in Proc. 36th Asilomar Conf. Signals, Systems, Computers, Pacific Grove (CA), Nov. 2002, pp. 284–288

FAST ITERATIVE DECODING OF LINEAR DISPERSION CODES FOR UNKNOWN MIMO CHANNELS

Harold Artés and Franz Hlawatsch

Institute of Communications and Radio-Frequency Engineering, Vienna University of Technology
Gusshausstrasse 25/389, A-1040 Vienna, Austria (Europe)
phone: +43 1 58801 38923, fax: +43 1 58801 38999, email: Harold.Artes@nt.tuwien.ac.at
web: http://www.nt.tuwien.ac.at/dspgroup/time.html

ABSTRACT

We propose an efficient iterative algorithm for demodulating/decoding linear dispersion (LD) codes in the case of an unknown *MIMO channel. The algorithm is based on the POCS (projections onto convex sets) technique and consists of alternating projections of an estimate of the transmit matrix onto two linear subspaces. In the noise-free case, the iteration is guaranteed to converge to the transmitted data whenever the LD code theoretically allows perfect data reconstruction. In the presence of noise, simulation results demonstrate the algorithm's good performance. We also propose a DFT-matrix based code design that yields a further substantial reduction of complexity, and a modified algorithm that exploits the presence of pilot symbols.*

1. INTRODUCTION

Background. Recently there has been considerable interest in space-time (ST) codes for the practically important case where the MIMO channel is unknown to both the transmitter and the receiver. *Unitary ST modulation* and *differential unitary ST modulation* [1–3] can achieve the capacity of the unknown MIMO channel, but this comes at the cost of a demodulation complexity that is prohibitive for a large blocklength and/or a large number of transmit antennas. In contrast, the ST transmission schemes proposed in [4–10] lead to equalization-type demodulation algorithms whose complexity is acceptable also if the blocklength and the number of antennas are medium-to-large. In [8], we introduced a particularly efficient iterative demodulation algorithm for a class of separable linear ST codes termed *ST matrix modulation* (here, "separable" means that the coding over space and the coding over time are done independently).

Contributions. This paper presents the following contributions.

- We extend our iterative demodulation algorithm to the general class of *linear dispersion (LD) codes* [11] for the case of an unknown MIMO channel. The iterative algorithm has significantly lower complexity than the SVD-based technique used in [4]. It is guaranteed to converge to the transmitted data (up to a factor) whenever an LD code theoretically allows perfect data reconstruction.
- We present a new DFT-matrix based code design that leads to a further significant reduction of demodulator complexity.
- We propose a modification of our iterative algorithm that exploits the presence of pilot symbols to resolve the unknown factor mentioned above.

Organization of paper. Section 2 reviews the structure of LD codes [11] and identifiability results for LD codes in the case of an unknown channel [4]. In Section 3, we extend the iterative demodulation algorithm from [8] to LD codes. A DFT-based code design allowing a further reduction of demodulator complexity is proposed in Section 4. In Section 5, we present a modified demodulation algorithm that exploits the presence of pilot symbols. Finally, simulation results provided in Section 6 demonstrate the robustness of our demodulation algorithm to channel noise.

2. REVIEW OF LINEAR DISPERSION CODES

In this section, we briefly review the definition of LD codes [11] and summarize the identifiability results from [4]. We consider an equivalent discrete-time, complex-valued, noise-free baseband MIMO channel with M_T transmit antennas and $M_R \geq M_T$ receive antennas. We assume that the channel is flat-fading and constant over a block of N time points. Thus, within such a block, it can be described by the matrix[1] $\tilde{\mathbf{H}} \in \mathbb{C}^{M_R \times M_T}$ that is assumed to have full rank. The channel's input-output relation is $\tilde{\mathbf{x}}[n] = \tilde{\mathbf{H}}\tilde{\mathbf{s}}[n]$ with the transmit vectors $\tilde{\mathbf{s}}[n]$ of length M_T and the receive vectors $\tilde{\mathbf{x}}[n]$ of length M_R. Combining N successive transmit vectors into a transmit matrix $\tilde{\mathbf{S}} \triangleq [\tilde{\mathbf{s}}[1]\ \tilde{\mathbf{s}}[2] \cdots \tilde{\mathbf{s}}[N]]$ of size $M_T \times N$ and N successive receive vectors into a receive ma-

Funding by FWF grant P15156.

[1] Throughout this paper, complex quantities will be marked by a tilde ˜.

trix $\tilde{\mathbf{X}} \triangleq [\tilde{\mathbf{x}}[1]\, \tilde{\mathbf{x}}[2] \cdots \tilde{\mathbf{x}}[N]]$ of size $M_R \times N$, we can write the input-output relation for a whole length-N block as

$$\tilde{\mathbf{X}} = \tilde{\mathbf{H}}\tilde{\mathbf{S}}. \tag{1}$$

Linear dispersion codes. For an LD code, K complex-valued data symbols $\tilde{d}_k$ are mapped onto the corresponding transmit matrix via [11]

$$\tilde{\mathbf{S}} = \sum_{k=1}^{K} \left[\alpha_k \tilde{\mathbf{A}}_k + j\beta_k \tilde{\mathbf{B}}_k\right], \tag{2}$$

with $\alpha_k = \mathrm{Re}\{\tilde{d}_k\}$, $\beta_k = \mathrm{Im}\{\tilde{d}_k\}$, and complex-valued "dispersion matrices" $\tilde{\mathbf{A}}_k$ and $\tilde{\mathbf{B}}_k$ of size $M_T \times N$. The expression (2) is generally nonlinear in the data vector $\tilde{\mathbf{d}} \triangleq (d_1 \cdots d_K)$. Here, we will find it convenient to use the following linear, real-valued formulation that is equivalent to (2):

$$\mathbf{S} = \sum_{k=1}^{K} [\alpha_k \mathbf{A}_k + \beta_k \mathbf{B}_k], \tag{3}$$

where

$$\mathbf{S} \triangleq \begin{bmatrix} \mathrm{Re}\{\tilde{\mathbf{S}}\} \\ \mathrm{Im}\{\tilde{\mathbf{S}}\} \end{bmatrix}, \quad \mathbf{A}_k \triangleq \begin{bmatrix} \mathrm{Re}\{\tilde{\mathbf{A}}_k\} \\ \mathrm{Im}\{\tilde{\mathbf{A}}_k\} \end{bmatrix}, \quad \mathbf{B}_k \triangleq \begin{bmatrix} -\mathrm{Im}\{\tilde{\mathbf{B}}_k\} \\ \mathrm{Re}\{\tilde{\mathbf{B}}_k\} \end{bmatrix}.$$

This can be rewritten as

$$\mathrm{vec}\{\mathbf{S}^T\} = \mathbf{L}\mathbf{d}, \tag{4}$$

with

$$\mathbf{L} \triangleq \left[\mathrm{vec}\{\mathbf{A}_1^T\} \cdots \mathrm{vec}\{\mathbf{A}_K^T\}\ \mathrm{vec}\{\mathbf{B}_1^T\} \cdots \mathrm{vec}\{\mathbf{B}_K^T\}\right],$$
$$\mathbf{d} \triangleq [\alpha_1 \cdots \alpha_K\ \beta_1 \cdots \beta_K]^T.$$

Note that $\mathrm{vec}\{\mathbf{S}^T\}$ is a linear function of the real data vector $\mathbf{d}$. Whereas in practice the data is taken from some finite alphabet, this will not be exploited or assumed here.

Furthermore, defining

$$\mathbf{X} \triangleq \begin{bmatrix} \mathrm{Re}\{\tilde{\mathbf{X}}\} \\ \mathrm{Im}\{\tilde{\mathbf{X}}\} \end{bmatrix}, \quad \mathbf{H} \triangleq \begin{bmatrix} \mathrm{Re}\{\tilde{\mathbf{H}}\} & -\mathrm{Im}\{\tilde{\mathbf{H}}\} \\ \mathrm{Im}\{\tilde{\mathbf{H}}\} & \mathrm{Re}\{\tilde{\mathbf{H}}\} \end{bmatrix},$$

we obtain the following equivalent real-valued formulation of the channel input-output relation (1):

$$\mathbf{X} = \mathbf{H}\mathbf{S}.$$

Identifiability results. In [4], it is shown that in the noise-free case and under certain identifiability conditions, the structure (3) of an LD code is strong enough to allow perfect data reconstruction at the receiver (up to an unknown real-valued factor) without knowledge of the channel $\mathbf{H}$. Specifically, consider the $2NM_T \times (2M_T M_R + 2K)$ stacked matrix

$$\mathbf{Q} \triangleq \left[\mathbf{I}_{M_T} \otimes \mathbf{X}^s\ \ -\mathbf{L}\right]$$

($\otimes$ denotes the Kronecker product), with

$$\mathbf{X}^s \triangleq \begin{bmatrix} \mathrm{Re}\{\tilde{\mathbf{X}}\}^T & -\mathrm{Im}\{\tilde{\mathbf{X}}\}^T \\ \mathrm{Im}\{\tilde{\mathbf{X}}\}^T & \mathrm{Re}\{\tilde{\mathbf{X}}\}^T \end{bmatrix}.$$

Then, it is stated in [4] that for perfect reconstruction, $\mathbf{Q}$ must have a right null space of dimension 1. (Note that this implies $N \geq M_R + K/M_T$.) In that case, the right null space of $\mathbf{Q}$ is spanned by the vector $\left[\mathrm{vec}\{\mathrm{Re}\{\tilde{\mathbf{W}}^T\}\}^T\ \mathrm{vec}\{\mathrm{Im}\{\tilde{\mathbf{W}}^T\}\}^T\ \mathbf{d}^T\right]^T$, where $\tilde{\mathbf{W}} = \tilde{\mathbf{H}}^{\#}$ (the pseudo-inverse of $\tilde{\mathbf{H}}$) is the unknown "equalizer matrix" that retrieves $\tilde{\mathbf{S}}$ from $\tilde{\mathbf{X}}$ according to $\tilde{\mathbf{S}} = \tilde{\mathbf{W}}\tilde{\mathbf{X}}$. Thus, in the presence of noise the equalizer $\tilde{\mathbf{W}}$ and the data $\mathbf{d}$ can be approximately recovered as the right singular vector of $\mathbf{Q}$ corresponding to the smallest singular value [4]. Since both dimensions of $\mathbf{Q}$ grow linearly with the blocklength N (note that for constant data rate, the number of data symbols K grows linearly with N), the computational complexity of this SVD-based algorithm becomes excessive for large N. This motivates the efficient iterative algorithm proposed next.

3. THE ITERATIVE ALGORITHM

We will now extend the efficient iterative demodulation algorithm that we originally proposed for ST matrix modulation codes (separable LD codes) in [8] to general LD codes.

Subspace structure. We first discuss two structural properties on which our algorithm is based. As before, we consider the noise-free case and assume knowledge of the receive matrix $\mathbf{X}$ but not of the channel matrix $\mathbf{H}$. Generalizing the results of [8], it can be shown that under the identifiability conditions of [4], the transmit matrix $\mathbf{S}$ is uniquely defined (up to an unknown factor) by the following two properties.

1. The transmit matrix $\mathbf{S}$ possesses the linear code structure (4).
2. The row-span of $\mathbf{S}$ lies in the row-span of $\mathbf{X}$ (this is due to $\mathbf{X} = \mathbf{H}\mathbf{S}$).

Because both properties have to be in force simultaneously, $\mathbf{S}$ satisfies

$$\mathbf{S} \in \mathcal{A} \cap \mathcal{B}, \tag{5}$$

where $\mathcal{A}$ denotes the linear subspace of all matrices with the LD code structure (4) and $\mathcal{B}$ denotes the linear subspace of all matrices whose row-span lies in the row-span of $\mathbf{X}$.

POCS algorithm. Since linear subspaces are special cases of convex sets, the formulation (5) suggests a *projections onto convex sets (POCS) algorithm* [12] for calculating and demodulating $\mathbf{S}$. This algorithm is iterative and consists of alternating orthogonal projections of the iterated transmit matrix estimate $\mathbf{S}^{(i)}$ onto $\mathcal{A}$ and $\mathcal{B}$ as visualized by Fig. 1. The POCS algorithm is guaranteed to converge to a point in the intersection $\mathcal{A} \cap \mathcal{B}$ [12]. If the relevant identifiability conditions are satisfied [4], i.e., if the LD code theoretically allows perfect reconstruction up to a factor, then the intersection $\mathcal{A} \cap \mathcal{B}$ is a one-dimensional linear subspace $\{c\mathbf{S}\}$

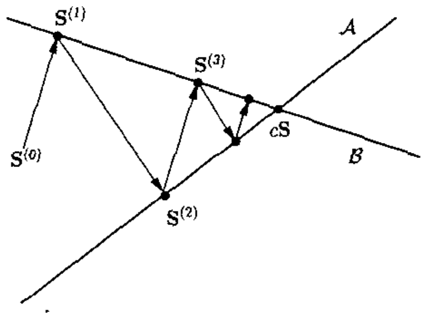

Fig. 1: The POCS algorithm. For any initialization $\mathbf{S}^{(0)}$, the matrices $\mathbf{S}^{(i)}$ generated by alternating orthogonal projections onto the subspaces $\mathcal{A}$ and $\mathcal{B}$ converge to a matrix in the intersection $\mathcal{A} \cap \mathcal{B}$.

with $c \in \mathbb{R}$. Thus, the convergence point in $\mathcal{A} \cap \mathcal{B}$ is guaranteed to be the desired (i.e., transmitted) $\mathbf{S}$ up to a factor.

Next, we will show how to calculate the orthogonal projections onto the subspaces $\mathcal{A}$ and $\mathcal{B}$.

- *Projection onto $\mathcal{A}$.* Any $\mathbf{S} \in \mathcal{A}$ has the form (4), i.e., $\mathrm{vec}\{\mathbf{S}^T\} = \mathbf{L}\mathbf{d}$ with some $\mathbf{d}$ and with $\mathbf{L}$ given. Let $\mathbf{S}^{(i-1)}$ denote the result of the previous iteration step, i.e., of the projection onto $\mathcal{B}$ (see below). Projecting $\mathbf{S}^{(i-1)}$ onto $\mathcal{A}$ then corresponds to calculating the vector $\mathbf{d}$ for which the model $\mathbf{L}\mathbf{d}$ best fits $\mathrm{vec}\{\mathbf{S}^{(i-1)T}\}$ in the least-squares (LS) sense. This vector is

$$\mathbf{d}^{(i)} = \mathbf{L}^{\#}\,\mathrm{vec}\{\mathbf{S}^{(i-1)T}\}. \qquad (6)$$

Thus, the orthogonal projection of $\mathbf{S}^{(i-1)}$ onto $\mathcal{A}$ is given by

$$\mathrm{vec}\{\mathbf{S}^{(i)T}\} = \mathbf{L}\mathbf{d}^{(i)} = \mathbf{L}\mathbf{L}^{\#}\,\mathrm{vec}\{\mathbf{S}^{(i-1)T}\}. \qquad (7)$$

The computational complexity of this projection is $O(NM_{\mathrm{T}}K)$ (note that $\mathbf{L}^{\#}$ can be precalculated before the iteration is started). This complexity increases quadratically with the blocklength N because K grows linearly with N if the data rate is held constant. In Section 4, we shall introduce a DFT-based code construction by which the complexity is reduced to $O(NM_{\mathrm{T}} \log(NM_{\mathrm{T}}))$.

- *Projection onto $\mathcal{B}$.* Any $\mathbf{S} \in \mathcal{B}$ can be written as

$$\mathbf{S} = \mathbf{W}\mathbf{X}, \qquad (8)$$

with some "channel equalizer matrix" $\mathbf{W} \in \mathbb{R}^{2M_{\mathrm{T}} \times 2M_{\mathrm{R}}}$. Let $\mathbf{S}^{(i-1)}$ denote the result of the previous iteration step, i.e., of the projection onto $\mathcal{A}$ as explained above. Projecting $\mathbf{S}^{(i-1)}$ onto $\mathcal{B}$ then corresponds to calculating the $\mathbf{W}$ for which the model $\mathbf{W}\mathbf{X}$ best fits $\mathbf{S}^{(i-1)}$ in the LS sense:

$$\mathbf{W}^{(i)} = \mathbf{S}^{(i-1)}\mathbf{X}^{\#}. \qquad (9)$$

Thus, the orthogonal projection of $\mathbf{S}^{(i-1)}$ onto $\mathcal{B}$ is obtained as

$$\mathbf{S}^{(i)} = \mathbf{W}^{(i)}\mathbf{X} = \mathbf{S}^{(i-1)}\mathbf{X}^{\#}\mathbf{X}.$$

Here, $\mathbf{X}^{\#}$ can be precalculated before starting the iteration. The complexity of this projection is $O(NM_{\mathrm{T}}M_{\mathrm{R}})$ and thus increases only linearly with the blocklength N.

In each iteration step, we obtain an estimate of the data vector $\mathbf{d}$ in (6) and an estimate of the channel equalizer matrix $\mathbf{W}$ in (9) (both up to a factor). In the noise-free case, it can be shown that the POCS algorithm is guaranteed to converge to the transmitted data (up to a single unknown factor) provided that the LD code theoretically allows perfect reconstruction. The rate of convergence depends on the angle between the subspaces $\mathcal{A}$ and $\mathcal{B}$ as well as on the initialization, $\mathbf{S}^{(0)}$. If some pilot symbols exist, they can be used to calculate a good initialization (see Section 5). Another way to speed up convergence is to use *relaxation* [12] and/or knowledge of the finite data alphabet. (The latter approach, however, introduces a nonconvex set and thus convergence to the desired solution is no longer guaranteed theoretically.)

4. DFT-MATRIX BASED CODE CONSTRUCTION

The projection onto $\mathcal{A}$ is the most complex part of the POCS algorithm. In this section, we propose a specific construction of the dispersion matrices $\mathbf{A}_k$ and $\mathbf{B}_k$ that allows to implement this projection by means of an FFT.

Suppressing the superscripts $^{(i)}$ and $^{(i-1)}$ for simplicity, we can rewrite (6) as

$$\mathbf{d} = \mathrm{Re}\{\tilde{\mathbf{L}}^{\#}\mathrm{vec}\{\tilde{\mathbf{S}}^T\}\}, \qquad (10)$$

with the $NM_{\mathrm{T}} \times K$ matrix $\tilde{\mathbf{L}}$ defined such that $\mathbf{L} = \left(\mathrm{Re}\{\tilde{\mathbf{L}}\}^T\ \mathrm{Im}\{\tilde{\mathbf{L}}\}^T\right)^T$. If we choose for the columns of $\tilde{\mathbf{L}}$ K different columns of the DFT matrix of size $NM_T \times NM_T$, the multiplication of $\mathrm{vec}\{\tilde{\mathbf{S}}^T\}$ by $\tilde{\mathbf{L}}^{\#}$ in (10) reduces to calculating an inverse DFT of $\mathrm{vec}\{\tilde{\mathbf{S}}^T\}$. The elements d_k of $\mathbf{d}$ are then the real parts of certain samples of this inverse DFT.

Specifically, we propose to choose the kth column of $\tilde{\mathbf{L}}$ as $\tilde{\mathbf{p}} \odot \tilde{\mathbf{f}}_{l_k}$, where $\tilde{\mathbf{f}}_{l_k}$ is the l_kth column of the DFT matrix of size $NM_{\mathrm{T}} \times NM_{\mathrm{T}}$ (thus, $l_k \in \{1, \cdots, NM_{\mathrm{T}}\}$), $\odot$ denotes the pointwise or Hadamard product, and $\tilde{\mathbf{p}}$ is a spreading vector that does not depend on k. Eq. (4) then becomes

$$\mathrm{vec}\{\tilde{\mathbf{S}}^T\} = \tilde{\mathbf{L}}\mathbf{d} = \sum_{k=1}^{K} d_k \tilde{\mathbf{p}} \odot \tilde{\mathbf{f}}_{l_k}.$$

The elements of $\tilde{\mathbf{p}}$ are chosen complex-valued with unit magnitude so that each data symbol d_k is spread with equal energy throughout the transmit matrix $\tilde{\mathbf{S}}$. We used $\tilde{\mathbf{p}} = \mathrm{vec}\{\tilde{\mathbf{P}}\}$ where $\tilde{\mathbf{P}}$ is an $N \times M_{\mathrm{T}}$ full-rank matrix with unit-magnitude entries. In our experiments, we observed this choice to result in identifiability.

With this construction of $\tilde{\mathbf{L}}$, (10) can be rewritten as

$$\mathbf{d} = \mathbf{\Pi}\,\mathrm{Re}\{\mathrm{DFT}_{NM_{\mathrm{T}}}^{-1}\{\tilde{\mathbf{p}}^* \odot \mathrm{vec}\{\tilde{\mathbf{S}}^T\}\}\},$$

where the $NM_T \times 2K$ matrix $\mathbf{\Pi}$ consists of the first $2K$ columns of an $NM_T \times NM_T$ permutation matrix that is determined by the index set $\{l_k\}$. More specifically, multiplication of $\mathrm{Re}\{\mathrm{DFT}_{NM_T}^{-1}\{\tilde{\mathbf{p}}^* \odot \mathrm{vec}\{\tilde{\mathbf{S}}^T\}\}\}$ by $\mathbf{\Pi}$ assigns the l_kth sample of $\mathrm{Re}\{\mathrm{DFT}_{NM_T}^{-1}\{\tilde{\mathbf{p}}^* \odot \mathrm{vec}\{\tilde{\mathbf{S}}^T\}\}\}$ to d_k, for $k = 1, \cdots, K$.

Hence, with this code construction the projection onto $\mathcal{A}$ can be implemented by means of an FFT of length NM_T. As a consequence, the computational complexity is only $O(NM_T \log(NM_T))$, instead of quadratic in N as obtained for the general case in Section 3. The degrees of freedom of this code construction lie in the choice of the DFT matrix column indices l_k and of the spreading vector $\tilde{\mathbf{p}}$.

5. USING PILOT SYMBOLS: AFFINE CODES

As is usual for blind demodulation techniques, our POCS algorithm allows to demodulate the data only up to a single real-valued factor. To remove this ambiguity, we have to transmit at least one pilot symbol that is known to the receiver. Therefore, following [4], we extend the linear code structure (3) to an affine structure that contains a $2M_T \times N$ pilot symbol matrix $\mathbf{P}$:

$$\mathbf{S} = \mathbf{P} + \sum_{k=1}^{K} [\alpha_k \mathbf{A}_k + \beta_k \mathbf{B}_k].$$

Equivalently,

$$\mathrm{vec}\{\mathbf{S}^T\} = \mathrm{vec}\{\mathbf{P}^T\} + \mathbf{L}\mathbf{d}. \qquad (11)$$

The entries of $\mathbf{P}$ are zero at the positions where no pilot symbols exist. In [4], conditions are given for the choice and placement of the pilot symbols such that the resulting code allows perfect reconstruction.

When pilot symbols are used, the expression (7) for the orthogonal projection onto $\mathcal{A}$ has to be modified. Let $\mathbf{S}^{(i-1)}$ denote the result of the previous projection onto $\mathcal{B}$. We have to calculate the vector $\mathbf{d}$ for which the model (11), $\mathrm{vec}\{\mathbf{P}^T\} + \mathbf{L}\mathbf{d}$, provides the LS fit to $\mathrm{vec}\{\mathbf{S}^{(i-1)T}\}$:

$$\begin{aligned}\mathbf{d}^{(i)} &= \operatorname*{argmin}_{\mathbf{d}} \left\| \mathrm{vec}\{\mathbf{S}^{(i-1)T}\} - (\mathrm{vec}\{\mathbf{P}^T\} + \mathbf{L}\mathbf{d}) \right\|^2 \\ &= \mathbf{L}^{\#}(\mathrm{vec}\{\mathbf{S}^{(i-1)T}\} - \mathrm{vec}\{\mathbf{P}^T\}).\end{aligned}$$

The orthogonal projection of $\mathbf{S}^{(i-1)}$ onto $\mathcal{A}$ is thus given by

$$\begin{aligned}\mathrm{vec}\{\mathbf{S}^{(i)T}\} &= \mathrm{vec}\{\mathbf{P}^T\} + \mathbf{L}\mathbf{d}^{(i)} \\ &= (\mathbf{I} - \mathbf{L}\mathbf{L}^{\#})\,\mathrm{vec}\{\mathbf{P}^T\} + \mathbf{L}\mathbf{L}^{\#}\mathrm{vec}\{\mathbf{S}^{(i-1)T}\}.\end{aligned}$$

Moreover, we can use the known pilot matrix $\mathbf{P}$ to calculate an initialization $\mathbf{S}^{(0)}$ as (cf. (8))

$$\mathbf{S}^{(0)} = \mathbf{W}^{(0)}\mathbf{X} \qquad \text{with } \mathbf{W}^{(0)} = \mathbf{P}\mathbf{X}^{\#}.$$

This can be expected to be better than random initialization.

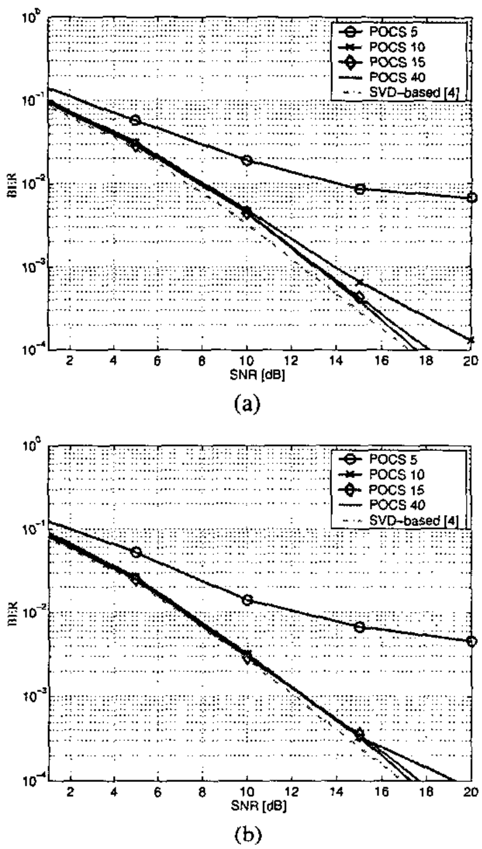

Fig. 2: BER vs. SNR obtained with the POCS algorithm and with the SVD-based algorithm from [4] using (a) a circulant code from [5] and (b) a DFT-matrix based code.

6. SIMULATION RESULTS

While a noise-free MIMO channel has been considered so far, we now present simulation results which demonstrate that our POCS algorithm is quite robust to channel noise.

BER vs. SNR. In Fig. 2, the BER vs. SNR performance[2] of the POCS algorithm is compared to that of the SVD-based algorithm from [4] in the presence of white Gaussian channel noise. The parameters are $M_T = 2$, $M_R = 3$, and $N = 40$ (cf. Example 1 in [4]). Fig. 2(a) shows the BER obtained with the POCS algorithm after 5, 10, 15, and 40 iteration steps[3] and with the SVD-based algorithm. The circulant code from [5] was used for this simulation. It can be seen that with just 15 iteration steps, the POCS algorithm performs similarly well as the SVD-based algorithm, however at a fraction of the computational complexity as will

[2] For determination of the BER, we assumed that the factor c is known.

[3] An iteration step now comprises both the projection onto $\mathcal{A}$ and the projection onto $\mathcal{B}$.

N	32	64	128	256
SVD-based	100	364	1436	5708
POCS	18	35	69	137
POCS-FFT	9	16	29	56

Table 1: Average computational complexity in kflops per transmitted symbol for the SVD-based algorithm, the POCS algorithm, and the POCS algorithm using an FFT.

be demonstrated presently. Fig. 2(b) shows a similar comparison using a DFT-matrix based code construction as described in Section 4. It is seen that here just 10 POCS iterations are sufficient.

Computational complexity. For the DFT-matrix based code, Table 1 compares the computational complexity per transmitted symbol (measured using MATLAB V5.3) of the SVD-based algorithm, the standard POCS algorithm, and the POCS algorithm using an FFT for the projection onto $\mathcal{A}$ (see Section 4). We chose $M_{\mathrm{T}} = 2$, $M_{\mathrm{R}} = 3$, and four different values of the blocklength ($N = 32$, 64, 128, and 256). Both POCS algorithms used 10 iteration steps. It is seen that within the N range considered, the complexity increase with N is roughly quadratic for the SVD-based algorithm, roughly linear for the standard POCS algorithm, and less than linear for the POCS algorithm using an FFT. (Note that now we consider the complexity per symbol, whereas in Sections 2 and 3 we considered the complexity per block of N symbols.) The gain in efficiency obtained with the POCS algorithms is significant for all N but most pronounced for large N.

Code with pilot symbols. Finally, we investigate the effect of using pilot symbols to remove the ambiguity related to the unknown factor c. We consider the POCS algorithm with the modification described in Section 5, a DFT-matrix based code, and parameters $M_{\mathrm{T}} = 2$, $M_{\mathrm{R}} = 3$, and $N = 40$. The pilot symbols were placed at the beginning of each block. Fig. 3 shows the BER vs. the number of pilot symbols for different values of the SNR. For comparison, we also display the BER for the case where c is known. It can be seen that for 3 or more pilot symbols, there is only a moderate loss in BER performance compared to the case of known c.

7. CONCLUSION

We presented a computationally efficient, POCS-based algorithm for the demodulation/decoding of linear dispersion codes in the case of an *unknown* MIMO channel. This algorithm features similar BER vs. SNR performance as the SVD-based algorithm from [4], however at a significantly reduced computational complexity. We also proposed a DFT-matrix based code design that yields a further reduction of complexity, and a modified demodulation algorithm that is suited to the use of pilot symbols.

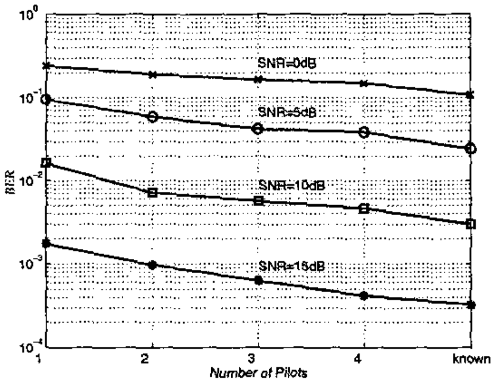

Fig. 3: BER vs. number of pilot symbols obtained with the POCS algorithm using a DFT-matrix based code. (For comparison, the case of known c is shown on the right side.)

REFERENCES

[1] B. Hochwald and T. Marzetta, 'Unitary space-time modulation for multiple-antenna communications in Rayleigh flat fading," *IEEE Trans. Inf. Theory*, vol. 46, pp. 543–564, March 2000.

[2] B. Hochwald and W. Sweldens, 'Differential unitary space-time modulation," *IEEE Trans. Comm.*, vol. 48, pp. 2041–2052, Dec. 2000.

[3] B. L. Hughes, 'Differential space-time modulation," *IEEE Trans. Inf. Theory*, vol. 46, pp. 2567–2578, Nov. 2000.

[4] A. L. Swindlehurst and G. Leus, 'Blind and semi-blind equalization for generalized space-time block codes," *IEEE Trans. Signal Processing*, vol. 50, pp. 2489–2498, Oct. 2002.

[5] A. L. Swindlehurst, 'Simultaneous channel estimation and decoding for diagonal space-time codes," in *Proc. Sensor Array & Multichannel Proc. Workshop*, (Cambridge, MA), pp. 173–177, 2000.

[6] G. Leus, P. Vandaele, and M. Moonen, 'Deterministic blind modulation-induced source separation for digital wireless communications," *IEEE Trans. Signal Processing*, vol. 49, pp. 219–227, Jan. 2001.

[7] P. Stoica and G. Ganesan, 'Space-time block codes: Trained, blind and semi-blind detection." submitted.

[8] H. Artés and F. Hlawatsch, 'Blind equalization of MIMO channels using deterministic precoding," in *Proc. IEEE ICASSP-2001*, (Salt Lake City, UT), pp. 2153–2156, May 2001.

[9] H. Artés and F. Hlawatsch, 'Space-time matrix modulation: Rank-defi cient channels and multi-user case,"in *Proc. IEEE ICASSP-2002*, (Orlando, FL), pp. 2225–2228, May 2002.

[10] H. Artés, G. Matz, D. Schafhuber, and F. Hlawatsch, 'Space-time multicarrier matrix modulation for unknown dispersive MIMO channels," in *Proc. 39th Allerton Conf. Commun., Contr., Comput.*, (Urbana, IL), pp. 466–475, Oct. 2001.

[11] B. Hassibi and B. Hochwald, 'High-rate codes that are linear in space and time," *IEEE Trans. Inf. Theory*, vol. 48, no. 7, pp. 1804–1824, 2002.

[12] P. L. Combettes, "The foundations of set theoretic estimation," *Proc. IEEE*, vol. 81, pp. 181–208, Feb. 1993.