Non-Coherent LDPC Decoding on Graphs

Ansgar Scherb, and Karl-Dirk Kammeyer
Department of Communications Engineering
University of Bremen
Otto-Hahn-Allee
D-28359 Bremen, Germany
Email:{scherb, kammeyer}@ant.uni-bremen.de

***Abstract*— Within this paper graph based non-coherent decoding algorithms for LDPC encoded systems are proposed. We study a class of LDPC codes suitable for non-coherent detection. On the basis of the related graphs a non-coherent decoding algorithm with variable trade-off between computational complexity and BER performance is derived. The proposed scheme is also capable to deal with pilot symbols if available. Finally, the excellent performance of the proposed methods is verified by simulations.**

I. Introduction

Non-coherent transmission concepts are attractive in situations where no information on the current channel phase is available at the receiver. Considering rotationally symmetric modulations, e.g. PSK or QAM, the transmitter has to preprocess the transmit signal, in order to enable the receiver to compensate the indeterminacy of the channel phase. Although differential pre-coding is most common, channel coding may serve for this purpose as well. Thus, in [1] appropriate channel codes for non-coherent transmission were searched by simulated annealing, and in [2] a class of convolutional codes mapped onto M-PSK symbols was presented under the term ”non-coherent coded modulation”. Recently, this approach was extended to non-coherent block coded trellis coded modulation [3].

An essential aspect of non-coherent systems is the effort which has to be spent by the receiver in order to obtain reliable estimates of the transmitted data. An incremental metric for performing maximum likelihood sequence detection was derived in [4]. Unfortunately, the entire history of the transmitted signal has to be taken into account for calculating the increment which yields very high computational complexity. Thus, the authors suggested to limit the size of the instantaneous observation window, in order to keep the effort low.

Recently, we observed that a certain class of low density parity check (LDPC) codes are also suitable for channel-blind data transmission [5]. LDPC codes invented by Gallager in 1963 [6] and rediscovered by MacKay in 1999 [7] are well known for their excellent error correction capabilities. As demonstrated in several contributions, e.g. [8], [9], near Shannon limit performance can be achieved using graph-based decoding techniques as e.g. the sum-product algorithm [10]. Whereas in [11], [5], [12] the redundancy of the LDPC code was exploited to blindly (without the assistance of pilot symbols) identify the channel, in this paper we will present a non-coherent graph-based decoding algorithm delivering directly soft estimates of the transmitted data without estimating the channel phase explicitly.

This paper is organized as follows. In Section II we present the system model. Preliminaries on non-coherent maximum a-posteriori (MAP) decoding are discussed in Section III. In Section IV we study appropriate settings of LDPC codes for non-coherent transmission. A class of non-coherent graph-based LDPC decoding algorithms is derived in Section V. The performance of the proposed method is evaluated by simulations in Section VI. Finally the paper is concluded in Section VII.

Throughout this paper we use the following notation: Column vectors and matrices are denoted by small and capital boldface letters, e.g. $\mathbf{a}$ and $\mathbf{A}$, respectively. $\mathbf{1}_m$ is the all-ones vector of size m. The upper indices “T” and “H” stand for the transpose and Hermitian operation, respectively. Calligraphic letters denote sets[1], e.g. $\mathcal{A}$. The union of two sets is denoted by the operator $\cap$, e.g.

$$\mathcal{A} \cap \mathcal{B} = \{a \in \mathcal{A}, b \in \mathcal{B}\}$$

and the union of I sets is denoted by

$$\bigcap_{l=0}^{I-1} \mathcal{A}_l = \mathcal{A}_0 \cap \cdots \cap \mathcal{A}_{I-1}.$$

The additive combination of two sets is denoted by the operator $\oplus$, e.g.

$$\mathcal{A} \oplus \mathcal{B} = \{a + b | a \in \mathcal{A}, b \in \mathcal{B}\}$$

and the additive combinations of I sets is denoted by

$$\bigoplus_{l=0}^{I-1} \mathcal{A}_l = \mathcal{A}_0 \oplus \cdots \oplus \mathcal{A}_{I-1}$$

.

II. System model

We consider a binary regular (λ, ρ, N, K)-LDPC code $\mathcal{C}$ of code rate K/N defined on the parity check matrix $\mathbf{H} \in \{0,1\}^{M \times N}$; $M \geq N - K$ with λ and ρ is the number of ones per column and row therein, respectively. The corresponding

[1] With exception of $\mathcal{CN}(\mu, \sigma^2)$, which denotes the complex normal distribution with mean μ and variance σ^2.

generator matrix is given by $\mathbf{G} \in \{0,1\}^{N\times K}; \mathrm{mod}_2(\mathbf{HG}) = \mathbf{0}$. $\mathbf{c} = [c_0, \cdots, c_{N-1}]^T$ is a binary code word from $\mathcal{C}$ and $\mathbf{s} = [s_0, \cdots, s_{N-1}]^T$ its BPSK representation in signal space via the mapping $0 \rightarrow 1$ and $1 \rightarrow -1$. Each admissible codeword $\mathbf{c}$ and, consequently, each corresponding sequence $\mathbf{s}$ is assumed to be transmitted with equal probability $1/|\mathcal{C}|$, where $|\mathcal{C}|$ is the cardinality of the code. The channel output can be expressed as

$$\mathbf{r} = \mathbf{s}e^{j\phi} + \mathbf{v}, \tag{1}$$

where ϕ is the unknown phase of the channel uniformly distributed within the interval $[0, 2\pi)$ and $\mathbf{v} = [v_0, \cdots, v_{N-1}]^T$ is i.i.d. zero mean complex gaussian noise with $v_n \sim \mathcal{CN}(0, \sigma_n^2)$. We define the subset $\mathcal{C}_n(a)$ by

$$\mathcal{C}_n(a) = \{\mathbf{c} \in \mathcal{C} | c_n = a\} \tag{2}$$

and, equivalently, the subset $\mathcal{S}_n(b)$ by

$$\mathcal{S}_n(b) = \{\mathbf{s} \leftarrow \mathbf{c} \in \mathcal{C} | s_n = b\}. \tag{3}$$

Note that with the mapping $a \rightarrow b$ the members of set $\mathcal{S}_n(b)$ are the signal space counterparts of the members of the set $\mathcal{C}_n(a)$.

III. Preliminaries

The a-posteriori log likelihood ratio (AP-LLR) w. r. t. c_n is defined by

$$L_n = \log\left(\frac{\Pr\{c_n = 0|\mathbf{r}\}}{\Pr\{c_n = 1|\mathbf{r}\}}\right). \tag{4}$$

Due to the Bayesian rule the a-posteriori probability $\Pr\{c_n|\mathbf{r}\}$ can be decomposed by

$$\begin{aligned}\Pr\{c_n = a|\mathbf{r}\} &= \sum_{\mathbf{c}\in\mathcal{C}_n(a)} \Pr\{\mathbf{c}|\mathbf{r}\} \\ &\propto \sum_{\substack{\mathbf{c}\in\{1,0\}^N \\ \forall c_n = a}} \Pr\{\mathbf{c}\}\, p(\mathbf{r}|\mathbf{c}),\end{aligned} \tag{5}$$

where $\Pr\{\mathbf{c}\}$ is the a-priori probability indicating whether a sequence $\mathbf{c} \in \{0,1\}^N$ is a member of the code $\mathcal{C}$ and $p(\mathbf{r}|\mathbf{c}) = p(\mathbf{r}|\mathbf{s}); \mathbf{c} \rightarrow \mathbf{s}$ is the non-coherent likelihood. Since all codewords $\mathbf{c} \in \mathcal{C}$ have to satisfy M parity check sums $\mathbf{h}_m^T\mathbf{c} = 0$, where $\mathbf{h}_m$ is the m-th column of $\mathbf{H}^T$, we have

$$\Pr\{\mathbf{c}\} = \frac{1}{|\mathcal{C}|}\prod_{m=1}^{M} \delta(\mathrm{mod}_2(\mathbf{h}_m^T\mathbf{c})), \tag{6}$$

where $\delta(.)$ is the Kronecker delta operator. The non-coherent likelihood function $p(\mathbf{r}|\mathbf{s}) = p(\mathbf{r}|\mathbf{c}); \mathbf{s} \rightarrow \mathbf{c}$ is given by

$$\begin{aligned}p(\mathbf{r}|\mathbf{s}) &= \int p(\mathbf{r}|\mathbf{s},\phi)p(\phi)d\phi \\ &= \frac{1}{(\pi\sigma_n^2)^N}\int_0^{2\pi} \exp\left(-\frac{\|\mathbf{r} - e^{j\phi}\mathbf{s}\|^2}{\sigma_n^2}\right)d\phi \\ &= \frac{1}{(\pi\sigma_n^2)^N}\exp\left(-\frac{\|\mathbf{r}\|^2 + \|\mathbf{s}\|^2}{\sigma_n^2}\right)\mathrm{I}_0\left(\frac{2|\mathbf{r}^H\mathbf{s}|}{\sigma_n^2}\right),\end{aligned} \tag{7}$$

where $\mathrm{I}_0(.)$ is the zeroth-order modified Bessel function of first kind. As the amplitude of the considered BPSK carries no information, i.e. $\|\mathbf{s}\|^2 = const.\ \forall\ \mathbf{s} \in \{1,-1\}^N$, and the modified Bessel function is monotonically increasing for positive arguments, the non-coherent likelihood function depends only on the transmitted sequence $\mathbf{s}$ by the magnitude of the inner product $\mathbf{r}^H\mathbf{s}$. Replacing (5), (6) and (7) in (4) we get

$$L_n = \log\left(\frac{\sum_{\mathbf{s}\in\mathcal{S}_n(1)} \mathrm{I}_0(2|\mathbf{s}^H\mathbf{r}|/\sigma_n^2)}{\sum_{\mathbf{s}\in\mathcal{S}_n(-1)} \mathrm{I}_0(2|\mathbf{s}^H\mathbf{r}|/\sigma_n^2)}\right). \tag{8}$$

Alternatively, the desired LLRs can be approximated by the max log likelihood function

$$\begin{aligned}L_n &\approx \log\left(\frac{\max_{\mathbf{s}\in\mathcal{S}_n(1)} \mathrm{I}_0(2|\mathbf{s}^H\mathbf{r}|/\sigma_n^2)}{\max_{\mathbf{s}\in\mathcal{S}_n(-1)} \mathrm{I}_0(2|\mathbf{s}^H\mathbf{r}|/\sigma_n^2)}\right) \\ &\approx 2\left(\max_{\mathbf{s}\in\mathcal{S}_n(1)} |\mathbf{s}^H\mathbf{r}| - \max_{\mathbf{s}\in\mathcal{S}_n(-1)} |\mathbf{s}^H\mathbf{r}|\right)/\sigma_n^2.\end{aligned} \tag{9}$$

The latter expression seems to be more reasonable for practical applications, since the number of summands in the numerator and the denominator of (8) may become very high. However, finding the sequence $\mathbf{s} \in \mathcal{S}_n(a)$ maximizing $|\mathbf{r}^H\mathbf{s}|$ for each $n = 0, \cdots, N-1$ still requires a high computational effort as pointed out in the introduction.

Furthermore, the non-coherent likelihood function (7) obviously is invariant of a change of sign, i.e. $p(\mathbf{r}|\mathbf{s}) = p(\mathbf{r}|-\mathbf{s})$. In order to avoid this sign ambiguity, the channel code has to be designed properly. Asymmetric LDPC codes are well suited for the purpose as we will see in the following section.

IV. Asymmetric LDPC Codes

An admissible signal sequence $\mathbf{s}$ is unique in sign, if the code satisfies the following necessary condition.

Necessary condition: Let $\mathbf{c} \in \mathcal{C}$ be an arbitrary admissible codeword associated to the sequence $\mathbf{s}$. Then, its negation $\bar{\mathbf{c}}$ associated to $-\mathbf{s}$ must not be within the code space, i.e. $\bar{\mathbf{c}} \notin \mathcal{C}$. We call codes excluding symmetric pairs $(\mathbf{c}, \bar{\mathbf{c}})$ "asymmetric codes".

From the next two theorems an easy rule for constructing LDPC codes satisfying the above condition can be deduced.

Theorem 1: Any binary linear code not containing the all-ones sequence is asymmetric.

Proof: The sum of symmetric binary sequences pair $(\mathbf{c}, \bar{\mathbf{c}})$ in GF(2) results in the all-ones sequence, i.e.

$$\mathbf{1}_N = \mathrm{mod}_2(\mathbf{c} + \bar{\mathbf{c}}). \tag{10}$$

Due to the linearity of the code each linear combination of sequences within the code space has to yield also a sequence within the code space. Therefore, any binary linear code not containing the all-ones sequence might either include $\mathbf{c}$ or $\bar{\mathbf{c}}$ but not the two in common.

Theorem 2: *A binary linear code is asymmetric, if the code is associated to a parity check sum with an odd number of summands.*

Proof: Let $\mathbf{c}$ be a sequence from $\mathcal{C}$ and $\mathbf{h}$ a related parity check vector containing ρ ones, i.e. $\mathrm{mod}_2(\mathbf{c}^T\mathbf{h}) = 0$. Then $\mathbf{c}$

has to contain at least one zero, if ρ is odd. Thus, any code corresponding to at least one parity check sum with an odd number of summands excludes the all-ones codeword.

The previous theorem induces that asymmetry may also hold for a local subsequence of the code, e.g. let $\tilde{\mathbf{c}}$ be a subsequence of $\mathbf{c} \in \mathcal{C}$ consisting only of those elements participating on an arbitrary odd parity check sum. Then $\tilde{\mathbf{c}}$ is asymmetric and, consequently, the corresponding signal space representation $\tilde{\mathbf{s}}$ is unique in sign. Moreover, if a code includes m disjoint asymmetric subsequences, then the minimum number of zeros within any admissible codeword $\mathbf{c}$ is equal or larger than m. We may refer this value as the degree of asymmetry.

Due to the above considerations LDPC codes with an odd number of ones per row in their parity check matrix $\mathbf{H}$ seem to be an appropriate choice for non-coherent block transmission. Herein, we consider a regular Gallager random design for the LDPC matrix, which can be described as follows. Let $\mathbf{H}_0$ be a $(N/\rho \times N)$ matrix with ρ subsequent and exclusive ones in each row, e.g.

$$\mathbf{H}_0 = \begin{bmatrix} 1 & \cdots & 1 & & & & & & \\ & & & 1 & \cdots & 1 & & & \\ & & & & & & \ddots & & \\ & & & & & & 1 & \cdots & 1 \end{bmatrix} \quad (11)$$

and $\mathbf{H}_1, \cdots, \mathbf{H}_{\lambda-1}$ randomly generated column permutations of $\mathbf{H}_0$. Then the parity check matrix of an (λ, ρ, N, K) LDPC code is given by

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_0 \\ \vdots \\ \mathbf{H}_{\lambda-1} \end{bmatrix} \quad (12)$$

V. Non-Coherent LDPC Decoding on Graphs

Recall that for exactly determining the desired APP-LLRs $L_n; n = 0, \cdots, N-1$ (cf. (4)) all inner products $\mathbf{s}^H\mathbf{r}$ have to be calculated and sorted with respect to the respective sets $\mathcal{S}_n(1)$ and $\mathcal{S}_n(-1)$. In this section, we derive a decoding method termed as non-coherent decoding on graphs (NCDG), which evolves in parallel all required inner products on a graph. As we will see, the proposed algorithm works perfectly if the graph is free of cycles, whereas it provides at least good approximates in the non-cycle-free case. The processing of our method as well as of the most common LDPC decoding techniques can be illustrated on a so called Tanner-graph (Fig. 1). Each of the N unknown variables c_n is represented by a variable node depicted by a circle and each of the M parity check sums is represented by a check node depicted by a square. Each check node is connected by an edge to those variable nodes, which participate in the particular parity check sum. The set $\mathcal{M}(n)$ contains the indices of the check nodes connected to the n-th variable node and the set $\mathcal{N}(m)$ contains the indices of the variable nodes connected to the m-th check node. NCDG is an iterative algorithm where information (messages) is alternately exchanged between variable and check nodes along the edges. The message transferred from check node m to variable node n regarding $c_n = x$ is represented by the set $\mathcal{E}_{m\to n}(x)$ and, equivalently, the message transferred from variable node n to check node m regarding $c_n = x$ is represented by the set $\mathcal{E}_{n\to m}(x)$. At initialization each message set is empty, whereas its cardinality increases with foregoing iterations. At each check node (and, similarly, at each variable node) the outgoing messages are updated by processing all incoming messages except for the incoming message coming from the direction of the outgoing message. Defining the single element sets $\mathcal{O}_n(x), n = 0, \cdots, N-1$ by

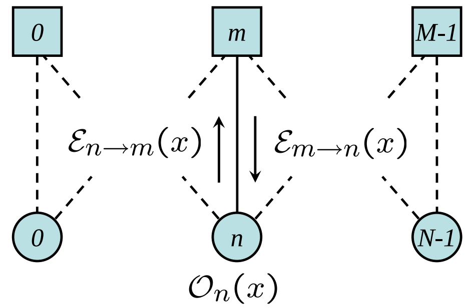

Fig. 1. Tanner-graph

$$\mathcal{O}_n(0) = -r_n \text{ and } \mathcal{O}_n(1) = r_n, \quad (13)$$

the message update rules at variable and check nodes are given in the following way: At each variable node the outgoing message sets are determined by

$$\mathcal{E}_{n\to m}(x) = \mathcal{O}_n(x) \oplus \left(\bigoplus_{\mu \in \mathcal{M}(n)/\{m\}} \mathcal{E}_{\mu\to n}(x) \right) \quad (14)$$

and at each check node by

$$\mathcal{E}_{m\to n}(x) = \bigcap_{\mathrm{mod}_2(\sum c_\nu)=x} \left(\bigoplus_{\nu \in \mathcal{N}(m)/\{n\}} \mathcal{E}_{\nu\to m}(c_\nu) \right). \quad (15)$$

Each set represents a selection of local inner products $\tilde{\mathbf{s}}^H\tilde{\mathbf{r}}$, where $\tilde{\mathbf{s}}$ and $\tilde{\mathbf{r}}$ are subsequences of $\mathbf{s}$ and $\mathbf{r}$, respectively. During iterations the cardinality of the message sets may grow with power of $\lambda - 1$ at each variable node update and with power of $\rho - 1$ at each check node update according to the length of the local subsequences. Thus, the computational complexity of the proposed method is very high and not feasible in practical systems. However, if the graph is free of cycles, after a finite number of iterations the cardinality of the message sets stays steady and the desired inner products can be obtained by

$$\{\mathbf{s}^H\mathbf{r} | \mathbf{s} \in \mathcal{S}_n(a)\} = \mathcal{O}_n(x) \oplus \left(\bigoplus_{\mu \in \mathcal{M}(n)} \mathcal{E}_{\mu\to n}(x) \right). \quad (16)$$

The NCDG approach is summarized in Alg. 1.

Algorithm 1 Non-Coherent Decoding on Graphs

Initialize $\mathcal{O}_n(x)$ by (13) and set the message sets to empty
repeat
  Variable node update: Calculate $\mathcal{E}_{n\to m}(x)$ by (14)
  Check node update: Calculate $\mathcal{E}_{m\to n}(x)$ by (15)
until stopping criterion is satisfied
Determine a proper selection of inner products by (16)
Calculate the desired LLRs by (8) or (9)

A. J-NCDG

In order to find a reasonable trade-off between computational effort and decoding performance, we suggest to limit the cardinality of the current message set at each variable and check node update. Let J denote the maximum tolerable cardinality of a message set. In order to keep only those J members in the particular message set which are the most promising ones, we define the clipping function

$$\mathcal{Y} = \text{clip}_{|.|}(\mathcal{X}, J), \tag{17}$$

where the output $\mathcal{Y}$ consists of those J elements in $\mathcal{X}$ with largest magnitude. Including this function in the variable and check node update yields Alg. 2 and Alg. 3. Note that the computational complexity of the modified update rules is linear in J.

Algorithm 2 Variable node update

Initialize $\mathcal{E}_{n\to m} := \mathcal{O}_n(x)$
for all $\mu \in \mathcal{M}(n)/\{m\}, x \in \{0,1\}$ **do**
  $\mathcal{E}_{n\to m}(x) := \mathcal{E}_{n\to m}(x) \oplus \mathcal{E}_{\mu\to n}(x)$
  $\mathcal{E}_{n\to m}(x) := \text{clip}_{|.|}(\mathcal{E}_{n\to m}(x), L)$
end for

Algorithm 3 Check node update

Initialize $\mathcal{E}_{m\to n}(0) = \{0\}$ and $\mathcal{E}_{m\to n}(1) = \{\}$
for all $\nu \in \mathcal{N}(m)/\{n\}$ **do**
  $\mathcal{E}_{m\to n}(0) := \mathcal{E}_{m\to n}(0) \oplus \mathcal{E}_{\nu\to m}(0)$
    $\cap\ \mathcal{E}_{m\to n}(1) \oplus \mathcal{E}_{\nu\to m}(1)$
  $\mathcal{E}_{m\to n}(0) := \text{clip}_{|.|}(\mathcal{E}_{m\to n}(0), J)$
  $\mathcal{E}_{m\to n}(1) := \mathcal{E}_{m\to n}(0) \oplus \mathcal{E}_{\nu\to m}(1)$
    $\cap\ \mathcal{E}_{m\to n}(1) \oplus \mathcal{E}_{\nu\to m}(0)$
  $\mathcal{E}_{m\to n}(1) := \text{clip}_{|.|}(\mathcal{E}_{m\to n}(1), J)$
end for

B. J-NCDG with pilot assistance

Up to now we assumed that the receiver has no pre-knowledge of transmitted data. However, if pilot symbols are available, the decoder may exploit this fact. Thus, in this section we propose an extension of the J-NCDG termed as J-NCDG with pilot assistance (J-NCDG-PA).

Let $\mathbf{s}_{\text{pilot}}$ be a pilot sequence, $\mathbf{r}_{\text{pilot}} = \mathbf{s}_{\text{pilot}} e^{j\phi} + \mathbf{n}_{\text{pilot}}$ be the corresponding observation and $q = \mathbf{s}_{\text{pilot}}^H \mathbf{r}_{\text{pilot}}$ be the inner product of the pilot sequence and the corresponding observation. On the one hand this inner product q has to be taken into account for the clipping criterion used at message updating (cf. Alg. 2 and Alg. 3), i.e. we modify the clipping function

$$\mathcal{Y} = \text{clip}_{|.+q|}(\mathcal{X}, J) \tag{18}$$

such that the output $\mathcal{Y}$ consists of those J elements of $\mathcal{X}$ with largest absolute $|x+q|, x \in \mathcal{X}$. Hence, the decision which elements in $\mathcal{X}$ are worth keeping is additionally supported by the pre-known pilot sequence. On the other hand the pilot sequence should be considered in a similar way at the calculation of the LLRs in (8) or (9) by adding q.

The variable node and the check node updates of the non-coherent decoding methods discussed in this paper are summarized in Table I.

TABLE I
NON-COHERENT LDPC DECODING ON GRAPHS

	NCDG	J-NCDG	J-NCDG-PA
V-update	Eq. (14)	Alg. 2, clip. (17)	Alg. 2, clip. (18)
C-update	Eq. (15)	Alg. 3, clip. (17)	Alg. 3, clip. (18)

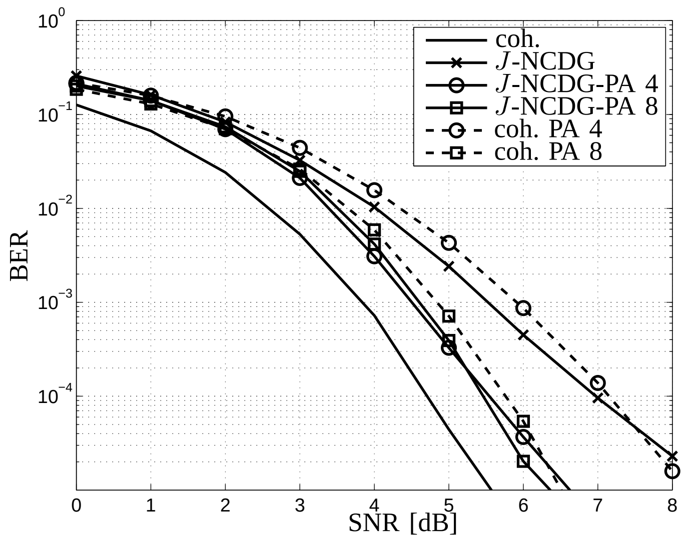

Fig. 2. BER vs. SNR. Simulation parameters: $(\lambda = 3, \rho = 5, N = 100, K = 40)$ random Gallager LDPC code; maximum message set size $J = 4$.

VI. Numerical Results

Fig. 2 and Fig. 3 shows the BER performance of J-NCDG, J-NCDG-PA and coherent receivers (least squares channel estimation, coherent MAP detection and LDPC-decoding via the sum-product algorithm). In Fig. 2 the transmitted data were encoded by a $(\lambda = 3, \rho = 5, N = 100, K = 40)$ LDPC code with random Gallager design, whereas in Fig. 3 a regular $(\lambda = 2, \rho = 3, N = 96, K = 33)$ LDPC code optimized in terms of a maximum girth [13] was used. In order to keep the comparison fair in terms of bandwidth efficiency, the LDPC code was punctured, whenever pilots are embedded. Thus, the total blocklength and the amount of information carried by a block of N symbols stay always constant.

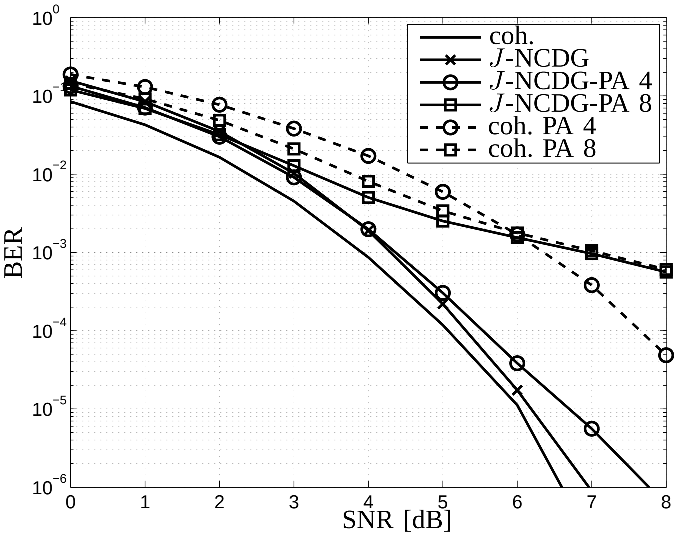

Fig. 3. BER vs. SNR. Simulation parameters: $(\lambda = 2, \rho = 3, N = 96, K = 33)$ LDPC code; maximum message set size $J = 4$.

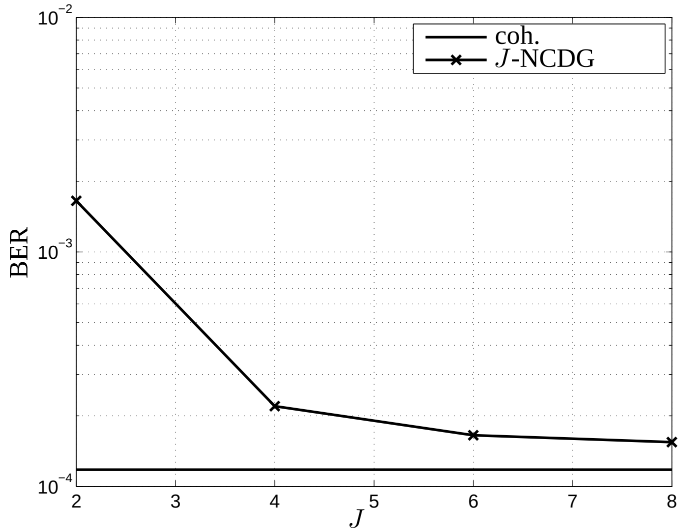

Fig. 4. BER vs. J. Simulation parameters: $(\lambda = 2, \rho = 3, N = 96, K = 33)$ LDPC code; SNR =5dB

As reference, we consider the coherent case with perfect channel state information at the receiver ("coh."). The coherent schemes with pilot assisted channel estimation are denoted as "coh. PA 4" and "coh. PA 8", where 4 and, respectively, 8 pilot symbols were embedded within the transmitted sequence. Similarly, "J-NCDG-PA 4" and "J-NCDG-PA 8" denote the pilot assisted version of NCDG, whereas "J-NCDG" is without pilot assistance. The maximum message set size is set to $J = 4$ for all NCDG schemes.

Considerable performance of the J-NCDG method can be recognized in Fig. 2. However, a gain of approximately 2dB can be achieved by using pilot symbols. In each case the pilot assisted version NCDG is superior to their coherent counterparts.

As observed in Fig. 3 the second LDPC code is more sensitive against puncturing. Thus, the performance becomes poor in case of 8 pilot symbols for both J-NCDG-PA and the coherent receiver. The effect of puncturing is moderate in case of only 4 pilots, though the quality of the channel seems to be not sufficient for the coherent scheme. Remarkable is the fact that J-NCDG without pilot assistance attains the best results and yields a gap from less than 0.5dB to the ideal case. Finally, in Fig. 4 the effect of the maximum message set cardinality J is examined. With increasing J the BER performance becomes better, though beyond $J = 4$ the gains are only moderate.

VII. Conclusions

We have presented a new non-coherent detection algorithm for LDPC encoded systems. The algorithm is variable in terms of a trade-off between computational complexity and BER performance. Moreover, the algorithm can be modified in such a way that pilot symbols can be utilized if available. The algorithm shows a superior performance in comparison to coherent receivers.

References

[1] J. Giese, and M. Skoglund, "Combined Coding and Modulation Design for Unknown Frequency Selective Fading Channels," in *IEEE International Symposium on Information Theory*, Yokohama, Japan, 2003.

[2] D. Raphaeli, "Decoding Algorithms for Noncoherent Coded Modulation," *IEEE Trans. Commun.*, vol. 44, pp. 312-323, Mar. 1996.

[3] R. Y. Wei, "Noncoherent Block-Coded MPSK," *IEEE Trans. Commun.*, vol. 53, No. 6, pp. 978-986, June 2005.

[4] G. Colavolpe and R. Raheli, "Noncoherent Sequence Detection," *IEEE Trans. Commun.*, vol. 47, No. 9, pp. 1376-1385, September 1999.

[5] A. Scherb, V. Kuhn, and K.-D. Kammeyer, "Blind Identification and Equalization of LDPC-encoded MIMO Systems," *IEEE Vehicular Technology Conference*, Stockholm, Sweden, 30 May-1 June 2005

[6] R G. Gallager, "Low-Density Parity-Check Codes," *Cambridge*, MA: M.I.T. Press, 1963.

[7] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," in *IEE Electronics Letters*, vol. 32, no. 18, pp. 1645-1655, 29th Aug. 1996.

[8] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," in *IEEE Trans. Inform. Theory*,vol. 45, pp. 399-431, March 1999

[9] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of Capacity Approaching Irregular Low-Density Parity-Check Codes," in *IEEE Transactions on Information Theory*, vol. 47(2), pp. 619637, February 2001.

[10] F. R. Kschischang, B. J. Frey, and H. A. Loeliger, "Factor Graphs and the Sum-Product Algorithm," in *IEEE Transactions on Information Theory*, vol. 47(2), pp. 498 - 519, February 2001.

[11] A. Scherb, V. Kuhn, and K.-D. Kammeyer, "On Phase Correct Blind Deconvolution exploiting Channel Coding," in *IEEE International Symposium on Signal Processing and Information Technology*, Darmstadt, Germany, 14-17 December 2003

[12] A. Scherb, V. Kuhn, and K.-D. Kammeyer, "On Phase Correct Blind Deconvolution of Flat MIMO Channels Exploiting Channel Encoding," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Philadelphia, PA, USA, March 18-23, 2005

[13] H. Zhang and J. M. F. Moura, "The Design of Structured Regular LDPC Codes with Large Girth," in *IEEE Globecom*, San Francisco, USA, December 2003