\hat{\mathbf{g}}_0 \ldots \hat{\mathbf{g}}_{\hat{L}}]^T$ with $\hat{\mathbf{g}}_p = [\hat{g}_p(n)\ldots\hat{g}_p(n-N_t+1)]^T$, and

$$\mathbf{QR} = \mathbf{E} = \begin{bmatrix} e^{j\hat{\varphi}_0 0} & \cdots & e^{j\hat{\varphi}_{\hat{L}-1}0} \\ \vdots & \ddots & \vdots \\ e^{j\hat{\varphi}_0(N_f-1)} & \cdots & e^{j\hat{\varphi}_{\hat{L}-1}(N_f-1)}\end{bmatrix} \tag{21}$$

is the "skinny" QR decomposition of the $N_f\times\hat{L}$ Fourier transform matrix $\mathbf{E}$. Let $\Delta\hat{\varphi}_i(n) = \hat{\varphi}_i(n) - \hat{\varphi}_i(n-1)$ denote the difference between the time-delay estimates at two consecutive time steps, and let $\boldsymbol{\Delta}_{\hat{\varphi}}(n) = \text{diag}\{e^{j\Delta\hat{\varphi}_0(n)},\ldots,e^{j\Delta\hat{\varphi}_{\hat{L}}(n)}\}$ be the diagonal $\hat{L}\times\hat{L}$ phase-difference matrix. Suppose that we have at hand the previous "skinny" QR decomposition of $\mathbf{E}(n-1) = \mathbf{Q}(n-1)\mathbf{R}(n-1)$, we can then write

$$\begin{aligned}\mathbf{E}(n) &= \mathbf{E}(n-1)\boldsymbol{\Delta}_{\hat{\varphi}}(n) \\ &= \mathbf{Q}(n-1)\mathbf{R}(n-1)\boldsymbol{\Delta}_{\hat{\varphi}}(n)\end{aligned} \tag{22}$$

which gives us $\mathbf{E}(n) = \mathbf{Q}(n)\mathbf{R}(n)$ where $\mathbf{Q}(n) = \mathbf{Q}(n-1)$ and $\mathbf{R}(n) = \mathbf{R}(n-1)\boldsymbol{\Delta}_{\hat{\varphi}}(n)$ which is still upper triangular. Thus, the only update to the QR decomposition is the multiplication of the diagonal phase-difference matrix to $\mathbf{R}(n-1)$, which can be performed in $\hat{L}^2/2$ operations. Finally, the new complex amplitude least-squares estimate is given as

$$\mathbf{R}(n)\hat{\mathbf{g}}(n) = \mathbf{Q}^H(n)\hat{\mathbf{h}}(n) \tag{23}$$

The above discussion assumes that the model order is static between iterations, which is the case most often encountered since the time delays are slowly varying. However, it is conceivable that there are small model order changes (e.g. 1 or 2) from one iteration to the next. We can use the same framework above in conjunction with efficient column deleting and appending procedures for the QR decomposition [17], which have complexity $O(\hat{L}^2)$ and $O(N_f\hat{L})$ respectively.

In tracking the Doppler-frequencies, we can use the same procedure for each path $p = 1,\ldots,\hat{L}$, with $\hat{\mathbf{g}}_p$ as the N_t-length observation vector for the pth path, and where we track the $\mathcal{M}_p$-dimension dominant subspace. This can be done with $O(N_t\mathcal{M})$ principal computational complexity, where $\mathcal{M} = \sum_{p=0}^{\hat{L}-1}\mathcal{M}_p$. To complete the channel prediction algorithm, we simply substitute the estimated parameters to the channel model in (1) with the future time index we wish to predict.

E. Computational complexity

Table III summarizes the computational complexity of the proposed adaptive channel prediction algorithm, initialized using [6]. We compare this with the adaptive algorithm using LMS and RLS proposed in [5], initialized with the reduced complexity MMSE TD prediction also proposed in [5]. Typically, $N > N_f \gg \mathcal{L} \geq \hat{L}$, $N_t \gg \mathcal{M}_p \geq \hat{M}_p$ and $N_{cp} > \hat{L}$, where N_{cp} is the cyclic prefix length. Thus, we can see that in the initialization stage, the proposed algorithm is more complex than that of [5]. In the tracking stage, our algorithm is slightly more complex than LMS, but is much less complex than RLS. In the prediction stage, [5] is more complex than our algorithm. The initialization stage only needs to be performed once, whereas both prediction and tracking are performed for each OFDM symbol to be predicted.

TABLE III

COMPUTATIONAL COMPLEXITY

Stage	Proposed Algorithm	Algorithm from [5]
Initialization	$O(N_f^3 + \hat{L}N_t^3)$ [6]	$O(N \log_2 N + N_{cp}N_t^2)$
Tracking	$O(N_f\mathcal{L} + N_t\mathcal{M})$	LMS - $O(N_{cp}N_t)$ RLS - $O(N_{cp}N_t^2)$
Prediction	$O(N\hat{L} + \hat{M})$	$O(N \log_2 N + N_{cp}N_t)$

TABLE IV

SIMULATION PARAMETERS

Parameter	Value	Parameter	Value
Subcarriers (N)	512	Used Subcarriers (N_u)	426
CP length (N_{cp})	16	Forgetting factor (β)	0.99
Bandwidth	5 MHz	TD pilots (N_t)	100
Sampling Freq.	5.7 Ms/s	FD pilots (N_f)	142
Carrier Freq.	2.6 GHz	TD pilot spacing (D_t)	22
Symbol Period	92.44 μs	FD pilot spacing (D_f)	3
Max. Delay ($\bar{\tau}$)	2.5 μs	Max. paths ($\mathcal{L}$)	10
Max. Dopp. (f)	180.6 Hz	Max. rays/path ($\mathcal{M}_p$)	20
Paths (L)	6	Estim. paths ($\mathbb{E}\{\hat{L}\}$)	6
Rays/path (M_p)	16	Estim. rays/path ($\mathbb{E}\{\hat{M}_p\}$)	16

IV. SIMULATION RESULTS

We provide simulation results for an outdoor mobile OFDM system based on the IEEE 802.16e standard [7] using time-division duplexing with a $T_{sym}D_t = 2$ ms frame length, whose parameters are given in Table IV. We estimate the channel during the preambles spaced $D_t = 22$ symbols apart, and we predict a whole frame (22 symbols) ahead. We assume a wireless channel following the ITU-Vehicular A [18] power delay profile with $L = 6$ paths, where each path is simulated using the modified Jakes method with $M_p = 16$ rays per path. We allow both time-delays and Doppler frequencies to vary linearly over time to simulate the non-stationary channel.

We initialized the adaptive prediction algorithm using [6] for $N_t = 100$ frames, and then used the adaptive algorithm for the next 500 frames. We then compared the performance of our proposed algorithm with the algorithm proposed in [5]. This method used an IFFT of the channel estimates from the $N_u = 426$ used subcarriers to derive the first N_{cp} TD channel taps. These TD channel taps are then used to perform Pth-order linear MMSE prediction. An FFT of the predicted TD taps are finally used to estimate the frequency response of the future channel. After designing the prediction filter, conventional LMS and RLS adaptive filters are then used to track the non-stationary channel. We ran this algorithm with $P = N_t = 100$ to match the complexity of our algorithm. Figure 1 shows the MSE performance for 20 channel realizations, with 500 frames per realization, for mobile velocities 30 and 75 kph. It can be seen that our algorithm outperforms [5] except for the case of very low SNR, where the MDL model-order selection is unable to detect the weakest paths, resulting in higher MSE when compared to using N_{cp} TD taps. In any case, the proposed algorithm presents a good MSE vs. computational complexity tradeoff for medium to high SNR regimes.

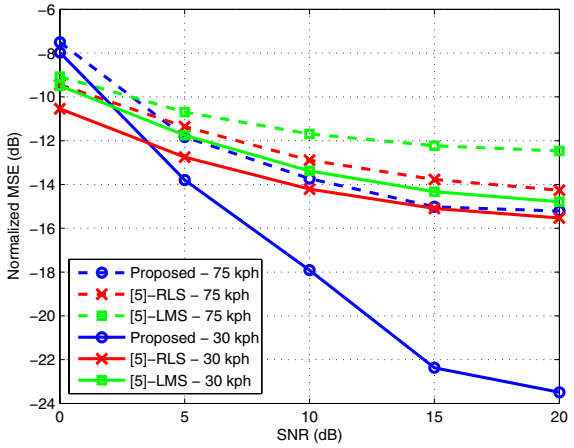

Fig. 1. Prediction MSE results for the proposed algorithm and the algorithms in [5] using simulation parameters given in Table IV.

REFERENCES

[1] T. Keller and L. Hanzo, "Adaptive multicarrier modulation: a convenient framework for time-frequency processing in wireless communications," *Proc. IEEE*, vol. 88, no. 5, pp. 611–640, May 2000.

[2] A. Duel-Hallen, S. Hu, and H. Hallen, "Long-Range Prediction of Fading Signals," *IEEE Signal Processing Mag.*, vol. 17, no. 3, pp. 62–75, May 2000.

[3] A. Forenza and R. W. Heath, Jr., "Link Adaptation and Channel Prediction in Wireless OFDM Systems," in *Proc. IEEE Midwest Symp. on Circuits and Sys.*, Aug 2002, pp. 211–214.

[4] M. Sternad and D. Aronsson, "Channel estimation and prediction for adaptive OFDM downlinks [vehicular applications]," in *Proc. IEEE Vehicular Technology Conference*, vol. 2, Oct 2003, pp. 1283–1287.

[5] D. Schafhuber and G. Matz, "MMSE and Adaptive Prediction of Time-Varying Channels for OFDM Systems," *IEEE Trans. Wireless Commun.*, vol. 4, no. 2, pp. 593–602, Mar 2005.

[6] I. C. Wong and B. L. Evans, "Joint Channel Estimation and Prediction for OFDM Systems," in *Proc. IEEE Global Telecommunications Conference*, St. Louis, MO, Dec 2005.

[7] *Air Interface for Fixed and Mobile Broadband Wireless Access Systems*, IEEE Std. 802.16e/D5, 2004.

[8] S. Barbarossa and A. Scaglione, "Theoretical bounds on the estimation and prediction of multipath time-varying channels," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Sig. Proc.*, vol. 5, Jun 2000, pp. 2545–2548.

[9] T. S. Rappaport, *Wireless communications : principles and practice*, 2nd ed. Prentice Hall, 2002.

[10] P. Strobach, "Bi-iteration SVD subspace tracking algorithms," *IEEE Trans. Signal Processing*, vol. 45, no. 5, pp. 1222–1240, May 1997.

[11] S. Ouyang and Y. Hua, "Bi-iterative least-square method for subspace tracking," *IEEE Trans. Signal Processing*, vol. 53, no. 8, pp. 2984–2996, Aug 2005.

[12] M. Wax and T. Kailath, "Detection of signals by information theoretic criteria," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 33, no. 2, pp. 387–392, Apr 1985.

[13] P. Strobach, "Fast recursive subspace adaptive ESPRIT algorithms," *IEEE Trans. Signal Processing*, vol. 46, no. 9, pp. 2413–2430, Sep 1998.

[14] R. Badeau, G. Richard, and B. David, "Adaptive ESPRIT algorithm based on the PAST subspace tracker," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Sig. Proc.*, vol. 6, Apr 2003, pp. VI–229–32.

[15] ——, "Fast adaptive ESPRIT algorithm," in *IEEE Workshop on Statistical Signal Processing*, Bordeaux, France, July 17-20 2005.

[16] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge University Press, 1985.

[17] D. S. Watkins, *Fundamentals of Matrix Computations*, 2nd ed. John Wiley, 2002.

[18] *Selection procedures for the choice of radio transmission technologies for the UMTS*, ETSI Std. TR 101 112 v. 3.2.0, 1998.