Complexity Reduction of Iterative Receivers Using Low-Rank Equalization

Guido Dietl, *Student Member, IEEE*, and Wolfgang Utschick, *Senior Member, IEEE*

***Abstract*—This paper covers the consideration of an *iterative* or *turbo receiver* where the nonlinear trellis-based detection of the interleaved and coded data bits is replaced by linear detection using the *Wiener filter* (WF), i.e., the optimal linear filter based on the *mean-square error* (MSE) criterion. The equalization of channels with multiple antennas at the receiver as well as frequency-selective transfer functions requires high-dimensional observation vectors which involve computationally intense detectors. We extend an optimal but computationally efficient algorithm, originally derived for single receive antenna systems, to *single-input multiple-output* (SIMO) channels. To further reduce computational complexity, we apply the suboptimal low-rank *multistage WF* (MSWF), i.e., the WF approximation in the low-dimensional Krylov subspace, and replace additionally second-order statistics of nonstationary random processes by their time-invariant averages. Complexity investigations reveal the enormous capability of the proposed algorithms to decrease computational effort. Moreover, the analysis based on *extrinsic information transfer* (EXIT) charts as well as *Monte Carlo* simulations show that compared with reduced-rank detection methods based on eigensubspaces, the reduced-rank MSWF behaves near optimum although the rank is drastically reduced to two or even one.**

***Index Terms*—Decoding, equalizers, intersymbol interference (ISI), iterative methods, multistage Wiener filter.**

I. Introduction

COMPENSATING *intersymbol interference* (ISI) caused by channels with frequency-selective transfer functions is a fundamental task of practical communication systems. Unfortunately, the optimal receiver which meets this requirement by performing jointly symbol detection and decoding is computationally not feasible. Thus, we investigate a receiver structure performing symbol detection and decoding alternating in an iterative process. This so-called *turbo equalizer* was introduced by Douillard *et al.* [1] and consists of an optimal *maximum a posteriori* (MAP) detector and a MAP decoder exchanging iteratively soft information about the coded data bits. Simulation results have shown that this procedure eliminates ISI after several iteration steps such that the *bit error rate* (BER) of coded transmission over the corresponding *additive white Gaussian noise* (AWGN) channel can be achieved. Note that the proposed turbo system can be interpreted as an iterative decoding scheme for serial concatenated codes [2] where the inner code is the channel and the inner decoder is the detector. Although the computational complexity of the iterative receiver is tremendously smaller than the one of the optimal receiver, the complexity is still very high for practical implementations. Therefore, Glavieux *et al.* [3] approximated the optimal nonlinear detector using the *Wiener filter* (WF), i.e., the linear filter with the *minimum mean-square error* (MMSE). Later, Wang and Poor [4] exploited this idea to reduce the complexity in a coded multiuser CDMA system. Since then, several methods have been proposed for further decreasing computational complexity such as *channel-shortening filters* [5], equalizer design based on the *fast Fourier transformation* [6], or *sequential Monte Carlo sampling techniques* [7].

This paper presents an alternative complexity reduction approach based on [4] which we apply to a single-user uplink scenario for mobile communications over a frequency-selective channel with multiple antennas at the receiver, i.e., a time-dispersive *single-input multiple-output* (SIMO) channel. In a first step, we reduce the high computational effort for the calculation of the WF weights [4] by extending the reduced-complexity algorithm introduced by Tüchler *et al.* [8] to the proposed SIMO channel. This method exploits the time dependency of the autocovariance matrix of the observation vector in order to decrease the computational complexity of its inversion by one order but is still optimal in the MMSE sense, i.e., it achieves the performance of the turbo equalizer proposed by [4]. Compared with adaptive *stochastic-gradient*-type algorithms [9] and methods based on *maximum-likelihood* estimation of statistics [10], all algorithms proposed in this paper compute statistics using a channel model. Although we assume perfect channel state information at the receiver, the presented methods can be applied straightforwardly to systems where the channel is estimated.

To further reduce computational complexity, we introduce suboptimal solutions using reduced-rank equalization. The *multistage WF* (MSWF) developed by Goldstein *et al.* [11] is a computationally cheap approximation of the WF. It has been shown by Honig *et al.* [12], [13] that the application of the MSWF is equivalent to Wiener filtering in the *Krylov subspace* of the autocovariance matrix of the observation vector and the cross-covariance vector between the observation and the desired signal. Therefore, the iterative *Lanczos algorithm* [14], [15], [13] can be used to compute efficiently the reduced-rank filter weights which converge very fast to the optimal solution.

Moreover, the approximation of second-order statistics of nonstationary random processes by their time-invariant averages yields suboptimal implementations with dramatically reduced computational effort although the performance decrease is tolerable. This idea was developed for full-rank WFs [8] but can also be applied to the reduced-rank MSWF. A comparison of the *floating-point operations* (FLOPs) needed to

Manuscript received February 28, 2005; revised May 2, 2006. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Hongya Ge.

The authors are with the Associate Institute for Signal Processing, Department of Electrical Engineering and Information Technology, Munich University of Technology (TUM), 80333 Munich, Germany (e-mail: dietl@tum.de; utschick@tum.de).

Digital Object Identifier 10.1109/TSP.2006.885698

calculate the filter coefficients gives an impression for the capability of the proposed linear detectors to reduce computational complexity.

Finally, the analysis of *extrinsic information transfer* (EXIT) charts and *Monte Carlo* simulations show close to optimal behavior of the proposed solutions despite of their tremendously reduced computational complexity. Especially, the rank-one MSWF, which is an easy-to-implement normalized *matched filter* (MF) followed by a scalar WF, turns out to have only a slightly smaller performance than the optimal WF. Besides, a comparison of the MSWF to approaches based on the approximation of the WF in an eigensubspace of the autocovariance matrix of the observation, i.e., the *principal component* (PC) [16] and the *cross-spectral* (CS) [17] method, reveals the superiority of Krylov-subspace-based algorithms.

Notation

Throughout the paper, vectors and matrices are denoted by lower and upper case bold letters, respectively. Random variables are written using *sans serif* font and their realizations with the corresponding font with serifs. The matrix $\boldsymbol{I}_n$ is the $n \times n$ identity matrix, $\boldsymbol{e}_\nu$ its νth column,[1] $\mathbf{0}_{m\times n}$ the $m \times n$ zero matrix, and $\mathbf{0}_n$ the n-dimensional zero vector. The operation "$\otimes$" denotes the Kronecker product, $\mathrm{E}\{\cdot\}$ expectation, $(\cdot)^*$ conjugate complex, $(\cdot)^{\mathrm{T}}$ transpose, $(\cdot)^{\mathrm{H}}$ Hermitian, i.e., conjugate transpose, $\|\cdot\|_2$ the Euclidean norm, and $\mathrm{O}(\cdot)$ the Landau symbol. The matrix $\boldsymbol{S}_{(\ell,M,N)} = [\mathbf{0}_{M\times\ell}, \boldsymbol{I}_M, \mathbf{0}_{M\times(N-\ell)}] \in \{0,1\}^{M\times(M+N)}$ is used to select M rows of a matrix beginning from the $(\ell+1)$th row by applying it to the matrix from the left-hand side. Analogous, if its transpose is applied to a matrix from the right-hand side, M columns are selected beginning from the $(\ell+1)$th column. Besides, the matrix $\boldsymbol{S}_{(\ell,M,N)}$ can be used to describe a convolutional matrix. The probability $\mathrm{P}(\mathsf{x} = x)$ denotes the likelihood that a realization of the random variable x is equal to x and $\mathrm{p}_{\mathsf{x}}(x)$ is the probability density function of x. The soft information of a binary random variable $\mathsf{d} \in \{0,1\}$ is represented by the *log-likelihood ratio* (LLR) [18] $l = \ln\left(\mathrm{P}\left(\mathsf{d}=0\right)/\mathrm{P}\left(\mathsf{d}=1\right)\right)$. The autocorrelation matrix of the complex-valued vector random process $\mathbf{u}[k]$ at time index k is $\boldsymbol{R}_{\mathbf{u}}[k] = \mathrm{E}\{\mathbf{u}[k]\mathbf{u}^{\mathrm{H}}[k]\}$. The power of the complex-valued scalar random process $\mathsf{v}[k]$ at time index k is the second-order moment $r_{\mathsf{v}}[k] = \mathrm{E}\{|\mathsf{v}[k]|^2\} \in \mathbb{R}_{0,+}$. The cross-correlation vector between $\mathbf{u}[k]$ and $\mathsf{v}[k]$ is $\boldsymbol{r}_{\mathbf{u},\mathsf{v}}[k] = \mathrm{E}\{\mathbf{u}[k]\mathsf{v}^*[k-\kappa]\}$ where κ denotes the latency time, e.g., introduced by an equalizer.[2] The mean of $\mathbf{u}[k]$ and $\mathsf{v}[k]$ is $\boldsymbol{m}_{\mathbf{u}}[k] = \mathrm{E}\{\mathbf{u}[k]\}$ and $m_{\mathsf{v}}[k] = \mathrm{E}\{\mathsf{v}[k]\}$, respectively. $\boldsymbol{C}_{\mathbf{u}}[k] = \boldsymbol{R}_{\mathbf{u}}[k] - \boldsymbol{m}_{\mathbf{u}}[k]\boldsymbol{m}_{\mathbf{u}}^{\mathrm{H}}[k]$ denotes the autocovariance matrix of $\mathbf{u}[k]$. The autocovariance or variance of $\mathsf{v}[k]$ computes as $c_{\mathsf{v}}[k] = \sigma_{\mathsf{v}}^2[k] = r_{\mathsf{v}}[k] - |m_{\mathsf{v}}[k]|^2 \in \mathbb{R}_{0,+}$, where $\sigma_{\mathsf{v}}[k]$ is the standard deviation of $\mathsf{v}[k]$. The cross-covariance vector between $\mathbf{u}[k]$ and $\mathsf{v}[k]$ is represented by $\boldsymbol{c}_{\mathbf{u},\mathsf{v}}[k] = \boldsymbol{r}_{\mathbf{u},\mathsf{v}}[k] - \boldsymbol{m}_{\mathbf{u}}[k]m_{\mathsf{v}}^*[k-\kappa]$. Omitting the argument "$[k]$" signalizes that the respective statistical moments are not time-variant, which is the case for stationary random processes.

[1]The dimension of the unit vector $\boldsymbol{e}_\nu$ is defined implicitly by the context.

[2]Note that the cross-correlation vector $\boldsymbol{r}_{\mathbf{u},\mathsf{v}}[k]$ is a function of the index k and the shift κ. Since we choose a fixed κ in the following sections, the dependence on κ is omitted in the given notation.

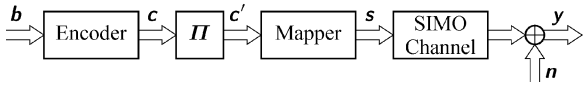

Fig. 1. Transmitter and channel model.

II. System Model

A. Transmitter and Channel Model

The model of the transmitter and SIMO channel is depicted in Fig. 1, based on the receiver's point of view, i.e., all signals are assumed to be random processes since they are obviously not known at the receiver.

The binary data block $\mathbf{b} \in \{0,1\}^B$, where B denotes the data block length, is encoded with code rate $r = B/(SQ)$. Then, the coded data block $\mathbf{c} \in \{0,1\}^{SQ}$ of length SQ is interleaved using the permutation matrix $\boldsymbol{\Pi} \in \{0,1\}^{SQ\times SQ}$, i.e., $\mathbf{c}' = \boldsymbol{\Pi}\mathbf{c} \in \{0,1\}^{SQ}$, and mapped to the complex symbol block $\mathbf{s} \in \mathbb{M}^S$ using the modulation alphabet $\mathbb{M}$ whose cardinality is 2^Q. Here, S denotes the symbol block length and Q is the number of bits used to represent one symbol. The mapper M can be described by the bijective function $M : \{0,1\}^Q \to \mathbb{M}$, $\mathbf{c}'_{k+1} \mapsto \mathsf{s}[k] = M(\mathbf{c}'_{k+1})$, mapping the Q bits in $\mathbf{c}'_{k+1} = \boldsymbol{S}_{(kQ,Q,(S-1)Q)}\mathbf{c}'$ to the symbol $\mathsf{s}[k] = \boldsymbol{e}_{k+1}^{\mathrm{T}}\mathbf{s}$, $k \in \{0,1,\ldots,S-1\}$. The mapper used throughout this paper is based on the *Gray code* (see, e.g., [19]), i.e., the bit codes corresponding to nearby symbols differ only in one binary digit. Afterwards, the symbol sequence $\mathsf{s}[k]$ is transmitted over the multipath SIMO channel of order L with the coefficients $\boldsymbol{h}_\ell \in \mathbb{C}^R$, $\ell \in \{0,1,\ldots,L\}$, received by R antennas and perturbed by stationary AWGN $\boldsymbol{\eta}[k] \in \mathbb{C}^R$ with the circular complex normal distribution $\mathcal{N}_{\mathbb{C}}\left(\mathbf{0}_R, \sigma_{\boldsymbol{\eta}}^2\boldsymbol{I}_R\right)$, i.e., $\boldsymbol{\eta}[k]$ is zero-mean, spatially white, and each of its elements has the variance $\sigma_{\boldsymbol{\eta}}^2$. The channel is assumed to be constant during one block, i.e., for the transmission of S symbols, but varies from block to block. The received signal vector $\mathbf{r}[k] \in \mathbb{C}^R$ can finally be written as

$$\mathbf{r}[k] = \sum_{\ell=0}^{L} \boldsymbol{h}_\ell \mathsf{s}[k-\ell] + \boldsymbol{\eta}[k]. \tag{1}$$

For the derivation of the optimal linear equalizer filter with K taps in Section III-A, we introduce the matrix-vector model

$$\mathbf{y}[k] = \boldsymbol{H}\mathbf{s}[k] + \mathbf{n}[k] \tag{2}$$

with the observation vector $\mathbf{y}[k] \in \mathbb{C}^N$, $N = RK$, the noise vector $\mathbf{n}[k] \in \mathbb{C}^N$, the symbol vector $\mathbf{s}[k] \in \mathbb{C}^{K+L}$, and the time-invariant channel convolutional matrix $\boldsymbol{H} \in \mathbb{C}^{N\times(K+L)}$ defined as follows:

$$\mathbf{y}[k] = \left[\mathbf{r}^{\mathrm{T}}[k], \mathbf{r}^{\mathrm{T}}[k-1], \ldots, \mathbf{r}^{\mathrm{T}}[k-K+1]\right]^{\mathrm{T}} \tag{3}$$

$$\mathbf{n}[k] = \left[\boldsymbol{\eta}^{\mathrm{T}}[k], \boldsymbol{\eta}^{\mathrm{T}}[k-1], \ldots, \boldsymbol{\eta}^{\mathrm{T}}[k-K+1]\right]^{\mathrm{T}} \tag{4}$$

$$\mathbf{s}[k] = \left[\mathsf{s}[k], \mathsf{s}[k-1], \ldots, \mathsf{s}[k-K-L+1]\right]^{\mathrm{T}} \tag{5}$$

$$\boldsymbol{H} = \sum_{\ell=0}^{L} \boldsymbol{S}_{(\ell,K,L)} \otimes \boldsymbol{h}_\ell. \tag{6}$$

The observation vector $\mathbf{y}[k]$ represents an N-dimensional subblock of the received signal block $\mathbf{y} \in \mathbb{C}^{R(S+L)}$ comprising all the received signal vectors $\mathbf{r}[k]$, $k \in \{0,1,\ldots,S+L-1\}$, of one block. Throughout the paper, we assume that the transmitted

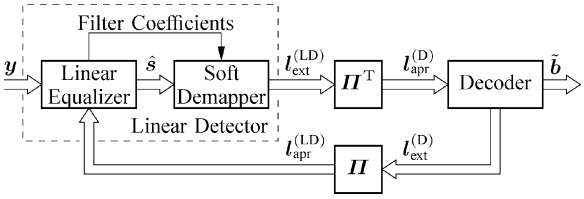

Fig. 2. Receiver structure.

symbols are uncorrelated to the noise, i.e., $\mathrm{E}\{\mathsf{s}[k]\mathsf{n}^{\mathrm{H}}[k]\} = \mathbf{0}_{(K+L)\times N}$, and that the noise is also temporally white, i.e., $\boldsymbol{R}_{\mathsf{n}}[k] = \sigma_{\boldsymbol{\eta}}^2 \boldsymbol{I}_N$. Note that $\mathsf{n}[k]$ is also zero-mean, i.e., $\boldsymbol{m}_{\mathsf{n}}[k] = \mathbf{0}_N$.

B. Receiver Structure

The aim of the receiver structure given in Fig. 2 is to process the observation signal block $\boldsymbol{y}$ such that its output $\tilde{\boldsymbol{b}}$ is best possible equal to the transmitted data block $\boldsymbol{b}$. Note that the observation vector block $\boldsymbol{y}$ is seen as a realization of the vector random variable $\mathbf{y}$ at the output of the transmitter and channel model in Fig. 1, as well as the data block $\boldsymbol{b}$ as a realization of $\mathbf{b}$ at the input of the model. Besides, we assume throughout the paper that the receiver has perfect channel state information. However, the proposed methods can be easily applied to systems where the channel is estimated.

The aim of best possible detection is approximately achieved by exchanging soft information about the coded data bits between the linear detector and the decoder in an iterative process. As will be discussed more detailed in Section III-B, the linear detector calculates the extrinsic information $\boldsymbol{l}_{\mathrm{ext}}^{(\mathrm{LD})} \in \mathbb{R}^{SQ}$ about the interleaved and coded data block $\mathbf{c}'$ using the observation signal block $\boldsymbol{y}$ and the *a priori* information $\boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{LD})} \in \mathbb{R}^{SQ}$ about $\mathbf{c}'$ obtained by interleaving of the extrinsic information $\boldsymbol{l}_{\mathrm{ext}}^{(\mathrm{D})} \in \mathbb{R}^{SQ}$ about the coded data block $\mathbf{c}$ computed by the decoder at the previous iteration step, i.e., $\boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{LD})} = \boldsymbol{\Pi}\boldsymbol{l}_{\mathrm{ext}}^{(\mathrm{D})}$.

Further, the MAP decoder computes the extrinsic information $\boldsymbol{l}_{\mathrm{ext}}^{(\mathrm{D})}$ about the coded data block $\mathbf{c}$ from its *a posteriori* LLR vector $\boldsymbol{l}_{\mathrm{apo}}^{(\mathrm{D})} \in \mathbb{R}^{SQ}$ using its *a priori* information $\boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{D})} \in \mathbb{R}^{SQ}$ which is the deinterleaved extrinsic information $\boldsymbol{l}_{\mathrm{ext}}^{(\mathrm{LD})}$ about the interleaved and coded data block $\mathbf{c}'$ at the output of the detector at the previous iteration step, i.e., $\boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{D})} = \boldsymbol{\Pi}^{\mathrm{T}}\boldsymbol{l}_{\mathrm{ext}}^{(\mathrm{LD})}$. Let $\boldsymbol{l}^{(\mathrm{D})} \in \mathbb{R}^{SQ}$ be the vector composed by the LLRs $l_i^{(\mathrm{D})} = \boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{l}^{(\mathrm{D})}$, $i \in \{1, 2, \ldots, SQ\}$, which can be either *a posteriori*, *a priori*, or extrinsic information. Clearly, these informations depend on the number of the current iteration. Nevertheless, an additional index is omitted due to simplicity. Then, the ith element of the *a posteriori* LLR vector $\boldsymbol{l}_{\mathrm{apo}}^{(\mathrm{D})}$ computes as [4]

$$l_{\mathrm{apo},i}^{(\mathrm{D})} = \ln \frac{\mathrm{P}\left(\mathsf{c}_i = 0 \,\middle|\, \mathbf{l}_{\mathrm{apr}}^{(\mathrm{D})} = \boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{D})}\right)}{\mathrm{P}\left(\mathsf{c}_i = 1 \,\middle|\, \mathbf{l}_{\mathrm{apr}}^{(\mathrm{D})} = \boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{D})}\right)} = \ln \frac{\sum\limits_{\substack{\boldsymbol{c}\in\{0,1\}^{SQ} \\ c_i=0}} \mathrm{p}_{\mathbf{l}_{\mathrm{apr}}^{(\mathrm{D})}}\left(\boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{D})} \,\middle|\, \mathbf{c} = \boldsymbol{c}\right) \mathrm{P}\left(\mathbf{c} = \boldsymbol{c}\right)}{\sum\limits_{\substack{\boldsymbol{c}\in\{0,1\}^{SQ} \\ c_i=1}} \mathrm{p}_{\mathbf{l}_{\mathrm{apr}}^{(\mathrm{D})}}\left(\boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{D})} \,\middle|\, \mathbf{c} = \boldsymbol{c}\right) \mathrm{P}\left(\mathbf{c} = \boldsymbol{c}\right)} \tag{7}$$

using Bayes' theorem and the assumption that the *a priori* LLR vector $\boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{D})}$ is a realization of the vector random variable $\mathbf{l}_{\mathrm{apr}}^{(\mathrm{D})}$.[3] Using the fact that the coded data bits $\mathsf{c}_i = \boldsymbol{e}_i^{\mathrm{T}}\mathbf{c}$, $i \in \{1, 2, \ldots, SQ\}$, in $\mathbf{c}$ are statistically independent due to the interleaver, it holds $\mathrm{P}(\mathbf{c} = \boldsymbol{c}) = \prod_{j=1}^{SQ} \mathrm{P}(\mathsf{c}_j = c_j)$ and finally [4]

$$l_{\mathrm{apo},i}^{(\mathrm{D})} = \underbrace{\ln \frac{\sum\limits_{\substack{\boldsymbol{c}\in\{0,1\}^{SQ} \\ c_i=0}} \mathrm{p}_{\mathbf{l}_{\mathrm{apr}}^{(\mathrm{D})}}\left(\boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{D})} \,\middle|\, \mathbf{c} = \boldsymbol{c}\right) \prod\limits_{\substack{j=1 \\ j\neq i}}^{SQ} \mathrm{P}\left(\mathsf{c}_j = c_j\right)}{\sum\limits_{\substack{\boldsymbol{c}\in\{0,1\}^{SQ} \\ c_i=1}} \mathrm{p}_{\mathbf{l}_{\mathrm{apr}}^{(\mathrm{D})}}\left(\boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{D})} \,\middle|\, \mathbf{c} = \boldsymbol{c}\right) \prod\limits_{\substack{j=1 \\ j\neq i}}^{SQ} \mathrm{P}\left(\mathsf{c}_j = c_j\right)}}_{l_{\mathrm{ext},i}^{(\mathrm{D})}} + \underbrace{\ln \frac{\mathrm{P}\left(\mathsf{c}_i = 0\right)}{\mathrm{P}\left(\mathsf{c}_i = 1\right)}}_{l_{\mathrm{apr},i}^{(\mathrm{D})}}. \tag{8}$$

Thus, the extrinsic information $\boldsymbol{l}_{\mathrm{ext}}^{(\mathrm{D})}$ can be obtained by subtracting the *a priori* information $\boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{D})}$ computed by the detector at the previous iteration step from the *a posteriori* LLR vector $\boldsymbol{l}_{\mathrm{apo}}^{(\mathrm{D})}$ offered by the decoder at the current iteration step.

During the iterative process, the *a priori* LLRs $l_{\mathrm{apr},i}^{(\mathrm{D})}$, $i \in \{1, 2, \ldots, SQ\}$, of the coded data bits c_i improve from iteration to iteration. The asymptotic values are $+\infty$ or $-\infty$ depending on the detected coded data bits $\tilde{c}_i$, i.e., $l_{\mathrm{apr},i}^{(\mathrm{D})} \to +\infty$ if $\tilde{c}_i = 0$ and $l_{\mathrm{apr},i}^{(\mathrm{D})} \to -\infty$ if $\tilde{c}_i = 1$. In other words, the probability $\mathrm{P}(\mathsf{c}_i = \tilde{c}_i)$ converges to one. Note that $\mathrm{P}(\mathsf{c}_i = \tilde{c}_i) \to 1$ does not mean that $\tilde{c}_i$ was actually transmitted, but the receiver *assumes* that $\tilde{c}_i$ is the transmitted coded data bit c_i.

After the last iteration, the MAP decoder provides the decoded data bits ($i \in \{1, 2, \ldots, B\}$) [4]

$$\boldsymbol{e}_i^{\mathrm{T}}\tilde{\boldsymbol{b}} = \underset{b\in\{0,1\}}{\operatorname{argmax}} \mathrm{P}\left(\boldsymbol{e}_i^{\mathrm{T}}\mathbf{b} = b \,\middle|\, \mathbf{l}_{\mathrm{apr}}^{(\mathrm{D})} = \boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{D})}\right). \tag{9}$$

III. Linear Detection

A. A Priori Based Linear Equalizer

For the derivation of the linear equalizer in Fig. 2, we use the channel model in (2) based on random processes. As can be seen at the end of this subsection, the mean $m_{\mathsf{s}}[k]$ and the variance $\sigma_{\mathsf{s}}^2[k]$ of the symbol sequence $\mathsf{s}[k]$ is a function of the *a priori* information about the interleaved and coded data bits. Thus, if we obtain *a priori* information from the decoder which is unequal to zero and varies with time index k, the random process $\mathsf{s}[k]$ must be assumed to be nonzero-mean and nonstationary. The estimate of the transmitted symbol sequence $\mathsf{s}[k]$ is obtained by linear filtering of the nonstationary vector random process $\mathbf{y}[k]$, i.e.,

$$\hat{\mathsf{s}}[k] = \boldsymbol{w}^{\mathrm{H}}[k]\mathbf{y}[k] + a[k] \tag{10}$$

[3]The modeling of the *a priori* LLR by a random variable is necessary to design the decoder based on the MAP criterion. Certainly, this model is reasonable since the actual *a priori* knowledge is estimated via the deinterleaved extrinsic information offered by the linear detector.

where $\boldsymbol{w}[k] \in \mathbb{C}^N$ and $a[k] \in \mathbb{C}$ are linear *time-variant* (TV) filter coefficients. The scalar $a[k]$ is necessary due to the mean $m_{\mathsf{s}}[k]$ of the symbol sequence $\mathsf{s}[k]$. The WF [20], [4] is the solution of the optimization

$$(\boldsymbol{w}[k], a[k]) = \underset{(\boldsymbol{w},a)}{\operatorname{argmin}}\, \xi_k(\boldsymbol{w}, a) \tag{11}$$

where $\xi_k(\boldsymbol{w}, a) = \mathrm{E}\{|\mathsf{s}[k-\kappa] - \hat{\mathsf{s}}_k|^2\}$ is the *mean-square error* (MSE) at time index k between the symbol $\mathsf{s}[k-\kappa]$ and its estimate[4] $\hat{\mathsf{s}}_k = \boldsymbol{w}^{\mathrm{H}}\mathbf{y}[k] + a$, i.e.,

$$\begin{aligned}\xi_k(\boldsymbol{w}, a) = 2\,\mathrm{Re}\left\{\boldsymbol{w}^{\mathrm{H}}\boldsymbol{m}_{\mathbf{y}}[k]a^* - \boldsymbol{w}^{\mathrm{H}}\boldsymbol{r}_{\mathbf{y},\mathsf{s}}[k] - a^* m_{\mathsf{s}}[k-\kappa]\right\} \\ + \boldsymbol{w}^{\mathrm{H}}\boldsymbol{R}_{\mathbf{y}}[k]\boldsymbol{w} + r_{\mathsf{s}}[k-\kappa] + |a|^2.\end{aligned} \tag{12}$$

Note that we assume the latency time κ to be fixed and not optimized with respect to the cost function in (12), although this would improve the performance (cf., e.g., [21] and [22]). Finding the optimum given by (11) is equal to solving jointly the equations $\partial\xi_k(\boldsymbol{w}, a)/\partial\boldsymbol{w} = \mathbf{0}_N$ and $\partial\xi_k(\boldsymbol{w}, a)/\partial a = 0$. Using the definitions of the autocovariance matrices and the cross-covariance vectors given at the end of Section I, the solution of the optimization given in (11) computes as

$$\boldsymbol{w}[k] = \boldsymbol{C}_{\mathbf{y}}^{-1}[k]\boldsymbol{c}_{\mathbf{y},\mathsf{s}}[k] \tag{13}$$

$$a[k] = m_{\mathsf{s}}[k-\kappa] - \boldsymbol{w}^{\mathrm{H}}[k]\boldsymbol{m}_{\mathbf{y}}[k]. \tag{14}$$

In order to ensure the adherence of the *turbo principle* [23], we choose the filter coefficients such that they do not depend on the *a priori* information corresponding to the symbol $\mathsf{s}[k-\kappa]$. Precisely speaking, we assume $\mathsf{s}[k-\kappa]$ to be uniformly distributed with zero-mean, i.e., $m_{\mathsf{s}}[k-\kappa] = 0$ and $\sigma_{\mathsf{s}}^2[k-\kappa] = r_{\mathsf{s}}[k-\kappa] = 2^{-Q}\sum_{s\in\mathbb{M}}|s|^2 =: \varrho_{\mathsf{s}}$. In Section III-C, we will explain in more detail why this assumption is necessary. If we recall additionally the transmission model defined in (2), (13) and (14) may be rewritten as

$$\boldsymbol{\omega}[k] = \left(\boldsymbol{H}\boldsymbol{\Gamma}_{\mathbf{s}}[k]\boldsymbol{H}^{\mathrm{H}} + \sigma_{\boldsymbol{\eta}}^2\boldsymbol{I}_N\right)^{-1}\boldsymbol{H}\boldsymbol{\Gamma}_{\mathbf{s}}[k]\boldsymbol{e}_{\kappa+1} \tag{15}$$

$$\alpha[k] = -\boldsymbol{\omega}^{\mathrm{H}}[k]\boldsymbol{H}\boldsymbol{\mu}_{\mathbf{s}}[k]. \tag{16}$$

Here, the matrix

$$\begin{aligned}\boldsymbol{\Gamma}_{\mathbf{s}}[k] = \mathrm{diag}\left\{\sigma_{\mathsf{s}}^2[k], \ldots, \sigma_{\mathsf{s}}^2[k-\kappa+1], \varrho_{\mathsf{s}},\right. \\ \left.\sigma_{\mathsf{s}}^2[k-\kappa-1], \ldots, \sigma_{\mathsf{s}}^2[k-K-L+1]\right\}\end{aligned} \tag{17}$$

in the following denoted as the *adjusted autocovariance matrix* of the symbol vector $\mathbf{s}[k]$, is the $(K+L)\times(K+L)$ diagonal autocovariance matrix $\boldsymbol{C}_{\mathbf{s}}[k]$, where the $(\kappa+1)$th diagonal element is replaced by ϱ_{s}, and the vector

$$\begin{aligned}\boldsymbol{\mu}_{\mathbf{s}}[k] = [m_{\mathsf{s}}[k], \ldots, m_{\mathsf{s}}[k-\kappa+1], 0, m_{\mathsf{s}}[k-\kappa-1], \\ \ldots, m_{\mathsf{s}}[k-K-L+1]]^{\mathrm{T}} \in \mathbb{C}^{K+L}\end{aligned} \tag{18}$$

denoted as the *adjusted mean* of $\mathbf{s}[k]$, is the mean $\boldsymbol{m}_{\mathbf{s}}[k]$, where the $(\kappa+1)$th element is set to zero. Nondiagonal elements in $\boldsymbol{C}_{\mathbf{s}}[k]$ and $\boldsymbol{\Gamma}_{\mathbf{s}}[k]$ vanish because the symbols $\mathsf{s}[k]$ are temporally uncorrelated due to the interleaver. Both $\boldsymbol{\Gamma}_{\mathbf{s}}[k]$ and $\boldsymbol{\mu}_{\mathbf{s}}[k]$ are computed using the *a priori* information $\boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{LD})}$. We define $\boldsymbol{l}^{(\mathrm{LD})}$ to be the vector composed by the LLRs $l_{kQ+q}^{(\mathrm{LD})} = \boldsymbol{e}_{kQ+q}^{\mathrm{T}}\boldsymbol{l}^{(\mathrm{LD})}$, $k \in \{0, 1, \ldots, S-1\}$, $q \in \{1, 2, \ldots, Q\}$, of the interleaved and coded data bits $\mathsf{c}'_{kQ+q} = \boldsymbol{e}_q^{\mathrm{T}}\mathbf{c}'_{k+1}$ which can be either *a priori* or extrinsic information (cf. Fig. 2). We have to find expressions for $m_{\mathsf{s}}[k]$ and $r_{\mathsf{s}}[k]$ in order to get finally $\boldsymbol{\Gamma}_{\mathbf{s}}[k]$ and $\boldsymbol{\mu}_{\mathbf{s}}[k]$. Remember that $\sigma_{\mathsf{s}}^2[k] = r_{\mathsf{s}}[k] - |m_{\mathsf{s}}[k]|^2$. It holds [8]

$$m_{\mathsf{s}}[k] = \sum_{s\in\mathbb{M}} s\,\mathrm{P}(\mathsf{s}[k] = s) \tag{19}$$

and

$$r_{\mathsf{s}}[k] = \sum_{s\in\mathbb{M}} |s|^2\,\mathrm{P}(\mathsf{s}[k] = s) \tag{20}$$

with

$$\mathrm{P}(\mathsf{s}[k] = s) = \mathrm{P}\left(\mathbf{c}'_{k+1} = \boldsymbol{d}(s)\right) = \prod_{q=1}^{Q}\mathrm{P}\left(\mathsf{c}'_{kQ+q} = \boldsymbol{e}_q^{\mathrm{T}}\boldsymbol{d}(s)\right) \tag{21}$$

where $\boldsymbol{d} : \mathbb{M} \to \{0,1\}^Q$, $s \mapsto \mathbf{c}'_{k+1} = \boldsymbol{d}(s)$ is the inverse function of the mapper M, i.e., $\boldsymbol{d}(M(\mathbf{c}'_{k+1})) = \mathbf{c}'_{k+1}$, and the elements of $\mathbf{c}'_{k+1}$ are statistically independent due to the interleaver. Finally, the probability $\mathrm{P}(\mathsf{c}'_{kQ+q} = \boldsymbol{e}_q^{\mathrm{T}}\boldsymbol{d}(s))$ can be expressed using the *a priori* information $l_{\mathrm{apr},kQ+q}^{(\mathrm{LD})}$ in the following way[5]:

$$\begin{aligned}&\mathrm{P}\left(\mathsf{c}'_{kQ+q} = \boldsymbol{e}_q^{\mathrm{T}}\boldsymbol{d}(s)\right) \\ &\quad= \frac{1}{2}\left(1 + \left(1 - 2\boldsymbol{e}_q^{\mathrm{T}}\boldsymbol{d}(s)\right)\tanh\frac{l_{\mathrm{apr},kQ+q}^{(\mathrm{LD})}}{2}\right).\end{aligned} \tag{22}$$

Thus, $\boldsymbol{\Gamma}_{\mathbf{s}}[k]$ and $\boldsymbol{\mu}_{\mathbf{s}}[k]$ and after all, the filter coefficients $\boldsymbol{\omega}[k]$ and $\alpha[k]$ can be computed using $\boldsymbol{l}_{\mathrm{apr}}^{(\mathrm{LD})}$.

For *quadriphase shift keying* (QPSK) modulation with

$$\mathbb{M} = \left\{\frac{1}{\sqrt{2}} + \frac{\mathrm{j}}{\sqrt{2}}, \frac{1}{\sqrt{2}} - \frac{\mathrm{j}}{\sqrt{2}}, -\frac{1}{\sqrt{2}} + \frac{\mathrm{j}}{\sqrt{2}}, -\frac{1}{\sqrt{2}} - \frac{\mathrm{j}}{\sqrt{2}}\right\}$$

used in the simulations of Section V, we get the simplified expressions [8], [24]

$$m_{\mathsf{s}}[k] = \frac{1}{\sqrt{2}}\left(\tanh\frac{l_{\mathrm{apr},2k+1}^{(\mathrm{LD})}}{2} + \mathrm{j}\,\tanh\frac{l_{\mathrm{apr},2k+2}^{(\mathrm{LD})}}{2}\right) \tag{23}$$

$$r_{\mathsf{s}}[k] = 1. \tag{24}$$

[4]The optimal estimate $\hat{\mathsf{s}}[k]$ is obtained by applying the optimal filter coefficients $\boldsymbol{w}[k]$ and $a[k]$, whereas $\hat{\mathsf{s}}_k$ denotes the estimate obtained by applying arbitrary filter coefficients $\boldsymbol{w}$ and a.

[5]It holds for the LLR l of the binary random variable $\mathsf{d} \in \{0,1\}$: $\tanh(l/2) = 2\mathrm{P}(\mathsf{d}=0) - 1 = 1 - 2\mathrm{P}(\mathsf{d}=1)$.

B. Soft Demapping

The $(kQ+q)$th element of the extrinsic information vector $\boldsymbol{l}_{\text{ext}}^{(\text{LD})}$ delivered by the linear detector is generated by *soft demapping* of the estimated symbol sequence $\hat{s}[k]$. By definition, we get (cf. (8))

$$l_{\text{ext},kQ+q}^{(\text{Det})} = \ln \frac{\sum\limits_{\substack{\boldsymbol{c}\in\{0,1\}^Q \\ \boldsymbol{e}_q^{\mathrm{T}}\boldsymbol{c}=0}} \mathrm{p}_{\hat{\mathsf{s}}[k]}\left(\hat{s}[k]\middle|\mathsf{c}'_{k+1}=\boldsymbol{c}\right) \prod\limits_{\substack{\ell=1 \\ \ell\neq q}}^{Q} \mathrm{P}\left(\boldsymbol{e}_\ell^{\mathrm{T}}\mathsf{c}'_{k+1}=\boldsymbol{e}_\ell^{\mathrm{T}}\boldsymbol{c}\right)}{\sum\limits_{\substack{\boldsymbol{c}\in\{0,1\}^Q \\ \boldsymbol{e}_q^{\mathrm{T}}\boldsymbol{c}=1}} \mathrm{p}_{\hat{\mathsf{s}}[k]}\left(\hat{s}[k]\middle|\mathsf{c}'_{k+1}=\boldsymbol{c}\right) \prod\limits_{\substack{\ell=1 \\ \ell\neq q}}^{Q} \mathrm{P}\left(\boldsymbol{e}_\ell^{\mathrm{T}}\mathsf{c}'_{k+1}=\boldsymbol{e}_\ell^{\mathrm{T}}\boldsymbol{c}\right)} \tag{25}$$

where q, $\ell \in \{1,2,\ldots,Q\}$, and $k \in \{0,1,\ldots,S-1\}$. From (22), we know that

$$\begin{aligned}\mathrm{P}\left(\boldsymbol{e}_\ell^{\mathrm{T}}\mathsf{c}'_{k+1}=\boldsymbol{e}_\ell^{\mathrm{T}}\boldsymbol{c}\right) &= \mathrm{P}\left(\mathsf{c}'_{kQ+\ell}=\boldsymbol{e}_\ell^{\mathrm{T}}\boldsymbol{c}\right) \\ &= \frac{1}{2}\left(1+\left(1-2\boldsymbol{e}_\ell^{\mathrm{T}}\boldsymbol{c}\right)\tanh\frac{l_{\text{apr},kQ+\ell}^{(\text{LD})}}{2}\right).\end{aligned} \tag{26}$$

The remaining part in calculating $l_{\text{ext},kQ+q}^{(\text{Det})}$ (cf. (25)) is the unknown probability density function $\mathrm{p}_{\hat{\mathsf{s}}[k]}(\hat{s}[k]\,|\,\mathsf{c}'_{k+1}=\boldsymbol{c})$, which we approximate assuming a Gaussian distribution, i.e.,

$$\mathrm{p}_{\hat{\mathsf{s}}[k]}\left(\hat{s}[k]\middle|\mathsf{c}'_{k+1}=\boldsymbol{c}\right) = \frac{\exp\left(-\frac{\left|\hat{s}[k]-\mu_{(\boldsymbol{c})}[k]\right|^2}{\sigma^2_{(\boldsymbol{c})}[k]}\right)}{\pi\sigma^2_{(\boldsymbol{c})}[k]} \tag{27}$$

with [8]

$$\begin{aligned}\mu_{(\boldsymbol{c})}[k] &= \mathrm{E}\left\{\hat{\mathsf{s}}[k]\,\middle|\,\mathsf{s}[k-\kappa]=M(\boldsymbol{c})\right\} \\ &= M(\boldsymbol{c})\boldsymbol{\omega}^{\mathrm{H}}[k]\boldsymbol{H}\boldsymbol{e}_{\kappa+1} \qquad (28)\\ \sigma^2_{(\boldsymbol{c})}[k] &= \mathrm{E}\left\{\left|\hat{\mathsf{s}}[k]\right|^2 \,\middle|\, \mathsf{s}[k-\kappa]=M(\boldsymbol{c})\right\} - \left|\mu_{(\boldsymbol{c})}[k]\right|^2 \\ &= \boldsymbol{\omega}^{\mathrm{H}}[k]\left(\boldsymbol{H}\left(\boldsymbol{\Gamma}_{\mathsf{s}}[k]-\varrho_{\mathsf{s}}\boldsymbol{e}_{\kappa+1}\boldsymbol{e}_{\kappa+1}^{\mathrm{T}}\right)\boldsymbol{H}^{\mathrm{H}}+\sigma_{\boldsymbol{\eta}}^2\boldsymbol{I}_N\right)\boldsymbol{\omega}[k]. \qquad (29)\end{aligned}$$

Here, we used $\hat{\mathsf{s}}[k] = \boldsymbol{\omega}^{\mathrm{H}}[k]\mathbf{y}[k] + \alpha[k]$ with the filter coefficient $\alpha[k]$ from (16) and the channel model defined in (2). Note that with $\mathsf{s}'[k] = \mathsf{s}[k] - \boldsymbol{\mu}_{\mathsf{s}}[k]$, $\mathrm{E}\{\mathsf{s}'[k]\,|\,\mathsf{s}[k-\kappa]=M(\boldsymbol{c})\} = M(\boldsymbol{c})\boldsymbol{e}_{\kappa+1}$ and

$$\begin{aligned}\mathrm{E}\left\{\mathsf{s}'[k]\mathsf{s}'^{,\mathrm{H}}[k]\middle|\mathsf{s}[k-\kappa]=M(\boldsymbol{c})\right\} \\ = \boldsymbol{\Gamma}_{\mathsf{s}}[k] - (\varrho_{\mathsf{s}} - |M(\boldsymbol{c})|^2)\boldsymbol{e}_{\kappa+1}\boldsymbol{e}_{\kappa+1}^{\mathrm{T}}.\end{aligned}$$

For QPSK modulation used in Section V, the elements of the extrinsic information $\boldsymbol{l}_{\text{ext}}^{(\text{LD})}$ can be easily simplified to

$$l_{\text{ext},2k+1}^{(\text{LD})} + \mathrm{j}\, l_{\text{ext},2k+2}^{(\text{LD})} = \frac{\sqrt{8}}{1-\boldsymbol{\omega}^{\mathrm{H}}[k]\boldsymbol{H}\boldsymbol{e}_{\kappa+1}}\hat{s}[k]. \tag{30}$$

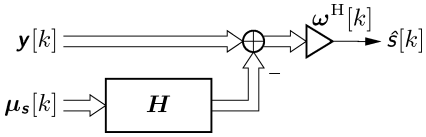

Fig. 3. Block diagram of linear equalizer.

C. Interpretation

In this subsection, we rewrite the estimate $\hat{\mathsf{s}}[k] = \boldsymbol{\omega}^{\mathrm{H}}[k]\mathbf{y}[k] + \alpha[k]$ such that it no longer depends on $\alpha[k]$. We will see that the result gives an interesting interpretation of the linear detector. Using (16), we get the expression

$$\hat{\mathsf{s}}[k] = \boldsymbol{\omega}^{\mathrm{H}}[k]\left(\mathbf{y}[k] - \boldsymbol{H}\boldsymbol{\mu}_{\mathsf{s}}[k]\right) \tag{31}$$

which is depicted in the block diagram of Fig. 3.

It can be seen that the adjusted mean vector $\boldsymbol{\mu}_{\mathsf{s}}[k]$, based on the *a priori* information in the way described by (19), (21), and (22), is used to subtract its influence after the transmission over the channel $\boldsymbol{H}$ from the observation vector $\mathbf{y}[k]$ to increase the degrees of freedom and therefore, the performance of the following linear filter $\boldsymbol{\omega}[k]$. This procedure is strongly related to the *decision feedback equalizer* [25], [26] or *interference canceler* [27], [28], where it is not the adjusted mean vector that is used to construct the interference, but the already decided symbols. Based on this analogy, the elements of the adjusted mean vector $\boldsymbol{\mu}_{\mathsf{s}}[k]$, defined via (19), can be interpreted as *soft decisions* or *soft symbols*. Note that the soft symbol is no soft information in terms of LLRs but a nonlinear transformation thereof (cf. (19), (21), and (22)).

The linear detector can also be derived by assuming the interference cancellation structure of Fig. 3 from the first (cf., e.g., [4], [29], and [10]). Moreover, this relationship between linear equalization and interference cancellation gives a good explanation why the mean $m_{\mathsf{s}}[k-\kappa]$ must be assumed to be zero in order to ensure the adherence of the turbo principle as stated in Section III-A. This assumption guarantees that we do not subtract the proportion of the signal of interest in the observation signal $\mathbf{y}[k]$ from it. However, the linear filter $\boldsymbol{\omega}[k]$ still depends on the available *a priori* information.

IV. Reduced-Complexity Linear Detection

Due to the inversion of the $N \times N$ matrix $\boldsymbol{\Gamma}_{\mathbf{y}}[k] = \boldsymbol{H}\boldsymbol{\Gamma}_{\mathsf{s}}[k]\boldsymbol{H}^{\mathrm{H}} + \sigma_{\boldsymbol{\eta}}^2\boldsymbol{I}_N$ in (15) at each time index $k \in \{0,1,\ldots,S-1\}$, the computational complexity of the filter calculation is $\mathrm{O}\left(SN^3\right)$, $N = RK$. This results in a computationally intense equalizer if the number of antennas R, the number of filter taps K to cover the order L channel, and/or the symbol block length S is high. In the following, we consider different strategies to reduce the computational complexity of the linear detector in such cases.

A. Optimal Reduced-Complexity Solution

At each time index k, the computational complexity can be reduced by one order if we exploit the time dependency of the diagonal matrix $\boldsymbol{\Gamma}_{\mathsf{s}}[k]$ in (15). From (17), we know that the last $K+L-1$ diagonal elements of $\boldsymbol{\Gamma}_{\mathsf{s}}[k]$ are equal to the first $K+L-1$ diagonal elements of $\boldsymbol{\Gamma}_{\mathsf{s}}[k-1]$. If we partition the adjusted autocovariance matrices at time indexes $i \in \{k, k-1\}$[6]

[6]Note that the given definitions for $i \in \{k, k-1\}$ are a compact notation for two types of decompositions with matrices of different sizes.

$$\boldsymbol{\Gamma}_{\mathbf{y}}[i] = \boldsymbol{H}\boldsymbol{\Gamma}_{\mathbf{s}}[i]\boldsymbol{H}^{\mathrm{H}} + \sigma_{\boldsymbol{\eta}}^2\boldsymbol{I}_N = \begin{bmatrix} \boldsymbol{D}[i] & \boldsymbol{E}[i] \\ \boldsymbol{E}^{\mathrm{H}}[i] & \boldsymbol{F}[i] \end{bmatrix} \tag{32}$$

and their inverses

$$\boldsymbol{\Gamma}_{\mathbf{y}}^{-1}[i] = \begin{bmatrix} \bar{\boldsymbol{D}}[i] & \bar{\boldsymbol{E}}[i] \\ \bar{\boldsymbol{E}}^{\mathrm{H}}[i] & \bar{\boldsymbol{F}}[i] \end{bmatrix}, \quad i \in \{k, k-1\} \tag{33}$$

such that $\boldsymbol{D}[k]$, $\boldsymbol{F}[k-1]$, $\bar{\boldsymbol{D}}[k]$, and $\bar{\boldsymbol{F}}[k-1]$ are $R \times R$ matrices, $\boldsymbol{E}[k]$, $\boldsymbol{E}^{\mathrm{H}}[k-1]$, $\bar{\boldsymbol{E}}[k]$, and $\bar{\boldsymbol{E}}^{\mathrm{H}}[k-1]$ are $R \times (N-R)$ matrices, and $\boldsymbol{F}[k]$, $\boldsymbol{D}[k-1]$, $\bar{\boldsymbol{F}}[k]$, and $\bar{\boldsymbol{D}}[k-1]$ are $(N-R) \times (N-R)$ matrices, and recall the structure of the channel matrix given in (6), it follows: $\boldsymbol{F}[k] = \boldsymbol{D}[k-1]$.

The fundamental idea of the *reduced-complexity* (RC) implementation [8] is to exploit the fact that $\boldsymbol{F}[k] = \boldsymbol{D}[k-1]$ in order to compute $\boldsymbol{\Gamma}_{\mathbf{y}}^{-1}[k]$, viz., $\bar{\boldsymbol{D}}[k]$, $\bar{\boldsymbol{E}}[k]$, and $\bar{\boldsymbol{F}}[k]$, using the block elements of $\boldsymbol{\Gamma}_{\mathbf{y}}[k]$ and the blocks of the already computed inverse $\boldsymbol{\Gamma}_{\mathbf{y}}^{-1}[k-1]$ from the previous time index. Finally, using the inversion lemma for partitioned matrices [20] and the matrix inversion lemma [20] yields the solution[7]

$$\bar{\boldsymbol{D}}[k] = \left(\boldsymbol{D}[k] - \boldsymbol{E}[k]\boldsymbol{F}^{-1}[k]\boldsymbol{E}^{\mathrm{H}}[k]\right)^{-1} \tag{34}$$

$$\bar{\boldsymbol{E}}[k] = -\bar{\boldsymbol{D}}[k]\boldsymbol{E}[k]\boldsymbol{F}^{-1}[k] \tag{35}$$

$$\bar{\boldsymbol{F}}[k] = \boldsymbol{F}^{-1}[k]\left(\boldsymbol{I}_{N-R} - \boldsymbol{E}^{\mathrm{H}}[k]\bar{\boldsymbol{E}}[k]\right) \tag{36}$$

where the inverse of $\boldsymbol{F}[k]$ can be expressed by the *Schur complement* [20] of $\bar{\boldsymbol{D}}[k-1]$ in $\boldsymbol{\Gamma}_{\mathbf{y}}^{-1}[k-1]$, i.e.,

$$\begin{aligned} \boldsymbol{F}^{-1}[k] &= \boldsymbol{D}^{-1}[k-1] \\ &= \bar{\boldsymbol{D}}[k-1] - \bar{\boldsymbol{E}}[k-1]\bar{\boldsymbol{F}}^{-1}[k-1]\bar{\boldsymbol{E}}^{\mathrm{H}}[k-1]. \end{aligned} \tag{37}$$

The enormous reduction in computational complexity is detailed investigated in Section IV-D. Note that the proposed RC algorithm is still optimal in the MMSE sense. The next subsections discuss suboptimal solutions.

B. Suboptimal Reduced-Rank Implementations

The *reduced-rank WF* $\boldsymbol{w}_D[k] \in \mathbb{C}^N$ denotes a rank $D < N$ approximation of the WF $\boldsymbol{w}[k]$ in a subspace spanned by the D columns of the prefilter matrix $\boldsymbol{T}_D[k] \in \mathbb{C}^{N \times D}$, i.e., $\boldsymbol{w}_D[k] = \boldsymbol{T}_D[k]\boldsymbol{g}_D[k] \in \mathbb{C}^N$, where $\boldsymbol{g}_D[k] \in \mathbb{C}^D$ denotes a *reduced-dimension WF* obtained from the optimization

$$\{\boldsymbol{g}_D[k], a_D[k]\} = \underset{\{\boldsymbol{g},a\}}{\operatorname{argmin}}\, \xi_k\left(\boldsymbol{T}_D[k]\boldsymbol{g}, a\right) \tag{38}$$

with the MSE function ξ_k from (12). The solution easily computes as

$$\boldsymbol{w}_D[k] = \boldsymbol{T}_D[k]\left(\boldsymbol{T}_D^{\mathrm{H}}[k]\boldsymbol{C}_{\mathbf{y}}[k]\boldsymbol{T}_D[k]\right)^{-1}\boldsymbol{T}_D^{\mathrm{H}}[k]\boldsymbol{c}_{\mathbf{y},\mathsf{s}}[k] \tag{39}$$

$$a_D[k] = m_{\mathsf{s}}[k-\kappa] - \boldsymbol{w}_D^{\mathrm{H}}[k]\boldsymbol{m}_{\mathbf{y}}[k]. \tag{40}$$

[7]The solution can easily be derived by rewriting the identities $\boldsymbol{\Gamma}_{\mathbf{y}}[i]\boldsymbol{\Gamma}_{\mathbf{y}}^{-1}[i] = \boldsymbol{I}_N$, $i \in \{k, k-1\}$, into eight equations using the block structures defined in (32) and (33).

Again, the filters $\boldsymbol{\omega}_D[k]$ and $\alpha_D[k]$, finally used in the linear detector of Fig. 2, are obtained by assuming $m_{\mathsf{s}}[k-\kappa] = 0$ and $\sigma_{\mathsf{s}}^2[k-\kappa] = \varrho_{\mathsf{s}}$, i.e.,

$$\boldsymbol{\omega}_D[k] = \boldsymbol{T}_D[k]\left(\boldsymbol{T}_D^{\mathrm{H}}[k]\boldsymbol{\Gamma}_{\mathbf{y}}[k]\boldsymbol{T}_D[k]\right)^{-1}\boldsymbol{T}_D^{\mathrm{H}}[k]\boldsymbol{\gamma}_{\mathbf{y},\mathsf{s}}[k] \tag{41}$$

$$\alpha_D[k] = -\boldsymbol{\omega}_D^{\mathrm{H}}[k]\boldsymbol{H}\boldsymbol{\mu}_{\mathbf{s}}[k] \tag{42}$$

with the adjusted autocovariance matrix $\boldsymbol{\Gamma}_{\mathbf{y}}[k] = \boldsymbol{H}\boldsymbol{\Gamma}_{\mathbf{s}}[k]\boldsymbol{H}^{\mathrm{H}} + \sigma_{\boldsymbol{\eta}}^2\boldsymbol{I}_N$ of the observation vector $\mathbf{y}[k]$ and the *adjusted cross-covariance vector* $\boldsymbol{\gamma}_{\mathbf{y},\mathsf{s}}[k] = \varrho_{\mathsf{s}}\boldsymbol{H}\boldsymbol{e}_{\kappa+1}$ between the observation $\mathbf{y}[k]$ and the desired signal $\mathsf{s}[k-\kappa]$ (cf. (15)), and where we used the channel model given in (2).

One category of reduced-rank methods is based on eigensubspaces. Consider the eigenvalue decomposition of the Hermitian and positive definite adjusted autocovariance matrix $\boldsymbol{\Gamma}_{\mathbf{y}}[k] = \boldsymbol{Q}[k]\boldsymbol{\Lambda}[k]\boldsymbol{Q}^{\mathrm{H}}[k]$, where $\boldsymbol{\Lambda}[k] = \operatorname{diag}\{\lambda_1[k], \lambda_2[k], \ldots, \lambda_N[k]\}$ is the diagonal matrix of the eigenvalues $\lambda_1[k] \geq \lambda_2[k] \geq \cdots \geq \lambda_N[k] > 0$ and $\boldsymbol{Q}[k] = [\boldsymbol{q}_1[k], \boldsymbol{q}_2[k], \ldots, \boldsymbol{q}_N[k]]$ is the matrix composed by the eigenvectors $\boldsymbol{q}_i[k] \in \mathbb{C}^N$, $i \in \{1, 2, \ldots, N\}$. The PC [16] method chooses the columns of the prefilter matrix $\boldsymbol{T}_D[k]$ to be the D eigenvectors $\boldsymbol{q}_i[k]$, $i \in \{1, 2, \ldots, D\}$, corresponding to the largest eigenvalues $\lambda_i[k]$, $i \in \{1, 2, \ldots, D\}$, of $\boldsymbol{\Gamma}_{\mathbf{y}}[k]$, i.e., $\boldsymbol{T}_D[k] = [\boldsymbol{q}_1[k], \boldsymbol{q}_2[k], \ldots, \boldsymbol{q}_D[k]]$. The motivation for this selection is to design the prefilter such that the signal power loss due to the dimension reduction from N to $D < N$ is as small as possible. Nevertheless, the PC algorithm does not differentiate if the signal power is due to the signal of interest or due to an interference signal. Thus, in systems with strong interferers, the PC method suffers from severe performance degradation. The more recently introduced CS [17] algorithm combats this lack of the PC method by selecting the columns of $\boldsymbol{T}_D[k]$ to be the D eigenvectors yielding the smallest MSE, hence, being the optimal eigensubspace method in the MMSE sense. Unfortunately, the CS method requires the computation of all eigenvectors. Thus, it has the same order of computational complexity as the full-rank approach. Since we are interested in computationally cheap implementations, the CS method is only presented as a performance bound for eigensubspace methods. More detailed derivations and investigations of the CS algorithm can be found in [30].

An alternative to eigensubspace methods, the so-called MSWF, has been introduced by Goldstein *et al.* in [11]. The MSWF is a stage-wise decomposition of the WF consisting of MFs, so-called *blocking matrices*, and scalar WFs.

Although this blocking-matrix-based derivation of the MSWF is more general, we concentrate on the version of the MSWF introduced by Joham *et al.* [13] depicted in Fig. 4. There, the first column $\boldsymbol{t}_1[k] \in \mathbb{C}^N$ of the prefilter matrix $\boldsymbol{T}_D[k] = [\boldsymbol{t}_1[k], \boldsymbol{t}_2[k], \ldots, \boldsymbol{t}_D[k]]$ is chosen to be the normalized MF $\boldsymbol{c}_{\mathbf{y},\mathsf{s}}[k]/\left\|\boldsymbol{c}_{\mathbf{y},\mathsf{s}}[k]\right\|_2$. Thus, its output $\mathsf{s}_1[k]$ is maximal correlated to the desired signal $\mathsf{s}[k-\kappa]$. The ith column $\boldsymbol{t}_i[k] \in \mathbb{C}^N$, $i \in \{2, 3, \ldots, D\}$, maximizes the real part of the correlation between its output $\mathsf{s}_i[k]$ and the output $\mathsf{s}_{i-1}[k]$ of the previous prefilter vector $\boldsymbol{t}_{i-1}[k]$, i.e., it fulfills the optimization

$$\boldsymbol{t}_i[k] = \underset{\boldsymbol{t}}{\operatorname{argmax}}\, \mathrm{E}\left\{\operatorname{Re}\left\{\mathsf{s}_i[k]\mathsf{s}_{i-1}^*[k]\right\}\right\} \quad \text{s.t.:} \quad \boldsymbol{t}^{\mathrm{H}}\boldsymbol{t} = 1 \quad \text{and} \quad \boldsymbol{t}^{\mathrm{H}}\boldsymbol{t}_j[k] = 0, \quad 1 \leq j < i \tag{43}$$

if we restrict the prefilter vectors $\boldsymbol{t}_i[k]$ to be orthonormal. The solution is identical to the well-known *Lanczos algorithm* [14], [15], [13]

$$\boldsymbol{t}_i[k] = \frac{\boldsymbol{P}_{i-1}[k]\boldsymbol{P}_{i-2}[k]\boldsymbol{C}_{\mathsf{y}}[k]\boldsymbol{t}_{i-1}[k]}{\|\boldsymbol{P}_{i-1}[k]\boldsymbol{P}_{i-2}[k]\boldsymbol{C}_{\mathsf{y}}[k]\boldsymbol{t}_{i-1}[k]\|_2} \tag{44}$$

with the projection matrices $\boldsymbol{P}_j[k] = \boldsymbol{I}_N - \boldsymbol{t}_j[k]\boldsymbol{t}_j^{\mathrm{H}}[k]$, $j \in \{i-1, i-2\}$, projecting onto the nullspace of $\boldsymbol{t}_j^{\mathrm{H}}[k]$. Note that the autocovariance matrix of the transformed observation vector $\mathsf{s}_D[k] = [\mathsf{s}_1[k], \mathsf{s}_2[k], \ldots, \mathsf{s}_D[k]]^{\mathrm{T}}$ is tridiagonal due the Lanczos-based prefilter matrix [15]. The following reduced-dimension WF $\boldsymbol{g}_D[k] \in \mathbb{C}^D$ can be decomposed [11] into the D scalar WFs $g_i[k] \in \mathbb{C}$, $i \in \{1, 2, \ldots, D\}$, estimating the output signal $\mathsf{s}_{i-1}[k]$ of the previous prefilter based on the observation $\varepsilon_i[k] = \mathsf{s}_i[k] - \hat{\mathsf{s}}_i[k]$ in the MMSE sense. Note that we define $\mathsf{s}_0[k] = \mathsf{s}[k-\kappa]$ and $\hat{\mathsf{s}}_D[k] = 0$. With the cross-covariance $c_{\varepsilon_i,\mathsf{s}_{i-1}}[k] = \mathrm{E}\{\varepsilon_i[k]\mathsf{s}_{i-1}^*[k]\} - m_{\varepsilon_i}[k]m_{\mathsf{s}_{i-1}}^*[k]$, the solution of the optimization $g_i[k] = \operatorname{argmin}_g \mathrm{E}\{|\varepsilon_i[k]|^2\}$ computes as $g_i[k] = \sigma_{\varepsilon_i}^{-2}[k]c_{\varepsilon_i,\mathsf{s}_{i-1}}[k]$ and can be computed recursively beginning with $g_D[k]$ (cf., e.g., [11]). Finally, Algorithm I summarizes the Lanczos-based computation of the MSWF[8] coefficients $\boldsymbol{\omega}_D[k]$, where we assume again $m_{\mathsf{s}}[k-\kappa] = 0$, i.e., $\boldsymbol{C}_{\mathsf{y}}[k]$ and $\boldsymbol{c}_{\mathsf{y},\mathsf{s}}[k]$ have to be replaced by $\boldsymbol{\Gamma}_{\mathsf{y}}[k]$ and $\boldsymbol{\gamma}_{\mathsf{y},\mathsf{s}}[k]$, respectively. In addition, the backward recursion is adjusted according to the Lanczos-based prefilter computation [13]. It remains to compute $\alpha_D[k]$ according to (42).

Algorithm I: Lanczos-based computation of $\boldsymbol{\omega}_D[k]$

$\boldsymbol{t}_0 \leftarrow \boldsymbol{0}_N$, $c_{0,1} \leftarrow 0$

$\boldsymbol{t}_1 \leftarrow \boldsymbol{\gamma}_{\mathsf{y},\mathsf{s}}[k] / \|\boldsymbol{\gamma}_{\mathsf{y},\mathsf{s}}[k]\|_2$, $c_{1,1} = \beta_1 \leftarrow \boldsymbol{t}_1^{\mathrm{H}}\boldsymbol{\Gamma}_{\mathsf{y}}[k]\boldsymbol{t}_1$

$g_{\mathrm{first}}^{(1)} = g_{\mathrm{last}}^{(1)} \leftarrow c_{1,1}^{-1}$

for $i \in \{2, 3, \ldots, D\}$ **do**

$\boldsymbol{v} \leftarrow \boldsymbol{\Gamma}_{\mathsf{y}}[k]\boldsymbol{t}_{i-1} - c_{i-1,i-1}\boldsymbol{t}_{i-1} - c_{i-2,i-1}\boldsymbol{t}_{i-2}$

$c_{i-1,i} \leftarrow \|\boldsymbol{v}\|_2$

$\boldsymbol{t}_i \leftarrow \boldsymbol{v}/c_{i-1,i}$

$c_{i,i} \leftarrow \boldsymbol{t}_i^{\mathrm{H}}\boldsymbol{\Gamma}_{\mathsf{y}}[k]\boldsymbol{t}_i$

$\beta_i \leftarrow c_{i,i} - c_{i-1,i}^2\beta_{i-1}^{-1}$

$\boldsymbol{g}_{\mathrm{first}}^{(i)} \leftarrow \begin{bmatrix} \boldsymbol{g}_{\mathrm{first}}^{(i-1)} \\ 0 \end{bmatrix} + \beta_i^{-1}\boldsymbol{e}_1^{\mathrm{T}}\boldsymbol{g}_{\mathrm{last}}^{(i-1)} \begin{bmatrix} c_{i-1,i}^2\boldsymbol{g}_{\mathrm{last}}^{(i-1)} \\ -c_{i-1,i} \end{bmatrix}$

$\boldsymbol{g}_{\mathrm{last}}^{(i)} \leftarrow \beta_i^{-1} \begin{bmatrix} -c_{i-1,i}\boldsymbol{g}_{\mathrm{last}}^{(i-1)} \\ 1 \end{bmatrix}$

end for

$\boldsymbol{T}_D \leftarrow [\boldsymbol{t}_1, \boldsymbol{t}_2, \ldots, \boldsymbol{t}_D]$

$\boldsymbol{\omega}_D[k] = \|\boldsymbol{\gamma}_{\mathsf{y},\mathsf{s}}[k]\|_2 \boldsymbol{T}_D\boldsymbol{g}_{\mathrm{first}}^{(D)}$

With the knowledge that the prefilter vectors can be generated using the Lanczos procedure, it was shown [12], [13] that the columns of the prefilter matrix $\boldsymbol{T}_D[k]$ are basis vectors of the D-dimensional *Krylov subspace* [15]

$$\begin{aligned}&\mathcal{K}^{(D)}\left(\boldsymbol{\Gamma}_{\mathsf{y}}[k], \boldsymbol{\gamma}_{\mathsf{y},\mathsf{s}}[k]\right)\\ &\quad = \operatorname{span}\left\{\boldsymbol{\gamma}_{\mathsf{y},\mathsf{s}}[k], \boldsymbol{\Gamma}_{\mathsf{y}}[k]\boldsymbol{\gamma}_{\mathsf{y},\mathsf{s}}[k], \ldots, \boldsymbol{\Gamma}_{\mathsf{y}}^{D-1}[k]\boldsymbol{\gamma}_{\mathsf{y},\mathsf{s}}[k]\right\}.\end{aligned} \tag{45}$$

Therefore, the MSWF can be seen as an MSE optimal approximation of the WF in the subspace $\mathcal{K}^{(D)}\left(\boldsymbol{\Gamma}_{\mathsf{y}}[k], \boldsymbol{\gamma}_{\mathsf{y},\mathsf{s}}[k]\right)$, which is not equal to the eigensubspace.

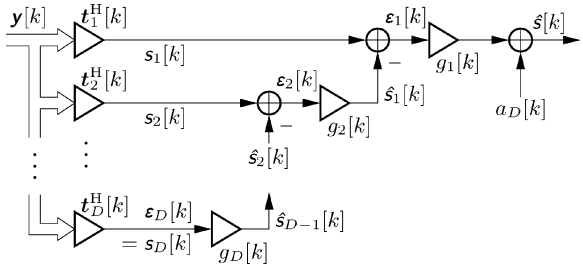

Fig. 4. MSWF as filter bank.

C. Suboptimal Time-Invariant Approach

To avoid the calculation of the TV filter coefficients at each time index k, they can be set *time-invariant* (TI) by approximating the TV adjusted autocovariance matrix $\boldsymbol{\Gamma}_{\mathsf{s}}[k]$ by its time average [8]

$$\bar{\boldsymbol{\Gamma}}_{\mathsf{s}} = \frac{1}{S}\sum_{i=0}^{S-1}\boldsymbol{\Gamma}_{\mathsf{s}}[i] \tag{46}$$

for each symbol block of length S. This method can be applied to either the optimal WF implementation (cf. Section IV-A) or the suboptimal reduced-rank solutions of Section IV-B. However, it is not applicable to the RC implementation of Section IV-A since the RC approach strongly depends on the time variance of $\boldsymbol{\Gamma}_{\mathsf{s}}[k]$.

D. Complexity Investigations

Table I shows the number of real-valued FLOPs needed for the filter computations of one block and one turbo iteration where only the important terms with N^3 and N^2 are presented. As defined in numerical linear algebra (cf., e.g., [31]), one real-valued FLOP is a real-valued multiplication, division, addition, subtraction, or square root.

The optimal method without any complexity reduction is based on a *Cholesky factorization* together with a *forward* and *backward recursion* [32] whereas the optimal RC approach applies the ideas of Section IV-A. For each block, the latter one needs one inversion of a $N \times N$ matrix, i.e., the inversion of $\boldsymbol{\Gamma}_{\mathsf{y}}[0]$, in order to initialize the recursion given by (33) to (37). Note that the highest order term of the inversion of a $N \times N$ matrix is $4N^3$, which is three times larger than the highest order term $4N^3/3$ of solving the corresponding equation system with N equations and N unknowns. However, the inversion in the RC algorithm has to be performed only once per block whereas the optimal approach without any complexity reduction solves the equation system at each time index k, i.e., S times. Thus, the computational complexity of the one matrix inversion per block can be neglected if $S \gg N$ and the RC enhancement

[8]Note that in the following, the term MSWF as well as WF denotes the filter coefficients computed using the adjusted auto-covariance matrix and the adjusted cross-covariance vector although this is not explicitly mentioned.

TABLE I
NUMBER OF FLOPS FOR FILTER COMPUTATIONS

	TV	TI
Optimal	$S(\frac{4}{3}N^3+9N^2)$	$\frac{4}{3}N^3+9N^2$
Optimal, RC	$4N^3+8(S-1)(2R+1)N^2+9N^2$	—
MSWF (D=2)	$16SN^2$	$16N^2$
MSWF (D=1)	$8SN^2$	$8N^2$

reduces the computational effort of the optimal approach by one order.

Table I presents the computational complexity of the MSWF summarized in Algorithm I only for ranks $D \in \{1,2\}$ since the simulations in Section V will show that the rank $D=2$ MSWF already achieves quasi-optimal performance. It can be seen that, compared to the RC approach, the computational effort needed to compute the MSWF coefficients has no third order term since no matrix inversion is implied. The highest order term $8SDN^2$ arises from the matrix-vector multiplication $\boldsymbol{\Gamma}_{\mathbf{y}}[k]\boldsymbol{t}_i$ (cf. Lines 2, 5, and 8 of Algorithm I) which has to be performed D times for each symbol $k \in \{0,1,\ldots,S-1\}$. Note that the multiplication of the $N \times N$ matrix $\boldsymbol{\Gamma}_{\mathbf{y}}[k]$ by the vector $\boldsymbol{t}_i \in \mathbb{C}^N$ requires N^2 complex-valued multiplications and N^2-N complex-valued additions. Thus, the number of real-valued FLOPs[9] needed to perform the given matrix-vector product once, is $8N^2-2N$. The computational complexity of the remaining operations in Algorithm I can be neglected since their complexity order is at most linear. The number of FLOPs of the eigensubspace-based PC and CS method is not shown in Table I since their performance is much worse than the one of the MSWF as will be seen in Section V. Nevertheless, the order of computational effort to compute the PC filter weights is the same as for the MSWF, i.e., $\mathrm{O}\left(SDN^2\right)$, since algorithms which compute eigenvectors of a Hermitian matrix are also based on the Lanczos algorithm. There, the Lanczos method is used to tridiagonalize the matrix before performing the eigenvalue decomposition. Note that computing the eigenvectors of a tridiagonal matrix has a negligible order of computational complexity (cf., e.g., [31]). As mentioned above, the CS method is only presented as a performance bound for eigensubspace methods since its computational complexity is the same as the one of the full-rank WF, i.e., $\mathrm{O}\left(SN^3\right)$.

Up to the optimal RC approach, the effort to compute the TV filters is S times larger than the one for the TI methods. This is due to the fact that the TI filter coefficients have to be computed only once per each block instead of S times since the time-varying adjusted autocovariance matrix is replaced by its time average. However, also the TI filters have to be recomputed from block to block because the channel is assumed to be block varying. Nevertheless, the TI approaches achieve an additional tremendous reduction in computational complexity, especially if $S \gg N$ whereas the resulting performance degradation is negligible as we will see in the simulations of Section V. Moreover, for $S \gg N$, the TI approximation of the optimal WF without any complexity reduction should be preferred to the RC approach since the number of FLOPs is smaller. If one is interested in very computationally cheap implementations, the TI MSWF is finally the adequate algorithm.

[9]One complex-valued multiplication is treated as four real-valued multiplications and two real-valued additions, and one complex-valued addition is treated as two real-valued additions.

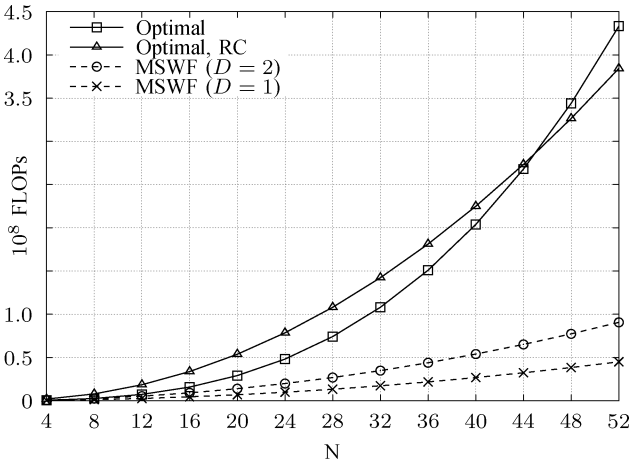

Fig. 5. FLOPs for computation of TV filter coefficients.

This statement is also approved by Fig. 5, which shows the exact number of FLOPs of the proposed TV filters for $R=4$ receive antennas and a symbol block length of $S=2048$. The figure demonstrates the computational efficiency of the suboptimal MSWFs with ranks $D \in \{1,2\}$ compared to the optimal solutions. Again, the MSWFs with ranks $D>2$ are not considered here since the simulation results of Section V will show that the rank $D=2$ MSWF achieves already near optimal performance in the given turbo scenario. Note that only for $N>45$, the RC approach needs less FLOPs than the optimal one due to the higher prefactor of the second-order term (cf. Table I). Again, the exact number of FLOPs to compute the TI filters are S times smaller than the one to compute the TV filter coefficients presented in Fig. 5.

V. SIMULATION RESULTS

In the sequel of this paper, we consider $B=2048$ bits per data block $\mathbf{b}$ which are encoded with a $(7,5)$-convolutional code, i.e., $r=0.5$, and randomly interleaved. The interleaved and coded bits are QPSK modulated, i.e., $Q=2$, before the transmission over the frequency-selective channel with an impulse response of order $L=4$ (cf. Section II-A). The channel is assumed to be known at the receiver and the *signal-to-noise ratio* (SNR) is defined as $E_\mathrm{b}/N_0 = \varrho_\mathrm{s}/(\sigma_{\boldsymbol{\eta}}^2 rQ) = \varrho_\mathrm{s}/\sigma_{\boldsymbol{\eta}}^2$ with the bit energy E_b and the noise power density N_0. The length of the linear equalizer filter is $K=7$ and the latency time is chosen fixed to $\kappa=5$. Besides, the decoder is implemented using the *Bahl–Cocke–Jelinek–Raviv* (BCJR) algorithm [33]. The following subsections present simulation results for two different channel types.

A. Fixed Channel Simulations

All simulations of this subsection assume $R=1$ antenna at the receiver, i.e., the observation vector $\mathbf{y}[k]$ has dimension $N=KR=7$, and TV filter coefficients. Hence, the computational complexity arises from the fact that the statistics is time-varying

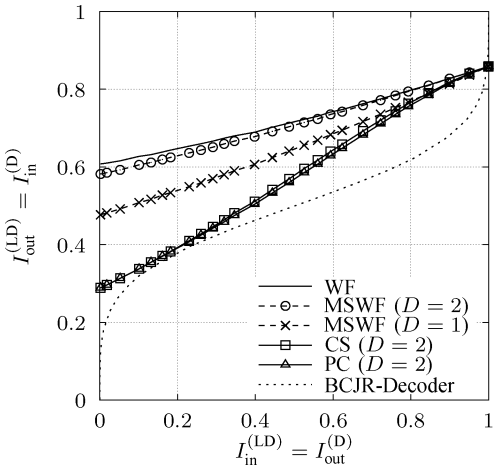

Fig. 6. EXIT chart for Porat channel at $10\log_{10}(E_{\text{b}}/N_0) = 5$ dB.

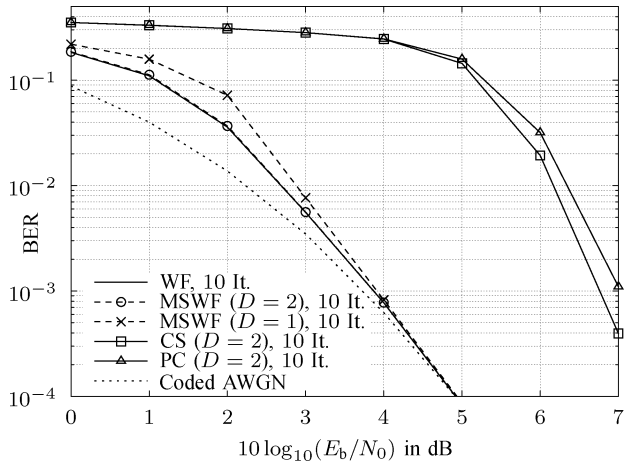

Fig. 7. BER plot for Porat channel.

although the channel is fixed (cf. Section IV-D). First, we investigate the performance of the proposed algorithms when applied to equalize a complex-valued channel defined by Porat *et al.* in [34] which behaves well in terms of interference. With the vector $\boldsymbol{h} = [h_0, h_1, h_2, h_3, h_4]^{\text{T}}$ of channel coefficients, the norm-one Porat channel is given by

$$\boldsymbol{h} = [0.49 + \text{j}\,0.10, 0.36 + \text{j}\,0.44, 0.24, 0.29 - \text{j}\,0.32, 0.19 + \text{j}\,0.39]^{\text{T}} \in \mathbb{C}^5. \quad (47)$$

For the Porat channel, the EXIT chart [35] with the proposed algorithms at $10\log_{10}(E_{\text{b}}/N_0) = 5$ dB is depicted in Fig. 6, where the mutual information $I_{\text{out}}^{(\text{LD})} = I(\mathsf{c}'; \mathsf{l}_{\text{ext}}^{(\text{LD})})$ between the interleaved and coded bits c' at the transmitter and the extrinsic information $\mathsf{l}_{\text{ext}}^{(\text{LD})}$ at the output of the linear detector, is plotted over the mutual information $I_{\text{in}}^{(\text{LD})} = I(\mathsf{c}'; \mathsf{l}_{\text{apr}}^{(\text{LD})})$ between c' and the *a priori* information $\mathsf{l}_{\text{apr}}^{(\text{LD})}$ at the input of the linear detector. Besides, Fig. 6 includes the transfer characteristic of the decoder, i.e., the dependence of the mutual information $I_{\text{out}}^{(\text{D})} = I(\mathsf{c}; \mathsf{l}_{\text{ext}}^{(\text{D})})$ between the coded bits c at the transmitter and the extrinsic information $\mathsf{l}_{\text{ext}}^{(\text{D})}$ at the output of the decoder, on the mutual information $I_{\text{in}}^{(\text{D})} = I(\mathsf{c}; \mathsf{l}_{\text{apr}}^{(\text{D})})$ between c and the *a priori* information $\mathsf{l}_{\text{apr}}^{(\text{D})}$ at the input of the decoder.[10] Note that $I_{\text{out}}^{(\text{LD})} = I_{\text{in}}^{(\text{D})}$ and $I_{\text{in}}^{(\text{LD})} = I_{\text{out}}^{(\text{D})}$ due to the iterative structure of the receiver (cf. Fig. 2) such that we can use the EXIT chart to analyze the convergence behavior of the turbo system [36].

In Fig. 6, it can be seen that all investigated algorithms achieve the same mutual information $I_{\text{out}}^{(\text{LD})}$ in case of full *a priori* knowledge, i.e., $I_{\text{in}}^{(\text{LD})} = 1$. Thus, for a sufficient high number of iterations, they perform equally if the SNR is high enough such that their EXIT curves do not intersect the one of the decoder. Nevertheless, the number of iterations, necessary to achieve the optimum, depends on the curvature of the corresponding EXIT graph and on the mutual information $I_{\text{out}}^{(\text{LD})}$ at $I_{\text{in}}^{(\text{LD})} = 0$ (no *a priori* information available) which corresponds to a coded system with a receiver performing no iterations. Consequently, the WF with the highest $I_{\text{out}}^{(\text{LD})}$ for all $I_{\text{in}}^{(\text{LD})}$ needs the lowest number of iterations. The proposed MSWF with rank $D = 2$ has only a small performance degradation compared to the WF although the computational complexity is reduced tremendously (cf. Section IV-D). Contrary to the Krylov-subspace-based MSWF, the eigensubspace-based PC or CS method with rank $D = 2$ behave even worse than the rank-one MSWF. Thus, in the given scenario, Krylov-subspace-based approximations of the WF should be preferred to WF approximations in eigensubspaces.

The observations made in Fig. 6 can be verified in the BER plot of Fig. 7 where the receiver iterated ten times. The lower bound in Fig. 7 is given by interference free transmission (coded AWGN). It can be seen that the optimal linear WF achieves the BER of the coded AWGN channel for $10\log_{10}(E_{\text{b}}/N_0) > 5$ dB. The MSWF with rank $D = 2$ performs identically to the WF and even the rank-one MSWF has only a slightly higher BER for low SNR values. Recall that the MSWF with $D = 1$ is equal to a normalized MF followed by a scalar WF. Thus, and because of the fact that the normalized MF $\boldsymbol{c}_{\mathsf{y},\mathsf{s}}[k]/\left\|\boldsymbol{c}_{\mathsf{y},\mathsf{s}}[k]\right\|_2$ has by definition TI filter coefficients, its computational complexity is reduced drastically compared with the full-rank WF. Note that contrary to the proposed rank-one MSWF which is used in any turbo iteration, the MF approaches introduced in [23] or [37] have a higher computational complexity since they are applied in a *hybrid manner*, i.e., the first iteration still applies a full-rank WF whereas only the following iterations are based on MF approximations. Contrary to the MSWF, the PC and CS method with $D = 2$ achieve the performance of the WF in Fig. 7 only at very high SNRs. Moreover, compared to the full-rank WF and the reduced-rank MSWFs, the eigensubspace-based algorithms still improve their performance if the number of iterations is further increased. This is due to the bottleneck between their and the decoder's EXIT curve which can be seen in Fig. 6, implicating a very high number of iterations in order to achieve the optimum at $I_{\text{in}}^{(\text{LD})} = 1$.

Finally, Fig. 8 depicts the EXIT chart at $10\log_{10}(E_{\text{b}}/N_0) = 5$ dB if we assume the real-valued Proakis channel [19]

$$\boldsymbol{h} = [0.227, 0.46, 0.688, 0.46, 0.227]^{\text{T}} \in \mathbb{R}^5. \quad (48)$$

[10] The index i used in Section II-B to refer to the position of $\mathsf{c}, \mathsf{c}', \mathsf{l}_{\text{apr/ext}}^{(\text{D})}$, and $\mathsf{l}_{\text{apr/ext}}^{(\text{LD})}$, in the vectors $\mathsf{c}, \mathsf{c}', \mathsf{l}_{\text{apr/ext}}^{(\text{D})}, \mathsf{l}_{\text{apr/ext}}^{(\text{LD})}$, respectively, is omitted here since the mutual information is assumed to be equal for all elements of the corresponding vectors.

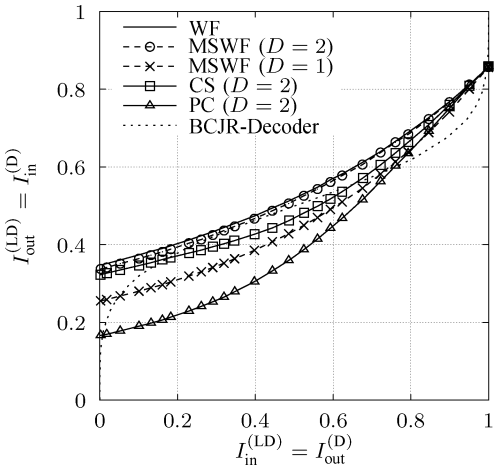

Fig. 8. EXIT chart for Proakis channel at $10\log_{10}(E_{\text{b}}/N_0) = 5$ dB.

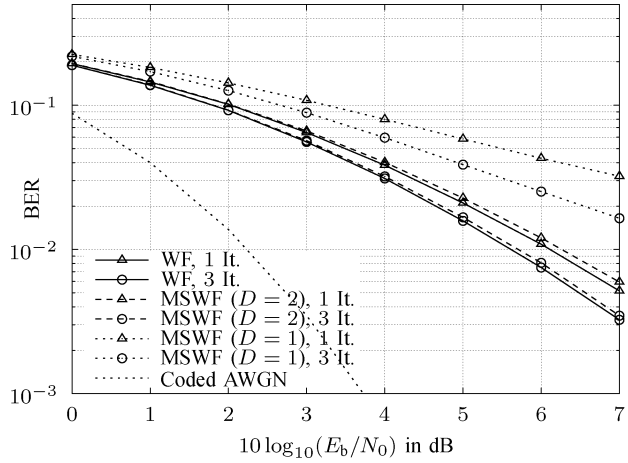

Fig. 9. BER for random channel ($R = 1$) and TV filter coefficients.

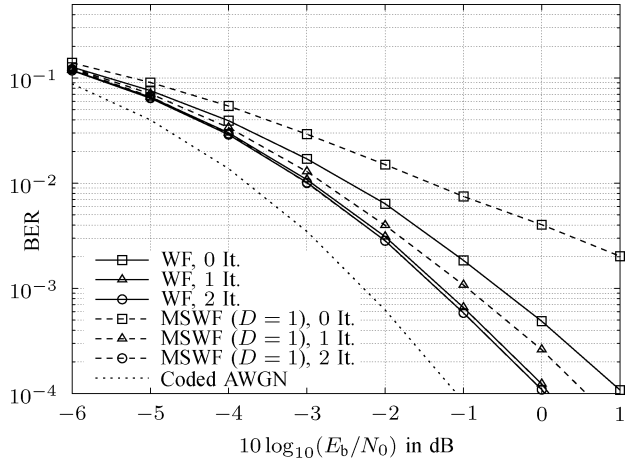

Fig. 10. BER for random channel ($R = 4$) and TV filter coefficients.

Due to its symmetric structure, the transfer function of the Proakis channel has zeros leading to severe ISI. Again, the norm of $\boldsymbol{h}$ is one. Due to the severe ISI, the EXIT curves of all algorithms start at a lower $I_{\text{out}}^{(\text{LD})}$ compared to the curves of Fig. 6 and their curvatures are positive, i.e., the number of iterations needed for convergence is higher compared to a nonpositive curved graph. Again, the optimum can only be reached for high SNR values, e.g., at the investigated log-SNR of $10\log_{10}(E_{\text{b}}/N_0) = 5$ dB, only the WF and the MSWF with rank $D = 2$ do not intersect the decoder curve, thus, achieving optimal performance after several iterations. In case of the Proakis channel, the rank $D = 2$ CS method as the MSE optimal eigensubspace algorithm is better than the rank-one MSWF but performs still worse than the MSWF with rank $D = 2$. Remember that the CS filter is only considered as a performance bound for eigensubspace methods since its computational complexity has the same order than the computational intense WF (cf. Section IV-D). Again, the EXIT curve of the computationally cheap MSWF with rank $D = 2$ is almost equal to the one of the full-rank WF.

B. Random Channel Simulations

In this subsection, we consider a random block-varying channel where each coefficient is taken from a circular complex normal distribution with variance $1/(L+1)$ and held constant during one block. Note that such channels with a uniform power delay profile can be seen as worst case scenarios since they produce more ISI than channels with exponential power delay profiles. Fig. 9 plots the BER versus SNR averaged over several channel realizations if we assume $R = 1$ antenna at the receiver, i.e., $N = KR = 7$, and TV filter coefficients. Note that the averaged BER of the WF does not further improve after three iterations. The BER curves achieve no longer the BER of a coded AWGN channel, even for an infinite number of iterations. Besides, in case of random channels with $R = 1$, the computationally cheap rank-one MSWF no longer reaches the WF performance. Nevertheless, the MSWF with rank $D = 2$ is still very close to the optimal linear filter.

In the sequel of this paper, we consider $R = 4$ antennas at the receiver. With a filter length of $K = 7$ for each antenna, we get an observation vector $\mathbf{y}[k]$ with dimension $N = KR = 28$. Figs. 10 and 11 show the BERs of TV and TI filters, respectively, again averaged over several channel realizations. Compared with the one-antenna case, the optimal linear equalizer achieves no further improvement after only one iteration. Due to the antenna gain, the BERs of the investigated algorithms are shifted about 6 dB to the left. The $D = 1$ MSWF solution with one iteration and a significant reduced complexity is only 0.5 dB away from the optimal curve. Performing an additional iteration leads to the performance of the WF at the cost of a higher computational complexity. Anyway, the $D = 2$ MSWF solution has already the same performance as the optimal solution after one iteration (cf. Fig. 12).

In Fig. 11, it can be seen that despite of their enormous computational efficiency, the performance of the TI filter coefficients is almost equal to the one of the corresponding TV implementation. Especially, in systems with large block sizes, it is recommended to first replace the TV filter coefficients by their TI approximations before applying alternative reduced-complexity methods. Note that the performance of the TI rank-one MSWF is not shown since the complexity reduction compared to its TV implementation is negligible because the time variance of the statistics affects only the scalar WF and not the normalized MF.

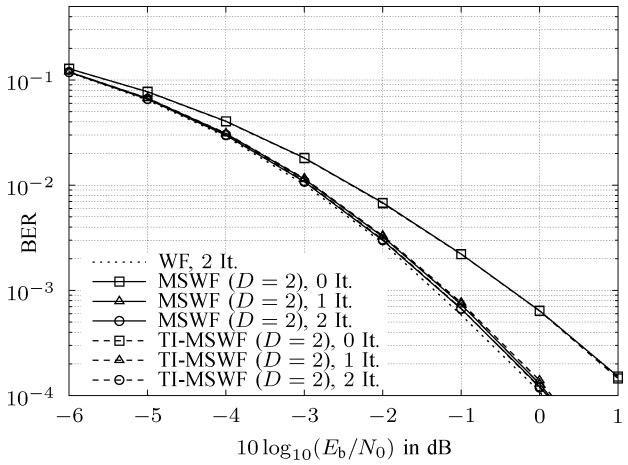

Fig. 11. Comparison to TI filter coefficients for random channel ($R = 4$).

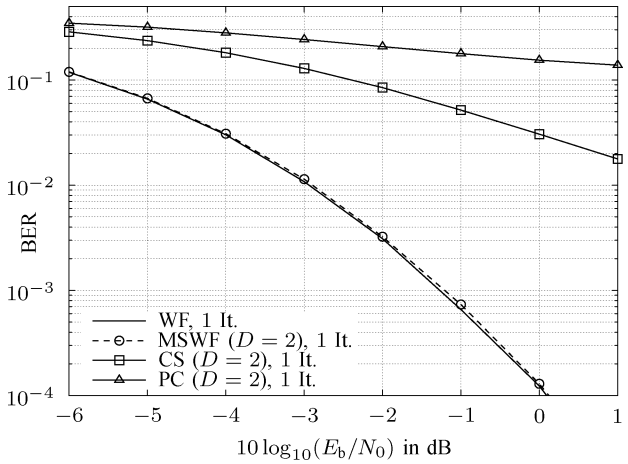

Fig. 12. Comparison to eigensubspace-based methods for random channel ($R = 4$).

Finally, Fig. 12 depicts an averaged BER comparison of the Krylov-subspace-based MSWF to eigensubspace-based methods performing only one turbo iteration. Again, Fig. 12 shows the superiority of the Krylov-subspace-based MSWF compared with the eigensubspace-based PC and CS method. Moreover, it can be seen that the MSWF with rank $D = 2$ has the same performance as the full-rank WF.

VI. Conclusion

In this paper, we have applied the reduced-rank MSWF to the linear detector of an iterative receiver. The analysis based on EXIT charts as well as *Monte Carlo* simulations of coded transmission over frequency-selective channels have shown that the reduced-rank MSWF achieves near optimal performance despite of a tremendous reduction in computational complexity compared with optimal reduced-complexity implementations. Even if the rank of the MSWF is reduced to one which results in an easy-to-compute normalized MF followed by a scalar WF, the performance of the turbo receiver is close to optimum if we assume several antennas at the receiver. Moreover, the MSWF outperforms eigensubspace methods with the same rank.

References

[1] C. Douillard, M. Jézéquel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: Turbo-equalization," *Eur. Trans. Telecommun.*, vol. 6, no. 5, pp. 507–511, Sep./Oct. 1995.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes (1)," in *Proc. 1993 IEEE Int. Conf. Communications*, May 1993, pp. 1064–1070.

[3] A. Glavieux, C. Laot, and J. Labat, "Turbo equalization over a frequency selective channel," in *Proc. Int. Symp. Turbo Codes Related Topics*, Sep. 1997, pp. 96–102.

[4] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, pp. 1046–1061, Jul. 1999.

[5] G. Bauch and N. Al-Dhahir, "Reduced-complexity space-time turbo-equalization for frequency-selective MIMO channels," *IEEE Tran. Wireless Commun.*, vol. 1, pp. 819–828, Oct. 2002.

[6] C. Laot, R. Le Bidan, and D. Leroux, "Low-complexity MMSE turbo equalization: A possible solution for EDGE," *IEEE Trans. Wireless Commun.*, vol. 4, pp. 965–974, May 2005.

[7] B. Dong and X. Wang, "Sampling-based soft equalization for frequency-selective MIMO channels," *IEEE Trans. Commun.*, vol. 53, pp. 278–288, Feb. 2005.

[8] M. Tüchler, A. C. Singer, and R. Koetter, "Minimum mean squared error equalization using a-priori information," *IEEE Trans. Signal Process.*, vol. 50, pp. 673–683, Mar. 2002.

[9] C. Laot, A. Glavieux, and J. Labat, "Turbo equalization: Adaptive equalization and channel decoding jointly optimized," *IEEE J. Sel. Areas Commun.*, vol. 19, pp. 1744–1752, Sep. 2001.

[10] M. L. Honig, G. K. Woodward, and Y. Sun, "Adaptive iterative multiuser decision feedback detection," *IEEE Trans. Wireless Commun.*, vol. 3, pp. 477–485, Mar. 2004.

[11] J. S. Goldstein, I. S. Reed, and L. L. Scharf, "A multistage representation of the Wiener filter based on orthogonal projections," *IEEE Trans. Inf. Theory*, vol. 44, pp. 2943–2959, Nov. 1998.

[12] M. L. Honig and W. Xiao, "Performance of reduced-rank linear interference suppression," *IEEE Trans. Inf. Theory*, vol. 47, pp. 1928–1946, Jul. 2001.

[13] M. Joham, Y. Sun, M. D. Zoltowksi, M. Honig, and J. S. Goldstein, "A new backward recursion for the multi-stage nested Wiener filter employing Krylov subspace methods," in *Proc. Milcom 2001*, Oct. 2001, vol. 2, pp. 1210–1213.

[14] C. Lanczos, "An iteration method for the solution of the eigenvalue problem of linear differential and integral operators," *J. Res. Nat. Bureau Standards*, vol. 45, no. 4, pp. 255–282, Oct. 1950.

[15] Y. Saad, *Iterative Methods for Sparse Linear Systems*. Philadelphia, PA: SIAM, 2003.

[16] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *J. Educ. Psychol.*, vol. 24, no. 6/7, pp. 417–441, 498–520, Sep./Oct. 1933.

[17] J. S. Goldstein and I. S. Reed, "Reduced-rank adaptive filtering," *IEEE Trans. Signal Process.*, vol. 45, pp. 492–496, Feb. 1997.

[18] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inf. Theory*, vol. 42, pp. 429–445, Mar. 1996.

[19] J. G. Proakis, *Digital Communications*. New York: McGraw-Hill, 1995.

[20] L. L. Scharf, *Statistical Signal Processing*. Reading, MA: Addison-Wesley, 1991.

[21] M. Joham, W. Utschick, and J. A. Nossek, "Latency time optimization for FIR and block transmit filters," in *Proc. 2003 Int. Symp. Signal Processing Its Applications*, Jul. 2003, vol. 1, pp. 273–276.

[22] F. Dietrich, G. Dietl, M. Joham, and W. Utschick, "Robust and reduced rank space-time decision feedback equalization," in *Smart Antennas—State of the Art*, ser. EURASIP Book Series on Signal Processing & Communications. New York: Hindawi Publishing Corp., 2005, ch. 10, Part I: Receiver.

[23] M. Tüchler, R. Koetter, and A. C. Singer, "Turbo equalization: Principles and new results," *IEEE Trans. Commun.*, vol. 50, pp. 754–767, May 2002.

[24] G. Dietl, C. Mensing, and W. Utschick, "Iterative detection based on reduced-rank equalization," in *Proc. 2004 IEEE 60th Vehicular Technology Conf.*, Sep. 2004, vol. 3, pp. 1533–1537.

[25] M. E. Austin, "Decision-feedback equalization for digital communication over dispersive channels," MIT Lincoln Laboratory, Lexington, MA, Tech. Rep. 437, 1967.

[26] G. Dietl, C. Mensing, W. Utschick, J. A. Nossek, and M. D. Zoltowski, "Multi-stage MMSE decision feedback equalization for EDGE," in *Proc. 2003 Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, Apr. 2003, vol. 4, pp. 509–513.
[27] A. Gersho and T. L. Lim, "Adaptive cancellation of intersymbol interference for data transmission," *Bell Syst. Tech. J.*, vol. 60, no. 11, pp. 1997–2021, Nov. 1981.
[28] M. S. Mueller and J. Salz, "A unified theory of data-aided equalization," *Bell Syst. Tech. J.*, vol. 60, no. 9, pp. 2023–2038, Nov. 1981.
[29] D. Reynolds and X. Wang, "Low-complexity turbo-equalization for diversity channels," *Signal Process.*, vol. 81, no. 5, pp. 989–995, 2001.
[30] G. Dietl and W. Utschick, "On reduced-rank approaches to matrix Wiener filters in MIMO systems," presented at the 2003 Int. Symp. Signal Processing Information Technology, Darmstadt, Germany, Dec. 14–17, 2003.
[31] L. N. Trefethen and D. Bau, III*, Numerical Linear Algebra*. Philadelphia, PA: SIAM, 1997.
[32] G. Golub and C. V. Loan*, Matrix Computations*. Baltimore, MD: The Johns Hopkins Univ. Press, 1996.
[33] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inf. Theory*, vol. IT-20, pp. 284–287, Mar. 1974.
[34] B. Porat and B. Friedlander, "Blind equalization of digital communications channels using high-order moments," *IEEE Trans. Signal Process.*, vol. 39, pp. 522–526, Feb. 1991.
[35] R. Otnes and M. Tüchler, "Exit chart analysis applied to adaptive turbo equalization," presented at the Nordic Signal Processing Symp., Trondheim, Norway, Oct. 2002.
[36] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, pp. 1727–1737, Oct. 2001.
[37] H. Omori, T. Asai, and T. Matsumoto, "A matched filter approximation for SC/MMSE iterative equalizers," *IEEE Commun. Lett.*, vol. 5, pp. 310–312, Jul. 2001.

Guido Dietl (S'01) received the Diploma and Ph.D. degrees (both *summa cum laude*) in electrical engineering from the Munich University of Technology (TUM), Munich, Germany, in 2001 and 2006, respectively.

Since 2001, he has been with the TUM, where he is currently working as a Research Engineer at the Associate Institute for Signal Processing. In winter 2000–2001 and summer 2004, he was a Guest Researcher at Purdue University, West Lafayette, IN. His main research interests are numerical linear algebra, reduced-rank signal processing, and iterative detection in wireless communications.

Dr. Dietl received the VDE Preis for his diploma thesis in 2001.

Wolfgang Utschick (M'97–SM'06) completed several industrial education programs before he received the diploma and doctoral degrees (both with honors) in electrical engineering from Munich University of Technology (TUM), Munich, Germany, in 1993 and 1998.

In 1993, he became a part-time Lecturer at a Technical School for Industrial Education. From 1998 to 2002, he was head of the Signal Processing Group at the Institute of Circuit Theory and Signal Processing at the TUM. He was a Guest Researcher at the ETH Zurich, Switzerland, in 2000, and since 2000 he has been instrumental in the 3rd Generation Partnership Project as an academic consultant in the field of multi-element antenna wireless communication systems. In October 2002, he was appointed Professor at the Department of Electrical Engineering of the TUM, where he is head of the Associate Institute for Signal Processing. He completed a biennial program for professional academic teaching and gives national and international undergraduate and graduate courses in the field of signal processing and communication systems. His research interests are statistical array signal processing, spectral theory, multiuser communication systems, multicriterion optimization theory, and kernel methods for approximation and detection.

Dr. Utschick held a scholarship of the Bavarian Ministry of Education for exceptional students and a scholarship of the Siemens AG from 1993 to 1998. He is senior member of the VDE/ITG and is instrumental as an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS.