



KTH Electrical Engineering

Wideband MIMO Channel Diagonalization in the Time Domain

© 2011 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

RASMUS BRANDT AND MATS BENGTTSSON

Stockholm 2011

Signal Processing
School of Electrical Engineering
KTH Royal Institute of Technology

IR-EE-SB 2011:052

Wideband MIMO Channel Diagonalization in the Time Domain

Rasmus Brandt and Mats Bengtsson
ACCESS Linnaeus Center
Signal Processing Laboratory
KTH Royal Institute of Technology
SE-100 44 Stockholm, Sweden
E-mail: firstname.lastname@ee.kth.se

Abstract—Methods for spatially diagonalizing wideband multiple-input multiple-output channels using linear finite impulse response (FIR) filters are investigated. The PSVD approach by applying the PQRD-BC algorithm for approximate singular value decomposition (SVD) of polynomial matrices is compared to the approach of performing a set of conventional SVDs in the Discrete Fourier Transform (DFT) domain, in terms of complexity and approximation error. Reduced order filters, based on the DFT-SVDs, are then obtained by optimizing the phases of the filters. Applying the phase optimized filters as linear filters then forms a benchmark on the accuracy attainable for any PSVD factorization, for the given filter length.

Simulations show that the DFT-SVD method has significantly lower complexity than the PSVD by PQRD-BC, but results in higher order filters. On the other hand, the PSVD by PQRD-BC yields filters which are close to being perfectly unitary for all frequencies. To achieve good performance, the reduced order filters are around one order of magnitude longer than the channel impulse response length. Therefore there is no gain in performing time domain diagonalization using a polynomial SVD, compared to using a multicarrier solution.

I. INTRODUCTION

A well-known transmission strategy for achieving the capacity of a narrowband multiple-input multiple-output (MIMO) channel is based on the singular value decomposition (SVD) of the channel matrix [1, p. 66]. Utilizing the unitary matrices obtained from the SVD for precoding and receive filtering, the effective channel becomes diagonal and the sub-channel link rates can be optimized. A similar approach can be taken for wideband MIMO channels, if the SVD is performed in the frequency domain. The effective channel is then perfectly diagonalized for each frequency bin. This is the basis for the spatial multiplexing by MIMO-OFDM [1] method, where the signaling is also performed in the frequency domain.

For a wideband MIMO single-carrier system, it may however still be of interest to operate over a spatially diagonalized channel. Instead of diagonalizing the channel in the frequency domain, another approach is to find an approximate SVD in the time domain. Such an approach has the benefit of approximately diagonalizing the channel for all frequencies, as opposed to only in specific points.

Recently [2]–[4], new methods for performing an approximate SVD of a polynomial matrix have been proposed. Applying the matrix sequences from the polynomial SVD (PSVD)

for precoding and receive filtering, the channel is transformed into an approximately diagonal polynomial matrix. This effective channel is frequency-selective, since the diagonal entries are finite impulse response (FIR) filters, and some form of equalization of the scalar sub-channels is therefore necessary. Another method is to apply time domain beamforming to a multicarrier system through cyclic filtering [5], an approach which, however, is beyond the scope of this paper.

It is the purpose of this paper to attempt to find a benchmark on the filter orders required to obtain a certain accuracy, by any PSVD factorization, as well as to draw general conclusions on the feasibility of time domain channel diagonalization. The benchmark will be based on performing multiple SVDs in the Discrete Fourier Transform (DFT) domain, called DFT-SVD. In order to decrease the approximation error when the DFT-SVD filters are employed as linear filters, reduced order filters are found by optimizing the filter phases [6].

Notation: Matrices and vectors will be written in upper-case and lower-case bold face, respectively. The operations $[\cdot]^*$, $[\cdot]^H$ and $\mathbb{E}(\cdot)$ represent complex conjugate, Hermitian transpose and the expectation operator. The Kronecker delta is $\delta(n)$. The space of complex polynomial matrices of spatial dimensions $p \times q$ is written as $\underline{\mathbb{C}}^{p \times q}$. The operators $*$ and \otimes represent linear and circular convolution, respectively.

II. PROBLEM FORMULATION

When modeling a block constant discrete-time and frequency-selective $p \times q$ complex baseband channel as a matrix FIR filter, its transfer function takes the form

$$\mathbf{H}(z) = \sum_{m=0}^{M-1} \mathbf{H}(m)z^{-m} \quad (1)$$

where $\mathbf{H}(m) \in \mathbb{C}^{p \times q}$ is the channel response at lag m and M is the length of the channel impulse response. For a transmitted signal $\mathbf{x}(k) \in \mathbb{C}^{q \times 1}$, and received signal $\mathbf{y}(k) \in \mathbb{C}^{p \times 1}$ the system model then takes the form

$$\mathbf{y}(k) = \sum_{m=0}^{M-1} \mathbf{H}(m)\mathbf{x}(k-m) + \mathbf{w}(k) \quad (2)$$

where $\mathbf{w}(k)$ is assumed to be temporally and spatially white additive Gaussian noise with zero mean and covariance $\mathbb{E}(\mathbf{w}(k)\mathbf{w}^H(m)) = \sigma_w^2 \mathbf{I} \delta(k-m)$.

For the system in (2), the goal here is to find linear FIR precoder/receive filters

$$\begin{aligned}\mathbf{x}(k) &= \sum_{m=0}^{L_t-1} \mathbf{T}(m)\mathbf{x}(k-m) = \mathbf{T}(k) * \mathbf{x}'(k) \\ \mathbf{y}'(k) &= \sum_{m=0}^{L_r-1} \mathbf{R}(m)\mathbf{y}(k-m) = \mathbf{R}(k) * \mathbf{y}(k)\end{aligned}\quad (3)$$

such that the effective channel

$$\mathbf{S}(k) = \mathbf{R}(k) * \mathbf{H}(k) * \mathbf{T}(k)$$

is diagonal and the noise characteristics are unchanged, i.e. $\mathbb{E}(\mathbf{w}'(k)\mathbf{w}^H(m)) = \mathbb{E}(\mathbf{w}(k)\mathbf{w}^H(m))$. The resulting spatially independent sub-channels can then be used for multiplexing of data streams, or diversity.

Infinite impulse response (IIR) precoder/receive filters can be found if an SVD of the channel in the discrete-time Fourier transform (DTFT) domain can be obtained:

$$\mathbf{H}(e^{j\omega}) = \mathbf{U}(e^{j\omega})\mathbf{D}(e^{j\omega})\mathbf{V}^H(e^{j\omega}), \quad 0 \leq \omega < 2\pi \quad (4)$$

where $\mathbf{U}(e^{j\omega}) \in \mathbb{C}^{p \times p}$, $\mathbf{V}(e^{j\omega}) \in \mathbb{C}^{q \times q}$ are unitary and $\mathbf{D}(e^{j\omega})$ has zero off-diagonal elements with $(d_1(e^{j\omega}) \ d_2(e^{j\omega}) \ \dots \ d_{\min(p,q)}(e^{j\omega}))$ along the diagonal. In time, (4) corresponds to the convolution

$$\mathbf{H}(n) = \mathbf{U}(n) * \mathbf{D}(n) * \mathbf{V}^H(-n), \quad -\infty < n < \infty \quad (5)$$

and the unitarity of $\mathbf{U}(e^{j\omega})$ and $\mathbf{V}(e^{j\omega})$ corresponds to

$$\begin{aligned}\mathbf{U}^H(-n) * \mathbf{U}(n) &= \mathbf{I}_{p \times p} \delta(n) \\ \mathbf{V}^H(-n) * \mathbf{V}(n) &= \mathbf{I}_{q \times q} \delta(n).\end{aligned}\quad (6)$$

For such a decomposition, precoder/receive filters selected as

$$\mathbf{T}(n) = \mathbf{V}(n), \quad \mathbf{R}(n) = \mathbf{U}^H(n), \quad -\infty < n < \infty$$

will perfectly diagonalize the channel. The noise term $\mathbf{w}'(k)$ will still be temporally and spatially white due to the unitarity of $\mathbf{U}(n)$.

In order to turn such a diagonalization procedure into a realizable FIR system, the SVD can either be performed in the DFT domain, corresponding to a sampling of (4), or the filter sequences in (5) can be approximated directly by finite-length sequences. By sampling in the frequency domain, the decomposition will be perfect in those sample points, but not necessarily in between. Approximating the decomposition in the time domain will lead to approximate decompositions at any frequency. In the next section, the method for performing the approximation in the time domain is described.

III. POLYNOMIAL SINGULAR VALUE DECOMPOSITION

A polynomial singular value decomposition is a non-unique decomposition of a polynomial matrix $\mathbf{H}(z) \in \mathbb{C}^{p \times q}$ such that

$$\mathbf{H}(z) \approx \mathbf{U}(z)\mathbf{D}(z)\mathbf{V}^H(z^{-*}) \quad (7)$$

where $\mathbf{D}(z) \in \mathbb{C}^{p \times q}$ is approximately diagonal in the sense that the magnitudes of the off-diagonal coefficients are close

to zero. The matrices $\mathbf{U}(z) \in \mathbb{C}^{p \times p}$, $\mathbf{V}(z) \in \mathbb{C}^{q \times q}$ are paraunitary, i.e. $\mathbf{U}^H(z^{-*})\mathbf{U}(z) = \mathbf{I}_{p \times p}$ and $\mathbf{V}^H(z^{-*})\mathbf{V}(z) = \mathbf{I}_{q \times q}$, but not necessarily causal. The decomposition in (7) is functionally equivalent to an FIR approximation of the infinite convolution in (5), and the paraunitarity of $\mathbf{U}(z)$, $\mathbf{V}(z)$ corresponds to (6). See [7, Ch. 13-14] for a thorough exposition on polynomial and paraunitary matrices.

If $\mathbf{H}(z)$ in (7) represents a wideband MIMO channel transfer function, then causal versions of $\mathbf{U}(z)$ and $\mathbf{V}^H(z^{-*})$ can be employed as the transmit and receive filters in (3). The effective channel will then approximately be $z^{-\alpha}\mathbf{D}(z)$ where the delay α is determined by the number of non-causal taps in $\mathbf{U}(z)$ and $\mathbf{V}(z)$. As $\mathbf{U}^H(z^{-*})$ is paraunitary, the second-order characteristics of the noise process $\mathbf{w}(z)$ stay the same.

The method for obtaining a PSVD of a polynomial matrix of interest here is the PSVD by PQRD-BC algorithm proposed in [2], which is based on an underlying polynomial QR decomposition (PQRD). Other approaches also exist, such as utilizing an eigenvalue decomposition algorithm for para-Hermitian matrices (PEVD) to form a PSVD [3] and a method based on generalized Kogbetliantz transformations [4]. The PEVD method of [3] has the drawback that the magnitudes of the off-diagonal coefficients in the approximately diagonal matrix cannot be ensured to be sufficiently small. The method in [4] does not suffer from that, but is expected to result in filters with high orders, just as the PSVD by PQRD-BC will be shown to do.

The main idea of the PSVD by PQRD-BC method is to apply paraunitary transformation matrices (PTMs), based on Givens rotations, to the polynomial matrix at hand. For the 2×2 case, such a PTM takes the form

$$\mathbf{G}_{2 \times 2}(z) = \begin{pmatrix} 1 & 0 \\ 0 & z^{-t} \end{pmatrix} \begin{pmatrix} ce^{j\alpha} & se^{j\phi} \\ -se^{-j\phi} & ce^{-j\alpha} \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & z^t \end{pmatrix} \quad (8)$$

where the inner matrix is a Givens rotation, and the outer matrices are elementary delay/advance matrices. The PTM in (8) will null the coefficient associated with the z^{-t} term of the $(2, 1)$ element of the matrix it is applied to from the left, if the Givens rotation parameters and t are selected appropriately [2]. The PTM for the general $p \times q$ case is obtained by forming an identity matrix, and replacing the elements at the intersections of rows and columns k, l with the corresponding elements of (8).

The PSVD is formed by subsequently applying PTMs from the left and the right of the original matrix, until the off-diagonal coefficients are sufficiently small, forming

$$\mathbf{D}(z) = \mathbf{G}_{K_l}^{(l)}(z) \dots \mathbf{G}_1^{(l)}(z) \mathbf{A}(z) \mathbf{G}_1^{(r),H}(z^{-*}) \dots \mathbf{G}_{K_r}^{(r),H}(z^{-*})$$

which is approximately diagonal [2]. See [2] for the convergence proof. Identifying

$$\begin{aligned}\mathbf{U}(z) &= \mathbf{G}_1^{(l),H}(z^{-*}) \dots \mathbf{G}_{K_l}^{(l),H}(z^{-*}) \\ \mathbf{V}(z) &= \mathbf{G}_{K_r}^{(r)}(z) \dots \mathbf{G}_1^{(r)}(z)\end{aligned}$$

and noticing that $\mathbf{U}(z)$ and $\mathbf{V}(z)$ are paraunitary, since they are the product of paraunitary matrices, a PSVD as

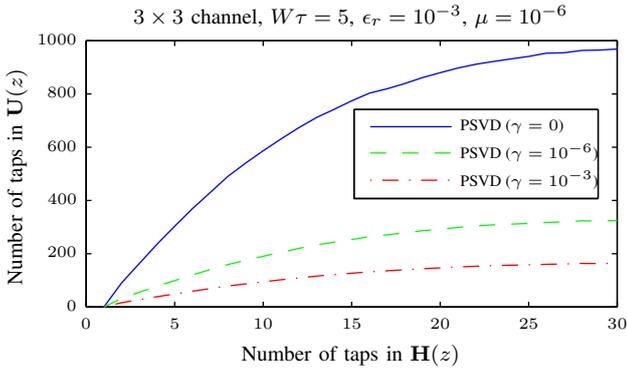


Fig. 1. Comparison of order of $\mathbf{H}(z)$ with corresponding orders of $\mathbf{U}(z)$, and truncated versions of $\mathbf{U}(z)$ were outer taps corresponding to at most a fraction of γ of the Frobenius norm sum were removed after convergence. The channels were generated from (12) and the resulting orders averaged over 4000 simulation runs.

in (7) is readily obtained. In this paper, the convergence criterion of the original PSVD by PQRD-BC algorithm has been slightly modified, so that the algorithm stops with $\sum_n \|\mathbf{D}(n)\|_F^2 / \sum_n \|\mathbf{A}_{\text{original}}(n)\|_F^2 < \epsilon_r$.

Each application of a PTM increases the order of the matrix it is applied to by $2|t|$. The orders of matrices involved in the PSVD by PQRD-BC operation therefore grow quickly, as made evident in Fig.1. To keep the orders down, a truncation step is performed during regular intervals in the algorithm. The outermost taps, corresponding to a fraction less than μ of the total Frobenius norm sum, are removed. Since the magnitudes of the coefficients affected generally are small, this has a limited effect on decomposition quality [2].

A further analysis of the PSVD by PQRD-BC algorithm is available in [8].

A. Complexity: Obtaining the Decomposition

The application of a PTM to a matrix $\mathbf{A}_i(z) \in \mathbb{C}^{p \times q}$ can be written as

$$\mathbf{G}(z)\mathbf{A}_{r,i}(z) = \mathbf{G}^{(-t)}z^{-t}\mathbf{A}_{r,i}(z) + \mathbf{G}^{(0)}\mathbf{A}_{r,i}(z) + \mathbf{G}^{(t)}\mathbf{A}_{r,i}(z)$$

where $\mathbf{A}_{r,i}(z)$, of length K_i , corresponds to the two rows of $\mathbf{A}_i(z)$ being affected by the PTM. From (8) it can be seen that $\mathbf{G}^{(-t)}$ and $\mathbf{G}^{(t)}$ only have one non-zero element each, and that $\mathbf{G}^{(0)}$ is diagonal. The application of one PTM therefore corresponds to four multiplications of a complex number with a sequence of complex numbers of length qK_i . Counting only complex multiplications, the complexity of the application of the PTM is then $4qK_i$ floating point operations (flops).

Since the PSVD by PQRD-BC algorithm iteratively applies PTMs, the total complexity is therefore lower bounded as

$$C_{\text{PSVD by PQRD-BC}} \geq 4q \sum_i K_i \quad (9)$$

where the sum is taken over all iterations, and the K_i s are the lengths of the matrices the PTMs are applied to.

B. Complexity: Filtering

For long data blocks, performing the filtering in (3) is efficiently implemented using the DFT convolution theorem and the Fast Fourier Transform. Specifically, this is more efficient than performing the time domain convolution when

$$\log_2(L_t + L_x - 1) < q(\min(L_t, L_x) - 1)$$

where L_x is the length of the signal sequence. The complexity of the filtering will then be $(L_t + L_x - 1)q^2 + (L_t + L_x - 1)q \log_2(L_t + L_x - 1)$ complex multiplications.

IV. SINGULAR VALUE DECOMPOSITION IN THE DFT DOMAIN

The decomposition in (4) can also be approximated by sampling the DTFT, yielding a decomposition on the form

$$\mathbf{H}(e^{j\omega_k}) = \mathbf{U}(e^{j\omega_k})\mathbf{D}(e^{j\omega_k})\mathbf{V}^H(e^{j\omega_k}) \quad k = 0 \dots N - 1 \quad (10)$$

where N is the number of frequency bins and $\omega_k = 2\pi k/N$. Such a decomposition can be obtained by taking the N -point DFT of the channel matrix sequence

$$\mathbf{H}(e^{j\omega_k}) = \sum_{m=0}^{N-1} \mathbf{H}_{zp}(m) e^{j2\pi km/N}$$

where $\mathbf{H}_{zp}(n)$ is a zero-padded version of $\mathbf{H}(n)$ if $N > M$, and performing a conventional SVD per frequency bin. The decomposition in (10) corresponds to circular convolution of the matrix sequences in the time domain, but are here used as linear filters such that

$$\mathbf{T}(n) = \mathbf{V}(n), \quad \mathbf{R}(n) = \mathbf{U}^H(n), \quad 0 \leq n \leq N - 1.$$

For small N , one would not expect this approach to result in an effective channel that is close to being diagonal. However, as N grows large, the DFT $\mathbf{U}(n)$, $\mathbf{V}(n)$ should converge to their DTFT counterparts, which diagonalize the channel perfectly.

This straight-forward approach has the drawback that the singular values may vary in-continuously between frequency bins, as the conventional SVD will order the singular values irrespectively of the ordering in neighbouring frequency bins [3]. The matrices $\mathbf{D}(e^{j\omega_k})$ have real elements, as opposed to the PSVD matrix $\mathbf{D}(z)$. This fact is not of great importance in communications, but could play a role in other applications.

A. Complexity: Obtaining the Decomposition

The number of flops needed, only counting complex multiplications, to find the N -point DFT on the left hand side of (10) is $Npq \log_2(N)/2$. From [9, p. 254] we find that the number of flops necessary to perform an SVD of a $p \times q$ matrix, assuming implementation using the Golub-Reinsch method, is approximately $4p^2q + 8pq^2 + 9q^3$. Together, this gives that the number of flops necessary to obtain the N SVDs is approximately

$$C_{\text{DFT-SVD}} = \frac{Npq \log_2(N)}{2} + N(4p^2q + 8pq^2 + 9q^3). \quad (11)$$

V. MODEL ORDER REDUCTION

The DFT-SVD decomposition corresponds to cyclic convolution in the time domain, but the filters obtained are applied as linear filters. The error in this approximation can be reduced, if the filters are transformed so that the operations of linear and cyclic convolution with themselves give approximately the same result. This section investigates such a method, which exploits the fact that the filter phases can be adjusted. The transformed filters have reduced orders, and as an effect of that also lower filtering complexity.

A. Phase Optimization and Truncation

Assuming a DFT-SVD as in (10), notice that replacing

$$\begin{aligned}\tilde{\mathbf{U}}(e^{j\omega_k}) &= \mathbf{U}(e^{j\omega_k})\mathbf{D}_u(e^{j\omega_k}) \\ \tilde{\mathbf{D}}(e^{j\omega_k}) &= \mathbf{D}_u^H(e^{j\omega_k})\mathbf{D}(e^{j\omega_k})\mathbf{D}_v(e^{j\omega_k}) \\ \tilde{\mathbf{V}}(e^{j\omega_k}) &= \mathbf{V}(e^{j\omega_k})\mathbf{D}_v(e^{j\omega_k})\end{aligned}$$

where $\mathbf{D}_u(e^{j\omega_k}) = \text{diag}(e^{j\phi_{k,1}^u}, e^{j\phi_{k,2}^u}, \dots, e^{j\phi_{k,p}^u})$, $\mathbf{D}_v(e^{j\omega_k}) = \text{diag}(e^{j\phi_{k,1}^v}, e^{j\phi_{k,2}^v}, \dots, e^{j\phi_{k,q}^v})$ forms a similar decomposition, where the diagonal matrix is no longer real. By selecting the phases such that the $N - L_u$ ($N - L_v$) last matrix coefficients of $\mathbf{U}(n)$ ($\mathbf{V}(n)$) are minimized in magnitude, the filter sequences can be truncated to length L_u (L_v), yielding reduced order filters that approximate the phase optimized filters. Finding such phases can be posed as a non-convex optimization problem, and a suboptimal solution method is proposed in [6].

If $L_u < \frac{N+1}{2}$, the last $N - L_u$ coefficients of $\tilde{\mathbf{U}}(n)$, corresponding to more than half of the total number of coefficients, will have magnitudes close to zero. For that reason, $\tilde{\mathbf{U}}^H(-n) \otimes \tilde{\mathbf{U}}(n) \cong \tilde{\mathbf{U}}^H(-n) * \tilde{\mathbf{U}}(n)$, i.e. the circular convolution of the sequence with itself is approximately equal to the linear convolution of the sequence with itself.

The phase optimization step is directly applicable to the DFT-SVD decomposition. It can also be applied to a PSVD, by zero-padding the involved matrix sequences such that the linear convolution between them is equivalent to a circular convolution. Taking the DFT of the zero-padded sequences then gives a decomposition on the form (10), and the phase optimization and truncation can be carried out from there.

B. Direct Truncation of PSVD

A straight-forward approach for reducing the filter order is to remove the outermost filter taps, keeping only L taps around the constant tap. It can be regarded as a special case of the method in Section V-A, if the matrices are multiplied in frequency by a linear phase $e^{-j2\pi km/N}$, $m = \lfloor L/2 \rfloor$, and then truncated.

VI. NUMERICAL RESULTS

In order to evaluate the performance of the two methods, simulations were performed. Channel matrices were generated on the form

$$\begin{aligned}\text{vec}(\mathbf{H}(m)) &\sim \mathcal{CN}(\mathbf{0}, ab^m \mathbf{I}_{pq \times pq}), \quad 0 \leq m \leq M - 1 \\ b &= e^{-1/W\tau}, a = 1 - b\end{aligned}\quad (12)$$

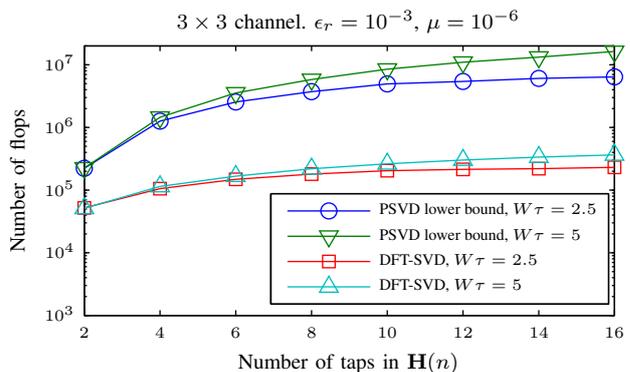


Fig. 2. Lower bound of complexity of PSVD by PQRD-BC compared to complexity of DFT-SVD for an effective channel error less than $10^{-3/2}$, averaged over 500 channel realizations.

where W is the transmitted signal bandwidth, τ is the RMS delay spread of the channel and \mathcal{CN} is a circularly symmetric Gaussian distribution. The product $W\tau$ is a measure of the frequency-selectivity of the channel. Two simulations were performed with $(M, W\tau) \in \{(7, 2.5), (15, 5)\}$. For those parameter values, the variance of the last channel tap was around 20dB lower than the variance of the first channel tap.

The following performance measures were used in the simulations. Denoting the relevant approximations with a hat, the effective channel error was calculated as

$$E_d = \sqrt{\frac{\sum_n \|\text{offdiag}(\hat{\mathbf{U}}^H(-n) * \mathbf{H}(n) * \hat{\mathbf{V}}(n))\|_F^2}{\sum_n \|\mathbf{H}(n)\|_F^2}}$$

and determines how close the effective channel is to being perfectly diagonal. Additionally, the unitarity error of the receive filter was calculated as

$$E_u = \sqrt{\frac{\sum_n \|\hat{\mathbf{U}}^H(-n) * \hat{\mathbf{U}}(n) - \mathbf{I}\delta(n)\|_F^2}{\|\mathbf{I}\|_F^2}}$$

This error determines how close the filtered received noise is to staying temporally and spatially white.

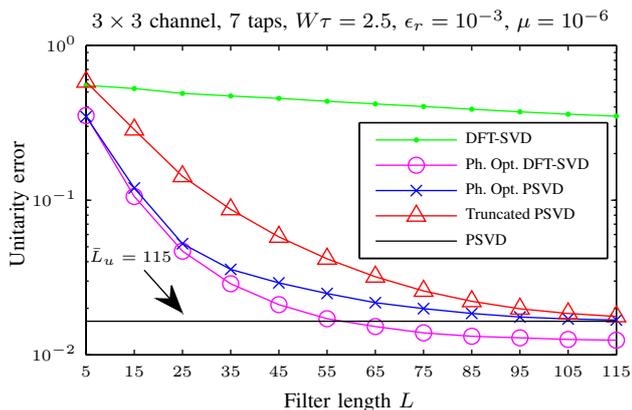
For sum rate comparisons of the two schemes, see [8].

A. Complexity

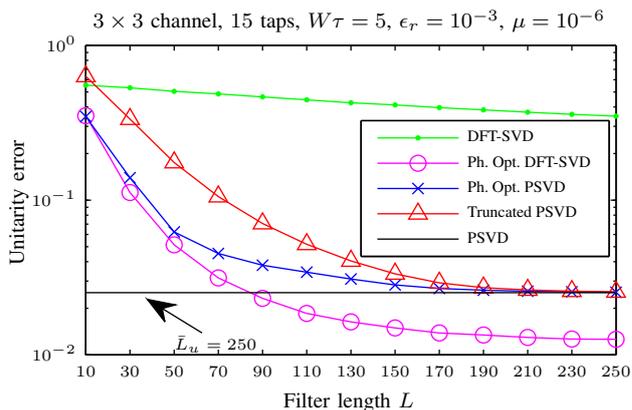
The number of flops necessary to reach $E_d < 10^{-3/2}$, obtained from (9) and (11), is plotted in Fig. 2 for PSVD by PQRD-BC and DFT-SVD. This corresponds to $\epsilon_r = 10^{-3}$ for PSVD by PQRD-BC and for the DFT-SVD, N was increased until the criterion was met. Clearly, the PSVD by PQRD-BC algorithm needs significantly more computations to converge, as compared to the other method.

B. Approximation Errors

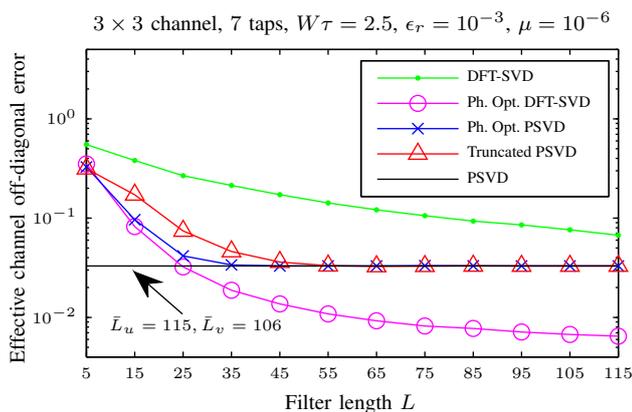
The approximation errors for the two methods, including phase optimized filters with $L_u = L_v = L \in \{10, \dots, 250\}$, was compared for the two channel scenarios and the results averaged over 100 channel realizations. Parameters used were $\epsilon_r = 10^{-3}$, $\mu = 10^{-6}$ for PSVD by PQRD-BC and $N = 2L$ for the phase optimized DFT-SVD.



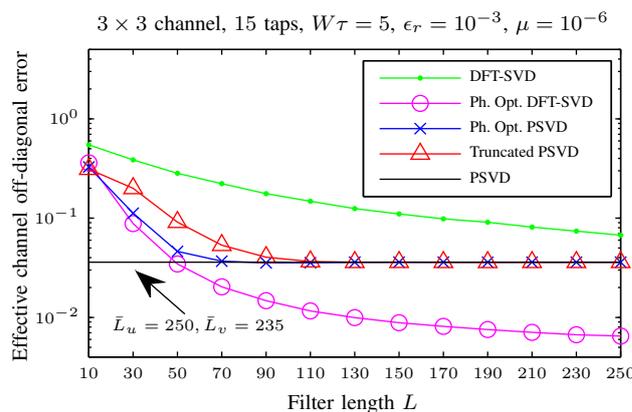
(a) Unitarity errors for the 7-tap $W\tau = 2.5$ channel



(b) Unitarity errors for the 15-tap $W\tau = 5$ channel



(c) Effective channel errors for the 7-tap $W\tau = 2.5$ channel



(d) Effective channel errors for the 15-tap $W\tau = 5$ channel

For $W\tau = 2.5$, the unitarity errors are shown in Fig. 3a and the effective channel errors are displayed in Fig. 3c. The phase optimized DFT-SVD performs better than the other methods in terms of both errors.

For the more frequency-selective $W\tau = 5$ case, the unitarity errors can be seen in Fig. 3b and the effective channel errors are plotted in Fig. 3d. The phase optimized DFT-SVD still performs best, but needs around twice as long filters to reach the same performance as for the $W\tau = 2.5$ scenario.

VII. CONCLUSIONS

Time domain strategies for diagonalization of frequency-selective channels were evaluated. To achieve the same effective channel error, the DFT-SVD needed significantly fewer flops than PSVD by PQRD-BC, but resulted in filters of higher order. The unitarity error performance of PSVD by PQRD-BC exceeds that of the DFT-SVD, for the same filter order.

The filter orders necessary grow as the frequency-selectivity and channel length increases. For the channel model investigated, filter lengths of one order of magnitude longer than the channel impulse response were needed for good performance. This shows that it is advisable to apply a multicarrier solution to the problem at hand instead of using a polynomial SVD, because any PSVD factorization will need very high filter

orders compared to the original channel length.

REFERENCES

- [1] A. Pauraj, R. Nabar, and D. Gore, *Introduction to Space-Time Wireless Communications*. Cambridge, U.K.: Cambridge University Press, 2003.
- [2] J. A. Foster, J. G. McWhirter, M. R. Davies, and J. A. Chambers, "An algorithm for calculating the QR and singular value decompositions of polynomial matrices," *IEEE Trans. Signal Process.*, vol. 58, pp. 1263–1274, Mar. 2010.
- [3] J. G. McWhirter, P. D. Baxter, T. Cooper, S. Redif, and J. Foster, "An EVD algorithm for para-hermitian polynomial matrices," *IEEE Trans. Signal Process.*, vol. 55, pp. 2158–2169, May 2007.
- [4] J. G. McWhirter, "An algorithm for polynomial matrix SVD based on generalised Kogbetliantz transformations," in *Proc. EUSIPCO'10*, Aalborg, Denmark, Aug. 2010, pp. 457–461.
- [5] Y. wen Liang, R. Schober, and W. Gerstacker, "Time-domain transmit beamforming for MIMO-OFDM systems with finite rate feedback," *IEEE Trans. Commun.*, vol. 57, pp. 2828–2838, Sep. 2009.
- [6] D. P. Palomar, M. A. Lagunas, A. Pascual, and A. P. Neira, "Practical implementation of jointly designed transmit receive space-time IIR filters," in *Proc. Symp. Signal Processing and its Applications, Sixth Int.*, Kuala Lumpur, Malaysia, Aug. 2001, pp. 521–524.
- [7] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice Hall, 1993.
- [8] R. Brandt, "Polynomial matrix decompositions: Evaluation of algorithms with an application to wideband MIMO communications," M.Sc. thesis, Uppsala University, Dec. 2010. [Online]. Available: <http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-134389>
- [9] G. H. Golub and C. F. van Loan, *Matrix computations*, 3rd ed. Baltimore, MD: Johns Hopkins University Press, 1996.