Abstract—Differential space–time modulation (DSTM) using unitary–matrix signal constellations is an attractive solution for transmission over multiple–input multiple–output (MIMO) fading channels without requiring channel state information (CSI) at the receiver. To avoid a high error floor for DSTM in relatively fast MIMO fading channels, multiple–symbol differential detection (MSDD) has to be applied at the receiver. MSDD jointly processes blocks of several received matrix–symbols and power efficiency improves as the blocksize increases. But since the search space of MSDD grows exponentially with the blocksize and also with the number of transmit antennas and the data rate, the complexity of MSDD quickly becomes prohibitive. In this paper, we investigate the application of tree–search algorithms to overcome the complexity limitation of MSDD. We devise a nested MSDD structure consisting of an outer and a number of inner tree–search decoders, which renders MSDD feasible for wide ranges of system parameters. Decoder designs tailored for diagonal and orthogonal DSTM codes are given, and a more power–efficient variant of MSDD, so–called subset MSDD, is proposed. Furthermore, we derive a tight symbol–error rate approximation for MSDD, which lends itself to efficient numerical evaluation. Numerical and simulation results for different DSTM constellations and fading channel scenarios show that the new tree–search MSDD achieves a significantly better performance–complexity tradeoff than benchmark decoders.

Index terms: Differential space–time modulation (DSTM), multiple–input multiple–output (MIMO) fading channels, multiple–symbol differential detection (MSDD), tree–search decoding, sphere decoding.

I. INTRODUCTION

Differential space–time modulation (DSTM) with unitary–matrix–signal constellations is an attractive solution for power– and/or bandwidth–efficient transmission over multiple–input multiple–output (MIMO) fading channels when channel state information (CSI) is not available at the receiver. Original work on DSTM in e.g. [1], [2], [3], [4] considered DSTM over block–wise constant fading (block–fading) channels and conventional differential detection (CDD) based on two consecutively received symbols. As shown in e.g. [2, Section IV.C], CDD incurs a loss of 3 dB in power efficiency compared to ST transmission with perfect CSI in block–fading channels. In case of relatively fast fading, CDD further suffers from a high error floor due to channel variations from one symbol to the other and DSTM performs even worse than single–antenna differential transmission [5]. To overcome this limitation, multiple–symbol differential detection (MSDD) for DSTM was proposed in [5], [6]. Analogous to MSDD for single–antenna differential phase–shift keying (DPSK) (cf. e.g. [7]), \( N – 1 \) data symbols are jointly detected based on \( N \) consecutively received symbols. The performance of MSDD improves for increasing observation window size \( N \) and approaches that of coherent detection with perfect CSI. As its complexity grows exponentially with \( N \), a number of suboptimal MSDD algorithms were designed for particular DSTM constellations and/or values of \( N \), cf. e.g. [8], [9], [10]. Furthermore, suboptimal decision–feedback differential detection (DFDD) and trellis–based noncoherent sequence detection were devised for DSTM with diagonal constellations [1], [2] and with orthogonal designs [11], [3] in [5], [12] and [13], respectively.

Recently, the authors presented an optimal maximum–likelihood (ML) MSDD algorithm for single–antenna DPSK [14]. This algorithm is a depth–first tree search or sphere decoding (SD) algorithm and was referred to as multiple–symbol differential sphere decoding (MSDSD) in [14]. The complexity of MSDSD was shown to be comparable to that of DFDD in most cases. In [15], a similar, Fano decoding algorithm for MSD of single–antenna DPSK was investigated. In this paper, we propose the application of tree–search decoding for DSTM transmission over MIMO fading channels and MSDD. In this context, we make the following contributions.

- Three tree–search decoders are devised for MSDSD. These include MSDSD and Fano decoding similar to the single–antenna schemes from [14], [15], as well as a new MSDSD scheme using a Fano–type metric for quicker termination of the search.
- The size of the DSTM constellation grows exponentially with the number of transmit antennas \( N_T \) and the data rate \( R \). To avoid a corresponding increase in decoding complexity, we propose to nest fast CDD algorithms into the outer tree–search MSD. Whereas sphere decoding is typically proposed for conventional CDD, cf. e.g. [4], [12], we show that stack decoding is better suited when CDD is integrated into MSDSD.

1 In the context of this paper, the terms “decoding” and “detection” refer to the same procedure and are used interchangeably. When referring to the signal set of DSTM, the terms “constellation” and “(space–time) code” are used interchangeably.
Considering a continuous MIMO fading channel model, we provide a tight symbol–error rate (SER) approximation for MSDD, which lends itself to efficient numerical evaluation. Thereby, we consider the individual SERs for the \( N - 1 \) consecutive data symbols jointly processed in one MSDD block. Numerical results show that the SERs for the individual symbols in one block vary significantly. This motivates the design of a more power–efficient variant of MSDD, which we refer to as subset MSDD.

- We present a thorough comparison of the proposed tree–search decoding schemes in terms of SER performance and computational complexity considering different DSTM constellations. Numerical and simulation results confirm that considerable improvements in power–efficiency over DFDD and CDD are achieved while decoding complexity remains moderate and, in fact, comparable to that of DFDD.

Organization: Section II introduces the DSTM system model. The tree–search MSDD algorithms are developed and optimized for DSTM in Section III. Expressions for the SER of MSDD are derived and subset MSDD is devised in Section IV. Numerical and simulated SER–performance and complexity results are presented and discussed in Section V, and conclusions are given in Section VI.

Notation: Bold upper case \( X \) and lower case \( x \) denote matrices and vectors, respectively. (\( \cdot \))^T, tr\{\cdot\}, |\cdot|, det\{\cdot\}, \( \otimes \), and \( \mathcal{E}\{\cdot\} \) denote Hermitian transposition, transposition, trace, Frobenius norm, determinant of a matrix, Kronecker product, and expectation, respectively. \( I_L \) is the \( L \times L \) identity matrix and \( \text{diag}\{X_1, \ldots, X_L\} \) is an \( LM \times L \) block–diagonal matrix with the \( M \times N \) matrices \( X_1 \) on its main diagonal. A symmetric \( L \times L \) Toeplitz matrix is defined by toeplitz\{\( x_1, \ldots, x_L \)\}.

II. SYSTEM MODEL

We consider a transmission scheme using \( N_T \) transmit and \( N_R \) receive antennas. At the transmitter \( N_T R \) bits are mapped to \((N_T \times N_T)\)–dimensional unitary matrices \( V[k] \) which are taken from a set \( \mathcal{V} \triangleq \{V^{(l)} \mid l \in \{1, \ldots, L\}, L \triangleq 2^{N+R}\} \). \( R \) is the data rate in bit per channel use. In order to facilitate noncoherent detection the data symbols \( V[k] \) are differentially encoded into transmit symbols

\[
S[k] = V[k]S[k - 1], \quad S[0] = I_{N_T}.
\]

At time \( \kappa = kN_T + i \), the transmitter radiates from antenna \( j \) the element \( s_{i,j}[k] \) in the \( i \)th row and \( j \)th column of \( S[k] \), \( 1 \leq i \leq N_T \), \( 1 \leq j \leq N_T \). The transmit power is independent of \( N_T \) at all times: \( \sum_{i=1}^{N_T} |s_{i,j}[k]|^2 = E_0R, \quad 1 \leq i \leq N_T, \) where \( E_0 \) is the energy per information bit and \( T \) is the modulation interval.

We assume a frequency–nonselective MIMO Rayleigh fading channel. The equivalent complex baseband (ECB) received signal \( r_{i,j}[k] \) at time \( \kappa = kN_T + i \) and antenna \( j \) is given by

\[
r_{i,j}[k] = \sum_{\nu=1}^{N_T} s_{i,\nu}[k]h_{\nu,j}[k] + n_j[k], \quad 1 \leq i \leq N_T, \quad 1 \leq j \leq N_R.
\]

III. TREE–SEARCH MULTIPLE–SYMBOL DIFFERENTIAL DECODING

In this section, we propose and design tree–search algorithms for efficient MSDD. To this end, we provide a suitable representation of the ML MSDD decision rule in Section III–A, based on which the tree–search decoding algorithms are developed in Section III–B. We then present inner decoders which are nested into the outer tree–search decoder and optimized for different DSTM constellations in Section III–C.

A. Maximum–Likelihood Multiple–Symbol Differential Detection (ML MSDD)

ML MSDD processes blocks \( \hat{R}[k] \) of \( N \) consecutively received matrix symbols to find estimates for the \( N - 1 \) data symbols \( V[k] \triangleq [V^T[k-N+2], \ldots, V^T[k]]^T \) corresponding to the \( N \) transmit symbols \( S[k] \triangleq [S^T[k-N+1], \ldots, S^T[k]]^T \). In analogy to the single–antenna case (cf. e.g. [7]) consecutively processed blocks \( \hat{R}[k] \) overlap by at least one matrix
symbol, i.e. the observation window of length $N$ moves forward by at most $N - 1$ symbols at a time. For the sake of readability, in the following we omit the reference [k] to time and address submatrices of the above block–matrices via subscripts with $X_i$ being the $i$th submatrix of $X$ and $1 \leq i \leq (N$ or $N - 1$).

Using the QSF model (4) and the fact that $S_D$ is an unitary matrix, the ML MSDD decision rule can be derived as [13, Section V]

$$\hat{V} = \arg\min_{V \in V^{N-1}} \left\{ \text{tr}\left\{ R^H S_D \left( C^{-1} \otimes I_{N_T} \right) S_D^H R \right\} \right\}, \tag{5}$$

where $C \triangleq (\Psi_{hh} + \sigma^2 \delta)I_N$ and $\Psi_{hh} \triangleq \text{toeplitz}\{\psi_{hh}[0], \psi_{hh}[N_T], \ldots, \psi_{hh}[N - 1]N_T\}$. Using the Cholesky factorization $C^{-1} = U^H U$ with upper triangular matrix, the ML MSDD decision rule can be derived as [13, Section V]

$$\hat{V} = \arg\min_{V \in V^{N-1}} \left\{ \sum_{n=1}^{N-1} \left\| V_n \bar{R}_n, n + X_n \right\|^2 \right\}, \tag{6}$$

with

$$X_n \triangleq S_{n+1} N \sum_{j=n+1}^{N} S_{j}^H \bar{R}_{n,j}, \tag{7}$$

where $\bar{R}_{n,j} \triangleq u_{n,j} R_j, 1 \leq n \leq N - 1, n \leq j \leq N$, and $u_{n,j}$ is the element of $U$ in row $n$ and column $j$. It is worth pointing out that $u_{n,j} = \frac{\sigma^2}{\sigma^2 n - n}$, where $\sigma^2$ and $\sigma^2 n$ denote the $j$th coefficient of the $n$th order linear backward minimum mean–

squared error (MMSE) predictor for the discrete time random process $H[k] + N[k]$ and the corresponding error variance, respectively, and $\delta = 1, \forall n$, cf. e.g. [18].

We can then compute the total error metric $E$ for all $2^{(N-1)N_T R} \hat{V} \left( S_N = I_{N_T} \right.$ can be chosen without loss of optimality), i.e. decoding complexity would be exponential in $N$, $N_T$, and $R$.

**B. Efficient Tree–Search Decoding Algorithms — Outer Decoder**

It can be observed that the ML metric in (6) is a sum of $N - 1$ non–negative scalar terms

$$\delta^2_n \overset{\Delta}{=} \left\| V_n \bar{R}_n, n + X_n \right\|^2, \quad 1 \leq n \leq N - 1, \tag{8}$$

which through $X_n$ and (1) depend on symbols $V_j, n + 1 \leq j \leq N - 1$. Thus, ML MSDD constitutes a tree–search problem and $\hat{V}$ from (6) corresponds to the best path in the tree. For an efficient (fast) tree search for MSDD, we concentrate on the application of depth–first search (DFS) and best–first search (BFS) algorithms, which achieve the best performance–complexity tradeoff (cf. e.g. [19] for coherent MIMO detection).

1) Sphere Decoding Algorithm — MSDSD: Sphere decoding algorithms are DFS–type algorithms. We refer to the application of SD to accomplish MSD as MSDSD (cf. [14] for DPSK).

To formulate the SD algorithm, let us define

$$d^2_n \overset{\Delta}{=} \sum_{k=n}^{N-1} \left\| V_n \bar{R}_n, + X_n \right\|^2 = d^2_{n+1} + \delta^2_n, \quad 1 \leq n \leq N - 1, \tag{9}$$

where $d^2_N = 0$ and $d^2_1$ is the ML metric in (6). Starting at $n = N - 1$, the SD algorithm selects candidates for symbols $V_n$ based on tentative decisions $V_j = V_j, n + 1 \leq j \leq N - 1$, and continues to decrement $n$ as long as the current metric $d_n$ does not exceed a given maximum metric (radius) $\rho$, i.e.

$$d_n \leq \rho. \tag{10}$$

If the decoder reaches $n = 1$, the metric of the currently best candidate $\hat{V}$ is used to further reduce the size of the search space by updating $\rho = d_1$. If $d_n$ exceeds $\rho$ for any value of $n$, $n$ is incremented and a new candidate for $V_n$ is examined. If the decoder returns to $n = N$, i.e. if $d_{N-1} > \rho$, it means that there are no further candidates inside the current sphere and that the ML solution $\hat{V}$ has been found. For the ordering of candidates for any $V_n$ we employ the Schnorr–Euchner (SE) enumeration strategy, i.e. we check candidates in order of increasing $\delta_n$, as this allows for an initialization with $\rho \rightarrow \infty$ and an (usually) fast convergence to the ML solution, cf. e.g. [14], [20], [21].

2) Sphere Decoding with Fano–Type Metric — MSDSD–FM: The sphere decoding structure described above is very efficient in finding the ML solution in moderate to high signal–

receiver ratios (SNRs). However, towards lower SNR the normalized predictor coefficients $p_j(\sigma^2 n)$ become very small and $d^2_n$ becomes less dependent on tentative decisions $V_j, n + 1 \leq j \leq N - 1$. In other words, $d^2_n$ resembles a sum of $N - n$ CDD metrics and thus scales almost linearly with $N - n$. It is therefore advisable to adjust the metric used in SD to take into account the length $N - n$ of the considered tree path, i.e. to introduce a bias $b$. This idea was first proposed by Fano for sequential decoding of convolutional codes (cf. e.g. [22]), and hence we refer to it as MSDSD with Fano–type metric (MSDSD–FM). MSDSD–FM uses the same search strategy as MSDSD but with Fano–type path metric

$$d^2_n \overset{\Delta}{=} d^2_n + (n-1)b. \tag{11}$$

with $d^2_n$ from (9). We propose to use the expected value of $\delta^2_n$ given $V_j = V_j, n \leq j \leq N - 1$, as bias:

$$b \overset{\Delta}{=} \mathbb{E}\left\{ \delta^2_n \right\} V_j = V_j, n \leq j \leq N - 1 \right\} = N_T N_R, \tag{12}$$

which leads to a simple implementation. The result in (12) is due to the fact that $-X_n$ is the MMSE estimate for $V_n \bar{R}_n, n + 1 \leq j \leq N - 1$, and that the filter coefficients are normalized by the corresponding error variance (cf. (6)). It should be noted that, different from MSDSD, MSDSD–FM is suboptimal in that the ML MSDSD solution is not necessarily found. Furthermore, it is worth mentioning that a SD with Fano–type metric was presented for coherent detection of space–time codes in [23].

3) Initial Search Radius: When using the SE enumeration, no initial radius is required, i.e. $p_{init} \rightarrow \infty$ can be assumed. Accordingly, our first strategy is to initialize the MSDSD algorithm with a very large value of $p_{init}$. The disadvantage of
this strategy is a potentially slow convergence or equivalently high complexity, in cases where the decoder makes false tentative decisions for symbols $V_n$ with $n$ close to $N$, before reaching $n = 1$ for the first time. Therefore, our second strategy is to use a relatively small $\rho_{\text{init}}$ for initialization and, in case no solution $V$ is found for this radius, to increase it gradually by steps of $\rho_{\text{init}}$, cf. e.g. [21]. Since for MSDD–FM the expected metric of the correct solution equals $E[||\hat{V}||] = (N - 1)N_T N_R c$, we choose the squared start radius $\rho^2_{\text{init}}$ proportional to that value:

$$\rho^2_{\text{init}} = (N - 1)N_T N_R c, \quad c > 0.$$  

(13)

The value of the constant $c$ influences the overall search time. Simulation results presented in Section V confirm that the use of a finite initial radius is beneficial especially for DSTM with large constellations.

4) Fano Decoding Algorithm — MSDD–Fano: The Fano decoding algorithm is a BFS type algorithm. It has recently been applied to MSDD of single–antenna DPSK [15] and to coherent MIMO detection [19]. In this paper, we propose its application to MSDD for DSTM using the metric of (11) and refer to the resulting algorithm as MSDD–Fano. We do not repeat the basic description of the basic Fano algorithm, whose details can be found in, e.g. [22].

C. Application to Different Signal Constellations — Inner Decoders

MSDD, MSDD–FM, and MSDD–Fano break the exponential dependence of decoder complexity on $N$. In order to also break exponential dependence of decoder complexity on $N_T$ and $R$ it is necessary to give some thought on the search for individual candidate symbols $V_n$. In this regard it is important to note that minimizing $\delta_n$ is equivalent to the problem of finding the ML solution for CDD. This observation allows us to implement MSDD for DSTM using a nested structure of an outer and $N - 1$ identical inner decoders. The outer decoder, which solves (6), initializes an inner decoder at stage $n$, $1 \leq n \leq N - 1$, with matrices $\tilde{R}_{n,n}$ and $X_n$ to find a new candidate $\hat{V}_n$ that minimizes $\delta_n$. It further provides the inner decoder with (a) a list of candidates that have been examined previously given the same set of tentative decisions $V_{j'}$, $n + 1 \leq j' \leq N - 1$, and thus to be excluded from the search, and (b) a start radius $r$ to force $\delta_n < r$ such that (10) is still fulfilled. In our investigations we considered diagonal (cyclic) and dicyclic group codes [24] and orthogonal [11] and Cayley [4] non–group codes. Due to space limitations however, we present details only for diagonal and orthogonal constellations.

1) Diagonal Constellations: Diagonal constellations are defined by the set [1], [2]

$$V_D \triangleq \left\{ \text{diag}\left(\begin{bmatrix} \frac{2\pi}{c_1}, \ldots, \frac{2\pi}{c_c} \end{bmatrix}, \ldots, \begin{bmatrix} \frac{2\pi}{c_{N_T}} \end{bmatrix}\right) \mid l \in \{0, \ldots, L - 1\} \right\}.$$  

(14)

A straightforward approach, which we will refer to as full search (FS) inner decoding, is to compute $\delta_n$ for all elements of $V_D$ and to process them in order of increasing $\delta_n$. In this case, MSDD benefits from the efficiency of the outer tree–search algorithm, but its complexity is still exponential in $N_T$ and $R$.

In order to overcome this exponential dependence of complexity on $N_T$ and $R$ we turn to a more sophisticated strategy using a lattice–based CDD algorithm proposed in [25] (cf. also [12]). We will refer to this approach as inner lattice decoding (LD). Using the cosine–approximation for small arguments, the problem of finding the minimizer for $\delta_n$ can be turned into a (degenerated) $N_T$–dimensional lattice–decoding problem of the form

$$\hat{x} = \arg\min_{x \in \mathbb{Z}^{N_T}} ||Bx - t||,$$

(15)

where the $N_T \times N_T$ matrix $B$ has non–zero entries $b_{i,j}$ only in the first column $j = 1$ and on the main diagonal $i = j$, and $V_n = \text{diag}\{\begin{bmatrix} \frac{2\pi}{c_1}, \ldots, \frac{2\pi}{c_{N_T}} \end{bmatrix}\}_\text{mod}(x,t)$ (see [25], [12] for details). Due to the cosine–approximation, $V_n$ is not necessarily the optimal solution, but as we will see in Section V only small performance degradations result. Since $\hat{x}$ in (15) and thus $V_n$ can be obtained from a tree search, we propose two tree–search algorithms, which are particularly suited for efficient inner decoding.

Sphere Decoding Algorithm: Due to the sparse, lower–triangular structure of $B$ in (15), a particularly efficient sphere decoder can be applied. The decoder may use the SE strategy for finding candidates for $x_1 \in \mathbb{Z}$ and increase $i$, $2 \leq i \leq N_T$, as long as $\mu^2_i \triangleq ||x_1 \in \mathbb{Z}||^2 + \sum_{j=2}^i \|b_{j,1} x_1 + b_{j,2} x_2 - t_j\| < r^2$ with $\tilde{x}_j \triangleq \{t_j - b_{j,1} x_1\} / b_{j,2}$. If $\mu_{N_T} < r$ the current $x_1$ is stored as tentative decoding result and the decoder checks the next candidate for $x_1$ with updated $r = \mu_{N_T}$. The decoder terminates, when $|b_{1,1} x_1 - t_1| > r$. Thus the decoder actually searches only one dimension of the alleged $N_T$–dimensional search space. We found that this implementation of the inner sphere decoder is more efficient than performing a QR–decomposition and subsequent SD as was advocated in [12] for CDD. The reason for this is that in MSDD previously examined candidates are excluded at the very beginning of the search as they correspond to $x_1$.

Stack Decoding Algorithm: One drawback of the SD algorithm is that the search restarts from $i = 1$ and $x_1 = [t_1 / b_{1,1}]$ every time the inner decoder is called to provide the next best candidate $V_n$ given the same tentative decisions $V_{j'}$, $n + 1 \leq j' \leq N - 1$. The stack algorithm (cf. e.g. [22]) seems to be an interesting alternative for the inner decoder, since it maintains a sorted list of examined paths and decoding can be continued if the inner decoder is called again.

As the regular stack algorithm is not directly applicable for solving (15), because $x_i$, $1 \leq i \leq N_T$, are not from a finite set, we propose a modified stack decoder. Instead of replacing the path $x^{(i)} = [x_1, \ldots, x_L]$ currently at the top of the stack with all of its extensions $x^{(i+1)} = [x^{(i)}, x_{i+1}]$, we only consider its best extension $x^{(i+1)}$ and, due to the sparse structure of the matrix in (15), the next candidate for $x_i$ only when $i = 1$ according to the SE strategy. The search can be terminated without loss of optimality, if the metric of the path at the top of the stack exceeds the start radius $r$. As for the regular stack algorithm, the first path of length $N_T$ to reach the top of
the stack is the optimal path corresponding to the (next–)best candidate for $V_n$. It remains to store the stack and to continue with this stack when the next candidate is required. While, theoretically, limitation of the stack size may lead to decoding errors, we found that limiting the maximum stack size to $2L$ does not incur a noticeable performance degradation.

2) Orthogonal Designs: A second important class of non–group DSTM codes are orthogonal designs (ODs). In particular, we consider DSTM based on Alamouti’s code \cite{al4}, where symbols in fast fading environments it can be expected that symbols error rates this intuitive reasoning, we consider the individual symbol–error rates $\tilde{P}_e$ of ML MSDD (Section IV–A). In particular, we consider the MSDD for DSTM with ODs has been presented in \cite{al1}. There, only a subset of all $N$ symbols can however not be extended to arbitrary fading channels as considered in this paper.

For ODs, it is relatively straightforward to show that $\delta_n^2$ can be expressed as

$$\delta_n^2 = \gamma_n + \text{Re} \{ a_n^* \} + \text{Re} \{ b_n^* \}, \quad 1 \leq n \leq N - 1,$$

with $a_n = \sqrt{2} \sum_{j=1}^{N} \tilde{r}_{n,n,1,j} \tilde{r}_{n,n,2,j} \nonumber + \tilde{r}_{n,n,2,j} \tilde{r}_{n,n,1,j}$ and $b_n = \sqrt{2} \sum_{j=1}^{N} \tilde{r}_{n,n,1,j} \tilde{r}_{n,n,2,j} \nonumber - \tilde{r}_{n,n,2,j} \tilde{r}_{n,n,1,j}$. Consequently, outer and inner decoder for MSDD can be integrated into one decoder searching a $(2N - 2)$–dimensional tree of $\sqrt{L}$–PSK symbols. If the MSDD algorithm from Section III-B.1 is applied, this decoder is a direct extension of single–antenna MSDD from \cite{al1} to DSTM. way as for single–antenna MSDD.

It is worth mentioning that recently, a similar description of MSDD for DSTM with ODs has been presented in \cite{al1}. There, for the special case of $L = 16$ (ODs with 4PSK symbols) and “block fading”, the detection problem is formulated as a $4(N – 1)$–dimensional search in variable symbols. The scheme proposed in \cite{al1} can however not be extended to arbitrary fading channels as considered in this paper.

IV. Symbol–Error Rate Analysis and Subset MSDD

In this section, we first derive an approximation for the SER of ML MSDD (Section IV-A). In particular, we consider the individual SERs for the $N – 1$ data symbols in each block $V$. Evaluation of these individual SERs suggests that decoding of only a subset of all $N – 1$ symbols can be greatly beneficial. The corresponding subset MSDD algorithm is presented in Section IV-B.

A. SER Analysis

It is intuitive that due to different correlations between the received samples in $\tilde{R}$, symbol decisions on the $N – 1$ data symbols in $\tilde{V}$ are not equally reliable. Especially in relatively fast fading environments it can be expected that symbols located in the center of the observation window can be detected more reliably than those at the edges. In order to substantiate this intuitive reasoning, we consider the individual symbol–error rates $\tilde{P}_e$ for $V_n$, $1 \leq n \leq N - 1$.

The pairwise error probability $\tilde{P}_e(\tilde{S} \rightarrow \hat{S})$ for detecting $\hat{S}$ while $\tilde{S} \neq \hat{S}$ was transmitted can be expressed as (cf. e.g. \cite{al5}, Section III)

$$\tilde{P}_e(\tilde{S} \rightarrow \hat{S}) = \sum_{R \neq \hat{S}} \text{Res} \left( \frac{1}{s} \prod_{i=1}^{N} \left( 1 + s \lambda_i \left( \Psi_{rr}(\hat{S}^\mathcal{P}) \right)^{-N} \right) \right),$$

where the summation in (18) is taken over all residues corresponding to poles located in the right–hand (RH) side of the complex $s$–plane. $\lambda_i(X)$ denotes the $i$th eigenvalue of $X$, and $\Psi_{rr}(\hat{S}^\mathcal{P})$ denotes the autocorrelation matrix of the regular fading–plus–noise process (cf. (2)). Depending on the multiplicities of the eigenvalues $\lambda_j(\Psi_{rr}(\hat{S}^\mathcal{P}))$, analytical or numerical evaluation of (18) may be preferable, e.g. \cite{al5, al5a, al5b, al5c}.

With this, the symbol–error rate SER$_n$ for $V_n$ can be upper bounded using the union bound, averaged over all $L^N$ transmit sequences:

$$\text{SER}_n \leq \frac{1}{L^N} \sum_{\hat{S}} \sum_{V_{n+1} \neq V_n} \tilde{P}_e(\tilde{S} \rightarrow \hat{S}).$$

Evaluating (23) is computationally tractable only for small constellations and observation window sizes $N$. In order to obtain a simpler approximation for $\text{SER}_n$, we examine the PEP of (18) more closely. It can be shown that it depends on $S$ and $\hat{S}$ only through the matrix

$$P(\tilde{C} \otimes I_{N_T}) \tilde{S} \tilde{D} \left( \tilde{C} \otimes I_{N_T} \right) \tilde{S} \tilde{D} - \left( \tilde{C} \otimes I_{N_T} \right) \tilde{S} \tilde{D} \tilde{S} \tilde{D}^{-1} \left( \tilde{C} \otimes I_{N_T} \right),$$

where we introduced $P \triangleq \tilde{S}_D^{\mathcal{P}} \tilde{S}_D$. 

1) Group Constellations: For $L$ being a power of two, diagonal (or cyclic) and dicyclic codes \cite{al15} fully represent full–rank unitary group constellations. In both cases $P$ is a sparse matrix with a single one in each row. Whereas $P$ is constant and independent of $S_D$ for diagonal constellations, there are $2^N$ different matrices $P$ in case of dicyclic constellations. This means that the sum of $L^N$ matrices $S_D$ can be partitioned into $K$ equivalence classes $\tilde{S}_D^{\mathcal{P}}$ with respect to $P$, $1 \leq k \leq K$, where $K = 1$ and $K = 2^N$ for diagonal and dicyclic constellations, respectively. Furthermore, since $\tilde{S}_D^{\mathcal{P}} = \text{diag}\{ \tilde{S}_1^{\mathcal{P}}, \ldots, \tilde{S}_N^{\mathcal{P}} \}$ with $\tilde{S}_n \in \mathcal{V}$, $1 \leq n \leq N$, for group constellations, the inner sum of (23) is independent of which class representative $\tilde{S} \in \tilde{S}_D^{\mathcal{P}}$ is chosen. We can thus limit the outer summation to only $K \ll L^N$ terms ($K = 1$ or $K = 2^N$).
To reduce the number of terms of the inner sum in (23), we propose to take only the dominating instead of all $L^{N-1} - 1$ relevant error events into account. Following a similar reasoning as in [28] for single-antenna DPSK and utilizing the results of [17] for the design of unitary space–time constellations, these dominant error events are found to correspond to matrix pairs $\{S, \breve{S}\}$ with the highest correlation $\zeta \triangleq ||S^H \breve{S}||$. A brief examination of $\zeta$ reveals that error events with only one non–identical transmit matrix $\breve{S}_n \neq S_n$ such that

$$\zeta_n \triangleq \text{Re} \left\{ \text{tr} \left\{ \hat{S}_n^H S_n \right\} \right\}, \ 1 \leq n \leq N,$$  

is maximized also maximize $\zeta$. Collecting all matrices $\hat{S} = [S_1^T \ldots S_{n-1}^T S_n^T \ldots S_{N-1}^T]^T$ with $\breve{S}_n \neq S_n$ maximizing $\zeta_n$ in sets $\hat{S}_n$, $1 \leq n \leq N$, we can approximate SER$_n$ by

$$\text{SER}_n \approx \frac{1}{K} \sum_{n=1}^{N} \left( \sum_{S \in \hat{S}_n} \text{PEP}(S \rightarrow \hat{S}) + \sum_{\breve{S} \in \hat{S}_{n+1}} \text{PEP}(\breve{S} \rightarrow \hat{S}) \right)$$

for $1 \leq n \leq N - 1$.

2) Non–Group Constellations: For non–group constellations, e.g. orthonogonal [11] and Cayley [4] codes, a simplification of (23) based on equivalence classes with respect to $P$ is not possible. Furthermore, it is not guaranteed that the quadruple $Q_n \triangleq \{S_n, \breve{S}_n, S_{n+1}, \breve{S}_{n+1}\}$ with $\breve{S}_n \neq S_n, \breve{S}_{n+1} \neq S_{n+1}$ maximizing (25) is admissible, i.e. that there is a $V_n \in V$ such that $\hat{S}_{n+1} = V_n \hat{S}_n$. Hence, the correlation $\zeta_n$ (25) is not necessarily an appropriate indicator for the dominant error events for every realization of $\hat{S}$. However, we found from numerical evaluations that the SER of non–group codes is approximated quite accurately using (26) with a few candidates $\hat{S}$ for which $Q_n$ is admissible and $S_n$ according to (25), $1 \leq n \leq N - 1$. For the performance evaluation in Section V, we use this approximation with the $L$ admissible matrices $\hat{S} = [I_{NT_1}, (V(i))^T, I_{NT_1}, (V(i))^T, \ldots]^T, 1 \leq l \leq L$.

3) Evaluation: At this point, it is illustrative to consider SER$_n$ for the example of diagonal DSTM with parameters $N_T = 3, R = 1, B_0 T = 0.03$, and $N_R = 1$. Fig. 1 shows numerical results from (26) and simulation results for the SNR in terms of $10 \log_{10}(E_b/N_0)$ required to achieve, respectively, SER$_n \approx 10^{-5}$ (solid lines) and the average error rate SER = $\frac{1}{N-1} \sum_{n=1}^{N-1} \text{SER}_n = 10^{-5}$ (dashed lines) as function of the position $n$, $1 \leq n \leq N - 1$, for different window sizes $N$. MSDD is implemented using MSDD with FS inner decoding (MSDD–FS). Also included is the SER for coherent detection assuming perfect CSI.

First, we observe a good agreement between the SER approximation from (26) and the simulated SER. Second, it can be seen that the individual error rates SER$_n$ are almost identical for symbols $V_n$, $2 \leq n \leq N - 2$, but significantly deteriorate for symbols at the edges of the observation window. More specifically, there are differences of $5 - 8$ dB in power efficiency when comparing edge and non–edge positions.

**B. Subset MSDD**

The observations from Fig. 1 strongly suggest a variant of MSDD, which we refer to as subset MSDD. For subset MSDD, only the decoding decisions for the $N' - 1 \leq N' \leq N - 1$ symbols $V_n$ located in the middle of the observation window, i.e. $(N - N')/2 < n < (N + N')/2$, $N \pm N'$ even, are passed to the sink, whereas the other $N - N'$ estimates are discarded. Consequently, the observation window must slide forward by $N' - 1$ received symbols at a time and the decoding complexity is increased by a factor of $(N - 1)/(N' - 1)$.

For the example considered above, the error–rate results in Fig. 1 suggest that only the decisions corresponding to the very edge symbols at positions $n = 1$ and $n = N - 1$ should be discarded, i.e. $N' = N - 2$ should be used. When comparing the resulting average SER of subset MSDD with that of MSDD, we observe from Fig. 1 gains in power efficiency of 6.8 dB, 3.0 dB, and 1.5 dB for $N = 4, N = 10$, and $N = 20$, while complexity is increased by a factor of $3, 1.29,$ and $1.12$, respectively. It is also interesting to note that subset MSDD with $N = 10$ and $N' = 8$ approaches coherent detection with perfect CSI within 2.0 dB, which is quite remarkable considering the low error rate SER = $10^{-5}$ and the large normalized fading bandwidth $B_0 T = 0.03$.

**V. PERFORMANCE EVALUATION**

In this section, we present numerical and simulated SER results (Section V-A) and a complexity comparison (Section V-B) for the different proposed MSDD implementations. Table I provides an overview of the different MSDD algorithms (see Sections III-B, III-C and IV-B), where check–marks indicate (a) that the subset MSDD variant can be applied, which we denote by a suffix “–S” in the following, and (b) that a particular inner decoder is applicable, which is denoted by an (additional) suffix “–FS” (all constellations), “–LD” (only for diagonal constellations), respectively. As benchmark
algorithms, we also consider (a) CDD \((N = 2)\), (b) DFDD [5], [12], and (c) the branch–intersect–detect MSD decoder (MSDD–BID) of [10], which is a suboptimal MSDD algorithm for diagonal constellations. Unless specified otherwise, the channel model according to (2) is applied for simulation.

We present SER results for three illustrative examples. In case of DSTM with diagonal constellations we used parameters \([c_1, \ldots, c_{N_c}]\) from [2, Table I].

1) Diagonal Constellation with \(R = 1\), \(N_T = 3\), \(N_R = 1\), and MSDD with \(N = 10\): Fig. 2 compares various decoders (with FS inner decoding) in terms of the SNR required to achieve \(\text{SER} = 10^{-5}\) as function of the normalized fading bandwidth \(B_h T\). Both numerical results according to Section IV-A (solid lines) and simulation results (markers) are plotted. As reference curves, the SNR for MSDD–FS under block–fading \((B_h T = 0)\) and for coherent detection with perfect CSI are also shown.

First, we note the good match of the SER approximation from Section IV-A and the simulated SER. The slight fluctuations of the numerically obtained curves are due to the particular fading autocorrelation according to the Bessel function. Second, we observe that CDD–FS suffers from a relatively high error floor already in moderately fast fading with \(B_h T \gtrsim 0.006\). The performance of MSDD–BID approaches that of ML MSDD for \(B_h T \lesssim 0.01\), but rapidly deteriorates as \(B_h T\) grows. MSDD–FS is considerably more robust to fast fading than MSDD–BID and also outperforms DFDD by about \(-2 \text{ to } -3 \text{ dB}\) depending on the fading bandwidth. Suboptimal MSDD–Fano–FS performs within \(-3 \text{ to } -2 \text{ dB}\) regardless of the \(N'\) of \(0\). Finally, it can be seen that subset MSDD (MSDD–S–FS) significantly improves performance in the fast fading regime. Almost all the gain in power efficiency is already accomplished with \(N' = 8\), which corresponds to a moderate complexity increase by a factor of \(1.29\) compared to MSDD with \(N = 10\).

In Fig. 3, we evaluate the performance with inner LD for an exemplary fading bandwidth of \(B_h T = 0.03\). MSDD, MSDD–S, and DFDD with \(N = 10\) and CDD are considered, and LD inner decoding (dashed lines) is compared with FS inner decoding (solid lines). It can be seen that the cosine–approximation involved in LD causes only small performance degradations in the order of \(-0.2 \text{ to } -0.3 \text{ dB}\) regardless of the

<table>
<thead>
<tr>
<th>MSDD Algorithm</th>
<th>Subset MSDD (Sec. IV-B)</th>
<th>Full Search (–FS)</th>
<th>Lattice Decoder (LD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSDSD, (Sec. III-B.1)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>MSDSD-FM, (Sec. III-B.2)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>MSDDD–Fano, (Sec. III-B.4)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>CDD</td>
<td>—</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>MSDD–BID [10]</td>
<td>(not considered in [10])</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

**TABLE I**

**Summary of algorithms for differential detection for unitary DSTM.**
particular detector. We also observed (not shown in Fig. 3) that even smaller performance losses occur at higher rates \( R > 1 \), since the cosine-approximation becomes more accurate.

2) Diagonal Constellation with \( R = 2 \), \( N_T = 3 \), \( N_R = 1 \), \( B_0T = 0.03 \), and MSDD with \( N = 6 \): The power efficiency of MSDSD with Fano-type metric (MSDSD–FM) is compared to optimal MSDSD and MSDSD–Fano in Fig. 4. As FS inner decoding is computationally complex for this scenario with \( L = 64 \), we consider LD in all cases. The respective curves for CDD–LD, DFDD–LD, and MSDD–BID are also included for comparison. MSDSD–FM–LD clearly outperforms CDD–LD and MSDSD–BID, which suffer from a high error floor in this fast fading scenario. We further observe that the use of the suboptimal Fano-type metric results in relatively small performance losses of \( 0.5 - 1.0 \text{ dB} \). As expected, the performance of MSDSD–FM–LD is very similar to that of MSDSD–Fano–LD. Finally, MSDSD–FM–LD and especially MSDSD–FM–S–LD achieve significant gains over DFDD–LD, which has been studied in detail in [12].

3) Orthogonal Design (OD) with \( R = 1 \), \( N_T = 2 \), \( N_R = 1 \), and MSDD with \( N = 10 \): Fig. 5 shows the performance of DSTM with ODs in terms of the SNR required to achieve SER = \( 10^{-5} \). MSDSD, using the efficient decoder described in Section III-C.2, is compared with DFDD and CDD (\( N = 2 \)), respectively. Numerical results (see Section IV-A, lines) and simulation results (markers) are plotted. Since for DSTM with ODs the \( N_T \) antennas transmit constantly, the QSFc model is only an approximation. In order to illustrate the degradation due to the deviation of the QSFc model from the underlying channel (2), numerical and simulation results assuming the QSFc model are also included in Fig. 5 (dashed lines).

First, we observe that SERs from the approximation in Section IV-A and simulated SERs closely match also for this non–group constellation. Furthermore, it can be seen that the performance is limited by channel variations during the transmission of one ST symbol, which are not accounted for in the QSFc model and thus cause an error floor regardless of the detection algorithm. MSDSD and DFDD can cope with much faster fading than CDD. The performance gains due to MSDSD over DFDD are between about 1 dB and 4 dB depending on \( B_0T \).

### B. Computational Complexity

We now compare the computational complexity of the proposed algorithms. In particular, we (a) consider the effects of a finite initial search radius \( \rho_{\text{init}} \) (see Section III-B.3) and the use of sphere and stack decoding algorithms for inner LD (see Section III-C) and (b) compare the complexity of the proposed MSDSD implementations with those of CDD, DFDD, and MSDD–BID. As measure of complexity, we choose the average number of real–valued flops per decoded symbol. Exemplarily, we present results for diagonal DSTM with \( R = 2 \), \( N_T = 3 \), \( N_R = 1 \), \( B_0T = 0.03 \), and MSDSD with \( N = 6 \) (cf. Fig. 4 for performance results).

1) Initial Radius: To study the effect of the initial radius we consider MSDSD–FS (the results are very similar for other MSDSD algorithms). Fig. 6 depicts the complexity for different SNRs as a function of the constant \( c \) introduced in (13). While complexity increases if the initial radius is chosen too small, substantial complexity savings of up to 60 % compared to \( \rho_{\text{init}} \rightarrow \infty \) can be obtained by choosing an optimal finite start radius. It is interesting to note that for the optimal parameter \( c \approx 1.7 \), the transmitted block \( V \) lies inside the sphere with radius \( \rho_{\text{init}} \) with a probability of about 95 % independent of the SNR. It can also be observed that, for this relatively fast fading example, the complexity of MSDSD with infinite initial radius may increase for larger SNR. This is due to the fact that for high SNR the radius after the first tentative decision \( V \) may be much larger than the metric for the ML decision. A finite start radius, on the other hand, leads to the
more intuitive result that the decoding complexity decreases monotonically with SNR. All of the following results assume the optimal finite start radius with \( c = 1.7 \).

2) **MSDD vs. MSDSD–FM and Inner Decoding with FS vs. LD:** Fig. 7 compares the different possibilities to reduce the complexity of MSDSD, i.e. MSDSD vs. MSDSD–FM and MSDSD–FM with inner FS vs. inner LD are considered. LD is implemented using sphere and stack decoder, respectively. It can be seen that the complexity of MSDSD decreases rapidly with increasing SNR, since the search quickly terminates for small enough noise. Inner LD leads to a significant complexity reduction, and the stack decoder is clearly advantageous over the sphere decoder. Furthermore, applying the Fano–type metric speeds up the search especially in the lower SNR range,

![Graph showing complexity vs. SNR for different decoding methods](image)

as was anticipated in Section III-B.2. Combining Fano–type metric with inner LD using the stack algorithm, i.e. MSDSD–FM–LD, yields a tremendous complexity reduction by a factor of about 10 to 50 compared to MSDSD–FS in the range of relevant SNR. This gain comes at the expense of a less than 1 dB loss in power efficiency (cf. Fig. 4).

3) **Comparison with Benchmark Decoders:** Fig. 8 compares the average complexity of MSDSD–FM–LD considered above with those of MSDSD–Fano–LD and the benchmark detectors CDD–LD, DFDD–LD, and MSDSD–BID, respectively. While achieving a similar power efficiency (cf. Fig. 4), MSDSD–FM–LD requires a lower complexity than MSDSD–Fano–LD. The complexity of MSDSD–FM–LD is also well in the range of those of DFDD–LD and CDD–LD, which are less power efficient and suffer from a high error floor, respectively. MSDSD–BID, which also shows a high error floor for this transmission scenario, is considerably more complex than MSDSD–FM–LD.

We thus conclude that MSDSD–FM–LD and its subset MSDD version with the stack algorithm for inner LD offer the overall best performance–complexity tradeoff.

VI. CONCLUSIONS

In this paper, we have investigated tree–search MSDD. We have proposed a nested decoder structure consisting of an outer and \( N - 1 \) inner tree–search decoders, where the inner decoders efficiently search the signal space of the potentially large DSTM constellation. Decoder designs tailored for diagonal and orthogonal DSTM codes have been given. Considering the underlying continuous MIMO fading channel, a tight SER approximation for MSDD has been obtained, which can be evaluated efficiently. The novel subset MSDD scheme further improves power efficiency of MSDS by discarding less–reliable decisions at the edges of the observation window. Numerical and simulation results presented for different

![Graph showing complexity vs. SNR for benchmark decoders](image)
DSTM constellations and fading channel scenarios have shown (a) to what extent MSDD outperforms benchmark decoders increasing the range of fading bandwidths for which DSTM is applicable, (b) that sphere decoding with a Fano–type metric (MSDSD–FM) achieves the best performance–complexity tradeoff with a complexity comparable to that of DFDD in most cases, (c) that inner stack decoding is favorably applied to provide an ordered sequence of candidate signal points for outer tree–search decoding, and (d) that subset MSDD yields impressive additional performance gains at a very moderate increase in complexity.

REFERENCES


