“Turbo DPSK” Using Soft Multiple-Symbol Differential Sphere Decoding

Volker Pauli, Student Member, IEEE, Lutz Lampe, Member, IEEE, and Robert Schober, Member, IEEE

Abstract—Coded interleaved differential $M$-ary phase-shift keying (M/DPSK) with iterative decoding, so-called “Turbo DPSK”, is known as power-efficient transmission format. Due to the rotational invariance of DPSK, it particularly enables detection without channel state information (CSI). However, the soft-input soft-output (SISO) component decoder for DPSK is the computational bottleneck if performance close to the ideal case of perfect CSI is desired. In this paper, we take a fresh look at SISO decoding without CSI and apply sphere decoding (SD) to reduce complexity. In particular, we devise a maximum a posteriori (MAP) multiple-symbol differential sphere decoder (MSDSD) which efficiently solves the high-dimensional search problem inherent to detection without CSI. Together with a soft-output generation device the MAP MSDSD algorithm forms a new SISO-MSDSD module for iterative decoding. We analyze the extrinsic information transfer (EXIT) characteristic of the novel module, by means of which we are able to design powerful encoder and decoder structures. For, respectively, the additive white Gaussian noise (AWGN) and the continuously time-varying Rayleigh fading channel without CSI these designs operate within 1.7-1.9 dB and 2.3-2.5 dB of channel capacity assuming perfect CSI. These figures compare favorably with results available in the literature, especially for reasonably high data rates of 1-2 bit/(channel use). Simulation studies of the average and the maximum complexity required by SISO MSDSD demonstrate the advantageous performance versus complexity trade-off of our approach.

Index Terms—Differential phase-shift keying (DPSK), iterative decoding, sphere decoding, maximum a posteriori decoding, EXIT charts, Rayleigh fading channels.

I. INTRODUCTION

CODING phase-shift keying (PSK) is a widely used transmission format for power-efficient digital communication. Since information is carried by the signal phase, the phase shift introduced by the transmission channel has to be estimated for decoding at the receiver side. One approach to accomplish channel estimation is to deliberately introduce phase shifts of several respects. First, the overall code becomes invariant to receiver parameters only. Alternatively, one can use the code properties to implicitly estimate the channel phase without the need for pilot symbols, cf. e.g. [3]. This approach is usually referred to as detection without channel state information (CSI). Detection without CSI is regarded as more natural as it avoids to explicitly dedicate energy to channel estimation instead of data detection, and it also appears to be more robust for transmission over time-varying fading channels. The frequency of pilot symbols necessary for adequate estimation quality crucially depends on the fading rate, which ideally would require the transmitter and the receiver to adapt to the current channel conditions. For the alternative approach of implicit channel estimation, different channel conditions can be accounted for by adaptation of receiver parameters only.

A pragmatic scheme for coded $M$-ary PSK transmission is shown in the top part of Figure 1, cf. also e.g. [4], [5], [6]. An outer channel encoder is via interleaving and mapping serially concatenated with an inner rate-1 differential encoder, i.e., $M$-ary differential PSK (MDPSK) results. The use of differential encoding as inner encoding is advantageous in several respects. First, the overall code becomes invariant to phase shifts of $2\pi/M$ and integer multiples thereof, which alleviates the channel estimation problem and, in particular, enables detection without CSI, i.e., without transmission of pilot symbols. Second, it is a recursive encoder, so that the overall serially concatenated coding scheme benefits from an interleaver gain [7]. Since it is a rate-1 encoder, reasonably “high” overall code rates are possible. Finally, the fact that it is a convolutional and not a block encoder facilitates receiver adaptation if explicit CSI is not available. The last point distinguishes DPSK from other block-based coding approaches for detection without CSI, cf. e.g. [8], [9], [10], [11], whose receiver processing window size $N$ (see below) is tied to the block length.

Since in the absence of CSI, the transmission channel exhibits memory, multiple-symbol processing of a preferably large number $N$ of received samples is advantageous [12]. On the other hand, $N$ cannot grow without bound as, in general, receiver complexity is proportional to $M^N$, i.e., exponential in this so-called observation window size $N$ [13]. For this reason, the efforts in designing decoders for serially concatenated coded transmission without CSI have concentrated on devising soft-input soft-output (SISO) inner decoding modules with feasible complexity. The literature in this area can be roughly divided into trellis-based and block-based approaches. The trellis-based designs basically consist of a Bahl-Cocke-Jelinek-Raviv (BCJR)-type algorithm [14] to decode DPSK, where the number of states is increased compared to the case with CSI. For example, BCJR-type algorithms with $M^{N-1}$ states were presented by Hoehler and Lodge [4] and by Marsland and Mathiopoulou [15]. Anastasopoulos and Chugg [16] and
Colavolpe et al. [17] devised reduced-state versions with only $M^{N-K}$ states, $1 \leq K \leq N - 1$. Peleg et al. [18] and Chen et al. [11] proposed to discretize the phase of the desired signal into $L$ equi-spaced values, so that the BCJR algorithm operates on an $L$ state trellis. $L$ usually depends on $M$, but is independent of $N$. Block-based SISO modules were presented by Peleg and Shamai [9] and by Su and Geraniotis [19]. Whereas the complexity of the module in [9] is exponential in $N$, a suboptimum and less complex solution is given in [19]. Recently, Motedayen-Aval and Anastasopoulos [20] showed that multiple-symbol detection over fading channels is possible with a complexity proportional to $N^{2q}$, where $q$ is the rank of the $N \times N$ channel correlation matrix and they devised a block-based SISO module with complexity proportional to $N^{2q}$ per soft output. This is an interesting solution for rank-deficient (i.e., slow fading) channels with $q \ll N$. Finally, we note that Mackenthun’s low-complexity block-based algorithm [21] cannot be applied for SISO decoding with a priori information.

In this paper, we consider concatenated coded $M$-ary PSK transmission with iterative decoding as shown in Figure 1. Due to the presence of differential encoder and iterative decoder, this structure was quite concisely referred to as “Turbo DPSK” by Hoehler and Lodge [4]. As outer encoders we use standard convolutional codes of various memory orders and rates. We study both the noncoherent AWGN and the frequency non-selective Rayleigh fading channel, where in case of Rayleigh fading the channel statistics are assumed to be known at the receiver. We emphasize that no pilot symbols are inserted to support detection (cf. also [22] for a very efficient approach with periodically inserted training symbols).

We focus on the development of an inner SISO module, which favorably trades reliability of the soft outputs and required computational complexity, both of which grow monotonically with $N$. Thereby, we are inspired by the block-based approach by Peleg and Shamai [9]. To overcome the prohibitive complexity of their method for large $N$, we suggest the application of sphere decoding (SD) (cf. e.g., [23], [24], [25]) to SISO decoding without CSI. In a related previous work [26], we have shown that hard-output maximum-likelihood (ML) multiple-symbol detection for uncoded transmission can be formulated as a shortest-vector problem, which is amenable to SD. The resulting algorithm was named multiple-symbol differential sphere decoding (MSDSD) [26]. In this paper, we derive a maximum a posteriori (MAP) MSDSD algorithm, which together with a reliability generation unit forms a novel block-based SISO-MSDSD module. Since this SISO module enjoys the low average complexity inherent to SD, cf. e.g., [23], [24], [25], the window size $N$ can assume desirably large values.

The proper choices of the parameter $N$ as well as of the outer code and the $M$PSK labeling are guided by an analysis of the iterative decoding process using ten Brink’s extrinsic information transfer (EXIT) chart technique [27]. In particular, we devise an adaptive adjustment of $N$ depending on available a priori information. From the EXIT chart analysis we obtain several encoder and decoder designs for overall data rates between 1 and 2 bit/(channel use), which perform within 1.7–1.9 dB (AWGN) and 2.3–2.5 dB (Rayleigh fading with normalized fading bandwidth $B_b T = 0.01$) of channel capacity assuming perfect CSI. These remarkable results manifest that the introduction of SD to iterative decoding without CSI (which is already interesting in its own right) enables highly power-efficient transmission for fairly bandwidth-efficient modulation. Especially for the Rayleigh fading channel these figures compare favorably with other schemes available in the literature, cf. e.g., [4], [19], [11], which are either less power-efficient or restricted to lower data rates and/or the block fading channel.

It is worth mentioning that variations of SD for SISO decoding were recently suggested by Vikalo and Hassibi [28], Hochwald and ten Brink [29], Bäro et al. [30], and Boutros et al. [31] in the context of multiple-antenna transmission with CSI. Especially the MAP-SD algorithm presented in [28] is structurally similar to the MAP-MSDSD algorithm derived in this paper.

This paper is organized as follows. Section II details the system model shown in Figure 1. The SISO-MSDSD module is described in Section III. By relating iterative SISO decoding
to hard-decision feedback decoding studied in [32], we also derive a tight approximation for the achievable bit-error rate. Section IV is concerned with the complexity of SISO MSDSD and its proper incorporation into an iterative decoder by means of EXIT chart analysis. Several designs for power-efficient transmission without CSI are given. Performance results from simulations are presented and compared with the ultimate capacity limits in Section V. It is also shown that imposing reasonable limitations on the maximum complexity of SISO MSDSD does practically not affect performance. Conclusions are drawn in Section VI.

Throughout the paper we use the following notation. Bold lower case \( x \) and upper case \( X \) indicate column vectors and matrices, respectively. The \( i \)th element of vector \( x \) is addressed by \( x_i \), and \( x_{ij} \) denotes the entry of matrix \( X \) in row \( i \) and column \( j \). \( I_N \) and \( 1_N \) represent the \( N \times N \) identity matrix and all-ones matrix, respectively. \( \text{diag} \{ x \} \) is a diagonal matrix with the components of \( x \) on its main diagonal. \( \det \), \( \mathcal{F} \), \( \mathcal{G} \), and \( \mathcal{H} \) refer to the determinant of a matrix, transposition, (componentwise) complex conjugation, and Hermitian transposition, respectively. \( \text{Pr} \{ \cdot \} \), \( \text{E} \{ \cdot \} \), \( \text{Re} \{ \cdot \} \), \( \lfloor \cdot \rfloor \), and \( \otimes \) denote the probability of the event in brackets, expectation, the real part of a complex number, the Frobenius norm, and the Kronecker product, respectively. Further, in order to keep the notation as simple as possible, we do not distinguish between random variables and particular realizations.

II. System Model

The block diagram of the transmission system in the equivalent complex baseband is depicted in Figure 1. At the transmitter side, we employ an outer binary encoder which delivers a sequence of binary coded symbols \( c^j[n] \). The interleaved sequence \( c^j[n] \) is mapped to \( M \)-ary PSK symbols \( v[k] \in \{ e^{j2\pi m/M} | m = 0, \ldots, M - 1 \} \), which are differentially encoded into transmit symbols

\[
s[k] = v[k]s[k-1].
\]

The cascade of outer encoder, interleaver, mapper, and differential encoder constitutes a serially concatenated coding scheme. The mapper and the differential encoder form the inner encoder. For outer encoding any kind of error correction code could be used, e.g. convolutional code, repeat-accumulate code or turbo code. However, we found that widely used standard convolutional codes allow for performance close to the capacity limit with reasonably low decoding complexity (see Sections IV and V). This observation is in accordance with results reported in the literature, e.g., [5], [6], [11]. Thus, when presenting design and performance results we concentrate on this class of codes.

We consider transmission over frequency non-selective fading channels with modulation interval \( T \). The normalized bandwidth \( B_T \) of the fading process is assumed sufficiently small, so that the equivalent discrete-time channel model

\[
r[k] = h[k]s[k] + n[k]
\]

can safely be used. \( r[k] \), \( h[k] \), and \( n[k] \) are, respectively, the received sample, the fading coefficient, and the noise sample. As usual, the noise process is assumed to be white with circularly symmetric Gaussian distributed \( n[k] \) (AWGN) with variance \( \sigma_n^2 \). As for the fading coefficient, we concentrate on the two important special cases of a Ricean fading model, which are most often considered in the literature: (i) the noncoherent AWGN channel\(^1\) with a constant phase term \( h[k] = e^{j\theta} \), where \( \theta \) is uniformly distributed in \([0, 2\pi]\), and (ii) the Rayleigh fading channel with zero-mean circularly symmetric Gaussian distributed \( h[k] \). For the latter case, we assume the autocorrelation function

\[
\varphi[k] = \text{E} \{ h[k + \kappa]h^*[k] \} = J_0(2\pi B_h T \kappa)
\]

according to the Clarke fading model [33] (\( J_0(\cdot) \): zeroth order Bessel function of the first kind). Throughout this paper we assume that the channel coefficients \( h[k] \) are unknown to the receiver side and that the transmitted sequence does not include pilot symbols for channel estimation.

The received samples \( r[k] \) enter an iterative decoder which, corresponding to the serially concatenated encoder, consists of two soft-input soft-output (SISO) modules. These inner and outer decoders produce extrinsic information about the binary coded symbols which is used as a priori information by the respective other decoder. The exchange of extrinsic information is repeated in a certain number of iterations, which is predetermined in the decoder design (see Section IV). The outer decoder is the optimum maximum a posteriori symbol-by-symbol (MAPSSE) decoder using the Bahl-Cocke-Jelinek-Raviv (BCJR) algorithm [14]. Alternatively, and without significantly compromising performance, suboptimum MAPSSE decoder versions could be employed, whose complexity is well in the order of the standard Viterbi decoder [34]. If perfect CSI of \( h[k] \) was available, the optimum inner decoder would also consist of a BCJR decoder operating on the \( M \)-state trellis of the differential encoder and a demapper converting \( M \)-ary symbol information into bit information. However, in the absence of CSI, we have to resort to suboptimum solutions to avoid that the decoding complexity grows exponentially with the length of the transmitted sequence. For this reason, the soft-input soft-output multiple-symbol differential sphere decoding (SISO-MSDSD) module, which is presented in detail in the following section, is proposed as inner decoder.

III. Soft-Input Soft-Output MSDSD

In this section, we introduce the SISO-MSDSD module, which operates on an \( N \)-symbol observation interval to account for the memory of the fading process and the differential encoder. For this purpose, the conditional log-likelihood required for ML and MAP \( N \)-symbol decisions without CSI is appropriately formulated in Section III-A. Based on this expression, we propose a MAP-MSDSD algorithm which accepts a priori information about differential symbols, i.e., soft input, in Section III-B. The generation of extrinsic information, i.e., soft output, from the estimates found with the MAP-MSDSD algorithm is described in Section III-C. The MAP-MSDSD algorithm of Section III-B together with the soft

\(^1\)For the sake of a unified framework, we adopt the point of view that the AWGN channel is a degenerated fading channel.
output generation of Section III-C form the SISO-MSDSD module. Finally, we present a tight approximation for the best-case bit-error rate achievable with the SISO-MSDSD module in Section III-D.

At this point, it is convenient to introduce the vectors
\[ r \triangleq [r[k - (N - 1)] \ldots r[k]]^T \]
\[ s \triangleq [s[k - (N - 1)] \ldots s[k]]^T \]
\[ v \triangleq [v[k - (N - 2)] \ldots v[k]]^T \]
\[ h \triangleq [h[k - (N - 1)] \ldots h[k]]^T \]
of, respectively, \( N \) received vector, \( N \) transmitted symbols, and \( N \) fading coefficients. For the sake of clarity, we also drop the time index \( k \) in this section and address the \( i \)th element of \( x \) by \( x_i \).

A. Log-Likelihood Expressions

The ML and the MAP multiple-symbol decisions rely on the evaluation of the conditional probability density function (pdf) \( p(r|s) \) of \( r \) given \( s \). For the Rayleigh fading channel we find the relation [13]
\[ -\log(p_{\text{awgn}}(r|s)) \propto r^H R^{-1}_{rr} r, \quad (4) \]
where “\( \propto \)” indicates that additive terms independent of \( s \) are neglected, and \( R_{rr} \) is the conditional correlation matrix of the received vector
\[ R_{rr} \triangleq \mathbb{E}\{rr^H|s\}. \]
Defining the correlation matrix
\[ C_{\text{awgn}} \triangleq \mathbb{E}\{hh^H\} + \sigma_n^2 I_N \]
and following the derivations in [26], we can write
\[ -\log(p_{\text{awgn}}(r|s)) \propto (\text{diag}\{r\} s^*)^H C^{-1}_{\text{awgn}} \text{diag}\{r\} s^*. \]

Since \( C^{-1}_{\text{awgn}} \) is positive definite, we apply the Cholesky factorization
\[ C^{-1}_{\text{awgn}} = U^H L L^H \quad (8) \]
and define
\[ U \triangleq (L^H \text{diag}\{r\})^*, \quad (9) \]
where \( L \) and \( U \) are lower and upper triangular \( N \times N \) matrices, respectively. Finally, using (8) and (9) in (7) gives
\[ -\log(p_{\text{awgn}}(r|s)) \propto \|Us\|^2. \quad (10) \]

It is interesting to note that for very slow fading, i.e., \( B_k T = 0 \), the inverse correlation matrix \( C^{-1}_{\text{awgn}} \) reads
\[ C^{-1}_{\text{awgn}} \big|_{B_k T = 0} = -\frac{1}{N + \sigma_n^2/\sigma_0^2} I_N + \frac{1}{\sigma_0^2} I_N. \quad (11) \]

In the case of the AWGN channel, we find \( (t_0(\cdot) \) zeroth order modified Bessel function of the first kind) [35]
\[ -\log(p_{\text{awgn}}(r|s)) \propto \frac{1}{\sigma_n^2} r^H r - \log(t_0(2r^H s/\sigma_0^2)) \quad (12) \]
To arrive at an expression similar to (10), it is convenient to use the small argument approximation for \( \log(t_0(x)) \) [35, i.e., \( \log(t_0(x)) \approx x^2/4 \), which is tight for \( \sigma_0^2/N \gg 1 \) (low signal-to-noise ratio (SNR)). With this approximation and the definition \( C^{-1}_{\text{awgn}} \)
\[ -\log(p_{\text{awgn}}(r|s)) \propto (\text{diag}\{r\} s^*)^H C^{-1}_{\text{awgn}} \text{diag}\{r\} s^*. \]
the approximate relation
\[ -\log(p_{\text{awgn}}(r|s)) \propto (\text{diag}\{r\} s^*)^H C^{-1}_{\text{awgn}} \text{diag}\{r\} s^*. \quad (14) \]
is obtained. We note (i) that for \( \sigma_0^2/N \gg 1 \), \( C^{-1}_{\text{awgn}} \) is asymptotically equal to the inverse correlation matrix \( C^{-1}_{\text{awgn}} \big|_{B_k T = 0} \) given in (11), and (ii) that \( C^{-1}_{\text{awgn}} \) is only positive definite for \( \sigma_0^2/N > 1 \). These observations motivate us to replace \( C^{-1}_{\text{awgn}} \) in (14) with \( C^{-1}_{\text{awgn}} \big|_{B_k T = 0} \), i.e., we use
\[ -\log(p_{\text{awgn}}(r|s)) \propto (\text{diag}\{r\} s^*)^H C^{-1}_{\text{awgn}} \big|_{B_k T = 0} \text{diag}\{r\} s^*. \quad (15) \]
In this way, since \( C^{-1}_{\text{awgn}} \big|_{B_k T = 0} \) is always positive definite, the Cholesky factorization (8) can be applied and an expression (10) is found for the AWGN channel with arbitrary ratio \( \sigma_0^2/N \). Of course, this extension of the low SNR case to general SNRs is not optimum. However, the numerical and simulation results presented in Sections IV and V nicely confirm the effectiveness of our approach. Since we use expression (10) for both Rayleigh and AWGN channel, it is convenient to drop the subscripts “awgn” and “awgn” in the following.

Apparently, the ML metric (10) is invariant to a phase shift common to all components of \( s \) such that the same differential vector \( \alpha \) results. Therefore, we have
\[ -\log\{p(r|s)\} = -\log\{p(r|s^*)\} \propto \|Us\|^2 \quad (16) \]
where we introduced the vector \( \alpha \triangleq [a_1 \ldots a_N]^T \) of accumulated differential symbols
\[ a_i = \left\{ \begin{aligned} N^{-1} \prod_{i = 1}^{N} v_i^*, & \quad 1 \leq i \leq N - 1 \\ 1, \quad i = N \end{aligned} \right. \quad (17) \]
Because of the unique relation between \( v \) and \( \alpha \), they are treated interchangeably in the following.

B. Maximum A Posteriori MSDSD

Finding the ML estimate \( v_{\text{ml}} \) which minimizes (16) with respect to \( v \) requires a search in an \( (N-1) \)-dimensional space, i.e., if a brute-force search is applied, complexity increases exponentially with \( N \). In [26], we identified the ML estimate as solution of a shortest vector problem, cf. e.g. [24], and proposed MSDSD to overcome the exponential complexity growth. The ML-MSDSD algorithm examines all candidate vectors \( v \) that satisfy
\[ \|Us\|^2 \leq R_s^2 \quad (18) \]
1Note that \( C^{-1}_{\text{awgn}} \) is not an inverse correlation matrix.
for some radius $R_3$. Thereby, the MSDSD algorithm exploits the upper triangular structure of $U$ and works successively through the components of $v$.

However, for SISO decoding we are interested in the MAP estimate

$$\hat{v}_{\text{map}} = \arg\min_v \{ -\log(Pr\{v|r\}) \}$$

(19)

and hence, the appropriate candidate vectors $v$ satisfy

$$-\log(Pr\{v|r\}) \propto -\log(p(r|v)) - \log(Pr\{v\})$$

$$\propto \|Ua\|^2 - \log(Pr\{v\}) \leq R^2$$

(20)

where the available a priori information $\log(Pr\{v\})$ is taken into account.

To formulate the MAP-MSDSD algorithm, we assume that the symbols $v_i$ are independent, i.e., that

$$\log(Pr\{v\}) = \sum_{i=1}^{N-1} \log(Pr\{v_i\})$$

(21)

holds. Then, we can rewrite (20) as

$$\sum_{i=1}^{N-1} \left( \sum_{i=1}^{N} u_{i;i}a_{i}^2 - \log(Pr\{v_i\}) \right) \leq R^2 - |u_{NN}|^2 \leq R^2$$

(22)

where $u_{i;i}$ denotes the entry of $U$ in row $i$ and column $i$. Expression (22) reveals that a valid signal point $v_i$ for component $i$, $1 \leq i \leq N-1$, must satisfy

$$d_i^2 \triangleq u_{i;i}a_{i}^2 + \sum_{i=1}^{N} u_{i;i}a_{i}^2 - \log(Pr\{v_i\}) +$$

$$+ \sum_{i=1}^{N-1} \left( \sum_{i=1}^{N} u_{i;i}a_{i}^2 - \log(Pr\{v_i\}) \right) \leq R^2$$

(23)

Hence, the MAP-MSDSD algorithm starts with component $i = N-1$, selects a signal point $v_{N-1}$ with $\Delta_{N-1}^2 = d_{N-1}^2 \leq R^2$, proceeds with component $i = N-2$ and checks $d_{N-2}^2 = \Delta_{N-2}^2 + d_{N-1}^2 \leq R^2$, and so forth. If the component $i = 1$ is reached, i.e., a vector $v = [v_1 \ldots v_{N-1}]^T$ is found, the search radius is updated to

$$R^2 := d_1^2 = \|Ua\|^2 - \log(Pr\{v\})$$

The search is repeated with this refined radius and starting with component $i = 2$. The SD algorithm tests only other hypothetical signal points $v_i$ and works, respectively, forwards (decreasing $i$) as long as and backwards (increasing $i$) until condition (23) is met. If the new search does not yield a better estimate, the previously found vector $v$ is the MAP estimate $\hat{v}_{\text{map}}$ from (19).

The proposed MAP-MSDSD algorithm is summarized as pseudocode in Appendix I. Its structure is similar to the SD algorithm proposed in [24, Section III.B] for the closest point search in lattices. For a quick termination of the search, we apply the Schnorr-Euchner strategy [36], which orders hypothetical symbols $v_i$ examined for the $i$th component according to monotonically increasing values $\Delta_i^2$ (see (23)). This ordering is accomplished in the function “sortDelta()” in Appendix I. In contrast to the ML search ([24], [25], [26]), it is not possible to select only the best candidate point $v_i$, which minimizes $\Delta_i^2$, and then to zigzag through the remaining $M - 1$ signal points. The incorporation of a priori information distorts the regular geometry of the signal constellation. The counter $n_i$ is introduced (line F-5) and incremented (lines 20, 28) to indicate the hypothetical symbol $v_i$ that is to be examined for component $i$. Due to the finite constellation size, the search is terminated if $n_i = M$ (line 27). Since the Schnorr-Euchner strategy abandons the need for an initial search radius, i.e., $R \to \infty$ can be assumed, we chose to initialize our MSDSD algorithm with a reasonably high value of $R$ for the results presented in Sections IV and V.

C. Soft-Output Generation

The MAP-MSDSD algorithm of the previous section is the key element to provide extrinsic information as input to the outer decoder. To generate this soft output we compare the MAP-MSDSD estimate $\hat{v}_{\text{map}}$ with appropriate alternate vectors $v$. This approach is in the spirit of [37] and similar to those advocated in [28], [29]. In contrast to [28], [29] we do not apply a “list” version of the sphere decoder, but constrain ourselves to only one alternative hypothesis for each coded symbol $c[k]$, i.e., consider only nearest neighbors [37].

Let us assume that a priori information in terms of log-likelihood ratios (LLRs)

$$e_c[k] = \log \left( \frac{Pr\{c[k] = b\}}{Pr\{c[k] = \bar{b}\}} \right), \quad b \in \{0, 1\},$$

(24)

is available from the outer decoder for all binary coded symbols (see Figure 1, $\bar{b}$ is the complement of $b$). Assuming that due to powerful interleaving the coded symbols can be regarded as independent, the a priori information $\log(Pr\{v\})$ for symbol vectors is easily obtained from $e_c[k]$ taking into account the particular mapping of binary to $M$-ary symbols. Now, suppose that we are interested in one particular symbol $c[k]$ and that the MAP-MSDSD estimate $\hat{v}_{\text{map}}$ corresponds to $c[k] = b$. Then, for SISO MSDSD the a posteriori log-likelihood ratio (LLR) of $c[k]$ is approximated by

$$\log \left( \frac{Pr\{c[k] = b\}}{Pr\{c[k] = \bar{b}\}} \right) \approx \log \left( \frac{\max_{c[k] = b} \{ \exp(-\|Ua\|^2 + \log(Pr\{v\})) \} }{\max_{c[k] = \bar{b}} \{ \exp(-\|Ua\|^2 + \log(Pr\{v\})) \} } \right)$$

$$= -\|Ua_b\|^2 + \log(Pr\{v_{b,\text{map}}\}) +$$

$$+ \|Ua_\bar{b}\|^2 - \log(Pr\{v_{\bar{b},\text{map}}\}) \triangleq l_c[k],$$

(25)

where $v_{b,\text{map}}$ and $v_{\bar{b},\text{map}}$ is the MAP estimate when constraining the MSDSD search to the $M^{-1/2}$ vectors $v$ with address bit $c[k] = b$. Finding this alternative vector is

\[4\] It should be noted that a list construction outside the iterative decoder as advocated in [29] is not applicable to SISO MSDSD due to the adaptive adjustment of the parameter $N$ (see Section IV-B).
easily accomplished by slightly modifying the MAP-MSDSD algorithm, such that vectors corresponding to \( c[\mu] = b \) are excluded from the search space. We note that expression (25) can be regarded as “max-sum” approximation of the correct LLR, e.g. [37]. Finally, the extrinsic information for \( c[\mu] \) is obtained by excluding the corresponding a priori information, 

\[
e_c[\mu] = l_c[\mu] - e_c[\mu], \tag{26}
\]

and is passed to the outer decoder (see Figure 1).

In general, \((N-1) \log_2(M)\) LLRs \(e_c[\mu]\) could be generated from the observation of \( r \). However, we observed that the reliability of the soft output considerably degrades for coded symbols \( c[\mu]\) corresponding to differential symbols located at the edges of the observation interval. Therefore, we chose to only calculate \( e_c[\mu]\) for the \( \log_2(M) \) corresponding to the centered symbol \( v_{N/2} \) and then to move forward the observation window by one symbol. This means, we apply maximally overlapping observation windows, which is different from other block detection approaches, e.g. [9]. Of course, the observation size \( N \) has to be restricted to even values.

In summary, our SISO-MSDSD module consists of the MAP-MSDSD algorithm together with the extrinsic-information generation (25), (26). Due to the use of maximally overlapping observation windows, the MAP-MSDSD algorithm is executed on average \((\log_2(M) + 1)/\log_2(M)\) times per soft output.

### D. Tight Performance Approximation

In this section, we derive a performance approximation for iterative decoding without CSI using the previously described SISO-MSDSD module. In general, we could apply upper bounds for (ensembles of) concatenated codes presented in the literature for the cases of perfect and imperfect CSI, cf. e.g. [7], [6], [38]. However, we adopt the point of view that the scheme in Figure 1 can be regarded as conventional bit-interleaved coded modulation (BICM) [39] with decoder input \( e_c[\mu] \). Furthermore, we borrow from the methodology used for hard decision-feedback iterative decoding in [32], [40], and consider SISO MSDSD under the assumption of perfect a priori information, i.e., \( \Pr {c[\mu] = b} = 1 \), where \( b \) is the transmitted binary symbol. Applying this assumption, we subsequently derive the Laplace transform \( \Phi_{e,[\mu]}(s) \) of the pdf of \( e_c[\mu] \) in (26). From \( \Phi_{e,[\mu]}(s) \) and [39, Eqs. (26), (48)], we immediately obtain tight upper bounds for the achievable bit-error rate (BER), i.e., tight approximations for the true BER with realistic SISO MSDSD.

To find an expression for \( \Phi_{e,[\mu]}(s) \), let us assume that \( c[\mu] = b \) corresponding to \( v_{N/2} = v_b \) has been transmitted, and that the alternative symbol with address bit \( c[\mu] = \bar{b} \) is denoted by \( v_{\bar{b}} \). Furthermore, let \( \alpha = [\alpha_1, \ldots, \alpha_N]^T \) denote the accumulated vector corresponding to the correct (transmitted) data sequence. Then, from (26) and (25) we directly obtain

\[
e_c[\mu] = \sum_{\eta=1}^{N/2} 2\text{Re} \left\{ \left( \sum_{i=\eta-N/2+1}^{N} \alpha_i^* u_{\eta i} \right) \left( \sum_{\eta=1}^{N/2} \alpha_i^* u_{\eta i} \right)^* \right\} \cdot (v_{\eta}^* v_d - 1). \tag{27}
\]

Using the relation between \( U \) and \( L \) in (9) and defining \( g_i \triangleq h_i + n_i s_i^* \), the identity

\[
a_i u_{\eta i} = s_N^* g_i^* \ell_{i\eta} \tag{28}
\]

follows, where \( \ell_{i\eta} \) is the element of \( L \) in row \( i \) and column \( \eta \). Introducing (28) in (27) yields

\[
e_{c}[\mu] = \sum_{\eta=1}^{N/2} 2\text{Re} \left\{ \left( \sum_{i=\eta-N/2+1}^{N} g_i^* g_i^* \ell_{i\eta} \right) \left( \sum_{\eta=1}^{N/2} g_i^* \ell_{i\eta} \right)^* \right\} (v_{\eta}^* v_d - 1). \tag{29}
\]

Arranging the terms in (29) appropriately in a vector \( x = [x_1 \ldots x_N]^T \) with elements

\[
x_{2\eta-1} \triangleq \sum_{i=\eta-N/2+1}^{N} g_i^* \ell_{i\eta}, \quad x_{2\eta} \triangleq \sum_{\eta=1}^{N/2} g_i^* \ell_{i\eta}, \quad 1 \leq \eta \leq N/2, \tag{30}
\]

and a matrix

\[
F \triangleq I_{N/2} \otimes \left[ \begin{array}{cc} 0 & v_{1} v_d^* - 1 \\ v_{1} v_d^* - 1 & 0 \end{array} \right], \tag{31}
\]

respectively, the soft output \( e_c[\mu] \) can compactly be written as

\[
e_c[\mu] = x^H F x. \tag{32}
\]

Since the elements of \( x \) are jointly circular-symmetric Gaussian distributed random variables with mean vector \( \mathbf{\mu} \) and correlation matrix \( \Psi_{xx} \), the Laplace transform \( \Phi_{e,[\mu]}(s) \) is found as [41, Appendix B]

\[
\Phi_{e,[\mu]}(s) = \frac{\exp(-s\mathbf{\mu}^H (F^{-1} + s \Psi_{xx})^{-1} \mathbf{\mu})}{\det (I_N + s \Psi_{xx} F)}. \tag{33}
\]

We note that due to the approximation in (15), expression (33) applies to both the AWGN and the Rayleigh fading channel and both can conveniently be treated in a unified framework. For the Rayleigh fading channel, \( \mathbf{\mu} \) is the all-zero vector.

Using (33) in [39, Eq. (49)] and evaluating [39, Eqs. (26), (48)], we obtain an upper bound for the best-case and an approximation for the true performance of iterative decoding. Assuming sufficient convergence of iterative decoding, this approximation is used in Section IV for encoder and decoder design and in Section V for verification of the simulation results.

### IV. Iterative Decoding with SISO MSDSD

Having established the SISO-MSDSD module for iterative decoding without CSI, we now turn to the task of appropriately embedding this module in an iterative decoder. To this end, we first address the issue of computational complexity in Section IV-A, and then present EXIT chart designs in Section IV-B.

For the following considerations and those in Section V we need to specify some system parameters. First of all, we aim at overall data rates of 1 to 2 bit/channel use to accomplish bandwidth-efficient transmission. For modulation and differential encoding we apply, respectively, (i) 4PSK constellations with Gray labeling (GL) and set-partitioning labeling (SPL), and (ii) 8PSK constellations with GL, SPL, mixed labeling
(ML), and semi set-partitioning labeling (SSPL) (see [42], [43] for details of the labelings). As a pragmatic and, in fact, favorable choice for outer encoding, we use rate 1/2 non-recursive convolutional encoders as specified in [44]. If higher code rates are desired, puncturing is applied. The outer decoder is implemented as BCJR decoder. Finally, to limit the number of adjustable parameters, we fix the fading rate of the Rayleigh fading channel to $B_h T = 0.01$.

A. Complexity

The main computational effort of our SISO-MSDSD module is to perform the MAP-MSDSD algorithm\(^5\), whose implementation and thus complexity is similar to that of ML MSDSD. Hence, to make an absolute judgement on complexity, we can consider the results for the ML MSDSD presented in [26]. In fact, in [26] the computational complexity of the ML-MSDSD algorithm is shown to be comparable to that of Mackenthun's algorithm [21] and decision-feedback differential detection (DF-DD) [45], which are well-known low-complexity multiple-symbol detection schemes. Accordingly, we can safely state that the MAP-MSDSD algorithm is a low-complexity solution for SISO multiple-symbol decoding. This claim is also supported by results reported by Vikalo and Hassibi [46], [28], who found the complexity of their MAP-SD algorithm (for multiple-antenna transmission with perfect channel state information (CSI)) to be often cubic in the number of dimensions.

In order to gain further insight into the complexity properties of the MAP-MSDSD algorithm, we consider the number of real-valued multiplicative operations (RMO) required to produce one soft-output value as reasonable complexity measure. According to the pseudocode in Appendix I, \(4(N - i)\) and 6 RMOs are required each time \(k_i\) (line 13) and \(\sigma_i^2\) (line F-3) are calculated, respectively, and the generation of the \(N \times N\) matrix \(U\) (9) involves \(N(N + 1)\) RMOs, if we assume a real-valued autocorrelation function \(\varphi[k]\) (3). Thereby, we note (i) that a lot of RMOs correspond to multiplications with \(M\) PSK symbols (line 13, line F-3), which especially for 4PSK can be efficiently implemented, and (ii) that the computational complexity of the implementation considered in Appendix I can be reduced by trading RMOs against memory requirements.

A lower bound for the complexity of the MAP-MSDSD algorithm is obtained, if we count the number of RMOs involved to find the first estimate \(\theta\) for \(\theta_{\text{MAP}}\) and to check only a single alternative signal point for each component \(e_i\), \(1 \leq i \leq N - 1\). With the above assumptions, this lower bound is quickly found to be \((3\log_2(M) + 2)N^2\) for large \(N\). Hence, in an ideal scenario where the SD terminates, e.g. due to available a priori information (see below), after the first estimate is obtained, complexity increases quadratically (as opposed to exponentially) with \(N\). It is interesting to compare this bound with the complexity of the simple hard-decision feedback differential demodulation (DFDM) proposed in [32], which is a low-complexity method similar to DF-DD. DFDM requires asymptotically \((9/\log_2(M) + 6)/4 \cdot N^2\) RMOs per soft output, which is of the same order as the lower complexity bound for MAP MSDSD.

Of course, the actual complexity will depend on the fading channel dynamics and the SNR (cf. [26] for ML MSDSD) and the available a priori information. Especially the influence of the latter is important, since it changes during the decoding iterations. As meaningful measure of a priori information we consider the average mutual information (AMI) \(I_{a}^{\text{MSDSD}}\) between the a priori input \(e_{\text{a}}[i]\) and the outer encoder output \(e[i]\) (see Figure 1, cf. [47], [27]). To illustrate the dependence on the input AMI, we plot the complexity exponent (cf. e.g. [48], [25])

\[
E_{C}(N) \triangleq \log_{2}(\text{average number of RMOs per output})
\]

obtained by simulation as function of the observation length \(N\) and parametrized with \(I_{a}^{\text{MSDSD}}\) in Figure 2. As representative example, we choose 4PSK with SPL and Rayleigh fading channel. The SNR is adjusted to \(10\log_{10}(E_s/N_0) = 4\) dB such that \(10\log_{10}(E_s/N_0) = 4\) dB is true for a rate 1/2 outer channel code (see design examples in Section IV-B).\(^6\) Thereby, the assumption of Gaussian distributed a priori LLRs is applied (cf. [27]). As reference curves, the lower complexity bound and the complexity of DFDM are also included.

Apparently, the behavior of the complexity exponent strongly depends on the amount of a priori information provided by the outer decoder. For \(I_{a}^{\text{MSDSD}} < 0.8\), \(E_{C}(N)\) tends to increase for \(N \geq 6...10\), so it is advisable to use observation lengths \(N \leq 6\) when operating in this range. On the other hand, for \(I_{a}^{\text{MSDSD}} \geq 0.8\) the required complexity is well in the order of the lower bound and relatively large observation windows \(10 \leq N \leq 20\) are applicable with only

\(^5\)We note that the Cholesky factorization (8) of the channel correlation matrix is performed only once and does practically not contribute to the overall complexity.

\(^6\)\(E_s\) and \(E_o\) denote the average received energy per symbol and information bit, respectively, and \(N_0\) is the two-sided equivalent baseband noise power density.
moderate expected complexity. In fact, with increasing $P_{\text{MSDSD}}$ the complexity of SISO MSDSD becomes very similar to that of DFDM [32]. Practically the same observations as for this example can be made for other constellations and labelings, and for the AWGN channel.

B. Design

For the design of the concatenated coding scheme in Figure 1, we have essentially three degrees of freedom: the choice of the outer encoder, the labeling of the signal constellation, and the size $N$ of the observation window. We note that the labeling changes the properties of the inner encoder, whereas the window size $N$ only affects the decoder.

As appropriate design tool for concatenated codes with iterative decoding we apply ten Brink’s EXIT chart technique, which graphically connects the transfer characteristics (TCs) of the individual SISO decoders [47], [27]. The TC of the SISO-MSDSD module is the output AMI $I_{\text{MSDSD}}^{\text{out}}$ between $e/\mu$ and $c/\mu$ as function of the input AMI $I_{\text{MSDSD}}^{\text{in}}$ between $e/\mu$ and $c/\mu$ (see Figure 1). Correspondingly, the outer BCJR decoder is described by the AMI $I_{\text{BCJR}}^{\text{e}}$ between $e'/\mu$ and $c'/\mu$ as function of the AMI $I_{\text{BCJR}}^{\text{c}}$ between $e'/\mu$ and $c'/\mu$. Identifying $I_{\text{MSDSD}}^{\text{in}} = I_{\text{BCJR}}^{\text{c}}$ and $I_{\text{MSDSD}}^{\text{out}} = I_{\text{BCJR}}^{\text{e}}$, both characteristics can be plotted in one EXIT chart (see [27] for a detailed discussion).

To obtain these TCs with low computational effort and thus, to greatly facilitate the design task, (i) the Monte Carlo technique presented in [49] and (ii) the assumption of Gaussian distributed a priori LLRs [27] are applied. The latter assumption might be violated by the extrinsic output of the SISO-MSDSD module, but avoids the cumbersome recording of histograms. Furthermore, we will show that the predictions from our EXIT chart designs agree very well with results from simulating the entire transmission system.

1) Labeling and Outer Encoder: Figure 3 depicts (i) the TCs of SISO MSDSD for 8PSK with different labelings, observation length $N = 6$, and Rayleigh fading with $10\log_{10}(E_s/N_0) = 10$ dB, and (ii) the TCs of the outer BCJR decoder for convolutional codes of rate $R_c = 1/2$ and $R_c = 2/3$ and memory lengths $\nu = 2$ and $\nu = 6$, respectively.

First, let us focus on the TC of SISO MSDSD. From the figure we can observe that depending on the labeling the slope of the TC significantly varies. GL achieves the highest output AMI $I_{\text{MSDSD}}^{\text{out}}$ for no a priori input, i.e., $I_{\text{MSDSD}}^{\text{in}} = 0$, but the smallest $I_{\text{MSDSD}}^{\text{out}}$ for perfect feedback, i.e., $I_{\text{MSDSD}}^{\text{in}} = 1$. The converse is true for SSPL. We further observed (not shown in the figure) that a change in SNR basically shifts the TCs vertically. Consequently, the adopted signal labeling has a strong impact on the convergence behavior of iterative decoding. For a given outer code, iterative decoding with GL is likely to converge for lower SNRs than with SSPL, since the tunnel between the TCs of MSDSD and the outer decoder opens earlier. On the other hand, if convergence is achieved, the lowest BER is expected for SSPL, since the intersection of the TCs is very close to $I_{\text{BCJR}}^{\text{c}} = 1$ (cf. also [47] for transmission with perfect CSI). ML and SPL offer a compromise between early convergence and low achievable BER.

Taking now the TCs for the convolutional codes into account and, in particular, comparing the slopes of the TCs, we note that GL is well matched to memory $\nu = 6$ codes, whereas SSPL matches quite well with the memory $\nu = 2$ codes. Similar statements hold for 4PSK and the Rayleigh fading channel, where GL is well suited to memory $\nu = 6$ codes and SPL works better with $\nu = 2$ codes. For the AWGN channel we observed that the TCs of SISO MSDSD have generally larger slopes than for the fading channel. Therefore, GL is the favorable choice also for $\nu = 2$ codes. Of course, the window size $N$ also shapes the TCs, which is discussed in the next section.

2) Observation Window Size: For the same 4PSK example as in Figure 2, the TCs for the SISO-MSDSD module are shown for different values of $N$ in Figure 4. Also depicted are the TCs for BCJR decoding of 4PSK with perfect CSI and for the memory $\nu = 2$ convolutional code of rate 1/2, respectively. Clearly, decoding with perfect CSI available at the receiver (cf. e.g. [5], [6]) is the absolute performance benchmark for our MSDSD approach. The step function (dashed line) will be addressed in Section IV-B.3.

We observe from Figure 4 that the initial value of $I_{\text{MSDSD}}^{\text{out}}$ for $I_{\text{MSDSD}}^{\text{in}} = 0$ barely improves with increasing $N$. But, interestingly, the slopes of the TC curves change considerably with $N$ and become similar to the one for perfect CSI as $N$ grows. This observation suggests that a relatively small window size is used in the first decoding iteration and that during the following iterations $N$ is gradually increased in order to finally approach the performance of decoding with CSI. In fact, in doing so a tunnel opens between the SISO decoder TCs and convergence of iterative decoding is achieved. Such a procedure also accommodates our goal of low decoding complexity. According to Figure 2, larger $N$ become especially attractive when more a priori information is available.

As can be seen from Figure 4, there are two types of
performance penalties compared to the ideal case of perfect CSI. First, perfect CSI raises the TC and a tunnel opens, i.e., convergence is achieved at lower SNRs. Second, the TC of MSDSD does not end in the point \( I_A^{MSDSD} = I_E^{MSDSD} = 1 \). This can be explained by the fact that for non-zero noise the unknown fading channel cannot be estimated perfectly for \( B_kT > 0 \) even with infinite \( N \). The gap to the point \((1, 1)\) is highly dependent on the fading rate (see Figure 6 for the AWGN channel for a comparison). Furthermore, MSDSD does not decode the entire inner rate-1 code, but only operates locally within the observation window. As a consequence, the BER remains nonzero also for asymptotically infinite interleaver (code) length. We can estimate this error rate and hence, the potential improvement in the BER due to increasing \( N \), by evaluating the BER approximation (bound for \( I_A^{MSDSD} = 1 \)) from Section III-D. The numerical results for the considered memory \( \nu = 2 \) code and the different \( N \) are shown on the right-hand side of Figure 4. With \( N = 20 \) we expect a BER of about \( 10^{-4} \) for the given SNR. In principle, this BER can be arbitrarily lowered by further increasing \( N \). For example, \( \text{BER} = 3.0 \cdot 10^{-3} \) can be achieved if \( N = 30 \) is chosen (not shown in Figure 4). This, in fact, is feasible, since computational complexity is practically quadratic in \( N \) at the considered operation point of \( I_A^{MSDSD} \approx 1 \).

3) Design Examples: Making use of the EXIT chart technique, favorable designs for the serial concatenation of convolutional codes with M/DPDPSK and iterative decoding with SISO MSDSD can readily be found. To keep decoding complexity at a moderately low level, (i) preferably small observation sizes \( N \) during the first iterations and (ii) a preferably small number of total iterations are aimed at. Table I shows the parameters of 8 particular designs that were found for both the AWGN and the Rayleigh fading channel. Thereby, the target transmission rate \( R \) varies between 1 and 2 bit/(channel use). The value of \( N \) in the final iteration is adjusted such that \( \text{BER} \lesssim 10^{-4} \) is expected at the design \( E_b/N_0 \). Again, we used the upper bound for perfect feedback \( (I_A^{MSDSD} = 1) \) from Section III-D to estimate these BERs. The designs for \( R = 1 \) bit/(channel use) (4PSK and \( R_c = 1/2 \)) are illustrated in Figs. 4, 5, and 6, respectively, where the step function represents the expected trajectory of iterative decoding.

It is illustrative to compare Designs 4 and 3 for the Rayleigh fading channel with each other, which employ codes of different memory and various labelings and are shown in Figs. 4 and 5, respectively. In compliance with the conclusions in Section IV-B.1, SPL is used in combination with the memory \( \nu = 2 \) code and GL is used for the memory \( \nu = 6 \) code. Following the respective decoding trajectories, we note that the memory \( \nu = 6 \) convolutional code allows for less decoding iterations than the memory \( \nu = 2 \) code. Thus, decoding complexity required for the outer component code is traded for overall decoding complexity in terms of number of iterations.
This trend is also true for the 8PSK designs presented in Table I. Furthermore, the more powerful $\nu = 6$ code is expected to achieve a lower BER for the same final window size $N$, as the intersection point of the respective TCs is moved to the right, which also corresponds to a lower BER bound for perfect feedback ($I_{k=\text{MSDSD}}^\text{a} = 1$).

From the design for the AWGN channel in Figure 6 we can observe the larger slope of the TCs of SISO MSDSD compared to the Rayleigh fading case. Here, the combination of GL with the $\nu = 2$ code is most effective. Since the point $(1,1)$ is almost reached in the EXIT chart, practically zero BER after the final iteration is achieved. This was confirmed in our simulations for the AWGN channel.

As mentioned earlier, the TC of SISO MSDSD depends on the transmission channel characteristics and in particular on the fading rate $B_h T$. Consequently, the optimum design of the SISO-MSDSD module, i.e., the choice of observation size $N$ as function of the iteration, changes with $B_h T$. We found that the given designs obtained for the nominal fading rate of $B_h T = 0.01$ work well in the range of $0 \lesssim B_h T \lesssim 0.02$. A more robust implementation could (a) select from designs optimized offline according to the elements of $L$ (8), which are the coefficients of a linear $(N-1)$st order minimum mean-squared error (MMSE) predictor (cf. [26]) and which can be obtained using adaptive algorithms [50], or (b) increase $N$ during the iterations when error rate does not further improve with the current $N$ based on an estimate of the error rate from the BCJR output, cf. e.g. [51].

V. Simulation Results

We simulated transmission over, respectively, the noncoherent AWGN channel with constant unknown phase and the Rayleigh fading channel without CSI for all 8 designs presented in Section IV-B.3 using exactly the parameters from Table I. To avoid spurious effects with respect to fading correlations and correlations among symbols in iterative decoding, code lengths corresponding to $10^5$ information symbols were chosen. As benchmark, we also simulated DPSK transmission assuming perfect CSI at the receiver. Furthermore, the constellation constrained rate-distortion capacities, i.e., the capacity limits taking the particular PSK constellation and the finite error rate into account [52], for the respective channels with perfect CSI are also considered for comparison.

A. Performance

The performance results for the 8 designs are summarized in Table II in terms of the $E_b/N_0$ required to achieve $\text{BER} = 10^{-4}$ in the simulations. Comparing the respective figures, we observe gaps in power efficiency between transmission with and without CSI of, respectively, 0.7-0.9 dB for the AWGN channel and 1.0-1.5 dB for the Rayleigh fading channel. These results are quite remarkable, recalling that the decoder with CSI employs the optimum BCJR algorithm as component decoder for DPSK. The gap for the AWGN channel and 4PSK/8PSK constellations is also well in the order of the 0.3 dB gap to coherent decoding obtained in [17, Fig. 9] for differential 2PSK using a noncoherent BCJR-type algorithm.7

The figures collected in Table II further reveal that the considered concatenated coding schemes with CSI operate rather close to the capacity limits. More specifically, the distance to capacity is 1.0 dB for the AWGN channel and 1.1-1.4 dB for the Rayleigh fading channel, when considering the most power-efficient design for each rate. Adding the above specified performance penalties due to absence of CSI and sub optimum MSDSD to these numbers, our proposed schemes provide performance within 1.7-1.9 dB (AWGN) and 2.3-2.5 dB (Rayleigh fading) of channel capacity assuming perfect CSI. These results should be compared with those

7In [17, Fig. 9], a rate-1/2 recursive convolutional code is used in combination with 2DPSK and thus, the absolute performance is worse than that achieved with our scheme (Design 2), which also achieves double the data rate.
TABLE II
PERFORMANCE RESULTS FOR 8 DESIGNS FROM TABLE I. COMPARISON WITH RATE-DISTORTION CAPACITY AND SIMULATION RESULTS FOR THE PERFECT CSI CASE, RESPECTIVELY, AND TARGET BER = 10^{-4}.

<table>
<thead>
<tr>
<th>Design #</th>
<th>Channel</th>
<th>Rate (bit/channel use)</th>
<th>required $E_b/N_0$ for BER = 10^{-4}</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AWGN</td>
<td>1.0</td>
<td>0.2 dB</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>2.0</td>
<td>2.7 dB</td>
</tr>
<tr>
<td>3</td>
<td>Rayleigh</td>
<td>1.0</td>
<td>1.9 dB</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>1.5</td>
<td>3.2 dB</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>2.0</td>
<td>5.4 dB</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

It is also insightful to compare the $E_b/N_0$ required for BER = 10^{-4} in Table II (no CSI case) with the respective design $E_b/N_0$ in Table I. From this we find that convergence of iterative decoding is achieved for an $E_b/N_0$ not more than 0.5 dB away from the predicted value. This slight deviation can be attributed to the suboptimum soft-output generation of MSDSD (25) and to the Gaussian approximation made for straightforward generation of the EXIT characteristics.

To further support the success of our EXIT chart based designs in Table I, the simulated BERs are displayed exemplary for Designs 3 (rate 1 bit/(channel use)) and 7 (rate 2 bit/(channel use)) in Figs. 7 and 8, respectively. As can be seen, the BER curves exhibit the “waterfall” behavior typical for iterative decoding. Without CSI, the performance approximations from Section III-D are well approached. Thus, these approximations can be seen as very useful tools to choose the observation window size $N$ required for a target BER. We also note that the BER can be lowered almost arbitrarily at the SNR of convergence by performing one or more additional iterations with appropriately increased window sizes $N$.

Figure 7 also shows a BER curve taken from [4, Fig. 6] obtained by Hoche and Lodge (H&L) for the same transmission parameters. Again, they used a BCJR-type SISO module similar to the one in [17] for decoding without CSI. A linear predictor of order $P = 3$, which corresponds to an observation window size $N = 4$, and decoding in a $(4^P = 64)$-state trellis and three decoding iterations are performed. We observe that the H&L-curve is similar to that with SISO MSDSD and $N = 4$. Since the BCJR-type module fully decodes the rate-1 DPSK code, its performance is not limited by the bounds from Section III-D. On the other hand, the complexity of the BCJR-type module grows exponentially with $P = N - 1$ and hence, closing the gap to the BER curve for perfect CSI.
is problematic. This complexity problem becomes even more acute for higher-order modulation, e.g., for 8PSK. Of course, reduced-state BCJR-type modules as devised in [17] for the AWGN channel constitute an alternative approach to SISO MSDSD.

Finally, we would like to mention a comparison of our scheme with the forward-backward combining method presented by Su and Geraniotis for 2PSK in [19]. In particular, we consider performance results for convolutional coded 2DPSK and $N = 10$ depicted in [19, Fig. 7]. We verified that the best BER curve in [19, Fig. 7] approaches the BER bound for perfect feedback and $N = 10$. Thus, we conclude that the method of [19] performs similar to iterative decoding with SISO MSDSD for the special case of 2PSK transmission, i.e., rates $R < 1 \text{bit/(channel use)}$. A generalization to $M$PSK is not given in [19] and appears to be not straightforward.

### B. Complexity

During the BER simulations, we monitored the number of RMOs required for SISO MSDSD as specified in Section IV-A. We found that, in general, the measured numbers match quite well with those predicted from the complexity exponent chart, e.g., Figure 2 for Design 4, even if the point of convergence slightly deviated from the design SNR. For Design 4, for example, we observed values of about $[30, 340, 1400, 1000, 900, 700, 600, 500, 430, 1400, 1000, 950, 5300]$ for the expected number of RMOs per output during iterations 1 to 13. When judging these absolute numbers, we should compare them with the lower complexity bound from Section IV-A, which is about 30, 150, 290, 640 and 2000 RMOs for $N = 2, 4, 6, 10,$ and 20, respectively. Furthermore, we note that BCJR-type SISO algorithms easily involve (several) hundreds of branches [18], [11].

Since it is also desirable to limit the maximum complexity, we use these numbers for average complexity to determine meaningful limits. Two strategies appear to be reasonable. In

the first strategy, the maximum number of RMOs $m_{\text{max}}$ varies over the iterations depending on the respective average number of RMOs $m_{av}$. In the second strategy, an absolute maximum number $m_{\text{max}}$ is imposed. For Design 4, Figure 9 displays the BER as function of the iteration and the parameter $m_{\text{max}}$. The $E_b/N_0$ is adjusted to 4.5 dB to ensure convergence of iterative decoding (see Table II). As implementation of the first strategy, $m_{\text{max}} = (1.0, 1.5) \cdot m_{av}$ are applied. For the second strategy, $m_{\text{max}} = 3000$ RMOs per output is chosen, which is close to the lower complexity bound for $N = 20$, the observation window size in the last iteration. The curve for $m_{\text{max}} = 10^7$ RMOs per output can be regarded as the BER for MSDSD with unconstrained maximum complexity.

The results in Figure 9 demonstrate that the maximum complexity $m_{\text{max}}$ can be safely reduced to a value that is very much in the order of the average complexity $m_{av}$. Especially the first strategy with $m_{\text{max}} = 1.5 \cdot m_{av}$ is rewarding if relatively low error rates are desired, as it accounts for the comparatively higher complexity in the last iteration(s) due to larger $N$. This from a practical point of view quite pleasant result has been confirmed for other design examples given in Table I.

### VI. CONCLUSIONS

In this paper, we have studied powerful coded $M$-ary DPSK transmission with iterative decoding, so-called “Turbo DSK”, with an emphasis on moderate-complexity yet highly power-efficient decoding in the absence of CSI. We have derived a novel SISO component decoder for MDPK, which efficiently accomplishes multiple-symbol and a priori information processing through the application of MAP MSDSD. Since this SISO-MSDSD module inherits the favorable complexity properties of sphere decoding, its introduction to iterative decoding without CSI is interesting in its own right.

By means of EXIT chart analysis the SISO-MSDSD module was appropriately embedded in the iterative decoding process. In particular, we suggested to adapt the observation window...
size $N$ to the quality of the MSDSD soft input, therewith keeping computational complexity as low as possible. The EXIT chart design was complemented by the BER approximations obtained for SISO MSDSD. Using the new iterative decoder and employing standard outer convolutional codes of various memory orders, remarkably power-efficient transmission in close vicinity of the capacity limits was achieved for reasonably bandwidth-efficient modulation. Simulation results further indicated that SISO MSDSD is not only advantageous in terms of average complexity, but that also the maximum number of required operations can be strongly limited without sacrificing performance.
Appendix I

Pseudocode for MAP-MSDSD

For clarity, we use $\vec{x}$ and $x_m$ to denote an $M$-dimensional vector and its $m$th input in this Appendix.

```plaintext
// U - N x N matrix (see (9)) with entry $u_{i,i}$ in row $i$ and column $i$
// $\vec{v}$ - $M$-dimensional vector of PSK symbols with entries $e_{m} = e^{2\pi(m-1)/M}$
// M, N, R - PSK constellation size, observation interval, and initial radius, respectively
// P - $(N-1) \times M$ matrix with entry $p_{im} = \log(\Pr\left\{ e_{i} = e^{2\pi(m-1)/M}\right\} )$ in row $i$ and column $m$

function $[\vec{v}_{map}, d_{i}^{2}] = \text{MAP-MSDSD}(U, \vec{v}, M, N, R, P)$

1. $a_{N} := 1$
2. $d_{N} := 0$
3. $i := N - 1$
4. $k_{i} := u_{(N-1)N}$
5. $[\vec{v}_{i}, \vec{a}_{i}, \Delta_{i}^{2}, n_{i}] = \text{sortDelta}(k_{i}, u_{ii}, \vec{a}_{i+1})$
6. (loop)
7. $d_{i}^{2} := (\Delta_{i}^{2})_{n_{i}} + d_{i+1}^{2}$
8. if $d_{i} < R$ and $n_{i} \leq M$
9. $i_{i} := \binom{(\vec{v}_{i})}{n_{i}}$
10. $v_{i} := (\vec{a}_{i})_{n_{i}}$
11. if $i \neq 1$
12. $i := i - 1$
13. $k_{i} := \sum_{i=i+1}^{N} u_{ii}a_{i}$
14. $[\vec{v}_{i}, \vec{a}_{i}, \Delta_{i}^{2}, n_{i}] = \text{sortDelta}(k_{i}, u_{ii}, \vec{a}_{i+1})$
15. } else if $i = 1$
16. $v_{map} := v$
17. $R := d_{i}$
18. $i := i + 1$
19. } else
20. $n_{i} := n_{i} + 1$
21. } do
22. } while $n_{i} = M$
23. } while $i = N - 1$
24. return $[\vec{v}_{map}, d_{1}^{2}]$
25. exit
26. } end
27. } subfunction $[\vec{v}_{i}, \vec{a}_{i}, \Delta_{i}^{2}, n_{i}] = \text{sortDelta}(k_{i}, u_{ii}, \vec{a}_{i+1})$
28. // Determine $\Delta_{i}^{2}$ and sort signal points for component $i$ according to increasing $\Delta_{i}^{2}$ (see (23))
29. // loop over $M$ symbols
30. // $m$th accumulated symbol
31. // calculate squared length, see (23)
32. // initialize counter of examined candidates
33. // sort all three vectors according to
34. // increasing entries of $\Delta_{i}^{2}$

for $m = 1$ to $M$

$\alpha_{m} := \alpha_{i+1} v_{m}^{*}$

$\delta_{m}^{2} := |u_{ii}a_{m} + k_{i}|^{2} - p_{im}$

$\delta_{m}^{2}$

$n_{i} = 1$

$[\vec{v}_{i}, \vec{a}_{i}, \Delta_{i}^{2}] = \text{qsort}(\vec{v}, \vec{a}, \delta_{m}^{2})$
```


Volker Pauli (S’04) was born in Neuendettelsau, Germany, in 1977. He received the Diplom (University) degree in electrical engineering from the University of Erlangen, Germany, in 2003. He was a Visiting Student at the École Polytechnique Fédérale de Lausanne, Switzerland, in 2002 and the University of British Columbia, Vancouver, BC, Canada, in 2003 and 2005. He is currently a Research Assistant in the Telecommunications Laboratory at the University of Erlangen. His research interests include noncoherent detection, iterative receiver structures, multiple-input multiple-output (MIMO) systems, multicarrier modulation and equalization. In 1999, he received the Lucent Global Science Scholarship. In 2004, he was recipient of the award for exceptional achievements in studying electrical engineering from the Institute for Electrical and Information Technologies of the University of Erlangen.

Lutz Lampe (S’98, M’02) received the Diplom (Univ.) and the Ph.D. degrees in electrical engineering from the University of Erlangen, Germany, in 1998 and 2002, respectively. He is currently an Assistant Professor in the Department of Electrical and Computer Engineering at the University of British Columbia. His main research interests lie in the areas of communications and information theory applied to wireless and power-line transmission. This includes coding and decoding techniques for power- and bandwidth-efficient transmission, space-time coded modulation, detection and estimation techniques, multicarrier modulation, and multiuser detection. He received the Best Student Paper Award at the European Wireless Conference 2000 and at the International Zurich Seminar 2002. In 2003, he received the Dissertation Award from the of the German Society of Information Techniques (ITG). He is an Associate Editor for the IEEE Transactions on Vehicular Technology, and serves as Vice-Chair of the IEEE Communications Society Technical Committee on power-line communications. He was the General Chair of the 2005 International Symposium on Power Line Communications and Its Applications (ISPLC 2005).

Robert Schober Biography text here.