Sphere-Decoder-First Channel Code Design

Frank Kienle
Microelectronic Systems Design Research Group
University of Kaiserslautern
Erwin-Schroedinger-Strasse, 67663 Kaiserslautern, Germany
Email: kienle@eit.uni-kl.de

***Abstract*—Multiple antenna systems are promising approaches to increase the data rate of wireless communication systems. There, modulated symbols are multiplexed on multiple transmission antennas. Typically an outer error correcting code is used additionally to ensure a desired quality of service for a given data rate. An appropriate communications performance can be achieved by iterative decoding, where probabilistic information are exchanged between outer channel decoder and the MIMO demodulator (closed loop). In this paper we present an architecture feasibility check for the outer channel decoder to reach high throughputs in the closed loop. We will show that current state of the art outer channel decoder implementations (at least for turbo codes) cannot provide these high throughput demands. We present a new methodology for channel code design to reuse the soft MIMO decoder for channel decoder processing. Thus, the sphere decoder engine can be reused for channel code processing for the open and closed loop system respectively. The closed loop and in particular the open loop system shows better communications performance than state-of-the-art schemes.**

I. Introduction

Multiple input multiple output (MIMO) antenna systems are a promising candidate to achieve high data rates on a scattering-rich wireless transmission channel. In a MIMO system a symbol stream is demultiplexed to multiple transmit antennas while the receiver side collects superposed and noise disturbed samples by multiple receive antennas. Many effort has been taken to reduce the complexity of the detection process by introducing constraints in space and/or time. One of the most famous space-time codes are the original Bell Labs layered space-time (BLAST) technique [1] or the well known Alamouti scheme [2].

Typically these techniques are concatenated with an additional channel code to ensure a desired quality of service. The overall data rate is thus determined by the space-time encoder and the channel encoder. The complexity of the receiver is determined by the MIMO APP demodulator and the channel decoder complexity.

Transmission rates near MIMO capacity can be achieved by a serial concatenation of an outer code, interleaver, and modulator. In this case the modulator is a spatial multiplexing of the symbol stream without introducing any further constraints and thus data rate loss. This simple concatenation can be seen as a classical bit interleaved coded modulation scheme (BICM). The coded bits are correlated by the channel code. An interleaver spreads the correlated bits, to ensure 'locally' uncorrelated bits which are mapped to symbols. The outer code has to be matched to the demodulator to achieve MIMO capacity approaching performance, which was shown in [3] for low density parity check (LDPC) codes. The decoding can be done in an iterative manner, where probabilistic (soft) information are exchanged iteratively between demodulator and channel decoder (**closed** loop). Without feedback between demodulator and channel code we call it an **open** loop system. The demodulator has to calculate maximum a posteriori probabilities (APP) for each bit which can be calculated by a modified sphere decoding [4]. The complexity of this soft input soft output MIMO demodulator depends on the number of bits mapped to a transmission vector. A transmission vector is defined as the symbols which are transmitted simultaneously via multiple antennas.

Three new aspects are presented in this paper. First, we introduce a scaling factor which is utilized after the MIMO demodulator output. This scaling can improve the communications performance of a closed loop system (up to 1dB). Second, we present a feasibility study about the mandatory architecture parallelism to yield high throughput receivers. Especially for turbo codes the derived architecture parallelism is 'today' out of scope for implementation. Furthermore the double iterative loop of the closed loop systems gives strong constraints for both instances the channel code decoder and demodulator. Thus it is worth to look if we can reuse the demodulator part for channel code decoding. This is achieved by the sphere-decoder-first channel code design. We design channel codes in a manner that they can be decoded by an instantiated sphere decoder. These channel codes are denoted as Low-Density MIMO codes (LDMC) and they can be decoded by the sphere decoder in the case of open and closed loop decoding. The third new result are the very good open loop communications performance of LDMC codes. We show for the 64QAM 4x4 antenna system that the **open** loop performance of LDMC codes are even better than the **closed** loop performance of the BICM scheme with WiMAX LDPC codes.

II. State of the art system

We assume the system is composed of M_T transmit and M_R receive antennas with M_T=M_R. In one time slot (one channel use) a sequence of M_T symbols is transmitted with $\mathbf{s} = [s_1, ..., s_{M_T}]^T$. Each symbol s_j is taken out of constellations of size 2^Q. Thus $M_T \cdot Q$ coded bits are transmitted per channel use. $\mathbf{s}$ is denoted as constellation or transmission

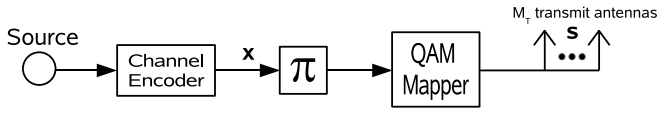

Fig. 1. Typical MIMO transmitter with the encoded codeword $\mathbf{x}$ and the constellation vector $\mathbf{s}$.

vector in the following. The received signal $\mathbf{y}$, denoted as channel vector, can be expressed by

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n}. \tag{1}$$

With $\mathbf{H}$ the channel matrix of dimension $M_T \mathrm{x} M_R$ and $\mathbf{n}$ the noise vector of dimension M_R. The entries in $\mathbf{H}$ are modeled as independent, complex, zero-mean, Gaussian random variables. Real and imaginary part are independent each with variance $\sigma^2 = N_0/2$. We assume $\mathbf{H}$ is perfectly known to the receiver and remains constant for an entire codeword of length N_c.

Figure 1 shows a typical MIMO transmitter. The source bits are encoded by an outer channel code of code rate R. The resulting codeword $\mathbf{x}$ has length N_c with K_c information bits. The encoded bits x_i are interleaved and mapped to one symbol s_j. The symbols are then multiplexed to the different antennas and transmitted via one transmission vector $\mathbf{s}$. The overall data rate is $\eta = R \cdot M_T \cdot Q$ which reflects the number of information bits per channel use.

The channel code can be a convolutional code, turbo-code (TC) or LDPC code respectively. This transmission scheme can be seen as bit-interleaved coded modulation (BICM) scheme.

Figure 2 shows a receiver structure with a demodulator concatenated with an outer channel decoder. For iterative decoding the demodulator calculates the a posteriori probability (APP) values of each bit x_i by evaluating

$$\lambda(x_i|\mathbf{y}) = ln\frac{P[x_i = +1|\mathbf{y}]}{P[x_i = -1|\mathbf{y}]}. \tag{2}$$

These log-likelihood ratios (LLR) λ which contain the intrinsic (channel) information are passed deinterleaved (π^{-1}) to the outer decoder. The channel decoder calculates a new maximum a posteriori information which is denoted with $\mathbf{\Lambda}$. During the iterative message exchange between demodulator and outer decoder the input message have to be subtracted which is well known from the turbo principle. $\mathbf{L^a}$ is the a priori information which is passed back to the MIMO APP detector. This information does not contain any intrinsic (channel) information. During the first demodulation there exist no a priori information ($\mathbf{L^a = 0}$). In the second iteration we obtain $\lambda^{\mathbf{e}}$ composed of the intrinsic information plus the extrinsic information which reflect the additional gain obtained by the MIMO APP detector. The major problem of this iterative decoding is as already mentioned the complexity of the MIMO APP demodulator.

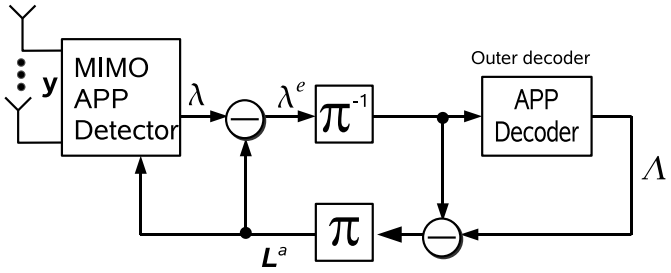

Fig. 2. Typical MIMO receiver with iterative loop with the a priori information $\mathbf{L}^a$ and the extrinsic information $\mathbf{L}^e$ passed to the outer decoder.

The LLR for a decoded bit x_i can be calculated by

$$\lambda(x_i|\mathbf{y}) = \log \frac{\sum_{\{\mathbf{s}:\mathbf{x_i}=+\mathbf{1}\}} \mathbf{e}^{-\frac{\mathbf{1}}{\mathbf{N_0}}||\mathbf{y}-\mathbf{H}\mathbf{s}||^{\mathbf{2}}+\sum_{\mathbf{k}=\mathbf{1}}^{\mathbf{M_T Q}} \mathbf{x_k}\frac{\mathbf{L_k^a}}{\mathbf{2}}}}{\sum_{\{\mathbf{s}:\mathbf{x_i}=-\mathbf{1}\}} \mathbf{e}^{-\frac{\mathbf{1}}{\mathbf{N_0}}||\mathbf{y}-\mathbf{H}\mathbf{s}||^{\mathbf{2}}+\sum_{\mathbf{k}=\mathbf{1},\mathbf{k}}^{\mathbf{M_T Q}} \mathbf{x_k}\frac{\mathbf{L_k^a}}{\mathbf{2}}}}, \tag{3}$$

with $\{\mathbf{s} : x_i = \pm 1\}$ the set of bit vectors $\mathbf{x}$ with $x_i = +1$ or $x_i = -1$ respectively. Equation 3 can be approximated by

$$\lambda(x_i|\mathbf{y}) \approx -\min_{\{\mathbf{s}:\mathbf{x_i}=+\mathbf{1}\}}\{\lambda_\mathbf{0}\} + \min_{\{\mathbf{s}:\mathbf{x_i}=-\mathbf{1}\}}\{\lambda_\mathbf{1}\} \tag{4}$$

λ_i is the metric of one possible path through the tree and evaluates to:.

$$\lambda_i = -\frac{1}{N_0}||\mathbf{y} - \mathbf{H}\mathbf{s}||^{\mathbf{2}} + \sum_{\mathbf{k}=\mathbf{1}}^{\mathbf{M_T Q}} \mathbf{x_k}\frac{\mathbf{L_k^a}}{\mathbf{2}}. \tag{5}$$

L_k^a are the a priori values calculated by the channel decoder.

Equation 4 is an approximation and always overestimates the calculated APP. Thus it is highly desirable to downscale this overestimation within the iterative process. The downscaling has to be performed only on the additional gain the APP decoder delivers between two iterations. Thus

$$L^e(x_i|\mathbf{y})^{\mathbf{iter}} = \lambda(\mathbf{x_i}|\mathbf{y})^{\mathbf{iter}} - \lambda(\mathbf{x_i}|\mathbf{y})^{\mathbf{iter-1}} \tag{6}$$

This new and better approximated APP value to the channel decoder is:

$$\begin{aligned} \lambda(x_i|\mathbf{y})^{iter} &= \lambda(x_i|\mathbf{y})^{iter-1} + L^e(x_i|\mathbf{y})^{iter} \cdot 0.75 \\ &= \lambda(x_i|\mathbf{y})^{iter} \cdot 0.75 + \lambda(x_i|\mathbf{y})^{iter+1} \cdot 0.25 \end{aligned} \tag{7}$$

This downscaling of the extrinsic information of the MIMO demodulator can improve the iterative decoding by up to 1dB which is shown in the results chapter. The downscaling is well known for turbo codes and their Max-Log MAP implementations. The argumentation for the APP calculation here is identical and thus in this paper not further described.

Solving Equation 4 is typically done by a sphere decoder. In this paper we do not show how to implement the sphere decoding algorithm. However, we assume that the realization will using a tree as most of the current implementation do.

III. Architecture Feasibility

Let us revisit some complexity considerations for the decoder system shown in Figure 2. We distinguish between the **closed loop** with feedback between channel decoder and

$\#cycles$	Number of cycles required to process one block
$P_{I/O}$	Parallelization of the input/output
$iter$	Number of iterations of the channel decoder
P	Parallelization of the decoder architecture, for turbo codes parallelization of the MAP architecture, LDPC codes the number of concurrently processed edges
$\overline{d_{VN}}$	Average variable node degree in the Tanner graph (LDPC codes)
(N, K)	(Block length, Number of information bits)
R	Code rate of the channel code
f_{clk}	Clock frequency
$\delta_{overhead}$	Additional fixed overhead due to architecture constraints (e.g. flush the pipeline)
$\#\frac{bits}{cycle}$	Normalized throughput: number of information bits decoded per clock cycle

MIMO demapper and the **open loop** system without feedback. For the open loop system we assume that the MIMO demapper provides soft-output information to the channel decoder. The communications performance gain of the closed loop system is about 3dB [3][4] compared to an open loop system.

For the realization of the closed loop one APP demodulator and one channel decoder can be instantiated, as shown in Figure 2. One block is decoded, while an equal balancing of the processing time between these two instances is assumed. Thus, each of them is 50 percent of the overall time in an idle mode. This equal balancing is just a starting point and relates to the typical iterative turbo processing where information is exchanged between two MAP components. Here, in Figure 2, the two components are the demodulator and outer channel decoder which are as well separated by an interleaver. We could process two blocks concurrently in this engine, while one is processed by the demodulator and the other is processed by the APP decoder. However, the different number of iterations of the channel decoders and the feedback loop respectively will result in a difficult scheduling and thus throughput losses. When processing two blocks in the engine, it is even better to instantiate two independent closed loop decoders with an appropriate throughput.

For the following throughput considerations we assume one instantiated MIMO detector and one APP detector where one block is processed. Let us summarize some throughput considerations for the outer channel decoder utilizing turbo decoder and LDPC decoder respectively.

The expected normalized throughput is

$$\#\frac{bits}{cycle} = \frac{K}{\#cylces} = \frac{N \cdot R}{\#cylces}. \tag{8}$$

The normalized throughput defines a good quality measure about an architecture. E.g LTE-advanced will require a turbo decoder architecture with $\#\frac{bits}{cycle} \sim 2$. For a typical frequency of $f_{cyc} = 300MHz$ this would yield to a payload (information bit) throughput of $T_{payload} = \#\frac{bits}{cycle} \cdot f_{cyc} = 600Mbit/s$.

A. Low-Density Parity-Check Decoder

The number of parallelism for LDPC codes is defined as the number of simultaneously processed edges. The normalsized throughput of a LDPC decoder can be approximated by:

$$\#\frac{bits}{cycle} \approx \frac{N \cdot R}{iter \cdot \frac{N \cdot \overline{d_{VN}}}{P}} = \frac{P \cdot R}{Iter \cdot \overline{d_{VN}}} \tag{9}$$

Most of the current partly parallel architecture use a so called layered architecture where a nearly continuous processing of the edges takes place. No overhead cycles ($\delta_{overhead}$) are present within the iterative loop, for more details see e.g. [5][6]. An average variable node degree of $\overline{d_{VN}} = 3.2$ (WiMAX LDPC) is assumed to derive the parallelism of a possible decoder architecture.

$$P = \left(\#\frac{bits}{cycle}\right) \cdot iter \cdot 3.2 \cdot \frac{1}{R} \tag{10}$$

Normalized Throughput	Turbo Codes K=6000		LDPC Codes $R = 1/2$	
	iterations	Parallelism	iterations	Parallelism
$\#\frac{bits}{cycle} = 1$	4 iter	8.35	5 iter	32
	6 iter	12.82	10 iter	64
	8 iter	17.49	20 iter	128
$\#\frac{bits}{cycle} = 2$	4 iter	17.49	5 iter	64
	6 iter	27.52	10 iter	128
	8 iter	38.58	20 iter	256

TABLE I
PARALLELIZATION OF AN **open loop** ARCHITECTURE FOR THE GIVEN ITERATIONS AND NORMALIZED THROUGHPUT.

B. Turbo Codes

P for turbo codes define the parallelization of the MAP architecture and thus, as well, the number of messages which are exchanged between the two component decoders. For Turbo decoding a normalized throughput can be expressed as

$$\#\frac{bits}{cycle} \approx \frac{P}{2 \cdot iter \cdot \left(1 + \delta_{overhead} \cdot \frac{P}{K}\right)} \tag{11}$$

Turbo decoding has two half iteration to process the two component codes, thus the $2 \cdot iter$. The $\delta_{overhead}$ cycles are currently a big obstacle for the turbo processing. Normally we have to flush the MAP pipeline before starting to process the next half iteration. Many research is going on to reduce this overhead cycles, however the new LTE Release8 has a 'insane' high puncturing which result into code rates $R > 0.9$ which hampers the reduction of these $\delta_{overhead}$ cycles. The bad thing about the turbo decoding is that for moderate length $K \sim 5000$ and increasing architecture parallelism P the additional term $\delta_{overhead} \cdot \frac{P}{K}$ will limit the desired throughput increase.

For the following calculations the turbo decoder parallelization is calculated with $\delta_{overhead} = 32$ and evaluates to:

$$P = \frac{\left(\#\frac{bits}{cycle}\right) \cdot 2 \cdot iter}{1 - \frac{\left(\#\frac{bits}{cycle}\right) \cdot 2 \cdot iter \cdot 32}{K}}. \tag{12}$$

Table I shows the expected parallelism for a Turbo decoder architecture and a LDPC decoder architecture for a open loop system. A normalized throughput of $\frac{bits}{cycle} = 1$ and $\frac{bits}{cycle} = 2$ is assumed. The parallelization for the Turbo Code is based on the Max-Log MAP decoder engine. E.g the MAP engine

Normalized Throughput	Turbo Codes K=6000		LDPC Codes $R = 1/2$	
	iterations	Parallelism	iterations	Parallelism
$\#\frac{bits}{cycle} = 1$	2 big - 3 code	26	2 big - 5 code	128
	3 big - 3 code	35	3 big - 5 code	192
	4 big - 3 code	52	4 big - 5 code	256
$\#\frac{bits}{cycle} = 2$	2 big - 3 code	55	2 big - 5 code	256
	3 big - 3 code	77	3 big - 5 code	384
	4 big - 3 code	110	4 big - 5 code	512

TABLE II
PARALLELIZATION OF A **closed loop** ARCHITECTURE FOR THE GIVEN BIG AND (CHANNEL) CODE ITERATIONS WITH RESPECT TO A NORMALIZED THROUGHPUT.

processes 8 bits per clock cycle for P=8. For LDPC codes the parallelization is the number of simultaneously processed edges.

Now we would like to bring an iterative turbo or LDPC decoder into a MIMO system with a feedback loop via the MIMO detector.

Table II shows the parallelization of a turbo decoder or LDPC decoder for a given normalized throughput assuming a closed loop system. The notation 2 big - 3 code means that the demodulator and channel code is active two times for each block, while the channel code is each time allowed to process 3 (inner) iterations. The huge number of P=26 comes from the already mentioned equal time balancing between demodulator and channel decoder. The mandatory parallelism for, e.g, a $\frac{bits}{cycle} = 1$ closed loop system with 2 big- 3 code iterations translates to a 6 channel decoder iteration in the open loop system. But this time with a normalized throughput of $\frac{bits}{cycle} = 2$.

The number of outer iterations and inner channel code iterations are pretty small in this examples. The parallelization even has to be increased in order to achieve the best possible communications performance.

State-of-the-art turbo decoder have a $P = 8$ architecture. For turbo codes there are of course as well further development targeting a parallelism up to maybe $P = 32$, however the resulting chip size will be very large. In our opinion this is currently a strong argument against a double iterative scheme with targets of $\#\frac{bits}{cycle} = 2$, especially in the case of turbo codes. For LDPC codes the requirements for the architecture parallelism seems to be more feasible. For LDPC decoders a parallelism of $P = 360$ was already presented in 2005 [7], today even a larger parallelism is possible.

The demodulator in the big loop has to provide as well a normalized throughout of $\#\frac{bits}{cycle} = 1$ or $\#\frac{bits}{cycle} = 2$. The advantage of the demodulator is that each received vector $\mathbf{y}$ can be decoded independently. Thus the throughput can be achieved by multiple instances. These multiple instances of the demodulator will result in a large area. To increase the overall architecture efficiency it highly desirable to reuse parts of the demodulator for the channel code processsing. Furthermore to achieve high throughputs it would be of great benefit to get rid of the double iterative scheme.

IV. SPHERE-DECODER-FIRST CHANNEL CODE DESIGN

The goal of the sphere-decoder-first code design is the reuse of the demodulator instances as well for the channel decoding part **and** to get rid of the double iterative decoding scheduling.

The major algorithmic kernel of the sphere decoder is the tree search procedure. In the following we show how to reuse the tree search to decode linear block codes and afterwards we present the overall channel code design.

A. Reuse of the sphere decoder engine

The decoding problem of the sphere decoder can be written in a more general form:

$$\lambda(x_i|\mathbf{y}) = \log \frac{\sum_{\{\mathbf{x}:x_i=+1\}} e^{\mathbf{\Lambda}(\mathbf{x})}}{\sum_{\{\mathbf{x}:x_i=-1\}} e^{\mathbf{\Lambda}(\mathbf{x})}}. \tag{13}$$

The formulation is taken from [8], there it is shown this general problem can be processed by a list sequential decoder. $\mathbf{\Lambda}(\mathbf{x})$ defines a path metric for a possible transmit sequence $\mathbf{x}$. Equation 13 can be used for different types of applications, like ISI channel equalization or multiuser detection [8]. It can be used as well to decode linear block codes with its approximation

$$\lambda(x_i|\mathbf{y}) \approx -\min_{\{\mathbf{C}:x_i=+1\}} \left\{ \sum_{k=1}^{K_c} x_k \frac{\Lambda_k}{2} \right\} + \min_{\{\mathbf{C}:x_i=-1\}} \left\{ \sum_{k=1}^{M_T Q} x_k \frac{\Lambda_k}{2} \right\}. \tag{14}$$

To decode a linear block code an observed bit x_i has to be an element of a valid codeword $\mathbf{x}$ out of the code space $\mathbf{C}$. We have to search over the entire set $\{\mathbf{C} : x_i = +1\}$ of the code space $\mathbf{C}$ which fulfills a linear block code constraint $H\mathbf{x}^T = 0$.

Imagine a small block code $H_e^{'}$ of block length $N_e \leq M_T Q$ is embedded in one transmission vector. Since the sphere decoder architecture can search the entire code tree with all its $2^{M_T Q}$ possibilities, the architecture is as well designed to search through the embedded channel code space.

Note that the code space $\mathbf{C}$ is smaller than the full search space to evaluate Equation 3 and depends on the code rate of the embedded code. This joint detection and decoding of a small linear block code was first proposed in [9].

Currently many paper and observations take place about the implementation of a sphere decoder which can handle soft input information and delivers soft output information [10][11][12]. A sphere decoder in hardware will instantiate all components we need to traverse a tree [10]. The only part missing to decode a small block code is the check of a valid path (codeword) during node extension. This check can be easily accomplished by checking the valid codeword constraint $H_e^{'}{\mathbf{x_e}}^T = 0$. This check is done all in binary and can been done within one clock cycle.

We have to distinguish between open loop and closed loop sphere processing. In a closed loop sphere processing the MIMO APP demodulation Equation 3 changes to

$$\lambda(x_i|\mathbf{y}) \approx -\min_{\{\mathbf{C}:\mathbf{x_i}=+\mathbf{1}\}} \{\lambda_\mathbf{0}\} + \min_{\{\mathbf{C}:\mathbf{x_i}=-\mathbf{1}\}} \{\lambda_\mathbf{1}\} \tag{15}$$

As mentioned with a much smaller set $\{\mathbf{C} : x_i = \pm 1\}$. The search space can be reduced by one order of magnitude with an embedded code which was shown in [13].

For the open loop processing we can still use the tree search and evaluate Equation 14, this time with the current APP information Λ_k.

B. Channel Code Design

Embedding small codes in each transmission vector enables the reuse of the tree search and reduces the search space for the demodulation. However we have to link them together to obtain a larger block code with an appropriate coding gain. In [13] we have denoted these codes as Low-Density MIMO Codes (LDMC). LDMC codes are linear block codes which can be described by a parity check matrix H_c and thus have to fulfill $H_c \mathbf{x}^T = 0$. The parity check matrix H_c has N_c columns and M_c rows and has to be of full rank. The LDMC parity check matrix can be described by two layers with

$$H_c = \begin{pmatrix} H_g \\ H_e \end{pmatrix} = \begin{pmatrix} & H_g & \\ H_e^{'} & \cdots & 0 \\ 0 & \ddots & 0 \\ 0 & \cdots & H_e^{'} \end{pmatrix} \qquad (16)$$

The first layer H_g is a sparse parity check matrix, while the second layer H_e defines multiple, unconnected sub-codes. Each sub-code $H_e^{'}$ has a codeword length of $N_e^{'} \leq M_T Q$.

As mentioned before, each transmission vector $\mathbf{s}$ carries the information of $M_T Q$ bits. For the transmission it has to be guaranteed that all bits of a sub-code $H_e^{'}$ are transmitted within one transmission vector. The LDMC code structure is shown in Figure 3. The allocation of one sub-code to one transmission vector is the additional constraint to generalized LDPC codes [14]. Without this allocation constraint the LDMC codes can be seen as generalized LDPC codes with component codes which are based on small block codes rather than single parity check codes. The design of quasi cyclic generalized LDPC codes with low error floors was shown in [15]. They show that generalized LDPC codes have an excellent performance in the single input single output antenna system. The design of quasi cyclic LDMC codes is presented in [13][16]. The decoding of LDMC codes in the closed loop has no double iterative loop since the embedded codes $H_e^{'}$ are solved implicitly during demodulation using Equation 15. The LDMC code in the result chapter have all code rate $R = 0.5$. The LDMC codes can be described by $H_e^{'}$ and a degree distribution for the sparse H_g part. The code parameters for the closed loop system where given in [16]. For the open loop system we have to change the H_g degree distribution to obtain the best performance, the matching can be done by EXIT chart anaylisis [13]. The result chapter uses two different antenna modulation schemes, 16-QAM and 64-QAM respectively. The degree distribution of H_g in the case of 16-QAM modulation is $f_{g[3,2,1]} = \{\frac{1}{16}, \frac{2}{16}, \frac{13}{16}\}$,$g_{g[7,6]} = \{\frac{2}{3}, \frac{1}{3}\}$, in the case of 64-QAM $f_{g[2,1,0]} = \{\frac{4}{24}, \frac{19}{24}, \frac{1}{24}\}$,$g_{g[7,6]} = \{\frac{3}{4}, \frac{1}{4}\}$ respectively.

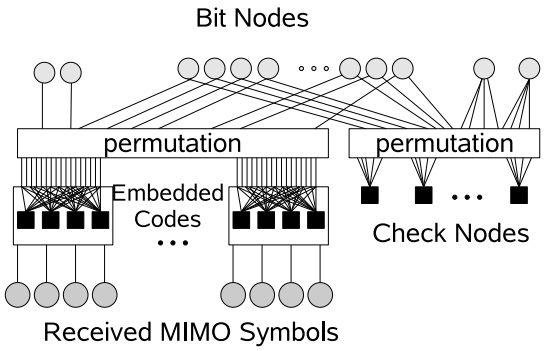

Fig. 3. Graph structure for a LDMC code: in a single-input single output antenna environment we receive APP values for the bit nodes, in the case of MIMO transmission the corresponding symbol nodes are received

V. Results

Figure 4 shows the communications performance for a 16-QAM 4x4 system (8 bits/channel use). Each E_b/N_0 point is simulated with 100k transmitted codewords of length $N_c = 1920$. As mentioned before we assume that the channel matrix $\mathbf{H}$ remains constant within one transmitted codeword, thus for 120 channel uses. The performance curve for the state-of-the-art BICM schemes utilizing WiMAX LDPC codes were performed with 4 big and 5 channel code iterations in the closed loop case. In the open loop case 20 LDPC iterations and 5 LDMC iterations are performed. These are in our opinion realistic iteration numbers which were derived within the feasibility study of Section III. For the closed WiMAX code, two performance curves are shown: one evaluating the normal Min search Equation 4, the other with scaling of the extrinsic information of the MIMO output according to Equation 8. A performance gain of 0.5dB is achieved with the Extrinsic MIMO scaling. This scaling is to the best of our knowledge the first time applied for MIMO demodulation. For more iterations the performance gain can be up to 1 dB depending on the antenna set up and modulation scheme. All following results are always shown with scaling, for LDMC codes and BICM performance curves.

The performance of the LDMC codes is shown with a maximum number of 4 iterations for closed and open loop respectively. The code parameters for the LDMC codes where given in [16] for the closed loop case and for the open loop case in the previous chapter. Note, no inner iterations of H_g are mandatory, the entire decoding is done with the tree search of the sphere decoding algorithm in the closed loop and open loop system set up. Thus with LDMC codes we get rid of the double iterative scheme and an implementation of the outer decoder. At a frame error rate (FER) of $FER = 10^{-2}$ the outage capacity can be approached by 2 dB and outperforms the BICM WiMAX scheme by 2 dB for this set up.

The most interesting and new result is the improved performance for the **open** loop system. The performance gain of the open loop LDMC code compared to the open loop BICM LDPC system is up to 2 dB. This is a very surprising

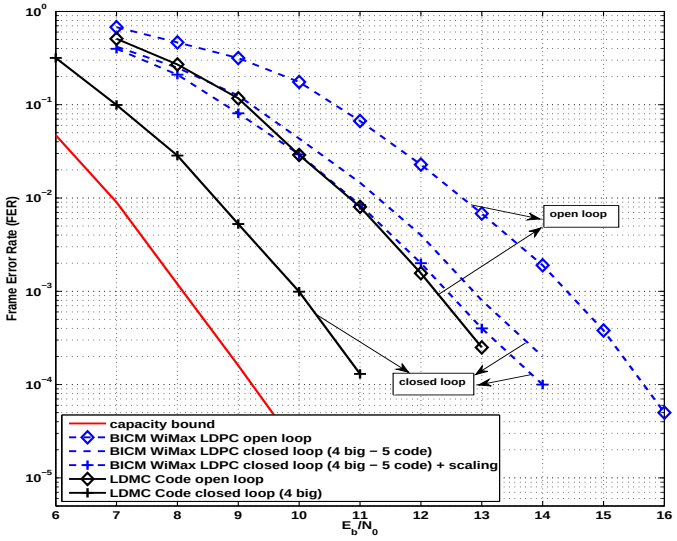

Fig. 4. Open and closed loop communications performance of LDMC code compared to a BICM system with a WiMAX LDPC code all with 4x4 antenna system 16QAM and 8 bits/s/Hz.

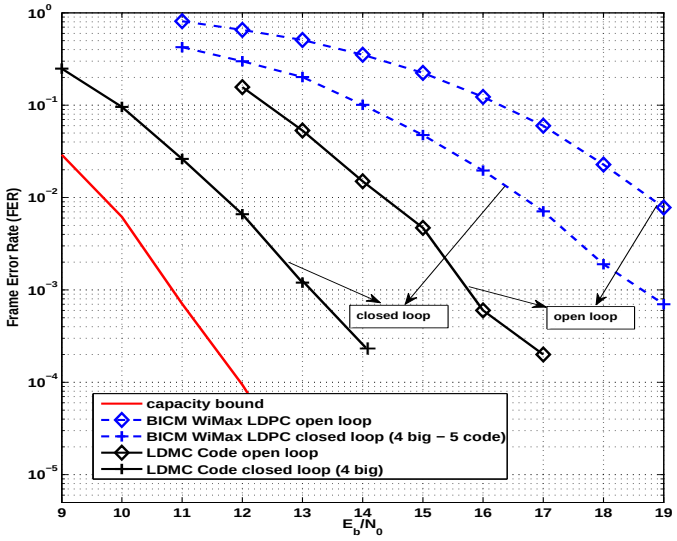

Fig. 5. Open and closed loop communications performance of LDMC code compared to a BICM system with a WiMAX LDPC code all with 4x4 antenna system 64QAM and 12 bits/s/Hz.

result. Initially one could think that we loose performance since we group correlated bits (correlated by H_e') within one transmission vector. However the reduced search space and the achieved information gain during MIMO demodulation is even more dominant.

Figure 5 shows as well the performance with 4x4 antennas and 64-QAM (12 bits/channel use) compared to the outage capacity bound.

Here the performance gain of the closed loop system utilizing LDMC codes is up to 5 dB compared to the closed loop BICM performance. The **open** loop performance of LDMC codes are even better than the **closed** loop performance of the BICM scheme with WiMAX LDPC codes.

VI. Conclusions

MIMO systems are one promising approach to increase the data rate in wireless communications systems. Crucial point for a good communications performance is the MIMO demodulation providing 'soft' values for an iterative decoding process. The presented sphere decoder first channel code design shows that it its possible to design channel codes which reuses the MIMO soft-demodulation for channel decoder processing while providing an excellent communications performance in open loop and closed loop system.

Acknowledgment

This work has in parts been supported by the UMIC Research Center, RWTH Aachen University.

References

[1] M. J. Gans, N. Amitay, Y. S. Yeh, H. Xu, R. A. Valenzuela, T. Sizer, R. Storz, D. Taylor, W. M. MacDonald, C. Tran, and A. Adamiecki, "BLAST system capacity measurements at 2.44 GHz in suburban outdoor environment," in *Vehicular Technology Conference, 2001. VTC 2001 Spring. IEEE VTS 53rd*, vol. 1, 6-9 May 2001, pp. 288–292vol.1.

[2] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *Selected Areas in Communications, IEEE Journal on*, vol. 16, no. 8, pp. 1451–1458, 1998.

[3] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *Communications, IEEE Transactions on*, vol. 52, no. 4, pp. 670–678, April 2004.

[4] H. Vikalo, B. Hassibi, and T. Kailath, "Iterative decoding for MIMO channels via modified sphere decoding," *Wireless Communications, IEEE Transactions on*, vol. 3, no. 6, pp. 2299–2311, Nov. 2004.

[5] S. Müller, M. Schreger, M. Kabutz, M. Alles, F. Kienle, and N. Wehn, "A novel LDPC decoder for DVB-S2 IP," in *Proc. DATE '09. Design, Automation. Test in Europe Conference. Exhibition*, Apr. 20–24, 2009, pp. 1308–1313.

[6] M. Alles, F. Berens, and N. Wehn, "A Synthesizable IP Core for WiMedia 1.5 UWB LDPC Code Decoding," in *Proc. IEEE International Conference on Ultra-Wideband (ICUWB 2009)*, Vancouver, Canada, September 2009.

[7] P. Urard, E. Yeo, L. Paumier, P. Georgelin, T. Michel, V. Lebars, E. Lantreibecq, and B. Gupta, "A 135Mb/s DVB-S2 compliant codec based on 64800b LDPC and BCH codes," in *Proc. Digest of Technical Papers Solid-State Circuits Conference ISSCC. 2005 IEEE International*, 10–10 Feb. 2005, pp. 446–609.

[8] J. Hagenauer and C. Kuhn, "The List-Sequential (LISS) Algorithm and Its Application," *IEEE Transactions on Communications*, vol. 55, no. 5, pp. 918–928, May 2007.

[9] H. Vikalo and B. Hassibi, "On joint detection and decoding of linear block codes on Gaussian vector channels," vol. 54, no. 9, pp. 3330–3342, Sept. 2006.

[10] C. Studer, A. Burg, and H. Bolcskei, "Soft-output sphere decoding: algorithms and VLSI implementation," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 2, pp. 290–300, February 2008.

[11] D. Garrett, L. Davis, S. ten Brink, B. Hochwald, and G. Knagge, "Silicon complexity for maximum likelihood MIMO detection using spherical decoding," *Solid-State Circuits, IEEE Journal of*, vol. 39, no. 9, pp. 1544–1552, Sept. 2004.

[12] B. Mennenga, R. Fritzsche, and G. Fettweis, "Iterative Soft-In Soft-Out Sphere Detection for MIMO Systems," in *Proc. IEEE 69th Vehicular Technology Conference VTC Spring 2009*, 26–29 April 2009, pp. 1–5.

[13] F. Kienle, "Low-Density MIMO Codes," in *Proc. 5th International Symposium on Turbo Codes and Related Topics*, Lausanne, Switzerland, Sep. 2008, pp. 107–112.

[14] J. Boutros, O. Pothier, and G. Zemor, "Generalized low density (Tanner) codes," in *Proc. IEEE International Conference on Communications ICC '99*, vol. 1, 6–10 June 1999, pp. 441–445.

[15] G. Liva, W. E. Ryan, and M. Chiani, "Quasi-cyclic generalized ldpc codes with low error floors," *IEEE Transactions on Communications*, vol. 56, no. 1, pp. 49–57, January 2008.

[16] F. Kienle, "On Low-Density MIMO Codes," in *Proc. IEEE International Conference on Communications ICC '09*, 14–18 June 2009, pp. 1–6.