

PRECONDITIONED AND RANK-FLEXIBLE BLOCK CONJUGATE GRADIENT IMPLEMENTATIONS OF MIMO WIENER DECISION FEEDBACK EQUALIZERS

Ingmar Groh*, Guido Dietl, and Wolfgang Utschick

Munich University of Technology (TUM),
Institute for Circuit Theory and Signal Processing,
80290 Munich, Germany
E-mail: {dietl, utschick}@tum.de

ABSTRACT

In this paper, we present two extensions of the *Block Conjugate Gradient* (BCG) algorithm, a method which exploits the concept of *block Krylov subspaces*. First, we extend the BCG algorithm such that it is more flexible concerning the dimension of the block Krylov subspace. Second, a computationally efficient *Preconditioned* BCG (PBCG) algorithm is introduced which turns out to outperform the standard BCG algorithm concerning the complexity-performance ratio. Hence, we provide a powerful implementation for reduced-rank signal processing in the *Minimum Mean Square Error* (MMSE) sense. Simulation results show the gain in rank-flexibility and convergence speed.

1. INTRODUCTION

The *Decision-Feedback Equalizer* (DFE) [1] structure is an extension of the principle of linear receive filters comprising both a *FeedForward* (FF) and a *FeedBack* (FB) filter. For our investigations, the filters are designed according to the MMSE criterion. The calculation of the FF and FB filter turns out to be computationally intense for observations of high dimensionality. Reduced-rank methods approximate the filter coefficients in a lower dimensional subspace, and thus, decrease computational complexity. A solution algorithm for the approximation of the FF and FB filter in a block Krylov subspace is the *Block Conjugate Gradient* (BCG) algorithm [2]. Up to now, the application of the BCG algorithm required the dimension of the block Krylov subspace to be an integer multiple of the signal dimension. In this paper, we modify the BCG algorithm such that this restriction is obsolete. The importance of this rank-flexibility is that it avoids zero-filling which would yield an unnecessary increase in computational complexity. The second contribution consists of the performance analysis of the *Preconditioned* BCG (PBCG) algorithm. Preconditioners have already been applied for the approximation of reduced-rank *Single Input Single Output* (SISO) Wiener DFEs [3]. Furtheron, the PBCG implementation in [4] is endowed with rank-flexibility, too. In spite of

causing an additional computational effort for the implementation, the PBCG outperforms the BCG algorithm concerning the performance-complexity ratio. Section 2 presents the channel model, Section 3 contains the DFE receiver structure together with the derivation of the Wiener DFE. Our main contributions are the extensions of the BCG algorithm in Section 4. In order to be able to judge the simulation results correctly, we will provide a complexity analysis of all BCG implementations. Throughout the paper, vectors and matrices are denoted by lower and upper case bold letters. In order to distinguish deterministic and random variables, we use *sans serif* font for vectors and matrices containing random entries. The matrix $\mathbf{I}_n$ is the $n \times n$ identity matrix, and $\mathbf{0}_{n \times m}$ the $n \times m$ zero matrix. $\mathbf{R}_u = \mathbb{E}\{\mathbf{u}\mathbf{u}^H\}$ denotes the auto-correlation matrix of a random vector $\mathbf{u}$ and $\mathbf{R}_{u,v} = \mathbb{E}\{\mathbf{u}\mathbf{v}^H\}$ the cross-correlation matrix between the vectors $\mathbf{u}$ and $\mathbf{v}$. The operation $(\cdot)_{ij}$ denotes access to the (i, j) -th entry of a matrix and $[\cdot]_{ij}$ access to the (i, j) -th block of a block matrix. Hence, the operation $[\cdot]_{i,j=1}^R$ builds up a block matrix where the indices i, j run from 1 to R . When applied to a diagonal matrix, $(\cdot)_k$ extracts the (k, k) -th element.

2. CHANNEL MODEL

In this paper, we consider the frequency-selective MIMO channel of length L with T transmitters and R receivers to be described by the channel matrix impulse response $\mathbf{H}[n] = \sum_{\ell=0}^{L-1} \mathbf{H}_\ell \delta[n - \ell] \in \mathbb{C}^{R \times T}$ with the unit impulse function $\delta[n]$. The weighting matrices of the propagation paths, $\mathbf{H}_\ell$, $\ell \in \{0, 1, \dots, L-1\}$, are realizations of the random matrix $\mathbf{H}$ having i. i. d. entries with $\mathcal{N}_c(0, 1/L)$. The receive vector

$$\mathbf{r}[n] = \mathbf{H}[n] * \mathbf{s}[n] + \boldsymbol{\eta}[n] \in \mathbb{C}^R \quad (1)$$

(‘ $*$ ’ denotes convolution) is perturbed by additive white Gaussian noise $\boldsymbol{\eta}[n] \in \mathbb{C}^R$ with the complex normal distribution $\mathcal{N}_c(\mathbf{0}_{R \times 1}, \sigma^2 \mathbf{I}_R)$. The transmit signal vector $\mathbf{s}[n]$ at time index n composes of T zero-mean i. i. d. symbols with variance one, yielding a total transmit power of $P_{\text{Tx}} = \text{tr}\{\mathbf{R}_s\} = T$. In order to compute the equalizer filters of length K (cf. Section 3), we derive an alternative matrix-vector model of the time-dispersive MIMO channel. The vector $\tilde{\mathbf{r}}[n] = [\mathbf{r}^T[n], \mathbf{r}^T[n-1], \dots, \mathbf{r}^T[n-K+1]]^T \in$

*Now with the German Aerospace Center (DLR), Institute of Communications and Navigation, 82234 Wessling, Germany.

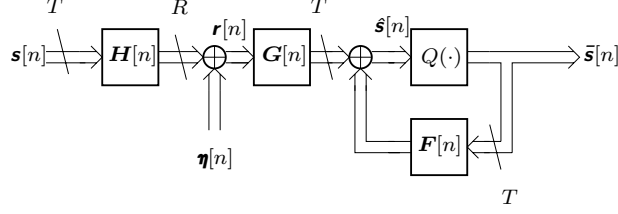


Fig. 1. Structure of the System With MIMO DFE

$\mathbb{C}^{KR}$ is composed of K adjacent receive vectors $\mathbf{r}[n]$. Using the block Toeplitz matrix $\tilde{\mathbf{H}} = \sum_{\ell=0}^{L-1} \mathbf{S}_{(\ell,K,L-1)} \otimes \mathbf{H}_\ell \in \mathbb{C}^{KR \times (K+L-1)T}$ ($\otimes$ denotes the Kronecker product and $\mathbf{S}_{(\ell,K,L-1)} = [\mathbf{0}_{K \times \ell}, \mathbf{I}_K, \mathbf{0}_{K \times (L-1-\ell)}] \in \{0,1\}^{K \times (K+L-1)}$ the selection matrix), Eq. (1) is equivalent to

$$\tilde{\mathbf{r}}[n] = \tilde{\mathbf{H}} \tilde{\mathbf{s}}[n] + \tilde{\boldsymbol{\eta}}[n] \in \mathbb{C}^{KR}. \quad (2)$$

Analogous to $\tilde{\mathbf{r}}[n]$, the vector $\tilde{\mathbf{s}}[n] \in \mathbb{C}^{(K+L-1)T}$ is composed of $K+L-1$ adjacent signal vectors $\mathbf{s}[n]$ and $\tilde{\boldsymbol{\eta}}[n] \in \mathbb{C}^{KR}$ of K adjacent noise vectors $\boldsymbol{\eta}[n]$.

3. WIENER DFE MODEL

In this section, we present the DFE structure in more detail (cf. Fig. 1) [3]. In contrast to linear receive filters, the DFE is endowed with a FF filter $\mathbf{G}[n] = \sum_{g=0}^{G-1} \mathbf{G}_g \delta[n-g] \in \mathbb{C}^{T \times R}$ as well as a FB filter $\mathbf{F}[n] = \sum_{f=1}^F \mathbf{F}_f \delta[n-f] \in \mathbb{C}^{T \times T}$. The estimate $\hat{\mathbf{s}}[k] \in \mathbb{C}^T$ at the quantizer input states

$$\hat{\mathbf{s}}[n] = \mathbf{G}[n] * \mathbf{r}[n] + \mathbf{F}[n] * \tilde{\mathbf{s}}[n], \quad (3)$$

where the vector $\tilde{\mathbf{s}}[n] \in \mathbb{C}^T$ denotes the estimated signal vector $\hat{\mathbf{s}}[n]$ after quantization. As we have already seen in Section 2, it is necessary to substitute the convolution representation of the DFE structure in Eq. (3) by adequate matrix-vector representations. If we define $\mathbf{G} = [\mathbf{G}_0, \dots, \mathbf{G}_{G-1}] \in \mathbb{C}^{T \times RG}$ and $\mathbf{F} = [\mathbf{F}_1, \dots, \mathbf{F}_F] \in \mathbb{C}^{T \times TF}$, we obtain

$$\hat{\mathbf{s}}[n] = \mathbf{G} \tilde{\mathbf{r}}[n] + \mathbf{F} \mathbf{S}_{(T(\nu+1), TF, T(\nu+1))} \tilde{\mathbf{s}}[n]. \quad (4)$$

Note, however, that the matrix-vector representation in Eq. (4) requires the FF and FB length to be $G = K$ and $F = L + K - \nu - 2$, respectively, and exploits the key assumption that the decisions of the DFE are correct with delay ν , i.e., $\tilde{\mathbf{s}}[k] = \mathbf{s}[k - \nu] \in \mathbb{C}^T$. Hence, the error vector $\mathbf{e}[k] \in \mathbb{C}^T$ can be written as

$$\mathbf{e}[n] = \tilde{\mathbf{s}}[n] - \hat{\mathbf{s}}[n] = \mathbf{s}[n - \nu] - \mathbf{W} \mathbf{u}[n] \in \mathbb{C}^T, \quad (5)$$

where we make use of the abbreviations $\mathbf{W} := [\mathbf{G}, \mathbf{F}] \in \mathbb{C}^{T \times (TF+RG)}$, $\tilde{\mathbf{s}}_\nu[n] := \mathbf{S}_{(T(\nu+1), TF, T(\nu+1))} \tilde{\mathbf{s}}[k] \in \mathbb{C}^{TF}$, and $\mathbf{u}[n] := [\tilde{\mathbf{r}}^T[n], \tilde{\mathbf{s}}_\nu^T[n]]^T \in \mathbb{C}^{TF+RG}$. The filter $\mathbf{W}$ is chosen to solve the optimization problem

$$\mathbf{W} = \underset{\mathbf{W}}{\operatorname{argmin}} \xi(\mathbf{W}) := \underset{\mathbf{W}}{\operatorname{argmin}} \mathbb{E} \{ \|\mathbf{e}[n]\|_2^2 \}, \quad (6)$$

whose solution $\mathbf{W}$ is determined by [5, 6]

$$\mathbf{R}_u \mathbf{W}^H = \begin{bmatrix} \mathbf{R}_{\tilde{\mathbf{r}}} & \mathbf{R}_{\tilde{\mathbf{r}}, \tilde{\mathbf{s}}_\nu} \\ \mathbf{R}_{\tilde{\mathbf{s}}_\nu, \tilde{\mathbf{r}}} & \mathbf{R}_{\tilde{\mathbf{s}}_\nu} \end{bmatrix} \mathbf{W}^H = \mathbf{R}_{u, \mathbf{s}}. \quad (7)$$

4. REDUCED-RANK IMPLEMENTATIONS OF THE WIENER DFE USING BCG IMPLEMENTATIONS

For sake of simplicity, we introduce the following abbreviations for the matrices $\mathbf{R}_u \in \mathbb{C}^{(TF+RG) \times (TF+RG)}$, $\mathbf{R}_{u, \mathbf{s}} \in \mathbb{C}^{(TF+RG) \times T}$, and $\mathbf{W}: \mathbf{R}_u \leftrightarrow \mathbf{A}, \mathbf{R}_{u, \mathbf{s}} \leftrightarrow \mathbf{B}, \mathbf{W}^H \leftrightarrow \mathbf{X}$, and we define $N = TF + RG, M = T$. A block Krylov subspace [7] defined as

$$\mathcal{K}_{\text{block}}^{(D)}(\mathbf{A}, \mathbf{B}) := \operatorname{span}\{[\mathbf{B}, \dots, \mathbf{A}^{d-2} \mathbf{B}, \mathbf{A}^{d-1} \mathbf{b}_1, \dots, \mathbf{A}^{d-1} \mathbf{b}_\mu]\}, \quad (8)$$

where $d+1 = \lceil D/M \rceil$, $\mu = D - dM$, and $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_M] \in \mathbb{C}^{N \times M}$ will be chosen as the D -dimensional subspace for determining the reduced-rank Wiener DFE. The reduced-rank BCG implementations minimize the cost function [2, 6] $f(\mathbf{X}) = \operatorname{tr} \{ \mathbf{X}^H \mathbf{A} \mathbf{X} - 2 \operatorname{Re} \{ \mathbf{X}^H \mathbf{B} \} \}$ iteratively, i.e., they solve the optimization problem

$$\mathbf{X} = \underset{\mathbf{X}}{\operatorname{argmin}} f(\mathbf{X}) \quad \text{s.t.} \quad \mathbf{X} \in \mathcal{K}_{\text{block}}^{(D)}(\mathbf{A}, \mathbf{B}). \quad (9)$$

Note that $f(\mathbf{X})$ is the same cost function as $\xi(\mathbf{W})$ in Eq. (6) apart from an additive constant.

4.1. Rank-Flexible Implementation of the BCG Algorithm

The standard BCG algorithm [2] generates a sequence of approximations $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_d$ for the solution of the optimization problem in Eq. (9). The implementation of the standard BCG algorithm in [2] required the dimension D of the block Krylov subspace to be an integer multiple of M . If this restriction shall be obsolete now (cf. the definition of $\mathcal{K}_{\text{block}}^{(D)}(\mathbf{A}, \mathbf{B})$ in Eq. (8), the BCG algorithm only needs to be modified when performing the last update $\mathbf{X}_d \rightarrow \mathbf{X}_{d+1}$. The transition from $\mathbf{X}_k$ to the next approximation $\mathbf{X}_{k+1}$, $k = 1, \dots, d-1$, is performed using the update rule [2]

$$\mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{D}_k \boldsymbol{\Phi}_k, k = 0, 1, \dots, d-1, \quad (10)$$

where $\mathbf{X}_{k+1}, \mathbf{X}_k, \mathbf{D}_k \in \mathbb{C}^{N \times M}$, and $\boldsymbol{\Phi}_k \in \mathbb{C}^{M \times M}$. The matrices $\mathbf{D}_k$ represent the searching directions along which the function $f(\mathbf{X})$ in Eq. (9) is minimized. The searching direction $\mathbf{D}_k$ of the k -th step and the optimum following searching direction $\mathbf{D}_{k+1}$ are $\mathbf{A}$ -conjugate, i.e., $\mathbf{D}_{k+1}^H \mathbf{A} \mathbf{D}_k = \mathbf{0}_{M \times M}$. Because of the modified structure of the block Krylov subspace $\mathcal{K}_{\text{block}}^{(D)}(\mathbf{A}, \mathbf{B})$, it is necessary to adjust the update rule for the last step of the optimization as $\mathbf{X}_{d+1} = \mathbf{X}_d + \mathbf{D}_{d+1} \boldsymbol{\Phi}_d$ where $\mathbf{X}_{d+1}, \mathbf{X}_d \in \mathbb{C}^{N \times M}$, $\mathbf{D}_{d+1} \in \mathbb{C}^{N \times \mu}$, and $\boldsymbol{\Phi}_d \in \mathbb{C}^{\mu \times M}$. In order to determine the optimum value for $\boldsymbol{\Phi}_d$, we plug the update rule of Eq. (10) into $f(\mathbf{X})$ of Eq. (9) and set the differentiation with respect to $\boldsymbol{\Phi}_k$ to zero yielding

$$\boldsymbol{\Phi}_d = (\mathbf{D}_{d+1}^H \mathbf{A} \mathbf{D}_{d+1})^{-1} \mathbf{D}_{d+1}^H \mathbf{R}_d. \quad (11)$$

Algorithm 1 shows the rank-flexible implementation of the BCG algorithm where the previous restriction that D is an integer multiple of M is obsolete. In Line 12 of Algorithm 1, the assignment $D_{d+1} \leftarrow D_d(:, 1 : \mu)$ defines the new searching matrix $D_{d+1} \in \mathbb{C}^{N \times \mu}$ as the first μ columns of the preceding searching matrix $D_d \in \mathbb{C}^{N \times M}$.

Algorithm 1 Rank-Flexible Version of the BCG Algorithm

```

Choose any  $X_0 \in \mathbb{C}^{N \times M}$ 
2:  $R_0, D_0 \leftarrow B - AX_0$ 
 $d \leftarrow \lceil D/M \rceil, \mu \leftarrow D - (d-1)M$ 
4: for  $k = 0, 1, \dots, d-1$  do
     $\Phi_k \leftarrow (D_k^H AD_k)^{-1} D_k^H R_k$ 
6:  $X_{k+1} \leftarrow X_k + D_k \Phi_k$ 
     $R_{k+1} \leftarrow R_k - AD_k \Phi_k$ 
8:  $\Psi_k \leftarrow (R_k^H R_k)^{-1} R_{k+1}^H R_{k+1}$ 
     $D_{k+1} \leftarrow R_{k+1} + D_k \Psi_k$ 
10: end for
 $D_{d+1} \leftarrow D_d(:, 1 : \mu)$ 
12:  $\Phi_d \leftarrow (D_{d+1}^H AD_{d+1})^{-1} D_{d+1}^H R_d$ 
 $X_{d+1} \leftarrow X_d + D_{d+1} \Phi_d$ 

```

4.2. Preconditioned BCG (PBCG) Algorithm

If we apply the proposed channel model in Section 2 to the Wiener DFE computation in Section 3, we observe that the matrix $R_{\tilde{r}} = \tilde{H}\tilde{H}^H + \sigma^2 I_{KR} =: A_{BT} \in \mathbb{C}^{RG \times RG}$ has *Block Toeplitz* (BT) structure. The preconditioning technique for BT systems [4], however, cannot be applied straightforward to $R_{\tilde{r}} = A_{BT}$ because preconditioning matrices can only be designed for Toeplitz matrices. Some preparatory transformations are needed.

For the derivation of the PBCG algorithm, we present the implementation of the PBCG algorithm for the BT system of equations, $A_{BT}Y = \tilde{B}$, $[A_{BT}]_{ij} = A_{j-i} \in \mathbb{C}^{R \times R}$, where A_{j-i} stands for a general matrix which is repeated in Toeplitz fashion, and $Y, \tilde{B} \in \mathbb{C}^{RG \times T}$ are the solution and right-hand side of the BT system of equations, respectively. Instead of $A_{BT}Y = \tilde{B}$, we consider the permuted system $T_{TB}X = B'$, where $T_{TB} = M^T A_{BT} M$, $X = M^T Y$, and $B' = M^T \tilde{B}$, using the permutation matrix

$$M = \begin{bmatrix} I_R \otimes e_1^T \\ \vdots \\ I_R \otimes e_G^T \end{bmatrix} \in \{0, 1\}^{RG \times RG} \quad (12)$$

($e_i \in \{0, 1\}^G$, $i = 1, \dots, G$, denotes the i -th unity vector of dimension G). Thereby, the system describes the conversion to a *Toeplitz Block* (TB) system. The so-called *Chan preconditioner* [8, 4] $C_F(T) \in \mathbb{C}^{G \times G}$ is defined as the circulant matrix $C = C_F(T)$, $(C)_{ij} = c_{j-i+G \text{ modulo } G}$, dependent on the Toeplitz matrix $(T)_{ij} = t_{i-j}$, $i, j = 1, \dots, G$, by

$$c_j = \begin{cases} \frac{j t_{j-G+(G-j)t_j}}{G}, & 0 \leq j < G, \\ c_{G+j}, & 0 < -j < G, \end{cases} \quad (13)$$

yielding those cyclic matrix $C_F(T)$ minimizing $\|C - T\|_F$. For a TB matrix $T_{TB} \in \mathbb{C}^{RG \times RG}$, we apply the construction of Eq. (13) to every block: $[C_F^{TB}]_{ij} =: C_F(T_{ij})$, $i, j = 1, \dots, R$, and $T_{ij} = [T]_{ij}$. Every iteration of the PBCG algorithm requires the determination of the preconditioned solution V from $C_F^{TB}V = Y$, $V, Y \in \mathbb{C}^{RG \times T}$. Of course, the circulant structure of the individual blocks of C_F^{TB} enables a computational cheap inversion using the *Fast Fourier Transform* (FFT). Since $C_F(T_{ij}) \in \mathbb{C}^{G \times G}$, $i, j = 1, \dots, R$, is circulant, we get from the diagonalization property $C_F(T_{ij}) = F^* \Lambda_{ij} F$, i.e. $\Lambda_{ij} \in \mathbb{C}^{G \times G}$ is a diagonal matrix containing all G eigenvalues of $C_F(T_{ij})$. By means of the definition $\tilde{F} := I_R \otimes F \in \mathbb{C}^{RG \times RG}$ ($F \in \mathbb{C}^{G \times G}$ denotes the G -dimensional Fourier matrix with $(F)_{k\ell} = \exp(-j2\pi(k-1)(\ell-1)/G)$, $k, \ell = 1, \dots, G$), we get

$$C_F^{TB} = [C_F(T_{ij})]_{i,j=1}^R = \tilde{F}^* [\Lambda_{ij}]_{i,j=1}^R \tilde{F}. \quad (14)$$

In the following, we make use of the matrix $D_k \in \mathbb{C}^{R \times R}$, $k = 1, \dots, G$, defined by $(D_k)_{ij} = (\Lambda_{ij})_k$, $i, j = 1, \dots, R$. The successive application of the permutation matrix M from the left and right hand side collapses these diagonal matrices to a block diagonal matrix. The whole Toeplitz block matrix can be decomposed into $C_F^{TB} = \tilde{F}^* M^T \text{diag}\{D_1, \dots, D_G\} M \tilde{F}$. Hence, the solution V to the linear system $C_F^{TB}V = Y$ can be computed via

$$V = \tilde{F}^* \left\{ M^T \left[\text{diag}\{D_1^{-1}, \dots, D_G^{-1}\} M \tilde{F} Y \right] \right\}. \quad (15)$$

Eq. (15) enables an efficient calculation of S_k , $k = 0, \dots, d-1$, in Lines 4 and 9 of Algorithm 2 and is therefore the key tool for decreasing complexity. Again, the crucial

Algorithm 2 Rank-Flexible Version of the PBCG Algorithm

```

Choose any  $X_0 \in \mathbb{C}^{N \times M}$ 
2:  $A \leftarrow \tilde{M}^T A M, B \leftarrow \tilde{M}^T B$ 
 $d \leftarrow \lceil D/M \rceil, \mu \leftarrow D - (d-1)M$ 
4:  $R_0 \leftarrow B - AX_0, D_0, S_0 \leftarrow C_F^{TB, -1} R_0$ 
for  $k = 0, \dots, d-1$  do
6:  $\Phi_k \leftarrow (D_k^H AD_k)^{-1} S_k^H R_k$ 
 $X_{k+1} \leftarrow X_k + D_k \Phi_k$ 
8:  $R_{k+1} \leftarrow R_k - AD_k \Phi_k$ 
 $S_{k+1} \leftarrow C_F^{TB, -1} R_{k+1}$ 
10:  $\Psi_k \leftarrow (R_k^H S_k)^{-1} R_{k+1}^H S_{k+1}$ 
 $D_{k+1} \leftarrow S_{k+1} + D_k \Psi_k$ 
12: end for
 $D_{d+1} \leftarrow D_d(:, 1 : \mu), S_{d+1} \leftarrow S_d(:, 1 : \mu)$ 
14:  $\Phi_d \leftarrow (D_{d+1}^H AD_{d+1})^{-1} S_{d+1}^H R_d$ 
 $X_{d+1} \leftarrow \tilde{M} (X_d + D_{d+1} \Phi_d)$ 

```

matrix $R_{\tilde{r}} = \tilde{H}\tilde{H}^H + \sigma^2 I_{KR}$ (cf. Eq. 7) has BT structure. Thus, the necessary conversion of $R_{\tilde{r}}$ from a BT matrix to a TB matrix has to be performed using (cf. Algorithm 2)

$$\tilde{M} = \begin{bmatrix} M & \mathbf{0}_{RG \times TF} \\ \mathbf{0}_{TF \times RG} & I_{TF} \end{bmatrix} \in \{0, 1\}^{(TF+RG) \times (TF+RG)},$$

Complexity Analysis for the BCG Algorithm in Flops	
Line 5	$N(2N - 1)M + 2(2N - 1)M^2 + (2 + \frac{1}{3})M^3$
Line 6	$NM + (2M - 1)NM$
Line 7	$(2N - 1)M^2 + NM$
Line 8	$M^2(2N - 1) + (2 + \frac{1}{3})M^3$
Line 9	$(2M - 1)NM + NM$

Table 1. Computational Complexity of the BCG Algorithm

Calculation of the Preconditioner	
R^2 preconditioners	$5GR^2$
G Cholesky factorizations	$\frac{1}{3}GR^3$
calculation of $\mathbf{D}_k _{k=1}^G$	$\frac{3}{2}R^2G \log_2 G$
Calculation in Every Iteration Step	
FFT and IFFT in Eq. (15)	$2(\frac{3}{2}G \log_2 G)RM$
GM for-/backward substitutions	$2GR^2M$

Table 2. Computational Complexity of the PBCG Algorithm in Flops Needed in Addition to the Complex Flops in Tab. 1

where $\mathbf{M} \in \{0, 1\}^{RG \times RG}$ is defined as in Eq. (12).

5. COMPLEXITY COMPARISON

This section will give the resulting complexity analysis of the presented reduced-rank equalization algorithms considering only terms of main order. Every complex addition (subtraction) as well as every complex multiplication (division) is assumed to have the computational complexity of one *floating-point operation* (flop). Tabs. 1 and 2 sum up the complexity of one iteration step of the BCG algorithm and the complexity needed additionally for the extension to Algorithm 2.

6. SIMULATION RESULTS

We consider QPSK transmission over a frequency-selective MIMO channel of length $L = 3$ with $T = 4$ transmitters and $R = 8$ receivers. The latency time of the frequency-selective channel is set to $\nu = L - 1 = 2$. Hence, if we choose the length of the FF filter to $G = K = 4$, the FB filter must have length $F = L + G - \nu - 2 = G - 1 = 3$. Fig. 2 depicts the uncoded *Bit Error Rates* (BERs) over *Signal to Noise Ratio* (SNR) $10 \log_{10}(P_{Tx}/\sigma^2)$ of the different reduced-rank Wiener DFEs based on the BCG and the PBCG algorithm as well as their rank-flexible implementations. Previous implementations of the BCG algorithm [2] and those of the PBCG algorithm [8, 4] did not offer the rank-flexibility which we can observe in Fig. 2, i.e., in our implementations the desired rank of the block Krylov subspace does not necessarily have to be an integer multiple of the observation length ($T = 4$ in this case). Hence, Algorithm 1 and 2 avoid zero-filling and prevent the BCG and the PBCG algorithm from becoming computationally inefficient. The considerations of Fig. 2 yield that the BCG Wiener DFEs of rank 8 and rank 12 are outperformed by the PBCG Wiener DFEs of rank 4 and rank 8, respectively. According to Tab. 1, the total computational complexities of

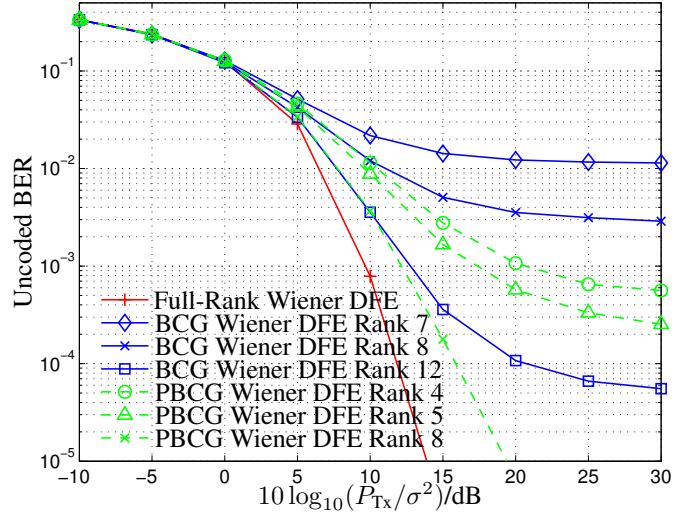


Fig. 2. Uncoded BER vs. SNR for Reduced-Rank Wiener DFEs for $T = 4$, $R = 8$, $L = 3$, $\nu = 2$, $G = K = 4$ and $F = 3$

the BCG Wiener DFEs of rank 8 and rank 12 are $0.7 \cdot 10^5$ and $1.05 \cdot 10^5$ flops, respectively, but, however, those of the PBCG Wiener DFEs of rank 4 and 8 are $0.45 \cdot 10^5$ and $0.85 \cdot 10^5$, respectively. We can conclude that despite of the additional complexity of the preconditioners (cf. Tab. 2), there is still a sufficient save of complexity when applying them and they even yield a better performance.

7. REFERENCES

- [1] P. Mosen, "Feedback Equalization for Fading Dispersive Channels," *IEEE Transactions on Information Theory*, vol. 17, no. 1, pp. 56–64, January 1971.
- [2] D. O'Leary, "The Block Conjugate Gradient Algorithm and Related Methods," *Linear Algebra and its Applications*, vol. 29, pp. 293–322, 1980.
- [3] P. Ding, M. D. Zoltowski, and M. Fimoff, "Preconditioned Conjugate Gradient based fast computation of indirect Decision Feedback Equalizer," *ICASSP 2004*, pp. (IV-1001)–(IV-1004), 2004.
- [4] T. F. Chan and J. A. Olkin, "Circulant Preconditioners for Toeplitz-block Matrices," *Numerical Algorithms*, pp. 89–101, 1994.
- [5] L. L. Scharf, *Statistical Signal Processing*, Addison-Wesley, 1991.
- [6] I. Groh and G. Dietl, "Reduced-Rank MIMO Receivers Based on Krylov Subspaces Random Matrices," Tech. Rep. TUM-LNS-TR-05-01, Munich University of Technology, April 2005.
- [7] Y. Saad, "Iterative Methods for Sparse Linear Systems," *SIAM*, 2003.
- [8] R. N. Chan and M. K. Ng, "Conjugate Gradient Methods for Toeplitz Systems," *SIAM Review*, vol. 38, no. 3, pp. 427–482, 1996.