

## Chapter 8

# Bit-Interleaved Coded MIMO System

In Chap. 2 we introduced the so called bit-interleaved coded modulation systems. In BICM systems a channel code is followed by an interleaver stage and the modulation. So far, the transmission of information was assumed to be done via one pair of antennas (one on the sender side and one on the receiver side), which means one symbol was sent in each time slot. However, as introduced in Sect. 2.4 it is possible to transmit multiple symbols via multiple antennas, while on the receiver side it is also possible to receive information via multiple antennas. In this chapter we will enhance the outer transceiver of the BICM system towards so called multiple-input multiple-output (MIMO) antenna systems. The resulting BICM-MIMO system is part of the outer transceiver, as a clear separation of the frequency modulation part is possible.

In a MIMO system a symbol stream is demultiplexed to multiple transmit antennas while the receiver side collects superimposed samples, which are additionally disturbed by channel noise, from multiple receive antennas. There are two reasons to use MIMO antenna systems: increasing the data rate and/or increasing the reliability of the transmission. Many techniques exist to reduce the complexity of the MIMO demodulation process, e.g. by introducing constraints in space and/or time, by trading off diversity gain and multiplexing gain. MIMO demodulator or MIMO detector are used as synonyms in the following. Two 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, all of these techniques have to be concatenated with an additional channel code to ensure a desired quality of service. The overall data rate of the transmission—the number of bits transmitted per channel use—is determined by the space-time encoder and the channel encoder independently. The overall complexity depends rather on the individual modules than on the integration of the two.

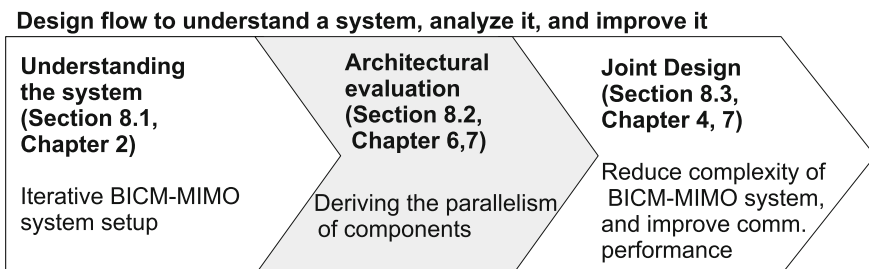
Transmission rates close to the theoretical capacity of a MIMO channel can be achieved by a simple encoder structure, by a serial concatenation of an outer code, interleaver, and modulator. In this case the modulator performs a spatial multiplexing of the symbol stream without introducing any further constraints and thus data rate

loss. This simple concatenation can be seen as a classical bit-interleaved coded-modulation scheme (BICM).

Approaching the MIMO capacity limit can be achieved by an iterative receiver, where probabilistic (soft) information are exchanged between MIMO detector and channel decoder [3, 4]. However, the demodulator has to calculate maximum a posteriori probabilities (APP) for each bit which can be computationally demanding. The complexity of the MIMO-APP demodulator depends on the number of transmit antennas and the size of the modulation alphabet.

Typically, the MIMO detector and the channel decoder are designed independently, while the overall complexity mainly depends on each individual part. There exist many different possibilities for the realization of the required soft-in soft-out MIMO detector. This chapter deals only with the one which can provide the best communications performance. This MIMO detector is based on the so called sphere detection algorithm. This chapter puts focus on a design flow to improve a system in terms of architectural efficiency and communications performance. The three steps are: understanding the system, deriving architectural constraints, and improve the system. The last step requires know-how from the algorithmic domain and the hardware domain to improve the system. The three steps are sketched in Fig. 8.1. Understanding all steps requires the knowledge of the previous chapters.

- First the system set up is explained and the state-of-the art performance of iterative BICM-MIMO systems is introduced. The basic MIMO detection algorithms are explained, but only the sphere-based algorithm is considered further. The presented soft-in soft-out sphere detector can calculate the optimal symbol-by-symbol MAP criterion.
- The second part deals with an architectural evaluation of the BICM-MIMO system. We show how to derive the necessary parallelism for a channel decoder with or without feedback loop. A high level exploration like the presented is always required before realizing an architecture. This is a classical top down approach at which a designer evaluates the parallelism of the data flow. Based on this the number of instances for the individual components can be derived.



**Fig. 8.1** Design steps for system improvement. Parts of this chapter and the related topics which are required for a joint design of algorithm and architecture with respect to BICM-MIMO systems

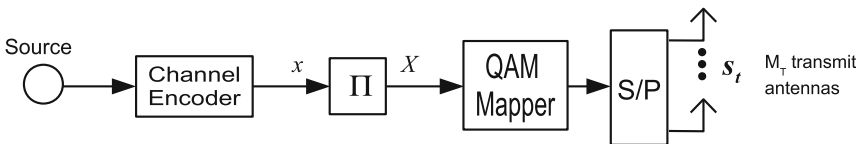
- Implementing a BICM-MIMO system in a top down approach results in independent implementations of the channel decoder and the MIMO detector. The overall complexity will be lower bounded by the VLSI footprints of the individual components. The basic philosophy of joint MIMO detector and channel code design, which is shown in the third part of the chapter, requires the knowledge of the algorithm and a basic understanding of architectural design. The goal is to reduce the complexity and to increase the communications performance at the same time. This can only be achieved when architectural know-how is taken into consideration in the early phases of system design.

## 8.1 State-of-the-Art BICM-MIMO Systems

In this section we will revise state-of-the-art BICM-MIMO systems. We assume for all MIMO system that they have a symmetrical number of antenna setup, i.e.  $M_T = M_R$ . The MIMO encoding and decoding processes are explained in the following paragraphs, furthermore the achievable communications performance is presented.

### 8.1.1 MIMO Transmitter

The entire encoding procedure is a bit-interleaved coded modulation (BICM) scheme and is shown in Fig. 8.2. The source bits are encoded by an outer channel code of code rate  $R$ . The resulting codeword is interleaved and then mapped to symbols. The symbols are then multiplexed to the different antennas and  $M_T$  symbols are transmitted simultaneously at each time step. In the following we will explain the notation, which differs slightly from the notation used in previous chapters. Instead of scalars each time slot now holds a vector of transmitted data. The source generates a random information word  $\mathbf{u}$  of length  $K$  which is encoded by the channel encoder. The resulting codeword  $\mathbf{x}$  consists of  $N$  bits which are grouped into  $N_s$  subblocks  $\mathbf{x}_n$ . In the following we combine the interleaver stage and this grouping in one stage with the resulting codeword matrix  $\mathbf{X}$



**Fig. 8.2** Typical MIMO transmitter with the encoded codeword  $\mathbf{x}$  and the transmission vector  $\mathbf{s}_t$ . The serial to parallel de-multiplexing stage is denoted as  $S/P$

$$\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n, \dots, \mathbf{x}_{N_s}). \quad (8.1)$$

Each subblock consists of  $Q$  coded bits ( $Q$  being the modulation size).

$$\mathbf{x}_n = (x_{1,n}, x_{2,n}, \dots, x_{q,n}, \dots, x_{Q,n}) \quad (8.2)$$

Each subblock  $\mathbf{x}_n$  is mapped to one complex symbol  $s$  chosen from a  $2^Q$ -ary QAM modulation scheme (Gray mapping). The advantage of this matrix notation is that it combines the interleaver and the allocation of the bit positions to symbol positions. At any given time  $M_T$  consecutive symbols are combined in one transmitted vector  $\mathbf{s}_t$ .

$$\mathbf{s}_t = (s_{1,t}, s_{2,t}, \dots, s_{M_T,t}) \quad (8.3)$$

The whole modulated sequence is represented by

$$\mathbf{S} = (\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_t, \dots, \mathbf{s}_T) \quad (8.4)$$

$T$  time slots are needed to transmit all symbols of one codeword. The transmission of one transmission vector  $\mathbf{s}_t$  in time step  $t$  is modeled by multiplying it with the channel matrix  $\mathbf{H}_t$  and adding Gaussian noise  $\mathbf{n}_t$ :

$$\mathbf{y}_t = \mathbf{H}_t \cdot \mathbf{s}_t + \mathbf{n}_t \quad (8.5)$$

The channel modeling and the difference between a quasi-static channel and an ergodic channel was already introduced in Sect. 2.4. For all presented communications performance curves in this chapter the type of channel model is stated explicitly.

The overall data rate of the presented transmission is  $\eta = RM_T Q$  which reflects the number of information bits per time slot. Often, the used channel codes in BICM-MIMO system are either convolutional codes, or turbo codes, or LDPC codes respectively.

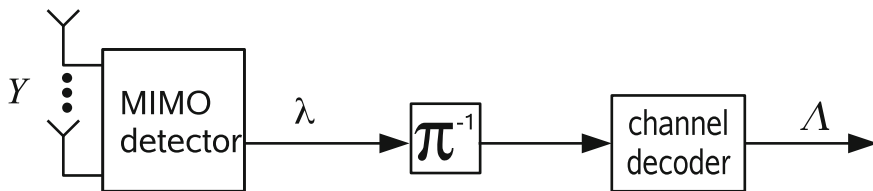
### 8.1.2 BICM-MIMO Receiver

We have to distinguish between BICM-MIMO receivers with an open loop structure and a closed loop structure. The different receiver types are shown in Figs. 8.3 and 8.4. We denote the information received via the  $M_R$  received antennas as a matrix  $\mathbf{Y}$ ,

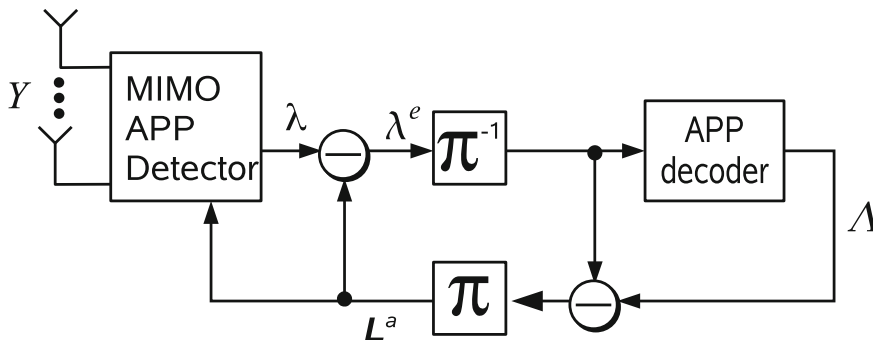
$$\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_t, \dots, \mathbf{y}_T) \quad (8.6)$$

with  $\mathbf{y}_t$  being the received samples in time slot  $t$ ,

$$\mathbf{y}_t = (y_{1,t}, y_{2,t}, \dots, y_{M_R,t}) \quad (8.7)$$



**Fig. 8.3** TBICM-MIMO receiver with **open loop** structure. The MIMO detector transforms the received information  $Y$  into LLRs ( $\lambda$ ) for each bit position. The interleaved MIMO detector output is passed to the channel decoder



**Fig. 8.4** BICM-MIMO receiver with **closed loop** structure, with the a priori information  $L^a$  and the extrinsic information  $\lambda^e$  passed to the outer decoder

As already stated throughout this manuscript it is always assumed that the channel ( $H_t$ ) is perfectly known by the receiver.

Assuming the open loop case of Fig. 8.3 we can calculate the detector output information using different criteria. The corresponding MIMO detectors are outlined shortly in the following. Note, that only systems with spatial multiplexing are assumed.

- Zero forcing detector: The received vector is multiplied by  $H^\dagger$  the pseudo-inverse of the channel matrix

$$\hat{z}^{ZF} = H^\dagger y_t = (H^H H)^{-1} H^H y_t. \tag{8.8}$$

The major problem of this approach is the amplification of the noise which results in a large degradation in communications performance. The zero forcing solution  $\hat{z}^{ZF}$  has a maximum diversity order of  $M_R - M_T + 1$  [5]. The hard decision symbols are obtained by quantizing the result to the closest constellation points.

- MMSE detector: The minimum mean square estimator calculates a filter matrix  $W$  which minimizes the following condition

$$W^{MMSE} = \arg \min_w \left\{ E\{ \|W^H y_t - s_t\|^2 \} \right\} \tag{8.9}$$

The resulting filter output  $\hat{z}^{MMSE}$  evaluates to

$$\hat{z}^{MMSE} = \mathbf{W}^{MMSE} \mathbf{y}_t = \left( \mathbf{H}^H \mathbf{H} + \frac{M_T}{SNR} \mathbf{I} \right)^{-1} \mathbf{H}^H \mathbf{y}_t, \quad (8.10)$$

with  $\mathbf{I}$  representing a diagonal matrix which is weighted by the corresponding noise. For large SNR values the MMSE solution approximates the ZF solution, thus a diversity order of  $M_R - M_T + 1$  is obtained [5].

- **ML detector:** ZF and MMSE are so called linear detectors while the ML detector is not. The ML detector calculates the maximum likelihood symbol estimation  $\hat{s}^{ML}$  which is defined as:

$$\hat{s}^{ML} = \arg \min_s \left\{ \|\mathbf{y}_t - \mathbf{H}_t s\|^2 \right\} \quad (8.11)$$

The ML detector has a diversity order of  $M_R$  [5] and provides hard-output values. Though correct, we can improve further the BICM system performance by calculating soft-output values.

- **APP detector:** Soft-output values can be obtained by applying an a posteriori probability (APP) criterion. The results is denoted as MIMO-APP to distinguish it from the APP detectors for single antenna systems introduced already in Sect. 2.2. The major difference for MIMO-APP detection is the conditional probability on a received vector  $\mathbf{y}_t$  which comprises the information of  $M_R$  symbols. The LLR value on each individual bit can be calculated by

$$\lambda(x_{t,q,m}) = \ln \frac{P(x_{t,q,m} = 0 | \mathbf{y}_t)}{P(x_{t,q,m} = 1 | \mathbf{y}_t)} \quad (8.12)$$

Since the channel decoder has its maximum achievable coding gain with APP information at its input we only concentrate on detectors which can provide APP information.

As mentioned, for the MIMO detection and channel decoding we have to distinguish between open loop and closed loop receiver structures.

- **Open loop:** Figure 8.3 shows a receiver structure with a demodulator concatenated with an outer channel decoder. The APP information of the MIMO detector is interleaved and directly passed to the channel code.
- **Closed loop:** Figure 8.4 shows a structure in which the demodulator and the channel decoder pass information back and forth. During the iterative message exchange between detector and outer decoder the input messages we have to ensure the extrinsic information principle, similar as done for turbo or LDPC decoding.  $\mathbf{L}^a$  is the a priori information which is passed to the MIMO-APP detector.  $\lambda^e = \lambda + \mathbf{L}^e$  comprises the information  $\lambda$  extracted from the received information and the additional gain  $\mathbf{L}^e$  obtained due to a priori input information. During the first demodulation there exist no a priori information, thus  $\mathbf{L}^a = 0$ .

Closed loop BICM-MIMO receivers can gain more than 3db in communications performance compared to open loop receivers [3]. The final gain depends on many system parameters like the used antenna system, modulation type, number of iterations, channel model and the channel code. Communications performance results are shown in Sect. 8.1.3.

In the following only the MIMO-APP demodulator capable for iterative processing is further considered. A MIMO-APP detector computes logarithmic likelihood values (LLRs) on each bit according to

$$\lambda(x_{q,m}) = \ln \frac{P(x_{q,m} = +1|\mathbf{y})}{P(x_{q,m} = -1|\mathbf{y})} \quad (8.13)$$

We have to evaluate this equation for each transmission time slot  $t$ , however, the index for the time slot  $t$  is skipped from now on.  $q$  is the index within a modulated symbol with  $q \in 1, \dots, Q$  and  $m$  the index with respect to the antenna layer with  $m \in 1, \dots, M$ . For independent  $x_{q,m}$ , the probability  $P(x_{q,m} = 0|\mathbf{y})$  is obtained by summing up the probabilities of all possible symbol vectors  $s$  which contain  $x_{q,m} = 0$ .

$$P(x_{q,m} = 0|\mathbf{y}) = \sum_{\forall s|x_{q,m}=0} P(s|\mathbf{y}) \quad (8.14)$$

$s|x_{q,m} = 0$  determines the symbol vector conditioned that the corresponding bit position is 0. This calculation is related to the demodulator example of Eq.D.9.

Using Bayes theorem,  $P(s|\mathbf{y})$  can be expressed as

$$P(s|\mathbf{y}) = \frac{P(s) \cdot P(\mathbf{y}|s)}{P(\mathbf{y})} \quad (8.15)$$

We can observe that the analyzed probability consists of three parts.  $P(s)$  takes into account that not every  $s$  is equally likely given the a-priori information  $L^a$  from the channel decoder. As the codeword is interleaved before the QAM mapping the bits  $x_{q,m}$  are assumed to be independent from each other. Therefore,  $P(s)$  is the product of the probabilities of the individual bits that were mapped into  $s$ :

$$P(s) = \prod_{\forall q,m} P(x_{q,m}) \quad (8.16)$$

The term  $P(\mathbf{y}|s)$  is the probability of receiving  $\mathbf{y}$  under the condition that the vector  $s$  was sent.  $P(\mathbf{y}|s)$  can be calculated via the corresponding Gaussian function, as an additive noise is assumed. The third part  $P(\mathbf{y})$  is constant during the detection of  $\mathbf{y}$  and is canceled out when applying (8.15) to calculate the LLRs of (8.13). Finally for the soft-input soft-output processing we have to evaluate

$$\lambda(x_{q,m}) = \ln \frac{\sum_{\forall \mathbf{s} | x_{q,m}=0} P(\mathbf{s}) \cdot e^{-\|\mathbf{y}-\mathbf{H}\mathbf{s}\|^2/N_0}}{\sum_{\forall \mathbf{s} | x_{q,m}=1} P(\mathbf{s}) \cdot e^{-\|\mathbf{y}-\mathbf{H}\mathbf{s}\|^2/N_0}} \quad (8.17)$$

Applying the Jacobian logarithm and ignoring the correction term results in the Max-Log-Map approximation. The detailed discussion can be found in [3, 6].

$$\lambda(x_{q,m}) \approx \min_{\forall \mathbf{s} | x_{q,m}=0} \left\{ \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 - N_0 \sum_{\forall q',m'} \ln P(x_{q',m'}) \right\} \\ - \min_{\forall \mathbf{s} | x_{q,m}=1} \left\{ \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 - N_0 \sum_{\forall q',m'} \ln P(x_{q',m'}) \right\} \quad (8.18)$$

An interpretation for (8.18) is that we derive the LLR value  $\lambda(x_{q,m})$  from the most likely symbol vectors  $\mathbf{s}$  with one bit  $x_{q,m}$  being 0 or 1 respectively. The expression  $N_0 \sum_{\forall q',m'} \ln P(x_{q',m'})$  determines the a priori information under the constraint of  $\forall \mathbf{s} | x_{q,m} = \pm 1$ .

The metric  $d(\mathbf{s})$  measures the likelihood that a specific vector  $\mathbf{s}$  has been sent:

$$d(\mathbf{s}) = \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 - N_0 \sum_{\forall q',m'} \ln P(x_{q',m'}). \quad (8.19)$$

Small metrics  $d(\mathbf{s})$  relate to a high probability of  $\mathbf{s}$  having been sent.

Calculating all possible  $d(\mathbf{s})$  to determine Eq. 8.18 quickly grows infeasible for higher antenna constellations and/or higher order modulations as the complexity grows with  $2^{QM}$ . Therefore, many sub-optimal algorithms with lower were devised. Most of them are based on a tree search. In order to map the metric calculations Eq. 8.19 on a tree, the channel matrix  $\mathbf{H}$  is decomposed into an unitary matrix  $\mathbf{Q}$  and an upper-triangular matrix  $\mathbf{R}$ . The Euclidean distance is rewritten as

$$\|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 = \|\mathbf{y}' - \mathbf{R}\mathbf{s}\|^2 \quad (8.20)$$

with  $\mathbf{y}' = \mathbf{Q}^H \mathbf{y}$ . Equation 8.19 is replaced by the equivalent metric

$$d(\mathbf{s}) = \|\mathbf{y}' - \mathbf{R}\mathbf{s}\|^2 - N_0 \sum_{\forall q',m'} \ln P(x_{q',m'}) \quad (8.21)$$

The triangular structure of  $\mathbf{R}$  allows the recursive calculation of  $d(\mathbf{s})$  which can be seen when we fully extend the term for the Euclidean distance in the equation:



$$\begin{aligned}
\|y' - \mathbf{R}s\|^2 &= \left\| \begin{pmatrix} y'_1 \\ y'_2 \\ \vdots \\ y'_M \end{pmatrix} - \begin{pmatrix} r_{1,1} & 0 & \cdots & 0 \\ r_{2,1} & r_{2,2} & & \vdots \\ \vdots & \vdots & \ddots & 0 \\ r_{M,1} & r_{M,2} & \cdots & r_{M,M} \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_M \end{pmatrix} \right\|^2 \\
&= \sum_{m=1}^M \left| y'_m - \sum_{j=1}^m r_{m,j} s_j \right|^2. \tag{8.22}
\end{aligned}$$

Using the partial symbol vector  $s^{(m)} = (s_1, s_2, \dots, s_m)$  the recursive calculation for each antenna layer  $m$  can be written as

$$d_m = d_{m-1} + \gamma_m(s^{(m)}). \tag{8.23}$$

$d_0 = 0$  is used for initialization. Including a priori information the partial distance metric of an antenna layer  $\gamma_m(s^{(m)})$  evaluates to:

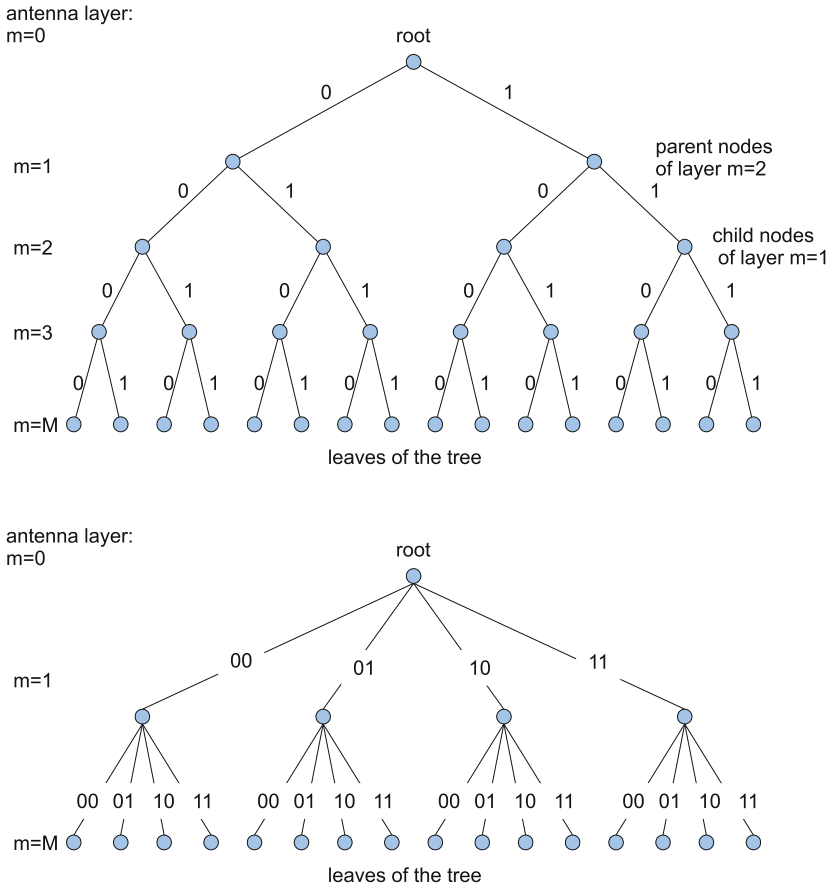
$$\gamma_m(s^{(m)}) = \left| y'_m - \sum_{j=1}^m r_{m,j} s_j \right|^2 - N_0 \sum_{q=1}^Q \ln P(x_{q,m}). \tag{8.24}$$

The recursive calculation can be represented by a tree with  $M + 1$  layers as shown in Fig. 8.5 for two different cases. The top figure represents a four antenna system with BPSK modulation (one bit per symbol), while the lower tree represents two antennas with two bits per symbol.

The root node corresponds to  $d_0$  and each leaf node corresponds to the metric  $d_M = d(s)$  of one possible vector  $s$ . Each layer corresponds to the detection of one symbol  $s_m$ . Branches are labeled with the corresponding bit pattern of the symbol. Each branch in the tree can be associated with a certain weight  $\gamma_m(s^{(m)})$  which depends on the path from the root node to the corresponding edge. Thus, when advancing from a parent node to a child node, the metric of the child node  $d_m$  is calculated from the metric of the parent node  $d_{m+1}$  and the branch metric  $\gamma_m(s^{(m)})$ , see Eq. 8.23.

Evaluating all possibilities of  $d(s)$  results in  $P = 2^{MQ}$  possibilities. E.g. for a  $4 \times 4$  antenna system with a 16-QAM modulation this results in  $P = 2^{MQ} = 65536$  possibilities reflecting all bit possibilities which are decoded within one transmission vector  $s_t$ .

Based on this tree search, many different MIMO detection algorithms exist. The main differences between the algorithms can be described by how the tree is traversed, e.g. breadth-first or depth-first, and how branches of the tree are pruned. In general, those algorithms achieve different results in terms of communications performance and implementation complexities.



**Fig. 8.5** Decision tree for MIMO detection, upper  $M=4$  antennas and one bit per modulated symbol, e.g. BPSK. Lower figure with 2 antennas and a two bit modulation, e.g. 4-AM

### 8.1.3 Communications Performance of State-of-the-Art Systems

All following communications performance results assume a BICM system with rate  $R = \frac{1}{2}$  LDPC code as channel coding scheme. We distinguish between the **closed loop** system with feedback between channel decoder and MIMO demapper and the **open loop** system without feedback. Figure 8.6 shows the communications performance for a 16-QAM  $4 \times 4$  system (8 bits/channel use). Each  $E_b/N_0$  point is simulated with 100k transmitted codewords of length  $N = 1920$ . Here we assume a quasi-static channel, i.e., the channel matrix  $\mathbf{H}_t$  remains constant within one transmitted codeword (120 channel uses). Four different performance curves are shown. The left most curve shows the outage capacity which is the theoretical lower bound for reliable transmission for this system. The curves are now explained starting

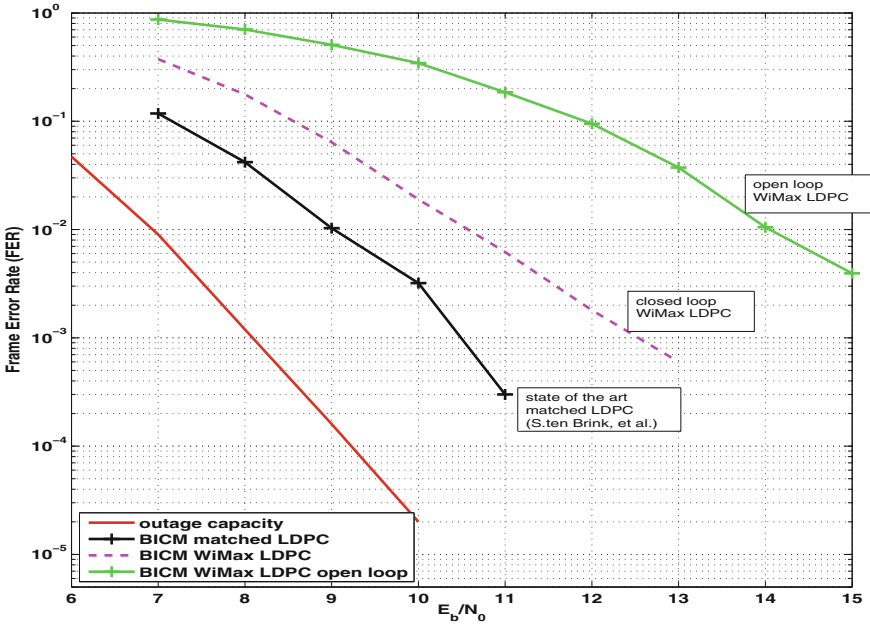
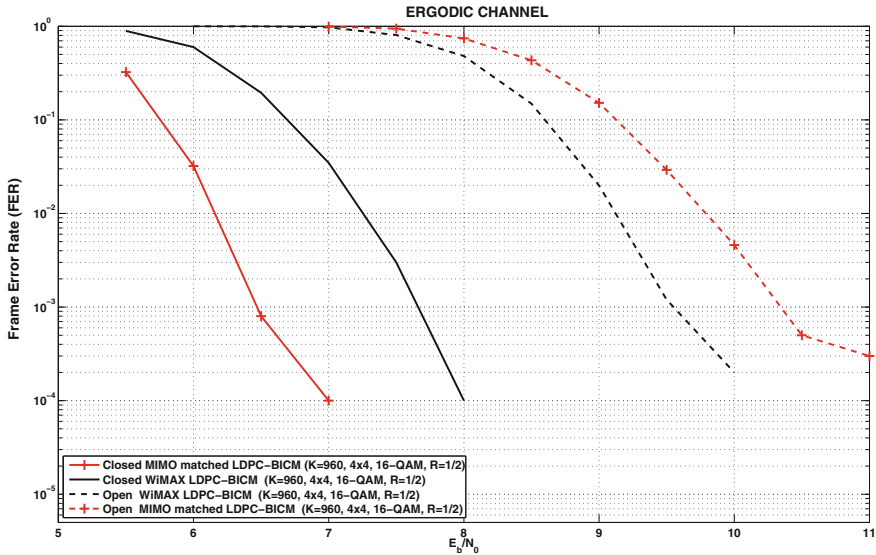


Fig. 8.6 State-of-the-art communications performance for quasi-static MIMO channel

with the one with worst performance. The right curve is achieved by performing the MIMO-APP demodulation and 40 iterations of an LDPC code, thus building an open loop system. A WiMAX type LDPC code is used with a degree distribution of  $f_{[6,3,2]} = \left\{ \frac{5}{24}, \frac{1}{3}, \frac{11}{24} \right\}$ ,  $g_{[7,6]} = \left\{ \frac{1}{3}, \frac{2}{3} \right\}$ . The next performance improvement step is to close the loop and do 5 outer loop iterations by evaluating Eq. 8.18. Iterations within an LDPC decoder are denoted as inner iterations, the feedback via MIMO-APP detector is denoted as outer iterations. For LDPC codes we can adjust the communications performance by its degree distribution as seen in Chap. 7. An LDPC code can be adjusted for iterative feedback loops by utilizing EXIT chart techniques, according to [7]. The new LDPC code is thinned out to obtain a different convergence performance. The resulting degree distribution here is  $f_{[6,3,2]} = \left\{ \frac{1}{8}, \frac{2}{8}, \frac{5}{8} \right\}$ ,  $g_{[6,5]} = \left\{ \frac{1}{2}, \frac{1}{2} \right\}$ . All LDPC codes presented here are quasi-cyclic to facilitate their implementation in hardware, as explained in Sect. 7.3.1. Especially for the matched LDPC design many variable nodes of degree two are present which gives a higher error floor. This degree distribution is a good trade-off between good convergence and a reasonable error floor. The performance gain of a closed loop system compared to an open loop system can be above 4 dB as seen in Fig. 8.6. The BICM system with matched LDPC codes is denoted as matched BICM in the following, the BICM with WiMAX LDPC code is denoted as WiMAX BICM, respectively.



**Fig. 8.7** Communications performance of open loop systems and a closed loop systems using two LDPC codes. One system utilizes an LDPC code which is matched with respect to the  $4 \times 4$  MIMO system, the other system uses the standard WiMAX code

Figure 8.7 shows the communications performance with the same set up ( $4 \times 4$  antennas, 16-QAM), but for an ergodic channel, i.e.,  $\mathbf{H}_t$  changes for each time slot. The graph shows the results for the open loop system and the closed loop system for the matched BICM and WiMAX BICM system. For the closed loop performance, using five outer iterations, we can see a 1 dB performance gain of the matched BICM system at  $FER = 10^{-2}$ . This gain between matched BICM and WiMAX BICM system is identical to that obtained under quasi-static channel conditions. However, for an open loop performance the WiMAX BICM system outperforms the matched BICM system. In summary two important aspects are highlighted:

- The gain in communications performance for different BICM-MIMO systems depends on the number of outer iterations. Achieving always the best communications performance within all outer iterations can not be achieved by a single code.
- Second important aspect which should be further discussed is the realistic number of outer loop iterations which can be performed in hardware designs. The important analysis of outer loop iterations for a BICM receiver architecture is done in the next section.

### 8.2 Architecture Feasibility BICM-MIMO

In this section we analyze complexity aspects for the BICM receiver system shown in Fig. 8.4. Again we distinguish between the closed loop system with feedback between channel decoder and 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.

For the realization of the closed loop system several possibilities for the architecture exist. Assuming a fixed number of outer iteration, the iterations can be unrolled and pipelined. The corresponding architecture is shown in Fig. 8.8 for three iterations. Now three blocks are processed simultaneously in this pipeline which implies the instantiation of three hardware instances of the APP demodulator and the APP channel decoder respectively. Unrolling the loop will result in a linear increase in terms of area, since all memories and the logic will be duplicated. The architecture is inflexible with respect to number of performed closed loop iterations.

In the following we assume that one MIMO demodulator instance and one channel decoder instance are used which operate on one coded block. This is shown in Fig. 8.4. Only one codeword of the channel code 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 relates to the typical iterative turbo code processing where information is exchanged between two MAP components, see Chap. 6. Here, in Fig. 8.4, the two components are the demodulator and outer channel decoder which are separated by an interleaver. We could also 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 problem which is not in the scope of this analysis.

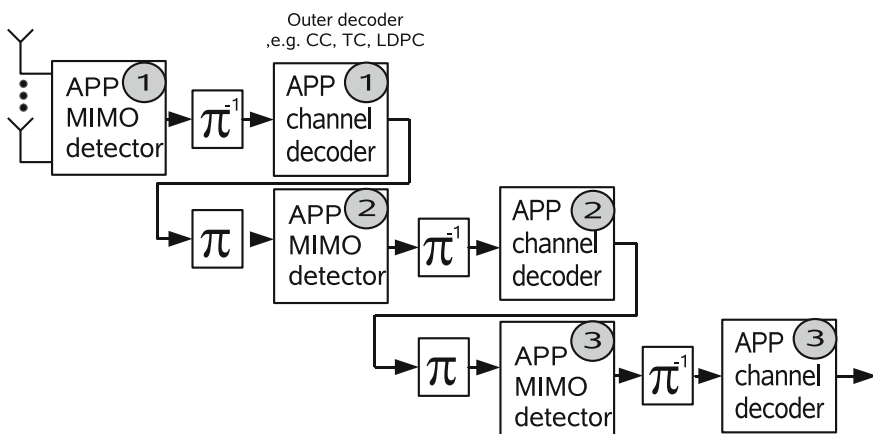


Fig. 8.8 Unrolled architecture for three outer iterations

A pragmatic solution for processing two blocks simultaneously is the instantiation of two independent closed loop receivers. An appropriate allocation of the codeword to be processed has thus to be done at a higher architectural level.

For the following discussions we assume one instantiated MIMO detector and one channel decoder where only one codeword is processed. We consider the throughput constraints for the outer channel decoder utilizing turbo decoders and LDPC decoders, respectively. The parameters to derive the throughput constraints are shown in following:

$\#cycles$	number of cycles required to process one block
$P_{I/O}$	parallelization of the input/output
$iter$	number of iterations of the channel decoder (half iterations for turbo decoding)
$P$	parallelization of the decoder architecture: for turbo codes parallelization of the MAP architecture, for 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 architectural overhead (e.g. for flushing the pipeline)
$\# \frac{bits}{cycle}$	normalized throughput: number of information bits decoded per clock cycle

The expected normalized throughput for a given architecture is

$$\# \frac{bits}{cycle} = \frac{K}{\#cycles} = \frac{N \cdot R}{\#cycles}. \quad (8.25)$$

The normalized throughput is a good performance metric of an architecture. For example, LTE advanced will require a turbo decoder architecture with  $\# \frac{bits}{cycle} \sim 2$ . For a typical frequency of  $f_{cyc} = 300$  MHz this yields a payload (information bit) throughput of  $T_{payload} = \# \frac{bits}{cycle} \cdot f_{cyc} = 600$  Mbit/s.

### Low-Density Parity-Check Decoder

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

$$\# \frac{bits}{cycle} \approx \frac{N \cdot R}{iter \cdot \frac{N \cdot \overline{d_{VN}}}{P}} = \frac{P \cdot R}{iter \cdot \overline{d_{VN}}} \quad (8.26)$$

Most of the current partly parallel architecture use the layered architecture where a nearly continuous processing takes place. No overhead cycles ( $\delta_{overhead}$ ) are present within the iterative loop, for more details see Chap. 7 and [8, 9]. An average variable node degree of  $\overline{d_{VN}} = 3.2$  (WiMAX LDPC) is assumed to derive the parallelism of a decoder architecture.

$$P = \left( \# \frac{bits}{cycle} \right) \cdot iter \cdot 3.2 \cdot \frac{1}{R} \quad (8.27)$$

### Turbo Code Decoder

For turbo code decoders we define the parallelism  $P$  of the architecture as the number of LLRs which are exchanged per clock cycle between the component decoders. For turbo decoding a normalized throughput can be expressed as

$$\# \frac{bits}{cycle} \approx \frac{P}{2 \cdot iter \cdot \left( 1 + \delta_{overhead} \cdot \frac{P}{K} \right)} \quad (8.28)$$

Turbo decoding needs two half iterations to process the two component codes, thus we need the term  $2 \cdot iter$ . The overhead  $\delta_{overhead}$  is a big obstacle to a further parallelization of turbo decoders. Reducing them is a research topic that is receiving a lot of attention. The LTE Release 8 standard supports very high code rates ( $R > 0.9$ ) which hampers the reduction of these  $\delta_{overhead}$  cycles. The problem for high throughput turbo decoder architectures is the limited throughput increase for moderate length  $K \sim 5000$  and increasing architecture parallelism  $P$ . In this case the term  $\delta_{overhead} \cdot \frac{P}{K}$  is significant. For the following calculations the turbo decoder parallelization is calculated with an overhead of  $\delta_{overhead} = 32$  and evaluates to:

$$P = \frac{\left( \# \frac{bits}{cycle} \right) \cdot 2 \cdot iter}{1 - \frac{\left( \# \frac{bits}{cycle} \right) \cdot 2 \cdot iter \cdot 32}{K}} \quad (8.29)$$

### Channel Coding Architecture in BICM-MIMO Systems

Table 8.1 shows the required parallelism for turbo decoder architectures and LDPC decoder architectures for an open loop system. A normalized throughput of  $\# \frac{bits}{cycle} = 1$  and  $\# \frac{bits}{cycle} = 2$  is assumed.

The required architecture parallelism depends on the number of iterations. For example, the presented turbo decoder of [10] has a throughput of 150Mbit/s at  $f_{cyc} = 300$ MHz. This turbo decoder uses an architecture of  $P = 8$  and results in a normalized throughput of  $\# \frac{bits}{cycle} = 0.5$  at 6.5 iterations. State-of-the-art turbo decoder architectures are already targeting a parallelism up to  $P = 32$  [11]. However, the resulting chip size is very large and a further increase in parallelism is inefficient due to the overhead cycles. Thus, a further increase of the throughput off turbo decoders can best be achieved on block level, which means by instantiating multiple turbo decoder instances. For LDPC decoders a parallelism of  $P = 360$  was already presented in 2005 [12], a larger degree of parallelism is possible. In summary for the open loop case we can say that there seems to be no practical obstacle to increasing

**Table 8.1** Parallelization (P) of an **open loop** architecture for the given iterations and normalized throughput

Normalized throughput	Turbo codes $K = 6000$		LDPC codes $R = 1/2$	
	Iterations	P	Iterations	P
$\# \frac{\text{bits}}{\text{cycle}} = 1$	4 iter	9	5 iter	32
	6 iter	13	10 iter	64
	8 iter	18	20 iter	128
$\# \frac{\text{bits}}{\text{cycle}} = 2$	4 iter	18	5 iter	64
	6 iter	28	10 iter	128
	8 iter	39	20 iter	256

the throughput. This can always be done by multiple decoder instances, if the latency constraints can be fulfilled.

Now we analyze the required parallelism of turbo decoder or LDPC decoders used within an iterative BICM-MIMO receiver. Table 8.2 shows the parallelization of a turbo decoder or LDPC decoder for a given normalized throughput assuming a closed loop system. The normalized throughput is defined for the BICM-MIMO receiver while we have now a double iterative system with inner channel code iterations and outer feedback iterations. For example, the notation ‘2 outer–3 inner’ means that the demodulator and channel code is active two times for each block, while the channel code performs three channel code iterations for each of these two times.

The parallelism for, e.g., a  $\# \frac{\text{bits}}{\text{cycle}} = 1$  closed loop system with 2 outer–3 inner iterations translates to a 6 channel decoder iteration in the open loop system. However, since the decoder is assumed to be idle 50% the parallelism to achieve the desired system throughput. This results in large parallelism of  $P=26$  for the simplest case of Table 8.2. The number of outer iterations and inner channel code iterations are rather small in these examples. The parallelization even has to be increased in order to achieve the best possible communications performance. Note, that the normalized throughput assumption of  $\# \frac{\text{bits}}{\text{cycle}} = 1$  or even higher required by upcoming standards, e.g. LTE advanced.

**Table 8.2** Parallelization (P) of a **closed loop** architecture for the given outer and (inner) code iterations with respect to a normalized throughput

Normalized throughput	Turbo codes $K = 6000$		LDPC codes $R = 1/2$	
	Iterations	P	Iterations	P
$\# \frac{\text{bits}}{\text{cycle}} = 1$	2 outer–3 inner	26	2 outer–5 inner	128
	3 outer–3 inner	35	3 outer–5 inner	192
	4 outer–3 inner	52	4 outer–5 inner	256
$\# \frac{\text{bits}}{\text{cycle}} = 2$	2 outer–3 inner	55	2 outer–5 inner	256
	3 outer–3 inner	77	3 outer–5 inner	384
	4 outer–3 inner	110	4 outer–5 inner	512



As mentioned, for turbo decoder architectures we can increase the throughput further by creating multiple instances. However, for a closed loop system this requires to handle multiple blocks within an iterative BICM-MIMO system. In our opinion this is currently a strong argument against a double iterative scheme with targets of  $\# \frac{\text{bits}}{\text{cycle}} = 2$ , especially in the case of turbo codes. For LDPC codes achieving the required architecture parallelism seems to be easier.

The MIMO demodulator in the outer loop has to provide a normalized throughput of  $\# \frac{\text{bits}}{\text{cycle}} = 1$  or  $\# \frac{\text{bits}}{\text{cycle}} = 2$ . The advantage of the MIMO detector is that each received vector  $\mathbf{y}_i$  can be decoded independently. Thus, the high throughput requirements for the MIMO demodulation can always be achieved by multiple instances.

In summary: the double iterative structure poses a big challenge for the architectural realization. We can extract two options to limit the architectural overhead.

- Either we should get rid of the double iterative scheme,
- or we should ensure a very good communications performance with a limited number of outer iterations, e.g. just 2 outer iterations.

Reducing the number of closed loop iterations, while providing a good communications performance requires a joint consideration of the MIMO detector and channel code design. One possible joint design is presented in the next section.

### 8.3 Joint Architecture-Algorithm Design

Implementing an iterative BICM-MIMO system in a straight forward manner results in an independent implementation of the channel decoder and the MIMO detector. This straight forward approach was treated in the previous section. A lower bound for the overall area is given by the sum of the independent realizations. In fact, additional memories for the iterative data exchange are required [13].

The goal of the techniques presented in this section is to reduce the complexity of the MIMO-APP detection without sacrificing the overall data rate or the capacity approaching communications performance. The goal is to reduce complexity while increasing communications performance. This can only be achieved when architectural know-how is taken into consideration in early phases of the system design.

The basic idea of the joint design approach is to reduce the search possibilities of the MIMO-APP detection. This can be achieved by a special design of the bit interleaver (Sect. 8.3.1) or by a dedicated code design (Sect. 8.3.2), which in part was published in [14, 15]. All channel codes used for the examples here are LDPC codes, however, it is possible to use the presented idea as well for turbo codes and convolutional codes.

The LDPC codes used in this section are also described by a parity check matrix  $\mathbf{H}_c$  and fulfill  $\mathbf{H}_c \mathbf{x}^T = \mathbf{0}$ . Note that the parity check matrix is here denoted with subscript  $c$  to make it distinguishable from  $\mathbf{H}$ , the channel matrix. The parity check matrix  $\mathbf{H}_c$  has  $N_c$  columns and  $M_c$  rows and has to be of full rank. The parity check matrix can be described by two layers with

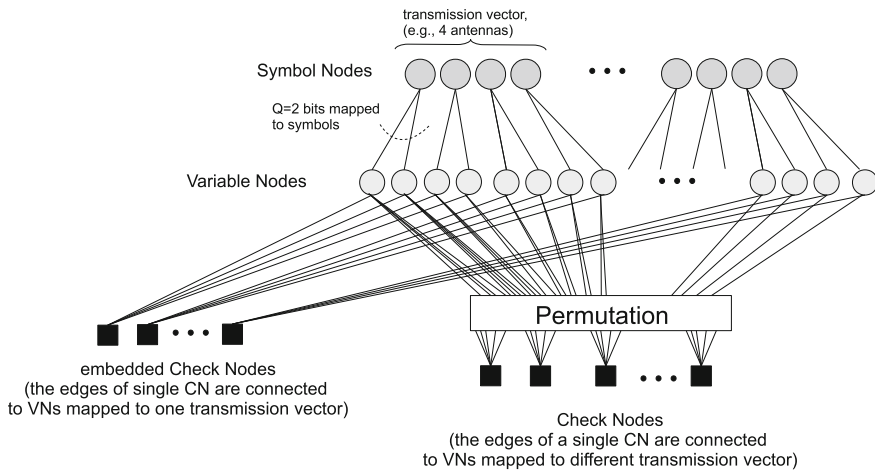
$$H_c = \begin{pmatrix} H_g \\ H_e \end{pmatrix} = \begin{pmatrix} H_g & & \\ H'_e & \cdots & 0 \\ 0 & \ddots & 0 \\ 0 & \cdots & H'_e \end{pmatrix} \tag{8.30}$$

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  $s_t$  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. Thus each transmission vector carries an embedded code  $H'_e$ . Embedded code or sub-code are used as synonyms in the following.

$H_g$  has the task to connect all embedded codes. The sparse layer  $H_g$  can be described by a degree distribution  $(f_g, g_g)$ . Where  $f_g$  represents the degree distribution of the variable nodes of the layer  $H_g$  and  $g_g$  defines the degree distribution of the check nodes respectively. The description of the second layer  $H_e$  can be done by defining one embedded code  $H'_e$ .

The graph structure of such an LDPC code is shown in Fig. 8.9. In this graph 4 symbol nodes of the transmission vector are connected to 8 variable nodes. These are linked to one embedded code  $H'_e$ , here, two check nodes. Each symbol node represents the information of one modulated symbol. Assuming a  $4 \times 4$  antenna system the received transmission vector  $y_t$  comprises the information of four symbols. The major advantage will be that the embedded constraints reduce the complexity of the



**Fig. 8.9** Generic graph structure for an LDPC code with symbol nodes connected to embedded codes. One symbol node represents the information of one modulated symbol

MIMO-APP detection while implicitly solving parts of the channel code. We will see that this will reduce the overall complexity while even a better overall communications performance can be achieved.

The most simple constraint on a sequence of bits is a single parity check constraint, which means  $H'_e$  results in a single parity check code. For MIMO-APP detection a new decision tree with one embedded check node results, which is shown in Fig. 8.10. Again two different decision trees are shown, both with one embedded single parity check node constraint. Figure 8.10 top represents four transmit antennas ( $M = 4$ ) and BPSK modulation ( $Q = 1$ ), the lower figure represents the decision tree for two transmit antennas and  $Q = 2$  bits per symbol.

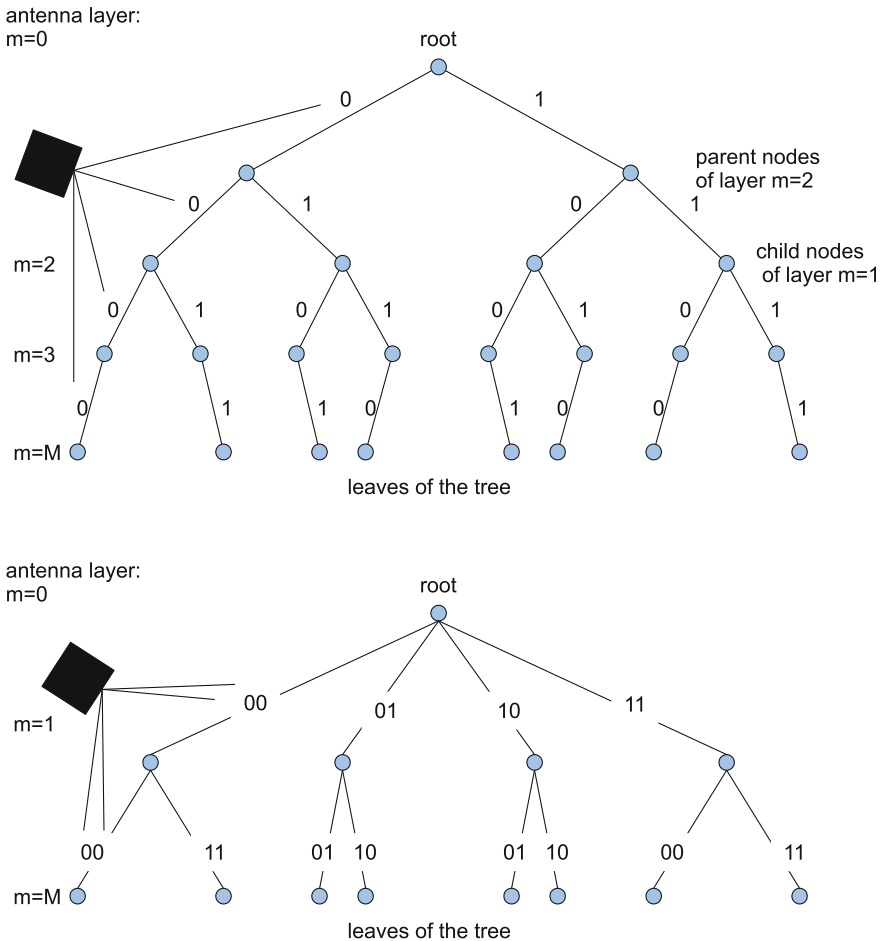


Fig. 8.10 Reduced decision tree with one embedded single parity check node

The check constraint (black square) in both cases is linked to all 4 bits which are simultaneously transmitted. This check eliminates paths in the decision tree, since the last bit has to fulfill the parity check equation. Thus the MIMO-APP demodulation Eq. 8.18 changes to

$$\lambda(x_{q,m}) \approx \min_{\forall s | \mathbf{c}_e, x_{q,m}=0} \left\{ \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 - N_0 \sum_{\forall q',m'} \ln P(x_{q',m'}) \right\} - \min_{\forall s | \mathbf{c}_e, x_{q,m}=1} \left\{ \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 - N_0 \sum_{\forall q',m'} \ln P(x_{q',m'}) \right\} \quad (8.31)$$

with the major difference of the term  $s | \mathbf{c}_e, x_{q,m} = 0$ , which means, the currently observed  $s$  is conditioned on bit  $x_{q,m}$  and  $\mathbf{c}_e$ . Each observed bit has to be an element of a valid codeword  $\mathbf{c}_e$ , while a valid codeword is defined via the embedded code constraint  $\mathbf{H}'_e \mathbf{c}_e^T = 0$ . Thus, we reduce the search space of the MIMO-APP demodulation while implicitly solving the second layer of  $\mathbf{H}_c$  during the MIMO-APP demodulation. If we embed one single parity check equation, the overall possibilities for demodulation downscales to  $P = 2^{MQ-1}$ . With  $q$  parity checks embedded within one transmission vector the number of possibilities for MIMO demodulation is reduced to  $P = 2^{MQ} / 2^q = P = 2^{MQ-q}$ .

It is important to distinguish the complexity reduction of the presented codes and the complexity reduction caused by algorithmic techniques. All algorithmic transformations, which are published for tree based MIMO-APP decoding, can be applied for the presented approach as well. In the following we will present two examples of how to enable the embedding of code constraints.

- Example one utilizes a standard WiMAX LDPC code (Sect. 8.3.1). By defining a well chosen bit interleaver we can embed parts of the defined parity checks within the transmission vector. The resulting BICM-MIMO will show a better communications performance while reducing the complexity of sphere decoding.
- Example two describes the design of quasi-cyclic LDPC codes which can be decoded by a standard compliant LDPC code decoder. Furthermore, the presented LDPC codes can largely decrease the search space of a sphere detector (Sect. 8.3.2).

### 8.3.1 Sphere Decoder Aware Bit Interleaver Design

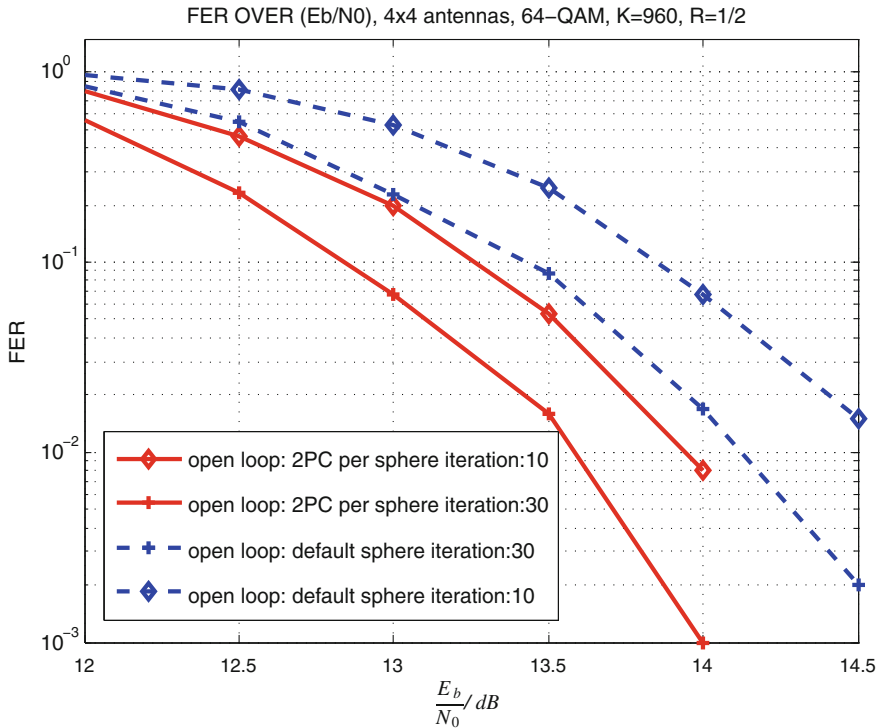
The joint design approach as presented in the previous section enables us to design an elaborated interleaver for the BICM-MIMO system which will decrease the complexity of the sphere detection while improving the communications performance.

This approach works for all LDPC codes which are quasi-cyclic. The resulting interleaver will be a quasi-cyclic interleaver. The motivation of this section is that we can reduce complexity and improve communications performance by simple



values in  $I^{offset}$  are the reversed permutation of the quasi-cyclic entries with respect to the last two groups of Eq. 8.32.

Figure 8.11 shows the open loop communications performance of a 64-QAM,  $4 \times 4$  antennas system. All curves use WiMAX LDPC codes, either with 10 or 30 iterations. The communications performance labeled with default sphere uses the bit interleaver defined in the WiMAX communications standard. The sphere decoder has to search through  $2^{24}$  branches of the tree. The improved communications performance is obtained by using the described bit to transmission vector mapping. The performance gain is up to 0.5 dB. The search space of the sphere detector is reduced by a factor of four (2 parity checks embedded).



**Fig. 8.11** Open loop communications performance, ergodic channel 64-QAM,  $4 \times 4$  antennas, both utilizing WiMAX LDPC codes. Improved communications performance and reduced complexity by a well chosen bit to transmission vector mapping. Note that the channel code is equivalent in both cases, while the number of leaves was reduced by a factor of four by embedding two parity checks (PC)

### 8.3.2 Sphere Decoder Aware LDPC Code Design

The second design example shows the design of LDPC codes with a special focus on the complexity reduction of the MIMO detector when using a tree search algorithm. Whenever considering a new design of an LDPC code it is beneficial when the new channel code can be processed by standard LDPC decoder architectures. Since nearly all wireless communications standards rely on quasi-cyclic LDPC codes we restrict the design example of this section to this type of codes. Here, only the basic idea is presented to introduce the potential of the approach of joint design of algorithm and architecture. Further results and how to derive the parity check matrix of the channel code are presented in [14, 15].

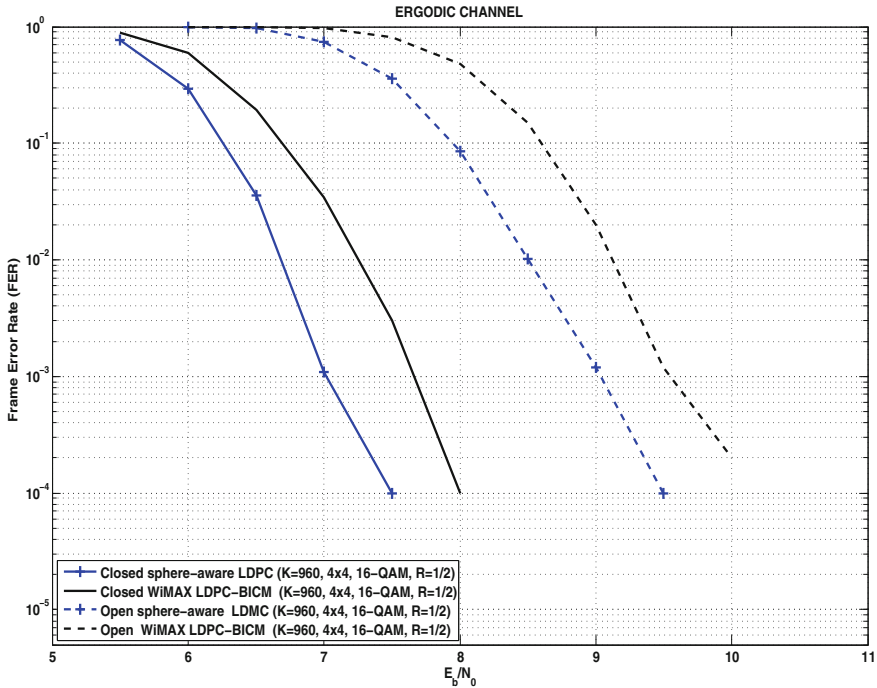
The transmission system assumed for the design example is a  $4 \times 4$  antenna system using a 16-QAM transmission scheme. The bit interleaver used in the example is a classical block interleaver with 16 columns and  $z$  rows. The number of rows of the block interleaver depends on the size of the identity matrix. In the following we assume one particular LDPC code with a block length of  $N = 1920$  bits and a code rate of  $R = 0.5$ .

$$H^{Macro} = \begin{pmatrix} 64 & 0 & 0 & 119 & 50 & 53 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 8 \\ 34 & 70 & 0 & 0 & 66 & 27 & 49 & 63 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 74 & 82 & 0 & 86 & 64 & 80 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 76 & 71 & 15 & 117 & 111 & 101 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{pmatrix} \quad (8.34)$$

The macro matrix with ( $z=120$ ) shown here is one realization which allows to embed four check nodes within each transmission vector. This is indicated by the separator between the top four and bottom four rows of the macro matrix. With the four rows all using identity matrices without permutation we can directly identify the embedded sub-code  $H'_e$ .

The block interleaver ensures always the correct codeword bit to transmission vector mapping. The bit positions which have to be mapped to the first transmission vector are  $[0 \ z - 1 \ 2z - 1 \ \dots \ 15z - 1]$  for the second transmission vector  $[1 \ z \ 2z \ \dots \ 15z]$ , and so on.

The communications performance results with respect to an ergodic channel and 16QAM  $4 \times 4$  system are shown in Fig. 8.12. The figure shows the open loop and the closed loop performance of the new LDPC code using embedded codes in comparison to the WiMAX simulations already presented in Fig. 8.7. Both schemes—WiMAX LDPC codes and joint LDPC code design—use 4 outer and 5 inner channel code iterations in the closed loop case. In the open loop case 20 LDPC iterations (layered) are performed. In both cases, the simulated performance of the open loop system as well as that of the closed loop system is better when using the joint design approach compared to the original WiMAX scheme.



**Fig. 8.12** Simulated communications performance (ergodic,  $4 \times 4$ , 16QAM) of an open loop system and a closed-loop system. The graph compares the original WiMAX code against a new LDPC code design. The new LDPC code can be processed by a WiMAX LDPC decoder architecture while reducing the size of the search tree for MIMO-APP detection by a factor of 16

In addition, when utilizing the new LDPC code design and a block interleaver the resulting tree for MIMO detection has only 4096 branches. Thus, the search tree for the sphere detector is reduced by a factor of 16 compared to standard case using WiMAX LDPC codes.

In summary the most important points when designing LDPC codes which use the knowledge of the sphere detector are:

- The properties of quasi-cyclic LDPC codes are used which enables the processing by standard decoder architectures.
- Parts of the channel code are implicitly solved during the MIMO-APP demodulation.
- The size of the search tree for MIMO-APP demodulation can be reduced by over a factor of ten.



- The achieved communications performance can be better than that of state-of-the-art BICM-MIMO schemes for open loop and closed loop simulations.

## References

1. Gans, M.J., Amitay, N., Yeh, Y.S., Xu, H., Valenzuela, R.A., Sizer, T., Storz, R., Taylor, D., MacDonald, W.M., Tran, C., Adamiecki, A.: 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, pp. 288–292 (2001). doi:[10.1109/VETECS.2001.944850](https://doi.org/10.1109/VETECS.2001.944850)
2. Alamouti, S.M.: A simple transmit diversity technique for wireless communications. *IEEE J. Sel. Areas Commun.* **16**(8), 1451–1458 (1998)
3. Hochwald, B., ten Brink, S.: Achieving near-capacity on a multiple-antenna channel. *IEEE Trans. Commun.* **51**(3), 389–399 (2003). doi:[10.1109/TCOMM.2003.809789](https://doi.org/10.1109/TCOMM.2003.809789)
4. Vikalo, H., Hassibi, B., Kailath, T.: Iterative decoding for MIMO channels via modified sphere decoding. *IEEE Trans. Wirel. Commun.* **3**(6), 2299–2311 (2004)
5. Paulraj, A.J., Gore, D.A., Nabar, R.U., Bölcskei, H.: An overview of MIMO communications—a key to Gigabit wireless. In: Proceedings of IEEE **92**(2), 198–218 (2004). doi:[10.1109/JPROC.2003.821915](https://doi.org/10.1109/JPROC.2003.821915)
6. Gimmler, C.: Dissertation in preparation. Ph.D. thesis, Department of Electrical Engineering and Information Technology, University of Kaiserslautern (2013)
7. ten Brink, S., Kramer, G., Ashikhmin, A.: Design of low-density parity-check codes for modulation and detection. *IEEE Trans. Commun.* **52**(4), 670–678 (2004). doi:[10.1109/TCOMM.2004.826370](https://doi.org/10.1109/TCOMM.2004.826370)
8. Müller, S., Schreger, M., Kabutz, M., Alles, M., Kienle, F., Wehn, N.: A novel LDPC decoder for DVB-S2 IP. In: Proceedings of the Design, Automation and Test in Europe Conference and Exhibition (DATE '09), pp. 1308–1313 (2009)
9. Alles, M., Berens, F., Wehn, N.: A synthesizable IP core for WiMedia 1.5 UWB LDPC code decoding. In: Proceedings of IEEE International Conference on Ultra-Wideband ICUWB 2009, pp. 597–601. Vancouver, Canada (2009)
10. May, M., Ilseher, T., Wehn, N., Raab, W.: A 150Mbit/s 3GPP LTE turbo code decoder. In: Proceedings of Design, Automation and Test in Europe, 2010 (DATE '10), pp. 1420–1425 (2010)
11. Sun, Y., Cavallaro, J.: Efficient hardware implementation of a highly-parallel 3GPP LTE/LTE-advance turbo decoder. *Integr. VLSI J.* (2010). doi:[10.1016/j.vlsi.2010.07.001](https://doi.org/10.1016/j.vlsi.2010.07.001)
12. Urard, P., Yeo, E., Paumier, L., Georgelin, P., Michel, T., Lebars, V., Lantreibeccq, E., Gupta, B.: A 135Mb/s DVB-S2 compliant codec based on 64800b LDPC and BCH codes. In: Proceedings of IEEE International Conference on Digest of Technical Papers Solid-State Circuits ISSCC, 2005, pp. 446–609 (2005). doi:[10.1109/ISSCC.2005.1494061](https://doi.org/10.1109/ISSCC.2005.1494061)
13. Gimmler-Dumont, C., Kienle, F., Wu, B., Masera, G.: A system view on iterative MIMO detection: dynamic sphere detection versus fixed effort list detection. *VLSI. Des. J.* Article ID 826350 (2012). doi:[10.1155/2012/826350](https://doi.org/10.1155/2012/826350)
14. Kienle, F.: Low-density MIMO codes. In: Proceedings of 5th International Symposium on Turbo Codes and Related Topics, pp. 107–112. Lausanne, Switzerland (2008)
15. Kienle, F.: On low-density MIMO codes. In: Proceedings of IEEE International Conference on Communications ICC '09, pp. 1–6 (2009). doi:[10.1109/ICC.2009.5199242](https://doi.org/10.1109/ICC.2009.5199242)