Tracking and Prediction of Multipath Components in Wireless MIMO Channels

Dmitriy Shutin, Gernot Kubin
Signal Processing and Speech Communication Laboratory
Graz University of Technology
Graz, Austria
dshutin@tugraz.at, gernot.kubin@tugraz.at

***Abstract*— Wireless systems are subject to fading - time variations of the receiving conditions caused by multipath propagation and transceiver movements. Prediction of fading allows to 'learn' the channel state information (CSI) in advance and adjust the transmission scheme accordingly. In this contribution we consider a framework to handle predictions of general fast- and non-flat fading MIMO wireless channels. The approach is based on modeling the dynamics of individual multipath components, extracted with the SAGE algorithm. This decreases the rate of variation of the channel thus allowing a greater prediction horizon and simpler predictor designs. The extracted components are then tracked using dynamical programming coupled with the multipath component distance measure, and component parameters are then predicted over time using adaptive predictors – hypermodels. We consider linear as well as nonlinear predictor designs. This prediction scheme is applied to MIMO impulse response measurements in 2GHz frequency band, tracked over the distance of ≈ 4m, achieving prediction horizons of 1.5λ.**

***Keywords*—fading prediction, channel parameter estimation, multipath component distance, dynamic programming**

I. Introduction

Wireless systems are subject to fading - time variations of the receiving conditions caused by multipath propagation and transceiver movements. The received power undergoes deep fades within a time-frame corresponding to one wave-length which, for a typical communication system, amounts to movements on the centimeter scale. For efficient transmission, the communication system has to be able to mitigate fading effects. Should the current channel state information (CSI) be known in advance, the transceiver could re-allocate internal resources in a better way or alter the transmission scheme in anticipation of the future conditions. This can be accomplished by predicting the CSI.

Fading mitigation by means of channel prediction has been studied and proved viable in a number of works [1]–[10]. These techniques were used to aid power control and resource allocation [3], [6], downlink diversity and adaptive modulation [1], [9]. The fading is often assumed to be a deterministic process with time-varying parameters. This process can be modeled using linear [2], [7] or nonlinear [3], [10] methods. Predictions are then made by propagating the learned dynamics into the future. These methods were studied for Single Input Single Output (SISO) narrowband [2], [4], [9], as well as wideband channels [8]. In [6] it has been recently proposed to combine different channels in a smart antenna system for prediction of the downlink received power, however, the authors only consider the narrowband case. In this contribution we discuss our latest results on predicting the Multiple Input Multiple Output (MIMO) channels that undergo fast non-flat fading.

Compared to the SISO case, multiple antenna systems deliver larger amounts of information about the propagation environment. Not exploiting this information might be too wasteful. Since fading results from the interaction between different multipath components, it can be advantageous to extract them from the channel using estimation techniques and treat each component as an individual channel. For example, in the Single Input Multiple Output (SIMO) case, each multipath can be described by a multipath complex gain a_l, delay τ_l, Doppler shift ν_l, and Angle-of-Arrival (AoA) ϕ_l[1]. The advantage of such decomposition is obvious: each individual channel will have a larger coherence time, coherence bandwidth, and coherence distance, and thus it is more immune to fading. Such approach can be generalized straightforwardly to the wideband (as well as narrowband) MIMO, SISO, and MISO systems.

In practical channels not only power (i.e., the multipath gain) but also the other multipath parameters vary with time. As we will see, this temporal variation obliges us to tackle several questions:

- How can the multipath components be estimated?
- How to keep the multipath structure up-to-date?
- How to make predictions?

In the following sections we will subsequently consider these questions. We will illustrate the principles of the algorithm based on the SIMO channels, since the presented approach can be easily extended to other channel configurations.

Throughout the text we will demonstrate the performance of the proposed algorithms on the measured MIMO channel data obtained by Forschungszentrum Telekommunikation Wien (FTW) in Vienna, Austria, under the supervision of Helmut Hofstetter [11]. The measurements were done with the MIMO capable wideband vector channel sounder RUSK-ATM, manufactured by MEDAV [12]. The sounder was specifically adapted to operate at the center frequency of 2GHz. The transmitted signal was generated in the frequency domain to yield a predefined spectrum over 120MHz bandwidth with an approximately constant envelope over time. Two simultaneously multiplexed antenna arrays have been used at the transmitter and receiver. The transmitter was a uniform circular array with 15 sensors spaced at ≈ 6.45cm. The receiver was a fixed uniform linear array, with 8 sensors spaced half a wavelength apart, $\lambda/2 \approx 7.5$cm. The measurements were performed outdoors, with the transmitter array mounted on the roof of a building and the receiver moving with a velocity of $\approx$ 1m/s. A MIMO channel snapshot was recorded every 20msec, thus resulting in a spatial resolution of $\approx \lambda/7$. For our purposes we will further select only a SIMO subset by taking a single transmitting antenna from the TX array.

The ensuing sections are organized as follows: In Section II we introduce the multipath channel model; Section III introduces and explains the main steps in predictor design. Finally, Section IV shows some application results for real measured channels.

[1]For simplicity, we do not account for the wave polarization and elevation angles.

II. Channel model and parameter estimation

Let us assume that the receiver (Rx) is equipped with an antenna array consisting of P sensors located at $\boldsymbol{r}_0, \ldots, \boldsymbol{r}_{P-1} \in \mathbb{R}^2$ with respect to an arbitrary reference point. The corresponding equivalent baseband SIMO channel can then be modeled as [13]:

$$\boldsymbol{h}(t,\tau) = \sum_{l=1}^{L} a_l(t)\boldsymbol{c}(\phi_l(t))\mathrm{e}^{j\nu_l(t)t}\delta(\tau - \tau_l(t)), \tag{1}$$

where $\boldsymbol{h}(t,\tau) \in \mathbb{C}^P$ is a vectorized representation of the time-varying SIMO channel impulse response (IR), $a_l(t)$ and $\tau_l(t)$ are the gain and delay of the lth multipath component at time t, respectively. The phase term $\mathrm{e}^{j\nu_l(t)t}$ accounts for the phase-shift induced by the corresponding time-varying Doppler frequency $\nu_l(t)$. The P-dimensional vector $\boldsymbol{c}(\phi_l(t))$ is known as the steering vector of the array and, provided the coupling between the antenna sensors can be neglected, it is represented as $\boldsymbol{c}(\phi_l(t)) = [c_0(\phi_l(t)), \ldots, c_{P-1}(\phi_l(t))]^T$, where $\phi_l(t)$ is the Angle-of-Arrival of the l-th multipath. The components of $\boldsymbol{c}(\phi_l(t))$ are given as

$$c_p(\phi_l(t)) = f_p(\phi_l(t)) \exp(j2\pi\lambda^{-1}\langle \boldsymbol{e}(\phi_l(t)), \boldsymbol{r}_p\rangle)$$

with λ, $\boldsymbol{e}(\phi_l(t))$, and $f_p(\phi_l(t))$ denoting the wavelength, the unit vector in $\mathbb{R}^2$ pointing in the direction $\phi_l(t)$, and the complex electric field pattern of the pth sensor, respectively.

The channel model (1) is a key to estimating multipath parameters since it allows to invoke model-based parameter estimation methods. This model also assumes explicitly the plane wave propagation. Spherical waves, however, can be approximated by (1), though for a short period of time. Thus, the wave sphericity will manifest itself by more pronounced multipath parameter variations.

We also stress that the multipath channel $\boldsymbol{h}(t,\tau)$ is a time-varying process. To capture it the channel measurements are repeated with the period T_r, producing consecutive channel snapshots $\boldsymbol{h}(nT_r,\tau)$ at $t = nT_r$. The channel acquisition period T_r automatically defines the sampling period for temporal variations of the multipath parameters. As such, it must be chosen so as to make sure that parameter variations are represented appropriately by the corresponding temporal samples. A sample power-delay profile of the measured wireless SIMO channel is shown in Fig. 1

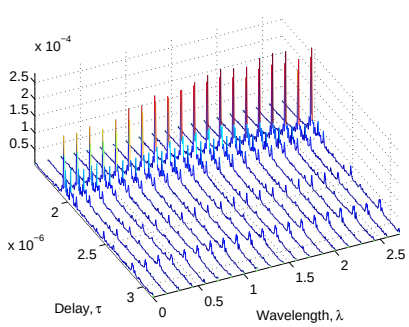

Fig. 1. Power-Delay profile of the measured SIMO channel.

A. Parameter estimation

Estimation of the multipath parameters $\boldsymbol{\theta}_l = \{a_l, \tau_l, \nu_l, \phi_l\}$ from the measurement data has been extensively studied [14]–[16]. Lately, a deterministic parameter estimation technique based on the SAGE algorithm [16] has become very popular. SAGE is an iterative estimation scheme that approximates the Maximum Likelihood (ML) solution for the model parameters. It replaces a resulting multi-dimensional optimization with a sequence of one-dimensional sub-optimizations, thus finding a solution more efficiently.

Unfortunately, the iterative nature of the SAGE algorithm requires a good initialization to avoid landing in a local maximum of the likelihood. Due to space limitations, the basic steps of the SAGE algorithm are not presented here. The interested reader will find a detailed algorithm description in [16].

Another aspect arising when using the SAGE algorithm is the number L of multipath components. Generally, for real measurements, the number of multipath components may vary with time. This presents a substantial difficulty not only for parameter estimation algorithms, but also, as we will see in Section III, for parameter tracking. The number of multipath components can be estimated 'on-line', if desired, using information-theoretic criteria [17]. Accommodation of this situation goes beyond the scope of the presented work. To account for it we will simply fix L to a number large enough to capture the most dominant waves.

Since the multipath channel generally varies with time, it is necessary to estimate multipath components for each new channel observation. This results in a temporal sequence of multipath estimates $\{\boldsymbol{\theta}_l[n]\}_{l=1}^{L}$. Unfortunately, parameter estimation algorithms do not provide any ordering information on how to associate the multipath components at time $n-1$ with those at time n. In order to be able to learn the parameter dynamics for prediction purposes, it is necessary to reconstruct this ordering, i.e., to associate the consecutive parameter estimates $\boldsymbol{\theta}_l[n]$ that correspond to the same physical multipath component over time.

The solution we propose is outlined in Section III.

III. Channel prediction

In general, parameter tracking/association is not a trivial problem since there is no *a priori* model that can be used to ease this task. However, this model can be constructed iteratively, as the algorithm proceeds. In fact, the sought predictor is an appropriate model of the track dynamics, and the predictor can be trained using the multipath track. We resolve this interdependency in the spirit of the classical sequential Bayesian estimation (see, for example, [18]).

Let us now assume that we want to reconstruct K tracks from the multipath estimates $\{\boldsymbol{\theta}_l[n]\}_{l=1}^{L}$, so that $K \leq L$. The block-diagram of the proposed sequential tracking and prediction scheme is depicted in Fig. 2.

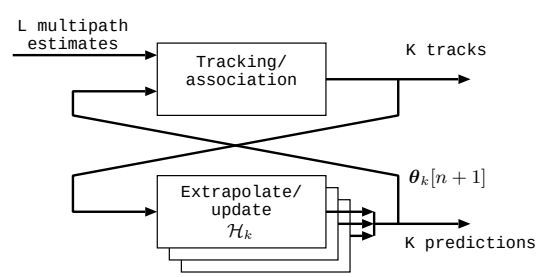

Fig. 2. Iterative tracking and adaptation of the track hypermodels.

The dynamics of each track is captured by a certain deterministic *hypermodel*, i.e., predictor, $\boldsymbol{H}_k(\cdot)$, in a sense that

$$\hat{\boldsymbol{\theta}}_k[n+1] = \boldsymbol{H}_k(\boldsymbol{\theta}_k[n], \boldsymbol{\theta}_k[n-1], \ldots). \tag{2}$$

The term hypermodel is used to stress that $\boldsymbol{H}_k(\cdot)$ encodes the dynamics of the 'underlying' channel component. Expression (2) is equivalent to the *prediction step* of Bayesian sequential estimation.

Once the prediction is obtained, we can define a distance measure $d(\cdot,\cdot)$ between the predictions $\hat{\boldsymbol{\theta}}_k[n+1]$ and newly obtained estimates $\{\boldsymbol{\theta}_l[n+1]\}_{l=1}^{L}$. The associations are then made so as to minimize the resulting distance between the predictions and the estimates. The obtained associations are then used to recursively update the hypermodels. This constitutes the *update step* of the sequential estimation. Note that the proposed scheme is not strictly Bayesian, but still implements a similar reasoning.

Now, let us consider how the association and hypermodel learning can be implemented.

A. Track association algorithm

For tracking purposes not all of the multipath parameters contribute equally to deciding among several alternative track continuations. Moreover, only a subset of the available parameters uniquely identifies a multipath component. In the simple SIMO case with a linear antenna array, these are a multipath delay τ_l, Doppler shift ν_l, and AoA ϕ_l. Although the multipath gain a_l is also estimated, it does not help to distinguish two different components.

Let $\boldsymbol{s}_l[n] \subseteq \boldsymbol{\theta}_l[n]$, $l = 1\ldots L$ denote a subset of estimated parameters used for tracking at the time n. In our case $\boldsymbol{s}_l[n] = [\tau_l[n], \nu_l[n], \phi_l[n]]^T$. Similarly, $\hat{\boldsymbol{s}}_k[n] \subseteq \hat{\boldsymbol{\theta}}_k[n], k = 1\ldots K$ is a subset of the predicted multipath parameters obtained as $\hat{\boldsymbol{\theta}}_k[n] = \boldsymbol{H}_k(\boldsymbol{\theta}_k[n-1], \boldsymbol{\theta}_k[n-2], \ldots)$. Then the track association can be cast as a standard dynamic programming problem.

Consider the graph in Fig. 3. Edges on the graph represent

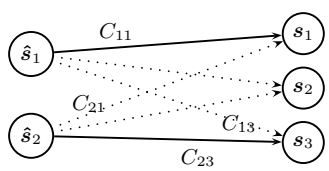

Fig. 3. Possible track continuations for $K = 2$ and $L = 3$

possible track continuations. Each connection between $\hat{\boldsymbol{s}}_k[n]$ and $\boldsymbol{s}_l[n]$ induces a cost

$$C_{kl}[n] = d(\hat{\boldsymbol{s}}_k[n], \boldsymbol{s}_l[n]) + \mu C_k[n-1]. \tag{3}$$

Here $C_k[n-1]$ is the cost accumulated by the k-th track at time $n-1$, and $0 \leq \mu \leq 1$ is a forgetting factor. Now, let us define a binary variable x_{kl} such that:

$$x_{kl} = \begin{cases} 1, & \text{if } \boldsymbol{s}_l[n] \text{ should be assigned to } \hat{\boldsymbol{s}}_k[n] \\ 0, & \text{otherwise} \end{cases}$$

Then, the optimal track association at time n should minimize the total induced cost Z:

$$\begin{aligned} &\underset{x_{kl}}{\operatorname{argmin}}\, Z = \sum_{k=1}^{K}\sum_{l=1}^{L} C_{kl}[n]x_{kl}, \quad \text{so that} \\ &\sum_{l=1}^{L} x_{kl} = 1,\ k = 1\ldots K, \quad \text{and} \quad x_{kl} \in \{0,1\}. \end{aligned} \tag{4}$$

Should we have just a single track, i.e., $K = 1$, the optimal solution could be found using the Viterbi algorithm. Solution to problem (4) is an optimal association for all K tracks simultaneously. It can be solved using standard linear programming techniques (see, for example, [19]).

The distance measure $d(\cdot,\cdot)$ can be effectively implemented using the multipath component distance (MCD) [20]. Basically, MCD is a weighted Euclidian distance measure, tailored to reflect the specifics of the multipath component parameters and it is particularly useful in our application.

B. Hypermodel learning and parameter prediction

Since the whole tracking/prediction approach is Bayesian-inspired, we employ Bayesian sequential methods for learning track hypermodels $\mathcal{H}_k$ as well. As we previously mentioned, a parameter subset $\boldsymbol{s}_k[n]$ is used in multipath tracking, and it requires a one-step-ahead predictor (2) to compute the cost (3). This prediction can be accomplished by a dedicated structure hypermodel $\mathcal{S}_k$. The ultimate application might however require long term forecasts of some of the multipath parameters, not necessarily used in tracking. These will be denoted as $\boldsymbol{\alpha}_k[n] \subseteq \boldsymbol{\theta}_k[n]$. As the result, we might need other hypermodels, denoted as $\mathcal{A}_k$, used specifically for this purpose.

1) Structure hypermodels $\mathcal{S}_k$: Since the hypermodels $\mathcal{S}_k$ are needed only for a one-step-ahead prediction, we can approximate the trajectory of the track $\boldsymbol{s}[n]_k$ by a damped local linear trend (DLLT) [21] and use the Kalman filter framework to learn the trend. For a single track, the state-space representation of this filter is given as:

$$\begin{cases} \begin{bmatrix} \hat{\boldsymbol{s}}_k[n+1] \\ \boldsymbol{v}_k[n+1] \end{bmatrix} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{I} \\ \boldsymbol{0} & \boldsymbol{\Delta}_k \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{s}}_k[n] \\ \boldsymbol{v}_k[n] \end{bmatrix} + \boldsymbol{\xi}_k[n] \\ \boldsymbol{s}_k[n] = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{s}}_k[n] \\ \boldsymbol{v}_k[n] \end{bmatrix} + \boldsymbol{\epsilon}_k[n], \end{cases} \tag{5}$$

where $\boldsymbol{I}$ is an identity matrix of the appropriate size, $\boldsymbol{v}_k[n]$ is a vector of estimated DLLT slopes, and $\boldsymbol{\Delta} = \operatorname{diag}([\delta_\tau, \delta_\nu, \delta_\phi])$ are fixed damping factors for each of the multipath parameters. The damping factors are chosen such that $0 \leq \delta_\tau, \delta_\nu, \delta_\phi \leq 1$. Practically, we select $\boldsymbol{\Delta}_k = 0.1\boldsymbol{I}$. Although for tracking we need only $\mathcal{L} = 1$ prediction, higher prediction horizons can be realized by recursive application of the transition equation (5) exactly $\mathcal{L}$ times.

The variance of the disturbance terms $\boldsymbol{\epsilon}_k[n]$ and $\boldsymbol{\xi}_k[n]$ in (5) are important design parameters. Since the multipath parameters cannot be estimated with zero variance, the observation noise $\boldsymbol{\epsilon}_k[n]$ can be related to residual estimation uncertainty of the SAGE algorithm. Due to the unbiasedness and consistency of the SAGE-obtained estimates [16], the disturbance $\boldsymbol{\epsilon}_k[n]$ can be treated as a white Gaussian estimation noise. State noise $\boldsymbol{\xi}_k[n]$, on the other hand, is left as a free design parameter. Practically, we choose it so as to make sure that the ratio between the variance of the state noise and that of the observation noise is ≈ 0.01.

The initialization of the hypermodels $\mathcal{S}_k$ is chosen so as to repeat the last seen value. This can be achieved by selecting $\boldsymbol{v}_k[0] = 0$, and setting $\boldsymbol{s}_k[0]$ to the estimated multipath parameters at $n = 0$. Assuming smooth parameter variations, the hypermodel predictions will not wander too far from the true future values. Such initialization is more likely to result in correct associations, and thus the proper values are going to be used to update the predictor coefficients.

2) Hypermodel $\mathcal{A}_k$: Multipath power prediction is often a desired output of the channel forecasting. In power prediction we are mostly interested in accurately modeling the evolution of the multipath gains and extrapolating it beyond the observation interval. Thus, $\boldsymbol{\alpha}_k[n] \equiv a_k[n]$. If we need long-term predictions, more complicated predictors are required. Again, we stick to the Bayesian sequential methods and utilize the predictor in the following state-space form:

$$\begin{cases} \begin{bmatrix} \hat{\boldsymbol{a}}_k[n+1] \\ \boldsymbol{w}_k[n+1] \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}(\hat{\boldsymbol{a}}_k[n], \boldsymbol{w}_k[n]) \\ \boldsymbol{I}\boldsymbol{w}[n] \end{bmatrix} + \boldsymbol{\eta}_k[n]. \\ a_k[n] = \begin{bmatrix} 1 & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{a}}_k[n] \\ \boldsymbol{w}_k[n] \end{bmatrix} + \varsigma_k[n], \end{cases} \tag{6}$$

where $\boldsymbol{a}_k[n] = [a_k[n], ..., a_k[n-Q+1]]^T \in \mathbb{C}^Q$ is a vector of delayed multipath gain samples, and $\boldsymbol{f}(\cdot,\cdot)$ is the state predictor, parametrized by the coefficients $\boldsymbol{w}_k[n] \in \mathbb{C}^Q$. The role of the disturbance terms $\varsigma_k[n]$ and $\boldsymbol{\eta}_k[n]$ in (6) is basically the same as of $\boldsymbol{\xi}_k[n]$ and $\boldsymbol{\epsilon}_k[n]$ in (5).

It can be seen that the formulation (6) forms a bilinear state-space representation, since both the states and the predictor coefficients are estimated. This necessitates application of the nonlinear versions of the Kalman filter. In some cases, it is possible to apply the extended Kalman filter (EKF) to jointly estimate $\hat{\boldsymbol{a}}_k[n]$ and $\boldsymbol{w}_k[n]$, which in the literature is known as the joint EKF [22, sec. 5]. In this work we will consider two realizations of $\boldsymbol{f}(\cdot,\cdot)$ that allow application of the EKF: 1) linear, when $\hat{a}_k[n+1] = \boldsymbol{w}_k^T[n]\hat{\boldsymbol{a}}_k[n]$, and 2) nonlinear, when $\boldsymbol{f}(\cdot,\cdot)$ is represented with a feedforward neural network.

To be able to obtain long-term predictions, i.e., $\mathcal{L} > 1$, from (6) we iterate the transition equation $\mathcal{L}$ times[2].

Initialization of the hypermodels $\mathcal{A}_k$ can be done similarly to the initialization of $\mathcal{S}_k$, making sure that initial hypermodels simply repeat the last seen value. However, since the models $\mathcal{A}_k$ do not take part in tracking, they might also be initialized randomly, by drawing model coefficients from a certain distribution. The initial covariance matrix should then be chosen to be quite large, to minimize the influence of the initial hypermodel coefficient values.

IV. Tracking and Prediction Results for the Measured Channels

To demonstrate the tracking results, we estimate $L = 20$ multipath components and reconstruct $K = 4$ tracks. The reconstructed trajectories of multipath components for these tracks are shown in Fig. 4. By taking Fig. 1 as a reference, we can clearly detect multipath

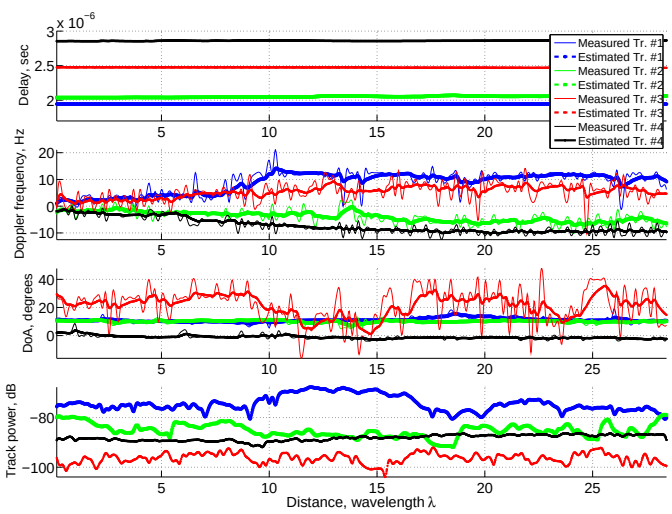

Fig. 4. Four reconstructed tracks.

components in the vicinity of the LOS (approx. 2μsec), and then a couple of distant reflections at 2.5μsec and 2.8μsec that are clearly identified as seen in Fig. 4. The strongest estimated components in these locations were used to initialize the tracking algorithm. In Fig. 4 we also plot the outputs of the track hypermodels. Recall that the latter acts as a predictor able to forecast component structure into the future. For tracking purposes we consider only a short-term prediction, but hypermodels can surely be utilized for making long-term forecasts by extrapolating the hypermodels into the future. In Fig. 4 we tracked the components over the distance of $\approx 28\lambda$. Taking into account that $\lambda \approx 0.15$m, we easily compute that the total walked distance is 4.2m, or equivalently, 4.2sec of tracking time.

To predict multipath gain, we utilize a linear predictor with $Q = 3$ coefficients, and a Neural Network with with 7 input neurons, and one hidden layer with 7 neurons with sigmoidal activation functions. In the case of the multipath gain, we can analyze the prediction quality off-line, once the tracks have been reconstructed. For this purpose we introduce the notion of the *naive predictor*. The naive predictor utilizes (6) but assumes that $\hat{a}_k[n+\mathcal{L}] = \hat{a}_k[n]$. In some sense this is a zero-order predictor.

An example of a gain prediction for $\mathcal{L} = 1$ and the linear predictor is shown in Fig. 5.

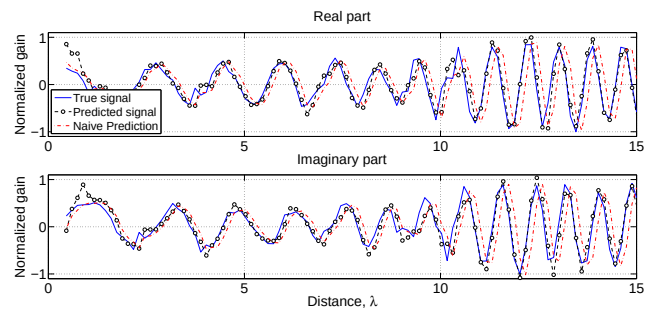

Fig. 5. Linear gain prediction for the Track #1. $\mathcal{L} = 1$.

Qualitatively, the prediction performance for $\mathcal{L} \geq 1$ can be assessed using the corresponding prediction error $e[n]$. Using $e[n]$, we can define the prediction gain (PG) G_{pred} as

$$G_{\text{pred}} = 10\log_{10}\left(\frac{\sigma^2_{\text{true}}}{\sigma^2_e}\right), \tag{7}$$

where σ^2_{true} is the variance of the true signal that we are predicting, and σ^2_e is the variance of the prediction error. We see that the prediction gain is an analogue of the classical signal-to-noise ratio, modified to reflect the prediction quality. In our case, however, the application of eq. (7) is not fully justified, since both $e[n]$ and the signal we predict might exhibit short-term transient behavior, i.e., generally they are nonstationary. A possible way to alleviate this is to consider a segmental prediction gain – an equivalent of the segmental SNR, often used in speech coding applications [23].

The basic idea behind the segmental PG is quite simple: the data sequence is sectioned into relatively small chunks of size $\approx 2\lambda$, over which signal stationarity can be assumed. For each chunk i the individual PG$_i$ is computed according to (7). The final Segmental PG is then found as an average over all the partial PG$_i$'s over the whole data sequence.

The evaluated PG as a function of the prediction horizon $\mathcal{L}$ over the distance of 4.2m is shown in Fig. 6. For both predictors we plot the best, as well as the averaged (over the 4 tracks) prediction performance.

The obtained plot shows that short-term prediction of the multipath gain variation is quite feasible. If we take a 3dB line as a minimum useful prediction quality, we see that we can forecast as far as 1.5λ into the future for the strongest track, which is the Track #1. Clearly, this track is the strongest one, as seen in Fig. 4, and thus, it is easier to build a proper track, and from that the correct hypermodel. The SNR for other tracks is lower and the corresponding prediction horizons

[2]It is of course possible to devise a learning algorithm for a fixed prediction horizon $\mathcal{L} > 1$, but this development goes beyond the scope of this paper

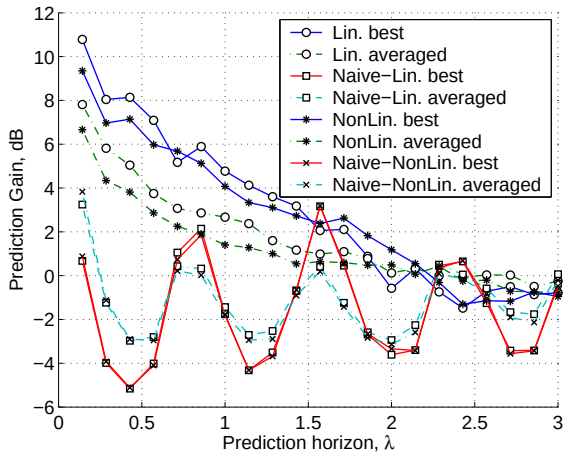

Fig. 6. Prediction gain for different prediction horizons.

shorter. To compare the achieved prediction quality, we refer to [6]. In this work the authors report the achievable prediction horizons for the measured data only to be $\approx \lambda/60$. However, we see that at least for some strong tracks we significantly outperform these results. It should also be stressed that in our case the spatial sampling is much lower, i.e., only 7 samples per wavelength, which adds to the efficiency of our approach. In our experiments we tried different orders and structures for both linear and nonlinear predictors. We noticed that linear hypermodels are generally better than the nonlinear ones. Linear structures are easier to update and they require less data to reliably estimate coefficients, as compared to the nonlinear models.

V. Conclusions

In the presented work we have considered a general framework for predicting wireless MIMO channels. The framework consists in several basic steps that include: 1) extraction of the multipath components from the measured channel (which is accomplished using a well-known SAGE algorithm), 2) tracking the estimated components over time, and, 3) learning and updating the dynamics of the multipath hypermodels. The latter two steps are solved jointly in the spirit of the Bayesian sequential estimation, by coupling the adaptive hypermodel construction with the dynamical programming to associate the estimated multipath components over time. The main emphasis of this work was put on steps 2) and 3). The tracking of components was solved using multipath component distance coupled with dynamical programming to find optimal associations. The latter relies on smooth parameter variation, which in turn can be ensured by increasing spatial sampling of the data. Since the multipath components might appear and disappear due to multiple factors, such as changing environment, the future development of the method must also consider track management strategies to remove weak tracks and search measured data for new ones.

The proper parameter association allows to adaptively construct track hypermodels. We introduced two types of predictors: a structure hypermodel that predicts the 'position' of the multipath component, needed for the tracking algorithm, and a gain hypermodel that we used to forecast multipath gain for possible power-controlled applications in the mobile. While the former is a simple linear trend predictor, the gain hypermodel is more complex. We studied linear as well as nonlinear structures. It was found that linear are better, since they require less data to adapt, and thus are less sensitive to transients and tracking errors, unlike the nonlinear structures. However, the iterative nature of the linear predictor requires stability check for $\mathcal{L} \gg 1$ The obtained simulation results prove the potential of the multipath-oriented framework to predict the dynamics of the multipath components, and thus of the whole channel, into the future. With a quite simple adaptive linear predictor we were able to forecast as long as 1.5λ at 3dB PG for the strongest track. However, we conjecture that efficient track management can bring this limit even higher.

References

[1] H. Duel-Hallen, A. Shengquan Hu Hallen, "Long-range prediction of fading signals," *IEEE Signal Processing Magazine*, vol. 17, no. 3, pp. 62 – 75, May 2000.

[2] J.K. Hwang and J.H. Winters, "Sinusoidal modeling and prediction of fast fading processes.," in *Global Telecom. Conf., GLOBECOM'98*. IEEE, 1998, vol. 2, pp. 892–896.

[3] T. Ekman, *Prediction of Mobile Radio Channels: Modeling and Design*, Ph.D. thesis, Signal Processing at Uppsala University, Nov. 2002.

[4] Eyceoz T., Duel-Hallen A., and Hallen H., "Deterministic channel modeling and long range prediction of fast fading mobile radio channels," *IEEE Communications Letters*, vol. 2, no. 9, pp. 254–256, Sep. 1998.

[5] Andersen J.B., Jensen J., Jensen S.H., and Frederiksen F., "Prediction of future fading based on past measurements," in *50th IEEE Conf. on Vehic. Tech.,VTC'99*, 1999, vol. 1, pp. 151 – 155.

[6] K.R. Guanghan Xu Arredondo, A. Dandekar, "Vector channel modeling and prediction for the improvement of downlink received power," *IEEE Trans. on Comm.*, vol. 50, no. 7, pp. 1121– 1129, Jul 2002.

[7] R. Vaughan, P. Teal, and R. Raich, "Short-term mobile channel prediction using discrete scatterer propagation model and subspace signal processing algorithms," in *52nd IEEE Conf. on Vehic. Tech.*, 2000, vol. 2, pp. 751 – 758.

[8] Liang Dong, Guanghan Xu, and Hao Ling, "Prediction of fast fading mobile radio channels in wideband communication systems," in *IEEE Global Telec. Conf., GLOBECOM '01.*, Nov 2001, vol. 6, pp. 3287 – 3291.

[9] S. Hu, H. Hallen, and A. Duel-Hallen, "Physical channel modeling, adaptive prediction and transmitterdiversity for flat fading mobile channel," in *IEEE Workshop on Signal Proc. Advances in Wireless Comm. SPAWC*, 1999, pp. 387–390.

[10] G. Ekman, T. Kubin, "Nonlinear prediction of mobile radio channels: measurements and MARS model designs," in *Proceedings of the IEEE Int. Conf. on Acoust., Speech, and Signal Proc.*, 1999, vol. 5, pp. 2667–2670.

[11] E. Bonek et al., "Double-directional superresolution radio channel measurements," *Proc. Conf. on Comm., Control, and Computing*, vol. 3, Oct. 2001.

[12] R. Thoma et al., "Identification of time-variant directional mobile radio channels," *IEEE Trans. on Instrumen. and Meas.*, vol. 49:2, pp. 357–364, Apr. 2000.

[13] Andreas F. Molisch, *Wireless Communcations*, IEEE Press, 2005.

[14] H. Krim and M. Viberg, "Two decades of array signal processing research," *IEEE Signal Processing Mag.*, p. 6794, July 1996.

[15] Harry L. Van Trees, *Optimum Array Processing*, Wiley, 2002.

[16] B.H. Fleury, M. Tschudin, R. Heddergott, D. Dahlhaus, and K. Ingeman Pedersen, "Channel parameter estimation in mobile radio environments using the SAGE algorithm," *IEEE Journal on Selected Areas in Comm.*, vol. 17, no. 3, pp. 434–450, March 1999.

[17] T. Wax, M. Kailath, "Detection of signals by information theoretic criteria," *IEEE Trans. on Acoustics, Speech, and Signal Proc.*, vol. 33, no. 2, pp. 387– 392, April 1985.

[18] Todd K. Moon and Wynn C. Stirling, *Mathematical Methods and Algorithms for Signal Processing*, Prentice-Hall, 2000.

[19] H.A. Taha, *Operations Research*, Prentice Hall International, 2002.

[20] M. Steinbauer, H. Özcelik, H. Hofstetter, C.F. Mecklenbräuker, and E. Bonek, "How to quantify multipath separation," *IEICE Transactions on Electronics, Special Issue on Signals,Systems and Electronics Technology*, pp. 552–557, March 2002.

[21] Andrew C. Harvey, *Forecasting, structural time series models and the Kalman filter.*, Cambridge University Press., 1989.

[22] Haykin S., Ed., *Kalman Filtering and Neural Networks*, John Wiley & Sons, Inc., 2001.

[23] Douglas O'Shaughnessy, *Speech Communication, Human and Machine*, IEEE Press, 2000.