Generalized Mutual Information-Maximizing Quantized Decoding of LDPC Codes with Layered Scheduling

(Extended Version)

Peng Kang[†], Kui Cai[†], Xuan He[†], Shuangyang Li[*], Jinhong Yuan[*]

[†]Singapore University of Technology and Design (SUTD), Singapore, 487372

[*]The University of New South Wales, Sydney, Australia, 2032

Abstract—In this paper, we propose a framework of the mutual information-maximizing (MIM) quantized decoding for low-density parity-check (LDPC) codes by using simple mappings and fixed-point additions. Our decoding method is generic in the sense that it can be applied to LDPC codes with arbitrary degree distributions, and can be implemented based on either the belief propagation (BP) algorithm or the min-sum (MS) algorithm. In particular, we propose the MIM density evolution (MIM-DE) to construct the lookup tables (LUTs) for the node updates. The computational complexity and memory requirements are discussed and compared to the LUT decoder variants. For applications with low-latency requirement, we consider the layered schedule to accelerate the convergence speed of decoding quasi-cyclic LDPC codes. In particular, we develop the layered MIM-DE to design the LUTs based on the MS algorithm, leading to the MIM layered quantized MS (MIM-LQMS) decoder. An optimization method is further introduced to reduce the memory requirement for storing the LUTs. Simulation results show that the MIM quantized decoders outperform the state-of-the-art LUT decoders in the waterfall region with both 3-bit and 4-bit precision over the additive white Gaussian noise channels. For small decoding iterations, the MIM quantized decoders also achieve a faster convergence speed compared to the benchmarks. Moreover, the 4-bit MIM-LQMS decoder can approach the error performance of the floating-point layered BP decoder within 0.3 dB in the moderate-to-high SNR regions, over both the AWGN channels and the fast fading channels.

Index Terms—Low-density parity-check (LDPC) codes, lookup table (LUT), mutual information-maximizing (MIM), quantization.

I. INTRODUCTION

Low-density parity-check (LDPC) codes [1] and their variations [2]–[6] have been widely applied to wireless communication and data storage systems, for their capacity approaching performance under iterative message passing decoding [7]. Many researchers have put their efforts in designing the LDPC coding schemes to enhance the system performance for different channels, such as the partial-response channels and the fading channels e.g., [8]–[10]. From the implementation perspective, the development of LDPC decoders also has drawn great attention in terms of the decoding algorithms [11]–[13] and architectures [14]–[17], for achieving a good trade-off between the error performance and decoding complexity. In practical systems, the LDPC decoders require inputs with finite precision and hence it is necessary to perform quantization on the messages exchanged within the decoders. To support applications such as vehicle-to-everything and automated driving

in recent fifth-generation (5G) systems and beyond, the design of LDPC decoders is devoted to achieving ultra-reliability and fast convergence speed with low decoding complexity. In this context, the LDPC decoders use coarsely quantized messages attract much attention in recent literature [18]–[24]. Due to the low-precision messages used for decoding, the complexity of the LDPC decoders can be significantly reduced. One approach to implement the LDPC decoders with coarse quantization is to adopt lookup tables (LUTs) for the variable node (VN) and the check node (CN) updates. For example, the non-surjective finite alphabet iterative decoder (NS-FAID) [18] optimizes a single LUT for the VN update based on the density evolution (DE) [7] to achieve a high throughput. In [19], the error performance of the NS-FAID was improved by using multiple LUTs with optimized usage probabilities. Moreover, the mutual information-maximizing LUT (MIM-LUT) decoders were proposed in [20]–[24]. These decoders replace the arithmetic computation for the node updates with purely table lookup operations for low decoding complexity. Note that the LUTs of these decoders are designed based on the DE by considering an optimal or a sub-optimal quantization scheme to maximize the mutual information between the quantized messages and the associated coded bits. More specifically, the LUTs constructed in [20] are designed based on the optimal quantization by using the dynamic programming (DP) [25]. In [21], the information bottleneck method is utilized to generate the LUTs for decoding regular LDPC codes. Following [21], Stark *et al.* proposed the message alignment method in [22] and extended the framework to decode irregular LDPC codes. The message alignment method was further improved in [23] by considering the degree distributions of the LDPC codes in the design of the LUTs. In addition, the min-LUT decoder [24] replaces the belief propagation (BP) algorithm [11] by the min-sum (MS) algorithm for the CN update to reduce the decoding complexity. However, as the size of LUTs grows exponentially with the node degrees, the large LUTs in [20]–[24] have to be decomposed into a series of cascaded small LUTs to achieve a manageable memory usage, where each LUT has two inputs and single output symbols. This degrades the performance of these decoders due to the loss of mutual information [26].

To improve the error performance, the mutual information-maximizing quantized BP (MIM-QBP) decoder was proposed in [27], and elaborated in [28] for regular LDPC codes. Instead

of considering only two input symbols in the LUT design, the MIM-QBP decoder reconstructs each incoming message into a specific integer number and computes all possible combinations of the messages for the node updates. Then an optimal quantizer is determined by the DP method in [29] to generate the threshold sets for the node updates, which maps the combined messages to the outgoing messages. As shown in [28], the MIM-QBP decoder with only 3-bit precision can outperform the floating-point BP decoder at the high signal-to-noise ratio (SNR) region for regular LDPC codes. Meanwhile, Wang *et al.* proposed a decoding method in [30] based on the reconstruction-computation-quantization (RCQ) approach to mitigate the performance degradation caused by LUT decomposition. Similar to the MIM-QBP decoder, the RCQ decoder maps each input message to a corresponding log-likelihood ratio (LLR) and computes the combined messages based on either the BP or the MS algorithm [24]. However, the quantization threshold sets for the RCQ decoder are designed in a sub-optimal way for less computational complexity. The reconstructed LLR values are naturally with floating-point precision and require extra quantization for implementing the decoder with finite precision.

On the other hand, the LUTs of the LDPC decoders [18]–[24] and [30] are commonly designed by considering the flooding schedule [31], where all the CN and VN messages are simultaneously updated and propagated along the edges of the Tanner graph [32]. However, the flooding schedule requires a relatively large number of iterations for a satisfying decoding error performance. This results in a slow convergence speed and high hardware complexity [33]. To accelerate the convergence speed of the decoder, the layered schedules have been investigated in [33]–[37] to generate the messages in a sequential order by using the latest available information. In particular, the horizontal layered schedules in [33], [34] update the CN messages sequentially while the vertical layered schedules in [35]–[37] was based on a sequential update of the VNs' messages. The analytical studies in [33]–[37] have proved that the layered schedules converge twice as fast as the flooding schedule at no cost in decoding complexity. However, designing the MIM-LUT LDPC decoders for the layered schedule is still a challenging problem. Although an overview of the layered LUT-based decoder using the information bottleneck method is presented in [38], there exist few works to provide a systematic way to construct the layered LUT-based LDPC decoders in detail.

In this paper, we present a generalized framework of the mutual information-maximizing (MIM) quantized decoding for LDPC codes with arbitrary degree distributions. The proposed decoding method only requires simple mappings and fixed-point additions, and hence facilitates easier hardware implementation. To further speed up the convergence of the decoder for quasi-cyclic LDPC (QC-LDPC) codes, we propose the framework of the MIM layered decoder, which can significantly benefit the applications with low-latency requirement such as the vehicular communications. The main contributions of this work are summarized below:

- We generalize the framework of the MIM quantized decoding for the LDPC codes with arbitrary degree

distributions. In particular, we derive the density evolution (DE) by using the MIM quantization [29] to construct the LUTs of the reconstruction and quantization mappings for the MIM-QBP decoder. To reduce the computational complexity of decoding, we approximate the CN update of the MIM-QBP decoder by the MS algorithm and obtain the MIM quantized MS (MIM-QMS) decoder. We also compare the differences between the proposed MIM quantized decoders and that of the existing LUT decoder variants.

- We analyze the decoding complexity of the proposed MIM quantized decoders. We point out that the MIM quantized decoders have slightly increased computational complexity but significantly reduce the memory requirement compared to the MIM-LUT decoders.

- We compare the error performance and the convergence behaviors of the MIM quantized decoders with 3-bit and 4-bit precision to those of the different MIM-LUT decoders over the additive white Gaussian noise (AWGN) channels. Simulation results show that the MIM quantized decoders achieve a faster convergence compared to the state-of-the-art MIM-LUT decoders for small decoding iterations. Moreover, the MIM-QBP decoder can outperform the benchmark decoders with the same message precision in the waterfall region. In addition, the 4-bit MIM-QMS decoder can even surpass the floating-point BP decoder in the error-floor region for the length-1296 IEEE 802.11n standard LDPC code with rate $1/2$.

- We further develop the layered MIM-DE (LMIM-DE) and use it to design the LUTs of the reconstruction and quantization mappings at each layer and each iteration for the MIM layered QMS (MIM-LQMS) decoder. To reduce the memory requirement for storing the LUTs by different layers and iterations, we further optimize the LUTs and make them only vary with different iterations with neglectable degradation of mutual information.

- We evaluate the error performance of the constructed MIM-LQMS decoders with 3-bit and 4-bit precision over the AWGN channels and the fast fading channels. Simulation results show that the 4-bit MIM-LQMS decoder achieves error performance less than 0.3 dB away from the floating-point layered BP decoder in the moderate-to-high SNR regions.

The rest of this paper is organized as follows. In Section II, we review the basic framework of the MIM-QBP decoder for decoding regular LDPC codes. In Section III, we present the generalized framework of the MIM quantized decoders for any ensemble of the LDPC codes that can be characterized by degree distributions. We also highlight the differences in the design of the proposed decoder compared to the existing MIM-LUT decoders. The decoding complexity of the MIM quantized decoders is discussed in Section IV. We also investigate the error performance of the proposed MIM quantized decoders and compare it with that of different LUT decoder variants. Section V introduces the design of the MIM-LQMS decoder, presents the LUT optimization method, and also illustrates the simulation results of the proposed MIM-LQMS decoders. Section VII concludes the paper.

TABLE I: List of Notations for the MIM-QBP Decoder

q_m	The precision (bit widths) of outer messages
q_c	The precision (bit widths) of inner messages at CNs
q_v	The precision (bit widths) of inner messages at VNs
$\mathcal{X}$	The alphabet set of coded bits
X	A random variable of the coded bit
$\mathcal{L}$	The alphabet set of the channel quantization output
L and l	A random variable and its realization of the channel quantization output
$P_{L\|X}$	The channel transition probability for output L conditioned on input X
$\mathcal{R}$	The alphabet set of variable-to-check (V2C) messages
R and r	A random variable and its realization of the V2C message
$P_{R\|X}$	The probability mass function (pmf) of V2C message R conditioned on X
$\mathcal{S}$	The alphabet set of check-to-variable (C2V) messages
S and s	A random variable and its realization of the C2V message
$P_{S\|X}$	The pmf of C2V message S conditioned on X

II. PRELIMINARIES

A. Notations

In this paper, we use the calligraphy capitals to define an alphabet set. The normal capitals denote the random variables. The lower-case letters denote the realization of a random variable. Boldface letters are used to define a vector. For convenience, we list the notations of the MIM-QBP decoder in TABLE I.

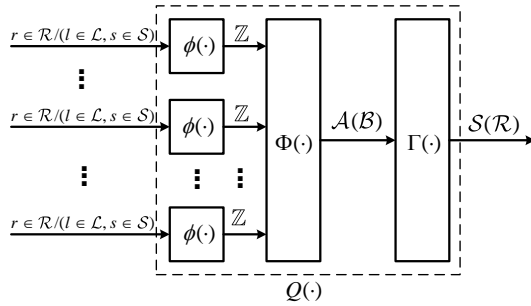### B. The MIM-QBP decoder for (d_v, d_c) LDPC codes

Fig. 1: The framework of the MIM-QBP decoder for decoding the (d_v, d_c) LDPC codes.

We briefly review the framework of the MIM-QBP decoder for (d_v, d_c) LDPC codes in [28]. As shown in Fig. 1, the node (both CN and VN) updates of the MIM-QBP decoder is divided into three steps: reconstruction, calculation, and quantization. We represent these three steps correspondingly by three functions, i.e., the reconstruction function $\phi(\cdot)$, the computing function $\Phi(\cdot)$, and the quantization function $\Gamma(\cdot)$. We call the messages at the reconstruction function input and the quantization function output as outer messages, and those related to the computing function as inner messages. The precision of the MIM-QBP decoder can be fully described by the bit widths q_m, q_c, and q_v. With the preset value of q_m, q_c, and q_v, the MIM-QBP decoder is designed under the discrete memoryless channel (DMC). Without loss of

generality, we obtain the DMC from a binary-input AWGN channel, where the coded bit X takes values from $\mathcal{X} = \{0, 1\}$. In the design procedure, the MIM-QBP decoder associates the LLRs of the messages exchanged between CNs and VNs to the symbols in the finite alphabet sets. More specifically, we let $\mathcal{L} = \{0, 1, \ldots, |\mathcal{L}| - 1\}$ be the alphabet set of the channel output, where $|\cdot|$ is the size of the alphabet set. We also consider the alphabet sets of V2C and C2V messages as $\mathcal{R} = \{0, 1, \ldots, |\mathcal{R}| - 1\}$ and $\mathcal{S} = \{0, 1, \ldots, |\mathcal{S}| - 1\}$, respectively. In general, we have $|\mathcal{R}| = |\mathcal{S}| = |\mathcal{L}| = 2^{q_m}$ and assume that the symbols in the alphabet set are sorted in a way such that their associated LLRs are in descending order. The design of the node updates for the MIM-QBP decoder is presented in detail as follows.

1) Reconstruction $\phi(\cdot)$: The reconstruction function $\phi(\cdot)$ maps each q_m-bit outer message to a specific number in the integer domain $\mathbb{Z}$. We denote the reconstruction function for the channel output by ϕ_{ch}, and those for the CN and VN updates by ϕ_c and ϕ_v, respectively. For the CN update, ϕ_c in [28] is designed to indicate the unreliability of the V2C message R such that larger $|\phi_c|$ means smaller magnitude of the associated LLR value. Let $g(r) = P_{R|X}(r|0) - P_{R|X}(r|1)$ for $r \in \mathcal{R}$. Then we have [28]

$$\phi_c(r) = \begin{cases} sgn(g(r)) \cdot \max\{1, \\ \left\lfloor \frac{2^{q_c-1}-1}{d_c} \cdot |\log(|g(r)|)| / \alpha + 0.5 \right\rfloor\}, & |g(r)| \neq 0, \\ \left\lfloor (2^{q_c-1} - 1)/d_c \right\rfloor, & |g(r)| = 0, \end{cases}$$
(1)

where $\alpha = \max\{|\log(|g(r)|)| : r \in \mathcal{R}, g(r) \neq 0\}$ is the normalized factor to ensure $\sum_{k=1}^{d_c} |\phi_c(r_k)| \leq 2^{q_c-1} - 1$, and $sgn(\cdot)$ refers to the sign of the message such that

$$sgn(\alpha) = \begin{cases} -1, & \alpha < 0, \\ 1, & \alpha \geq 0. \end{cases}$$

For the VN update, ϕ_{ch} and ϕ_v in [28] are designed to reflect the reliability of the C2V message S and the channel output L, where the larger magnitude of the reconstruction functions means the larger magnitude of the associated LLR

values. Let $h(l) = \log(P_{L|X}(l|0)/P_{L|X}(l|1))$ for $l \in \mathcal{L}$, and $h(s) = \log(P_{S|X}(s|0)/P_{S|X}(s|1))$ for $s \in \mathcal{S}$, respectively. The reconstruction functions ϕ_{ch} and ϕ_v is given by [28]

$$\begin{cases} \phi_{ch}(l) = sgn(h(l)) \cdot \left\lfloor \frac{2^{q_v-1}-1}{d_v+1} \cdot |h(l)|/\beta + 0.5 \right\rfloor, \\ \phi_v(s) = sgn(h(s)) \cdot \left\lfloor \frac{2^{q_v-1}-1}{d_v+1} \cdot |h(s)|/\beta + 0.5 \right\rfloor, \end{cases} \quad (2)$$

where $\beta = \max(\{|h(s)| : s \in \mathcal{S}\} \cup \{|h(l)| : l \in \mathcal{L}\})$ is the normalized factor to refine $|\phi_{ch}(l)| + d_v \cdot |\phi_v(s)| \leq 2^{q_v-1} - 1$.

2) Calculation $\Phi(\cdot)$: Following reconstruction, the computing function $\Phi(\cdot)$ is used to calculate the inner message with q_c-bit or q_v-bit precision. We denote the computing functions for the CN update and the VN update by Φ_c and Φ_v, respectively. Let $\mathbf{R} \in \mathcal{R}^{d_c-1}$ be the vector of V2C messages at a CN of degree d_c. For a realization $\mathbf{r} = (r_1, r_2, \ldots, r_{d_c-1}) \in \mathcal{R}^{d_c-1}$, Φ_c in [28] is designed based on the BP algorithm as

$$\Phi_c(\mathbf{r}) = \left(\prod_{k=1}^{d_c-1} sgn(\phi_c(r_k)) \right) \sum_{k=1}^{d_c-1} |\phi_c(r_k)|. \quad (3)$$

Define $\mathbf{S} \in \mathcal{S}^{d_v-1}$ as the vector of C2V messages at a VN of degree d_v. For a realization $\mathbf{s} = (s_1, s_2, \ldots, s_{d_v-1}) \in \mathcal{S}^{d_v-1}$, $\Phi_v(\cdot)$ in [28] operates as

$$\Phi_v(l, \mathbf{s}) = \phi_{ch}(l) + \sum_{k=1}^{d_v-1} \phi_v(s_k). \quad (4)$$

We denote the alphabet sets of the inner message for the CN update and the VN update by integer sets $\mathcal{A} = \{a_1, a_2, \ldots, a_{|\mathcal{A}|}\} = \{\Phi_c(\mathbf{r}) : \mathbf{r} \in \mathcal{R}^{d_c-1}\}$, and $\mathcal{B} = \{b_1, b_2, \ldots, b_{|\mathcal{B}|}\} = \{\Phi_v(l, \mathbf{s}) : (l, \mathbf{s}) \in \mathcal{L} \times \mathcal{S}^{d_v-1}\}$, respectively. The elements in $\mathcal{A}$ and $\mathcal{B}$ are listed in a certain order such that their associated LLRs are (or almost) decreasing. Specifically, the elements in $\mathcal{A}$ are labeled to satisfy $a_1 \succ a_2 \succ \cdots \succ a_{|\mathcal{A}|}$, where $\succ$ refers to a binary relation on $\mathbb{Z}$ such that $\alpha \succ \beta \iff sgn(\alpha) > sgn(\beta)$ or $(sgn(\alpha) = sgn(\beta)$ and $\alpha < \beta)$ for $\alpha, \beta \in \mathbb{Z}$. Here we give an example of the alphabet set $\mathcal{A}$ of size $|\mathcal{A}| = 6$ to clarify the binary relation $\succ$ such that $\mathcal{A} = \{a_1 = 4, a_2 = 10, a_3 = 30, a_4 = -30, a_5 = -10, a_6 = -4\}$. For the alphabet set $\mathcal{B}$, we have $b_1 > b_2 > \cdots > b_{|\mathcal{B}|}$.

3) Quantization $\Gamma(\cdot)$: Based on $\mathcal{A}$ and $\mathcal{B}$, we quantize the q_c and q_v bits inner message to q_m bits outer message by an optimal q_m-bit sequential deterministic quantizer (SDQ) [29]. Assume that the input message Y takes values from $\mathcal{Y} = \{y_1, y_2, \ldots, y_N\}$, where we have $\mathcal{Y} = \mathcal{A}$ for the CN update and $\mathcal{Y} = \mathcal{B}$ for the VN update. A q_m-bit SDQ for Y can be expressed as

$$Q(x) = \begin{cases} 0, & x \in \{y_1, y_2, \ldots, y_{\lambda_1}\}, \\ 1, & x \in \{y_{\lambda_1+1}, y_{\lambda_1+2}, \ldots, y_{\lambda_2}\}, \\ \vdots & \vdots \\ 2^{q_m}-1, & x \in \{y_{\lambda_{2^{q_m-1}}+1}, y_{\lambda_{2^{q_m-1}}+2}, \ldots, y_N\}, \end{cases} \quad (5)$$

where $0 < \lambda_1 < \cdots < \lambda_{2^{q_m}-1} < N$ are the $2^{q_m} - 1$ element indices to form the threshold set $\Gamma = \{\gamma_k : \gamma_k = y_{\lambda_k}, k = 1, 2, \ldots, 2^{q_m} - 1\}$. The threshold set of the optimal q_m-bit SDQ is determined by the DP method [29], which considers the MIM criterion such that the mutual information between

the coded bit X and the C2V (resp. V2C) message S (resp. R) can be maximized. We denote the MIM quantization for the CN and VN updates by

$$\arg\max_Q I(X; S), \text{ and } \arg\max_Q I(X; R), \quad (6)$$

respectively. We denote the threshold set for the CN update by Γ_c, and denote the threshold set for the VN update by Γ_v, which operate as

$$\Gamma_c(x) = \begin{cases} 0, & x \succeq \gamma_1, \\ 2^{q_m}-1, & \gamma_{2^{q_m}-1} \succ x, \\ k, & \gamma_k \succ x \succeq \gamma_{k+1}, \end{cases} \quad \Gamma_v(x) = \begin{cases} 0, x \geq \gamma_1, \\ 2^{q_m}-1, x < \gamma_{2^{q_m}-1}, \\ k, \gamma_k > x \geq \gamma_{k+1}. \end{cases} \quad (7)$$

Here $\succeq$ is defined as a binary relation on $\mathbb{Z}$ such that $\alpha \succeq \beta \iff \alpha \succ \beta$ or $\alpha = \beta$ for $\alpha, \beta \in \mathbb{Z}$. To design the MIM-QBP decoder, we first discretize a binary-input AWGN channel by considering the MIM quantization. The threshold set for quantizing the channel output is denoted by Γ_{ch}, which has the same operations as Γ_v. The channel parameter of the AWGN channel, denoted by σ_d, is the design noise standard deviation assumed to be known at the receiver. Based on the constructed DMC, we perform reconstruction, calculation, and quantization for the node updates and track the evolution of $P_{R|X}$ and $P_{S|X}$ to construct the reconstruction functions and threshold sets at each iteration accordingly. [1] The associated reconstruction functions and threshold sets are stored as the LUTs for implementing the decoding. Define the mapping functions $Q_c : \mathcal{R}^{d_c-1} \to \mathcal{S}$ and $Q_v : \mathcal{L} \times \mathcal{S}^{d_v-1} \to \mathcal{R}$ for the CN and the VN updates, respectively, which consist of the above three steps. In addition, the mapping function for bit decision $Q_e : \mathcal{L} \times \mathcal{S}^{d_v} \to \mathcal{X}$ is designed similarly to Q_v based on the same reconstruction functions ϕ_v and ϕ_{ch}. However, the alphabet set of the inner messages becomes $\{\Phi_v(l, \mathbf{s}) : (l, \mathbf{s}) \in \mathcal{L} \times \mathcal{S}^{d_v}\}$ and we have $|\Gamma_v| = 1$ for hard decision. We ignore the details here due to space limitations.

III. GENERALIZED MIM QUANTIZED DECODING FOR IRREGULAR LDPC CODES

In this section, we extend the design principle of the MIM-QBP decoder to irregular LDPC codes and propose the generalized framework of the MIM quantized decoding for the LDPC codes with arbitrary degree distributions. As shown in Section II, the reconstruction functions and threshold sets of the MIM-QBP decoder are constructed by quantizing the pmf between the coded bits and the messages sent from the CNs or VNs. Therefore, we first derive the MIM density evolution (MIM-DE) to trace the evolution of the pmf when decoding the irregular LDPC codes. The main difference between our MIM-DE and the conventional DE [7] is that we consider the node update with reconstruction-calculation-quantization architecture, and aim to maximize the mutual information between the coded bits and the message from the CN or VN, while the conventional DE uses uniform quantization and considers the asymptotic error probability. To reduce the

---

[1]In this paper, we consider reconstruction functions and threshold sets varying with iterations. Here we do not specify these notations for the associated iterations because once the decoder design for one iteration is determined, it is straightforward to generalize it for the other iterations.
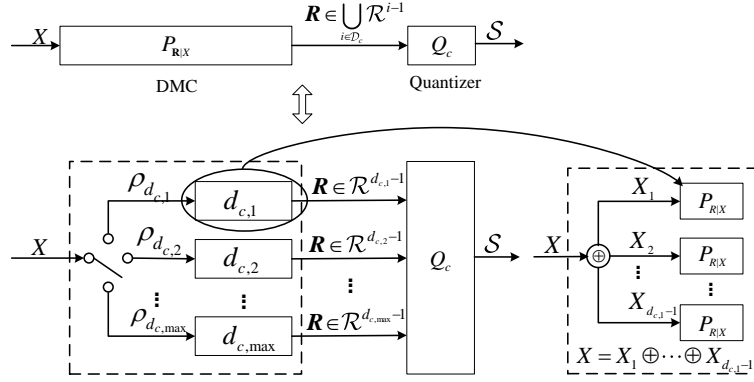
Fig. 2: The MIM-QBP decoder design at CN and its equivalence to quantization for DMC.

decoding complexity, we further simplify the CN update of the MIM-QBP decoder by the MS algorithm [24], which leads to the MIM-QMS decoder. We refer to either the MIM-QBP decoder or the MIM-QMS decoder as the MIM quantized decoder. In addition, we also compare the proposed design method of the MIM quantized decoders with those of the prior state-of-art LUT decoder variants.

Denote the sets of the CN and VN degrees by $\mathcal{D}_c = \{d_{c,1}, d_{c,2}, \ldots, d_{c,\max}\}$ with $d_{c,1} < d_{c,2} < \cdots < d_{c,\max}$ and $\mathcal{D}_v = \{d_{v,1}, d_{v,2}, \ldots, d_{v,\max}\}$ with $d_{v,1} < d_{v,2} < \cdots < d_{v,\max}$, respectively. An ensemble of binary LDPC codes can be characterized by the degree distributions as

$$\rho(x) = \sum_{i \in \mathcal{D}_c} \rho_i x^{i-1}, \; \theta(x) = \sum_{j \in \mathcal{D}_v} \theta_j x^{j-1}, \qquad (8)$$

where ρ_i and θ_j are the fractions of edges incident to the CNs with degree-i and the VNs with degree-j, respectively. In the following, we illustrate the design of the MIM-QBP decoder for irregular LDPC codes.

A. MIM-DE at CN Update

As depicted in Fig. 2, the design of the mapping function Q_c for the CN update of the irregular LDPC codes is equivalent to solving the DMC quantization problem as

$$Q_c : \bigcup_{i \in \mathcal{D}_c} \mathcal{R}^{i-1} \to \mathcal{S}$$

such that $I(X; S)$, $S \in \mathcal{S}$ is maximized. Note that the DMC output can be characterized by $|\mathcal{D}_c|$ independent alphabets corresponding to different CN degrees, where we have $|\mathcal{D}_c| = 1$ for regular LDPC codes. From Section II, we know that updating the degree-i ($i \in \mathcal{D}_c$) CNs requires $i - 1$ V2C messages from its connected VNs. Define $\mathbf{R} \in \mathcal{R}^{i-1}$ as the vector of the V2C messages sent to the degree-i CN. Based on the independence and identical distribution (i.i.d) assumption in the DE [7], each V2C message has the same pmf $P_{R|X}$. Therefore, the joint pmf $P_{\mathbf{R}|X}$ of $\mathbf{R}$ conditioned on the coded bit X is given by

$$P_{\mathbf{R}|X}(\mathbf{r}|x) = \left(\frac{1}{2}\right)^{i-2} \sum_{\mathbf{x}: \oplus \mathbf{x} = x} \prod_{k=1}^{i-1} P_{R|X}(r_k|x_k), \qquad (9)$$

where $x \in \mathcal{X}$ is a realization of coded bit X, and $\mathbf{x} = (x_1, x_2, \ldots, x_{i-1})$ is the vector of input coded bits associated with the CN's connecting VNs. Note that $\oplus \mathbf{x} = x$ refers to the case where the checksum of the CN is satisfied.

Let $\mathcal{A}_i = \{a_{i,1}, a_{i,2}, \ldots, a_{i,|\mathcal{A}_i|}\} = \{\Phi_c(\mathbf{r}) : \mathbf{r} \in \mathcal{R}^{i-1}, i \in \mathcal{D}_c\}$ be the alphabet set of the inner message calculated for the degree-i CNs, where $\Phi_c(\cdot)$ is given by (3). Denoted by A_i the random variable for the inner message taking values from $\mathcal{A}_i$, we obtain the conditional probability

$$P_{A_i|X}(a_{i,k}|x) = \sum_{\substack{\mathbf{r} \in \mathcal{R}^{i-1}, \\ \Phi_c(\mathbf{r}) = a_{i,k}}} P_{\mathbf{R}|X}(\mathbf{r}|x), \; k = 1, 2, \ldots, |\mathcal{A}_i|. \qquad (10)$$

According to (8), we obtain $\mathcal{A} = \cup_{i \in \mathcal{D}_c} \mathcal{A}_i$ by considering all $|\mathcal{D}_c|$ independent alphabets for the CN update. Define $A \in \mathcal{A}$ as the inner message computed for all CNs of different degrees. Based on the fraction ρ_i of the degree-i CNs, we have $A \in \mathcal{A}_i$ with probability ρ_i. Let $P_{A|X}$ denote the pmf of A conditioned on the coded bit X, which is given by

$$P_{A|X}(a|x) = \sum_{i \in \mathcal{D}_c} \rho_i \cdot P_{A_i|X}(a|x), \qquad (11)$$

where $a \in \mathcal{A}$ is a realization of A. Based on $\mathcal{A}$ and $P_{A|X}$, we perform the MIM quantization by using the DP method [29] to determine the threshold set Γ_c and the associated pmf $P_{S|X}$, which is denoted by

$$[P_{S|X}, \Gamma_c] = \mathbf{DP}(\mathcal{A}, P_{A|X}). \qquad (12)$$

Consequently, the reconstruction functions ϕ_{ch} and ϕ_v for current iteration can be constructed according to (2) for the VN update.

B. MIM-DE at VN Update

As shown by Fig. 3, the design of the mapping function Q_v for the VN update of irregular LDPC codes is equivalent to solving the quantization problem over DMC such that

$$Q_v : \mathcal{L} \times \bigcup_{j \in \mathcal{D}_v} \mathcal{S}^{j-1} \to \mathcal{R},$$

which maximize $I(X; R)$, $R \in \mathcal{R}$. Note that the DMC output is classified into $|\mathcal{D}_v|$ independent alphabets corresponding to different VN degrees, where $|\mathcal{D}_v| = 1$ is a special subcase
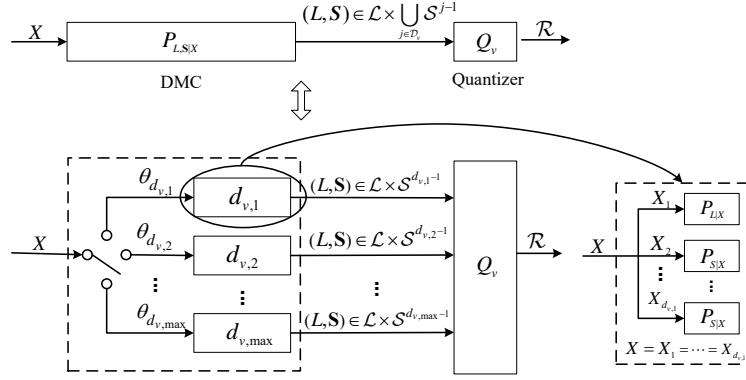
Fig. 3: The MIM-QBP decoder design at VN and its equivalence to quantization for DMC.

corresponding to regular LDPC codes. To update a degree-j ($j \in \mathcal{D}_v$) VN, we need the message from the channel output and $j-1$ C2V messages from its connected CNs. Define $\mathbf{S} \in \mathcal{S}^{j-1}$ as the vector of the C2V messages sent to the degree-j VN. We denote the vector of the C2V messages and the channel output by $(L, \mathbf{S}) \in \mathcal{L} \times \mathcal{S}^{j-1}$. Similar to the case of the CN update, we have the pmf $P_{S|X}$ for each C2V message and $P_{L|X}$ for the channel output. As a result, the joint pmf $P_{L,\mathbf{S}|X}$ can be computed by

$$P_{L,\mathbf{S}|X}(l,\mathbf{s}|x) = P_{L|X}(l|x) \prod_{k=1}^{j-1} P_{S|X}(s_k|x), \ j \in \mathcal{D}_v. \quad (13)$$

Let $\mathcal{B}_j = \{b_{j,1}, b_{j,2}, \ldots, b_{j,|\mathcal{B}_j|}\} = \{\Phi_v(l,\mathbf{s}) : l \in \mathcal{L}, \mathbf{s} \in \mathcal{S}^{j-1}\}$ be the alphabet set of the inner message computed for the VNs of degree-j, where $\Phi_v(\cdot)$ is given by (4). Denoted by B_j the random variable for the inner message taking values from $\mathcal{B}_j$, we obtain

$$P_{B_j|X}(b_{j,k}|x) = \sum_{\substack{(l,\mathbf{s}) \in \mathcal{L} \times \mathcal{S}^{j-1}, \\ \Phi_v(l,\mathbf{s})=b_{j,k}}} P_{L,\mathbf{S}|X}(l,\mathbf{s}|x), \ k = 1,2,\ldots,|\mathcal{B}_j|.$$
$$(14)$$

Based on (8), we obtain $\mathcal{B} = \cup_{j \in \mathcal{D}_v} \mathcal{B}_j$ by considering all $|\mathcal{D}_v|$ independent alphabet sets for the VN update. Define $B \in \mathcal{B}$ as the random variable for the inner message calculated by considering all VNs of different degrees. With the fraction θ_j of the degree-j VNs, we have $B \in \mathcal{B}_j$ with probability θ_j. Let $P_{B|X}$ denote the pmf of B conditioned on the coded bit X, which is given by

$$P_{B|X}(b|x) = \sum_{j \in \mathcal{D}_v} \theta_j \cdot P_{B_j|X}(b|x), \quad (15)$$

where $b \in \mathcal{B}$ is a realization of B. In the end, the MIM quantization [29] is conducted based on $\mathcal{B}$ and $P_{B|X}$, which is denoted by

$$[P_{R|X}, \Gamma_v] = \mathbf{DP}(\mathcal{B}, P_{B|X}). \quad (16)$$

As a result, the threshold set Γ_v for the current iteration is determined and the reconstruction function ϕ_c for the next iteration can be built based on (1) and $P_{R|X}$. In addition, the mapping function for the bit decision is $Q_e : \mathcal{L} \times \bigcup_{j \in \mathcal{D}_v} \mathcal{S}^j \to$

$\mathcal{X}$ for irregular LDPC codes. Since the design of Q_e is similar to that of Q_v, we ignore the details to save space. Note that the proposed MIM-DE does not consider the all-zero codeword assumption because it may result in asymmetrical pmf when using the DP method to realize the quantization [20].

Algorithm 1 Deisgn Flow of the MIM-QBP Decoder

Input: σ_d, $\rho(x)$, $\theta(x)$, q_m, q_c, q_v, $I_{\max}$.
Output: Γ_{ch}, $\Gamma_c^{(t)}$, $\Gamma_v^{(t)}$, $\phi_c^{(t)}$, $\phi_v^{(t)}$, $\phi_{ch}^{(t)}$.
1: Compute $P_{L|X}$ and Γ_{ch} with MIM quantization
2: $P_{R|X}^{(0)} = P_{L|X}$
3: **for** $t = 1 : I_{\max}$ **do**
4:     Construct $\phi_c^{(t)}$ with $P_{R|X}^{(t-1)}$ based on (1)
5:     Calculate $\mathcal{A}^{(t)}$ and $P_{A|X}^{(t)}$ using (3) and (11), respectively
6:     $[P_{S|X}^{(t)}, \Gamma_c^{(t)}] = \mathbf{DP}(\mathcal{A}^{(t)}, P_{A|X}^{(t)})$
7:     Construct $(\phi_v^{(t)}, \phi_{ch}^{(t)})$ with $P_{S|X}^{(t)}$ and $P_{L|X}$ based on (2)
8:     Calculate $\mathcal{B}^{(t)}$ and $P_{B|X}^{(t)}$ using (4) and (15), respectively
9:     $[P_{R|X}^{(t)}, \Gamma_v^{(t)}] = \mathbf{DP}(\mathcal{B}^{(t)}, P_{B|X}^{(t)})$
10:     Repeat 7-9 with $\mathbf{s} \in \mathcal{S}^j, j \in \mathcal{D}_v$ and $q_m = 1$ to construct $\Gamma_e^{(t)}$
11: **end for**

Inspired by the observations investigated empirically in [20]–[24], [28], the design noise standard deviation σ_d is selected based on the mutual information-maximizing criterion. More precisely, we perform the MIM-DE and choose σ_d to make the decoder converge at the maximum decoding iteration $I_{\max}$. In this paper, σ_d is determined via the bisection search. After selecting σ_d, the pmf $P_{L|X}$ and the threshold set Γ_{ch} of the associated DMC are obtained by performing the MIM quantization [29]. Then the pmfs $P_{S|X}$ and $P_{R|X}$ are iteratively updated by using the MIM-DE to construct ϕ_c, ϕ_v, ϕ_{ch}, Γ_c, and Γ_v, respectively. We summarize the design flow of the MIM-QBP decoder in **Algorithm 1**. The superscript t indicates the iteration number. Note that our proposed method can design the parameters of the decoders systematically based on the MIM-DE, given the degree distributions of the target

LDPC codes and the design noise standard deviation σ_d. This means that the LUTs of the reconstruction functions and the threshold sets are constructed depending on the selected σ_d, and they do not change with the received SNRs when implementing decoding. Moreover, our proposed method has the potential to be extended to other channels whose channel transition probabilities can be expressed explicitly. In addition, we would like to point out that the computational complexity of the design process mainly originates from the calculation and quantization steps. As motivated by [28], the computational complexity is $O(2^{q_c+q_m} \cdot (d_{c,\max} + 2^{q_c}))$ for the CN update and $O(2^{q_v+q_m} \cdot (d_{v,\max} + 2^{q_v}))$ for the VN update at each iteration. This is not a big issue for the offline process.

C. The MIM-QMS Decoder with Reduced-Complexity

Compared to the outer messages with q_m-bit precision, the computing function given by (3) requires a large q_c-bit precision and a large amount of fixed-point additions especially for the LDPC codes with high CN degrees. To reduce the computational complexity, we approximate (3) by the MS algorithm [24]. In this case, we define a function $f(\cdot)$ to reflect the relationship of the relative LLRs of the V2C messages for MS operation. More specifically, $f(\cdot)$ maps the V2C messages in $\mathcal{R} = \{0, 1, \ldots, 2^{q_m} - 1\}$ to an integer set $\{2^{q_m-1}, \ldots, 1, -1, \ldots -2^{q_m-1}\}$. To differentiate the updating rule for the CN, we denote the computing function based on the BP algorithm and the MS algorithm by Φ_c^{BP} and Φ_c^{MS}, respectively. For a degree-i $(i \in \mathcal{D}_c)$ CN, Φ_c^{MS} is given by

$$\Phi_c^{\mathrm{MS}}(\mathbf{r}) = f^{-1}\left(\prod_{k=1}^{i-1} sgn(f(r_k)) \cdot \min_{k \in 1,2,\ldots,i-1}(|f(r_k)|)\right),$$
(17)

where $f^{-1}(\cdot)$ is the inverse function of $f(\cdot)$. We call the decoder designed based on Φ_c^{MS} as the MIM quantized MS (MIM-QMS) decoder. In this paper, we compute Φ_c^{MS} recursively on $k = 1, 2, \ldots, i - 1$ to achieve low complexity. A similar recursive method is adopted by [39] for the DE to select a proper channel scaling factor of the MS decoder. Note that the inner messages computed by the MS operation have the same cardinality as that of the outer messages, i.e., $q_c = q_m$, for all iterations. Therefore, Γ_c is not required and the pmf $P_{S|X}$ in (12) can be directly derived from $P_{\mathbf{R}|X}$ as

$$P_{S|X}(s|x) = \sum_{i \in \mathcal{D}_c} \rho_i \cdot \sum_{\substack{\mathbf{r} \in \mathcal{R}^{i-1}, \\ \Phi_c^{\mathrm{MS}}(\mathbf{r})=s}} P_{\mathbf{R}|X}(\mathbf{r}|x), \qquad (18)$$

where $P_{\mathbf{R}|X}$ is the joint pmf given by (9). Compared to the MIM-QBP decoder, the MIM-QMS decoder requires less bit widths used for the CN update. Due to $q_c = q_m$, the precision of the MIM-QMS decoder can be simply determined by q_m and q_v. Moreover, the MIM-QMS decoder also reduces the computational complexity caused by the fixed-point additions in Φ_c^{BP}.

D. Remarks

We now discuss the difference between the design of the MIM quantized decoders and the LUT decoder variants [18], [19], [22]–[24], [30] that utilize the DE techniques to design the LUTs for irregular LDPC codes. In [18] and [19], the DE is employed to optimize one or multiple iteration-invariant LUTs for the VN update, which aims to minimize the asymptotic error probability with a preset memory constraint of the decoder. Instead of considering the error probability, the MIM quantized decoders are designed from the perspective of maximizing the mutual information between the coded bits and the messages from the CN or VN.

Furthermore, the MIM-LUT decoders in [22]–[24] adopt DE to construct the LUTs for node updates based on the tree-like decomposition structure with $d_{c,\max}$ or $d_{v,\max}$ levels. Note that each level of the tree-like structure is corresponding to the nodes with a specific degree. More precisely, the message alignment method is adopted in [22] as a post-processing step by considering the degree distributions after designing the LUTs for all levels. The implicit message alignment [23] and the min-LUT decoder [24] take the degree distributions into account to design the LUTs for the VN update at every j-th level, where $j \in \mathcal{D}_v$. Obviously, the design methods in [22]–[24] deteriorates the performance of the MIM-LUT decoders since table decomposition causes the loss of mutual information [26]. It is worth noting that our proposed MIM quantized decoders utilize fixed-point additions to handle all inner messages simultaneously without using the decomposition technique. In this way, the MIM quantized decoders can minimize the degradation of the error performance.

Apart from the aforementioned decoders, Wang et al. proposed the RCQ decoder in [30] based on the DE, which has a similar framework of the MIM quantized decoders in terms of reconstruction, calculation, and quantization. However, the reconstruction functions of the RCQ decoder are designed in the floating-point domain, and the quantization scheme considered in the RCQ decoder is sub-optimal compared to the optimal DP method [29].

IV. COMPLEXITY AND PERFORMANCE ANALYSIS OF MIM QUANTIZED DECODERS

In this section, we investigate the decoding complexity of the proposed MIM quantized decoders, which is a critical issue that needs to be considered due to system constraints. More specifically, we discuss the computational complexity of the MIM quantized decoders, and compare the implementation complexity in terms of the memory requirement with the MIM-LUT decoders. Moreover, the error performance of the MIM quantized decoders and the MIM-LUT decoders with low precision is evaluated and compared via Monte-Carlo simulations.

A. Computational Complexity

As known from Section II, the computational complexity of the MIM quantized decoders mainly originates from the calculation and quantization steps since the reconstruction step is a one-to-one mapping by the corresponding reconstruction functions. Denote the average CN and VN degrees by

$$\bar{d}_c = 1 \Big/ \sum_{i \in \mathcal{D}_c} \frac{\rho_i}{i} \text{ and } \bar{d}_v = 1 \Big/ \sum_{j \in \mathcal{D}_v} \frac{\theta_j}{j},$$

TABLE II: Computational Complexity of One Node per Iteration for the MIM Quantized Decoders

Mapping functions	QBP	QMS
Q_c	$\bar{d}_c \cdot (q_m + 2) - 1$	$2\bar{d}_c + \lceil \log_2(\bar{d}_c) \rceil - 2$
Q_v	$\bar{d}_v \cdot (q_m + 2)$	
Q_e	1	

TABLE III: Memory Demand per Iteration for Various Decoders

Decoders	MIM-QBP	MIM-QMS	MIM-LUT [22]–[24]
Q_c	$2^{q_m+1} - 1$	–	$2 \cdot \lfloor \log_2(d_{c,\max}) \rfloor \cdot 2^{2q_m}$
Q_v	$2^{q_m+2} - 2$		$2 \cdot \lfloor \log_2(d_{v,\max}) \rfloor \cdot 2^{2q_m}$
Q_e	1		$2 \cdot \lfloor \log_2(d_{v,\max} + 1) \rfloor \cdot 2^{2q_m}$

respectively. According to [40], the calculation of $\Phi_{\underline{c}}^{\mathrm{BP}}$ for the CN update of the MIM-QBP decoder requires $\bar{d}_c - 1$ addition operations to compute the summation of $\bar{d}_c$ inner messages, and $\bar{d}_c$ subtraction operations to compute the $\bar{d}_c$ outer messages. Moreover, the table lookup operations for q_m-bit outer messages based on Γ_c require $\bar{d}_c \cdot q_m$ comparisons. Therefore, we have the average computational complexity of $\bar{d}_c \cdot (q_m + 2) - 1$ for one CN per iteration. For the MIM-QMS decoder, the MS operation can be implemented by an efficient method such as [41], which leads to a computational complexity $2\bar{d}_c + \lceil \log_2(\bar{d}_c) \rceil - 2$ for one CN per iteration. Furthermore, the computation of Φ_v for the VN update requires $\bar{d}_v$ addition and subtraction operations to compute the summation of $\bar{d}_v + 1$ inner messages and the $\bar{d}_v$ outer messages, respectively. The table lookup operations based on Γ_v result in $\bar{d}_v \cdot q_m$ comparisons. Consequently, the average computational complexity is $\bar{d}_v \cdot (q_m + 2)$ for one VN per iteration. In addition, there is only one extra comparison operation required for the mapping function Q_e because the summation computed by Φ_v can be utilized for the hard decision. We summarize the computational complexity of the MIM quantized decoders in TABLE II. Although extra bit widths (q_c and q_v) are required for the computing functions, it is notable to point out that the choice of q_c and q_v merely impacts on the computational complexity in the design procedure, but not in the decoding process.

B. Memory Requirements

Apart from the computational complexity, the implementation complexity is also of significant importance for various applications. Due to table lookup operations adopted in the node updates, the memory requirements complexity of the MIM-LUT decoders and the MIM quantized decoders is mainly affected by the memory requirement for storing the LUTs. This depends on the number of LUTs used for decoding and the number of entries in each LUT. In [22]–[24], the tree-like structure of table lookup operations leads to the memory requirement for storing various LUTs for updating the nodes with different degrees. Each LUT has the same number of entries. Given q_m-bit precision of the messages, the number of entries in one LUT is 2^{2q_m}. According to [22], there are $2 \cdot \lfloor \log_2(d_{c,\max}) \rfloor$ and $2 \cdot \lfloor \log_2(d_{v,\max}) \rfloor$ LUTs required for the CN and the VN update, respectively. As a result, the memory requirement at each iteration for the tree-like structure is $2^{2q_m+1} \cdot \lfloor \log_2(d_{c,\max}) \rfloor$ for the CN update, and $2^{2q_m+1} \cdot \lfloor \log_2(d_{v,\max}) \rfloor$ for the VN update.

For our proposed MIM quantized decoders, we use the same LUTs for the reconstruction functions and threshold sets to update the nodes of different degrees. By considering

the q_m-bit outer messages, the LUTs of one reconstruction function and one threshold set have 2^{q_m} and $2^{q_m} - 1$ entries at each iteration, respectively. Therefore, the total memory requirement for the CN update of the MIM-QBP decoder with ϕ_c and Γ_c is $2^{q_m+1} - 1$ for each iteration. For the CN update of the MIM-QMS decoder, the function $f(\cdot)$ is totally determined by q_m. Thus we do not need to store the LUT for $f(\cdot)$. Moreover, the LUTs of the reconstruction functions (ϕ_v, ϕ_{ch}) and the threshold sets (Γ_v, Γ_{ch}) for the VN update have $2^{q_m+2} - 2$ entries in total for each iteration. Since the same reconstruction functions ϕ_v and ϕ_{ch} are used for the mapping function Q_e, there is only one entry in the threshold set Γ_e for hard decision at each iteration. We summarize the memory requirements of various decoders in TABLE III. Compared to the MIM-LUT decoders [22]–[24], our proposed MIM quantized decoders significantly reduce the memory requirement for the decoding process. More importantly, the memory demand for the MIM quantized decoders is only affected by the precision of the outer message, which is independent of the degree distributions of the LDPC codes. This benefits the decoding of the LDPC codes especially for those with high code rates. This is because the LDPC codes with high code rates incorporate very high node degrees, resulting in an undesirable growth in memory demand for the MIM-LUT decoders.

C. Performance Comparison

We further investigate the frame error rate (FER) performance of the proposed MIM quantized decoders via Monte-Carlo simulations and compared to that of the MIM-LUT decoders. Assume that binary LDPC codewords are modulated by binary phase-shift keying (BPSK) and transmitted over the AWGN channels. For convenience, we use the tuples (q_m, q_c, q_v) and (q_m, q_v) to represent the precision of the MIM-QBP decoder and the MIM-QMS decoder, respectively. Note that we consider the low precision quantization, i.e., $q_m = 3$ and 4 for practical implementations. The floating-point precision is denoted by "∞". The irregular LDPC code is selected from the IEEE 802.11n standard [42] with length 1296 and code rate $1/2$. The degree distributions of the simulated code are

$$\rho(x) = 0.8140x^6 + 0.1860x^7,$$
$$\theta(x) = 0.2558x + 0.3140x^2 + 0.0465x^3 + 0.3837x^{10}.$$

We summarize the design noise standard deviation σ_d for the associated MIM quantized decoders in TABLE IV. At least 300 error frames are collected for each simulated SNR. (See Appendix A in [43] for details of the constructed LUTs.)

TABLE IV: Noise Standard Deviation σ_d for MIM Quantized Decoders

(q_m, q_c, q_v) MIM-QBP, σ_d		(q_m, q_v) MIM-QMS, σ_d	
$(3, 12, 12)$	$(4, 12, 12)$	$(3, 12)$	$(4, 12)$
0.8778	0.9003	0.8501	0.8998

Example 1: Fig. 4 depicts the FER performance of the proposed MIM quantized decoders with the maximum number of iterations $I_{\max} = 50$. For comparison, we also present the FER performance of the 3-bit/4-bit min-LUT decoders [24], the $(4, 4, 12)$ RCQ decoder in [30], and the BP(∞) decoder. It is shown that the FER performance of our proposed $(3, 12, 12)$ and $(4, 12, 12)$ MIM-QBP decoder can approach that of the BP(∞) decoder by about 0.1 dB and 0.3 dB, respectively. For $E_b/N_0 < 2$ dB, the proposed $(3, 12, 12)$ MIM-QBP decoder has a performance gain of about 0.2 dB compared to the 3-bit min-LUT decoder. Furthermore, the $(4, 12, 12)$ MIM-QBP decoder performs better than both the 4-bit min-LUT decoder and the $(4, 4, 12)$ RCQ decoder when $E_b/N_0 < 2$ dB. In addition, the $(3, 12)$ MIM-QMS decoder achieves about 0.1 dB gain compared to the 3-bit min-LUT decoder and its FER performance is around 0.4 dB away from the BP(∞) decoder for $E_b/N_0 < 2$ dB. When $E_b/N_0 \geq 2$ dB, the FER performance of the $(3, 12)$ MIM-QMS decoder approaches that of the $(3, 12, 12)$ MIM-QBP decoder and the BP(∞) decoder with a difference of less than 0.1 dB and 0.3 dB, respectively. It is notable that the $(4, 12)$ MIM-QMS decoder can surpass the $(4, 4, 12)$ RCQ decoder and achieves the FER performance of less than 0.1 dB away from the BP(∞) decoder when $E_b/N_0 < 2.2$ dB. When $E_b/N_0 \geq 2.2$ dB, the $(4, 12)$ MIM-QMS decoder even outperforms the BP(∞) decoder by about 0.1 dB.

Due to the existing of the degree-2 VNs in the code graph, the BP(∞) decoder and the MIM-QBP decoders have higher error floor than the MIM-QMS decoders for the simulated LDPC code. This is because the cycles confined among the degree-2 VNs result in low-weight codewords, which become the most detrimental objects for the BP updating rule [44]. Similar phenomena are also observed in [20] and [30], which show that using the MIM quantization is capable of mitigating the effect of certain low-weight codewords or trapping sets.

We also investigate the convergence behaviors of the proposed MIM quantized decoders in decoding the same LDPC code at $E_b/N_0 = 2.2$ dB over AWGN channels. Fig. 5 shows the FER-versus-iteration performance for the proposed MIM quantized decoders. It can be seen that the $(3, 12, 12)$ MIM-QBP decoder and the $(3, 12)$ MIM-QMS decoder outperform the 3-bit min-LUT decoder for all decoding iterations. For decoding iteration ≤ 20, the $(4, 12, 12)$ MIM-QBP decoder has a convergence speed close to that of the BP(∞) decoder, and it achieves a faster convergence speed compared to both the min-LUT decoder and the RCQ decoder with the 4-bit precision. Moreover, the $(4, 12)$ MIM-QMS decoder outperforms the $(4, 4, 12)$ RCQ decoder and converges almost as fast as that of the 4-bit min-LUT decoder for decoding iteration ≤ 30. When decoding iteration ≤ 30, the $(4, 12)$ MIM-QMS decoder shows
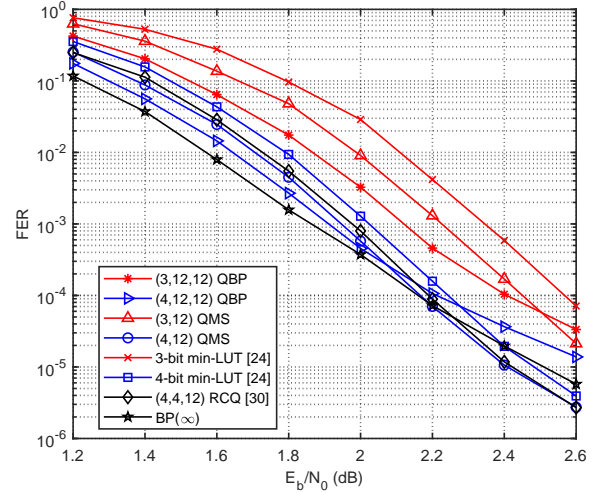
Fig. 4: FER of the length-1296 IEEE 802.11n LDPC code [42] with code rate $1/2$ over AWGN channels, $I_{\max} = 50$.
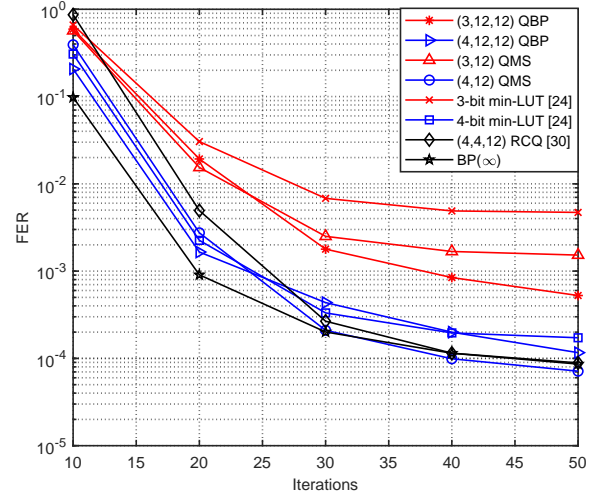
Fig. 5: FER-versus-iteration performance of the length-1296 IEEE 802.11n LDPC code [42] with code rate $1/2$ over AWGN channels, $E_b/N_0 = 2.2$ dB.

almost the same FER performance compared to the BP(∞) decoder for the same decoding iterations.

V. THE LAYERED SCHEDULE FOR MIM-QMS DECODER

Compared to the MIM-LUT decoders, the proposed MIM-QMS decoder achieves a better trade-off between the error performance and the decoding complexity. However, there are still a relative large number of iterations required to obtain a desirable error performance due to the flooding schedule [31] used for decoding. To reduce the number of decoding iterations, the layered schedule has been adopted in many designs, e.g., [33]–[37], to achieve a faster convergence speed of the decoder. However, we observe that there is a mismatch between the reconstruction functions and the threshold sets designed by the MIM-DE with those needed for layered decoding. More specifically, the MIM-DE is derived originally based on the flooding schedule at each iteration, and it constructs the LUTs of the reconstruction functions and threshold sets by assuming that all outer messages sent to the CN or VN have

the same pmfs. However, the pmfs of the outer messages need to be updated sequentially at each iteration when the layered schedule is adopted.

Motivated by this observation, in this section, we propose the design method of the MIM layered QMS (MIM-LQMS) decoder for the QC-LDPC codes. Note that we consider QC-LDPC codes because they are favorable for the implementation with layered scheduling due to the regularity in the parity check matrices. Particularly, we derive the layered MIM-DE (LMIM-DE) to construct the LUTs of the reconstruction functions and threshold sets for each layer and each iteration without mismatch. We call those reconstruction functions and threshold sets the layer-specific parameters. However, this design method increases the memory requirement of the MIM-LQMS decoder because the number of LUTs grows with the number of layers. To reduce the memory requirement, we further introduce an optimization method for the MIM-LQMS decoder by considering the iteration-specific reconstruction functions and threshold sets instead of the layer-specific ones.

A. The Design of MIM-LQMS Decoder

Let H be the parity-check matrix of size $M \times N$ for an LDPC code, which can be represented by a Tanner graph [32] containing N VNs and M CNs. We define $\mathcal{N}(c_m) \backslash v_n$ as the set of VNs connected to CN c_m ($1 \leq m \leq M$) with the node v_n excluded and define $\mathcal{M}(v_n) \backslash c_m$ as the set of CNs connected to VN v_n ($1 \leq n \leq N$) with the node c_m excluded. The QC-LDPC codes are a class of the structured LDPC codes which can be constructed from an $M_b \times N_b$ compact matrix H_b. To obtain H, each element in H_b is expanded by either a circulant permutation matrix [45] or an all-zero matrix of size $Z \times Z$, where Z is known as the lifting factor [3]. Therefore, we have $Z = M/M_b = N/N_b$.

Compared to the flooding schedule, the layered schedule updates the message in a sequential order to achieve a faster convergence speed. In this paper, we focus on the vertical layered schedule [35] because it inherently has short critical paths and can minimize the loading latency of the messages [36]. Denote the C2V message and the V2C message between a node $c_m \in \mathcal{M}(v_n)$ and v_n by S_{mn} and R_{nm}, respectively. At each layer and each iteration, the vertical layered decoder based on the MS algorithm performs [36]

$$S_{mn} = \prod_{v_{n'} \in \mathcal{N}(c_m) \backslash v_n} sgn(R_{n'm}) \cdot \min_{v_{n'} \in \mathcal{N}(c_m) \backslash v_n} |R_{n'm}|,$$
(19)

$$R_{nm} = L(v_n) + \sum_{c_{m'} \in \mathcal{M}(v_n) \backslash c_m} S_{m'n},$$
(20)

where $L(v_n)$ is the channel output message for the node v_n. At the end of each iteration, a posterior information of the node v_n, denoted by L_n, is given by

$$L_n = R_{nm} + S_{mn}.$$
(21)

The QC-LDPC codes are well suited to the vertical layered schedule by partitioning the VNs into N_b groups, so-called the layers. Each layer consists of Z VNs and the VNs of layer
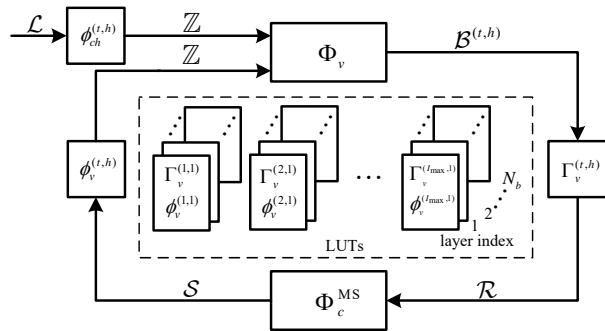
Fig. 6: The architecture of MIM-LQMS decoder.

h ($h = 1, 2, \ldots, N_b$) are corresponding to the $(h \cdot Z - Z + 1)$-th to ($h \cdot Z$)-th consecutive columns in H. Note that any two columns in each layer have at most one nonzero element in the same row. In this way, implementing the vertical layered schedule over H is equivalent to performing the decoding on the base matrix H_b, where layer h is corresponding to the h-th column in H_b. Moreover, we consider the decoding proceeds layer after layer in ascending order of layer index h.

In Fig. 6, we present the architecture of the MIM-LQMS decoder designed based on H_b. For convenience, we use the superscript (t, h) to represent the parameter at layer h and iteration t. Compared to the MIM-QMS decoder, the MIM-LQMS decoder has a similar framework to the MIM-QMS decoder but differs in the sense that the reconstruction functions and the threshold sets vary with both the associated layer and iteration. This is because the pmfs of the messages are different at each layer and each iteration due to the sequential updating order adopted by the layered scheduling. We denote the reconstruction function for the channel output and the VN update by $\phi_{ch}^{(t,h)}$ and $\phi_v^{(t,h)}$, respectively. We also denote the threshold set for the VN update at layer h and iteration t by $\Gamma_v^{(t,h)}$.

Similar to the case of MIM-QMS decoder, designing $\phi_{ch}^{(t,h)}$, $\phi_v^{(t,h)}$ and $\Gamma_v^{(t,h)}$ for the MIM-LQMS decoder requires the pmfs between the coded bits and the outer messages at each layer for different iterations. Since the MIM-DE is only capable of tracking the evolution of the pmfs under the flooding schedule, it is necessary to develop the LMIM-DE to obtain the pmfs at each layer and each iteration. Define $P_{R|X}^{(t,h)}$ and $P_{S|X}^{(t,h)}$ as the pmf of the V2C message R and C2V message S conditioned on the coded bit X at layer h and iteration t, respectively. In the following, we illustrate the design of the MIM-LQMS decoder based on the LMIM-DE in detail.

1) LMIM-DE at CN Update: At iteration t, a degree-i ($i \in \mathcal{D}_c$) CN connected to a VN in layer h receives the V2C messages from the layers that have already been updated (layer $h' < h$) at iteration t, and from the layers that were updated at iteration $t - 1$ (layer $h' > h$). Therefore, we consider the expected pmf of the V2C message received at layer h as

$$\tilde{P}_{R|X}^{(t,h)}(r|x) = \frac{1}{N_b - 1} \left(\sum_{h'=1}^{h-1} P_{R|X}^{(t,h')}(r|x) + \sum_{h'=h+1}^{N_b} P_{R|X}^{(t-1,h')}(r|x) \right).$$
(22)

Note that we have $P_{R|X}^{(0,h)} = P_{L|X}$ for $h = 1, 2, \ldots, N_b$. By replacing $P_{R|X}$ with $\tilde{P}_{R|X}^{(t,h)}$ in (9), the joint pmf at layer h and iteration t is

$$\tilde{P}_{\mathbf{R}|X}^{(t,h)}(\mathbf{r}|x) = \left(\frac{1}{2}\right)^{i-2} \sum_{\mathbf{x}:\oplus\mathbf{x}=x} \prod_{k=1}^{i-1} \tilde{P}_{R|X}^{(t,h)}(r_k|x_k). \quad (23)$$

Inspired by (18), the pmf of the C2V message at layer h and iteration t can be computed by

$$P_{S|X}^{(t,h)}(s|x) = \sum_{i \in D_c} \rho_i \cdot \sum_{\substack{\mathbf{r} \in \mathcal{R}^{i-1}, \\ \Phi_c^{\mathrm{MS}}(\mathbf{r})=s}} \tilde{P}_{\mathbf{R}|X}^{(t,h)}(\mathbf{r}|x). \quad (24)$$

2) LMIM-DE at VN Update: The VN update for the MIM-LQMS decoder consists of the reconstruction by the reconstruction functions $\phi_{ch}^{(t,h)}$ and $\phi_v^{(t,h)}$, the calculation with Φ_v according to (4), and the MIM quantization using the DP method. Based on $P_{S|X}^{(t,h)}$ and $P_{L|X}$, the reconstruction functions $\phi_{ch}^{(t,h)}$ and $\phi_v^{(t,h)}$ can be constructed accordingly by (2). Due to the DE assumptions in [7], we assume that the C2V messages entering a degree-j ($j \in \mathcal{D}_v$) VN in the same layer are independent and identically distributed. Thus, the joint pmf at layer h and iteration t can be calculated by

$$P_{L,\mathbf{S}|X}^{(t,h)}(l, \mathbf{s}|x) = P_{L|X}(l|x) \prod_{k=1}^{j-1} P_{S|X}^{(t,h)}(s_k|x). \quad (25)$$

After reconstruction, the computing function Φ_v calculates the alphabet set $\mathcal{B}^{(t,h)}$ of the inner message for each realization $(l, \mathbf{s}) \in \mathcal{L} \times \mathcal{S}^{j-1}$. Define $P_{B|X}^{(t,h)}$ as the pmf of the inner message B at layer h and iteration t. For a realization $b \in \mathcal{B}^{(t,h)}$, we have

$$P_{B|X}^{(t,h)}(b|x) = \sum_{j \in D_v} \theta_j \cdot \sum_{\substack{(l,\mathbf{s}) \in \mathcal{L} \times \mathcal{S}^{j-1}, \\ \Phi_v(l,\mathbf{s})=b}} P_{L,\mathbf{S}|X}^{(t,h)}(l, \mathbf{s}|x). \quad (26)$$

With $\mathcal{B}^{(t,h)}$ and $P_{B|X}^{(t,h)}$, we perform the MIM quantization and obtain $P_{R|X}^{(t,h)}$ and $\Gamma_v^{(t,h)}$:

$$[P_{R|X}^{(t,h)}, \Gamma_v^{(t,h)}] = \mathbf{DP}(\mathcal{B}^{(t,h)}, P_{B|X}^{(t,h)}). \quad (27)$$

By implementing LMIM-DE, the pmfs $P_{R|X}^{(t,h)}$ and $P_{S|X}^{(t,h)}$ are updated at each layer and each iteration to construct $\phi_v^{(t,h)}$, $\phi_{ch}^{(t,h)}$, and $\Gamma_v^{(t,h)}$, respectively. These layer-specific parameters need to be stored as the LUTs for decoding.

B. Iteration-Specific Optimization

As shown in the previous sub-section, the main disadvantage of the MIM-LQMS decoder is that storing the LUTs of layer-specific parameters significantly increases the memory requirement especially for a large number of layers. To reduce the memory demand for storing the LUTs, we optimize the proposed LMIM-DE and design the iteration-specific reconstruction functions and threshold sets for the MIM-LQMS decoder. More specifically, we first conduct the layer-specific design to obtain the pmf $P_{S|X}^{(t,h)}$. After that, a post-processing is performed, where we consider the expected pmf of the C2V

Algorithm 2 The Iteration-Specific Design Flow of MIM-LQMS Decoder

---

Input: σ_d, $\rho(x)$, $\theta(x)$, q_m, q_v, $I_{\max}$.
Output: Γ_{ch}, $\Gamma_e^{(t)}$, $\Gamma_v^{(t)}$, $\phi_v^{(t)}$, $\phi_{ch}^{(t)}$.

1: Compute $P_{L|X}$ and Γ_{ch} with MIM quantization in [29]
2: $P_{R|X}^{(0,h)} = P_{L|X}$, $h = 1, 2, \ldots, N_b$
3: **for** $t = 1 : I_{\max}$ **do**
4:     **for** $h = 1 : N_b$ **do**
5:         Compute $\tilde{P}_{R|X}^{(t,h)}$ and $P_{S|X}^{(t,h)}$ using (22) and (24), respectively
6:         Construct $\phi_{ch}^{(t,h)}$ and $\phi_v^{(t,h)}$ based on (2)
7:         Calculate $\mathcal{B}^{(t,h)}$ and $P_{B|X}^{(t,h)}$ using (4) and (26), respectively
8:         $[P_{R|X}^{(t,h)}, \Gamma_v^{(t,h)}] = \mathbf{DP}(\mathcal{B}^{(t,h)}, P_{B|X}^{(t,h)})$
9:     **end for**
10:     Compute $\tilde{P}_{S|X}^{(t)}$ using (28)
11:     Construct $\phi_{ch}^{(t)}$ and $\phi_v^{(t)}$ based on (2) with $\tilde{P}_{S|X}^{(t)}$
12:     Construct $\Gamma_v^{(t)}$ and $\Gamma_e^{(t)}$ using (4), (15), and (16) with $\tilde{P}_{S|X}^{(t)}$
13:     **for** $h = 1 : N_b$ **do**
14:         Perform steps 5 and 7 with $\phi_{ch}^{(t)}$ and $\phi_v^{(t)}$
15:         Update $P_{R|X}^{(t,h)}$ based on $\mathcal{B}^{(t,h)}$, $P_{B|X}^{(t,h)}$, and $\Gamma_v^{(t)}$
16:     **end for**
17: **end for**

---

message from all N_b layers at iteration t as

$$\tilde{P}_{S|X}^{(t)}(s|x) = \frac{1}{N_b} \sum_{h=1}^{N_b} P_{S|X}^{(t,h)}(s|x). \quad (28)$$

By replacing $P_{S|X}^{(t,h)}$ with $\tilde{P}_{S|X}^{(t)}$ in (2), we can construct the iteration-specific reconstruction functions $\phi_{ch}^{(t)}$ and $\phi_v^{(t)}$, and the threshold set $\Gamma_v^{(t)}$ following the steps 7-9 in **Algorithm 1**. Then the layer-specific design need to be conducted again to update the pmf $P_{R|X}^{(t,h)}$ for each layer with the iteration-specific parameters $\phi_{ch}^{(t)}$, $\phi_v^{(t)}$, and $\Gamma_v^{(t)}$. Note that $\phi_v^{(t)}$, $\phi_{ch}^{(t)}$, and $\tilde{P}_{S|X}^{(t)}$ can be directly used to determine the threshold set $\Gamma_e^{(t)}$ for the bit decision with $\{\Phi_v(l, \mathbf{s}) : (l, \mathbf{s}) \in \mathcal{L} \times \mathcal{S}^j\}$ in (26) and $q_m = 1$. We summarize the iteration-specific design flow of the MIM-LQMS decoder in **Algorithm 2**. By using the iteration-specific parameters, the MIM-LQMS decoder has the same memory requirement for storing the LUTs as that for the MIM-QMS decoder with the same precision. Similar to the MIM-QMS decoder, these LUTs of the iteration-specific parameters are fixed for the decoding process and do not change with the variation of the received SNRs.

In Fig. 7, we demonstrate the mutual information evolution of the MIM-LQMS decoders with both layer-specific and iteration-specific parameters for the irregular LDPC code in *Example* 1. For comparison, we present the mutual information evolution of the MIM-QMS decoder and the MIM-LQMS
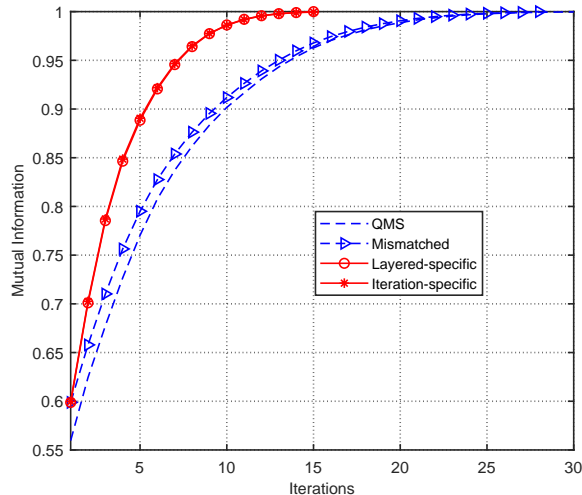
Fig. 7: The Comparison of Mutual Information Evolution.

decoder with mismatched parameters. Note that the simulated SNR for all decoders are selected as $E_b/N_0 = 1.4$ dB. For the MIM-LQMS decoder with the layer-specific parameters, we present the average mutual information of N_b layers at each iteration. Simulation results show that the mismatched reconstruction functions and threshold sets incur a significant loss of mutual information. However, the MIM-LQMS decoders reduce the number of iterations to converge into half compared to the MIM-QMS decoder, which is consistent with the results shown in [33]–[35]. More importantly, the MIM-LQMS decoder with iteration-specific parameters has nearly the same convergence speed with neglectable loss of the mutual information compared to the layer-specific design. This indicates that the MIM-LQMS decoder with iteration-specific parameters has a high potential for practical implementation.

C. Simulation Results of MIM-LQMS Decoders

In this section, we evaluate the FER performance of the $(3, 12)$ and $(4, 12)$ MIM-LQMS decoders for both the AWGN channels and the fast fading channels via Monte-Carlo simulations. We assume the BPSK modulation is adopted and the j-th received signal of the fading channel is given by

$$y_j = h_j \cdot x_j + n_j,$$

where h_j is the fading coefficient following $h_j \sim \mathcal{N}(0, 1)$, and n_j is the AWGN with $n_j \sim \mathcal{N}(0, N_0)$. Note that N_0 is the single-sided noise power spectral density. We consider three examples, where the simulated LDPC codes in each example have short, moderate, and long block lengths, respectively. More specifically, a length-560 5G LDPC code is considered in *Example* 2, which has a code rate of $1/2$ after rate matching [46]. In *Example* 3, the length-1296 LDPC codes are selected from the IEEE 802.11n standard [42] with code rates $1/2, 2/3$, and $5/6$, respectively. Furthermore, we also consider the length-17664 IEEE 802.3ca standard LDPC code [47] in *Example* 4, which has a code rate of 0.826 without puncturing and shortening. For comparison, we include the FER performance of the floating-point layered BP (LBP) decoder in [35], the 4-bit layered normalized min-sum (LNMS) decoder [12], and the MIM-QMS decoders as references. We

set the maximum number of iterations $I_{\max} = 15$ for all layered decoders, and $I_{\max} = 30$ for the MIM-QMS decoders. Following the notations in Section IV-C, we summarize the degree distributions and N_b of the LDPC codes and also present σ_d for the corresponding MIM quantized decoders in TABLE V. (See Appendix B in [43] for details of the constructed LUTs.)
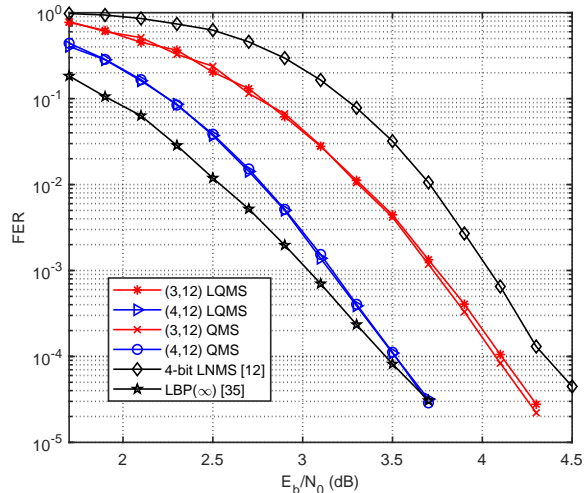
Fig. 8: FER of the length-560 5G LDPC code [46] with code rate $1/2$ over the AWGN channels.
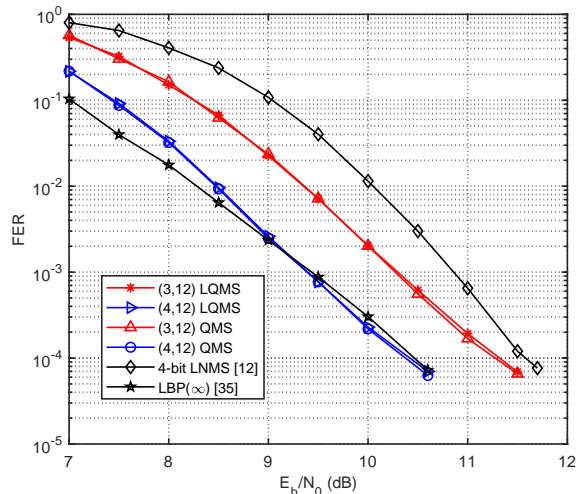
Fig. 9: FER of the length-560 5G LDPC code [46] with code rate $1/2$ over the fast fading channels.

Example 2: Fig. 8 and Fig. 9 show the FERs of the length-560 5G LDPC code [46] over the AWGN channels and the fast fading channels, respectively. The scaling factor of the LNMS decoder is optimized as 0.7. We can see that the $(3, 12)$ and $(4, 12)$ MIM-LQMS decoders have almost identical FER performance compared to the MIM-QMS decoders with the same precision. The $(3, 12)$ MIM-LQMS decoder outperforms the 4-bit LNMS decoder by about 0.4 dB and 0.6 dB at FER $= 10^{-2}$, over the AWGN channels and the fast fading channels, respectively. Moreover, the $(4, 12)$ MIM-LQMS decoder achieves performance close to the LBP(∞) decoder within 0.25 dB and surpasses the 4-bit LNMS decoder by about 0.9

TABLE V: Degree Distributions, N_b, and Noise Standard Deviation σ_d for MIM quantized Decoders

	Code rates	Degree distributions $(\rho(x), \theta(x))$	N_b	MIM-LQMS, σ_d		MIM-QMS, σ_d	
				(3, 12)	(4, 12)	(3, 12)	(4, 12)
Example 2	1/2	$\rho(x) = 0.1039x^3 + 0.1948x^4 + 0.2338x^5 + 0.2078x^7 + 0.2597x^9$ $\theta(x) = 0.1039 + 0.0260x + 0.1169x^2 + 0.2078x^3 + 0.1299x^4 + 0.1818x^6 + 0.2338x^8$	22	0.9821	1.0042	0.9837	1.0115
Example 3	1/2	$\rho(x) = 0.8140x^6 + 0.1860x^7$ $\theta(x) = 0.2558x + 0.3140x^2 + 0.0465x^3 + 0.3837x^{10}$	24	0.8362	0.8750	0.8379	0.8761
	2/3	$\rho(x) = x^{10}$ $\theta(x) = 0.1591x + 0.4091x^2 + 0.1591x^6 + 0.2727x^7$		0.6889	0.7112	0.6914	0.7112
	5/6	$\rho(x) = 0.7412x^{20} + 0.2588x^{21}$ $\theta(x) = 0.0706x + 0.1765x^2 + 0.7529x^3$		0.5413	0.5502	0.5420	0.5511
Example 4	0.826	$\rho(x) = 0.0800x^{21} + 0.9200x^{22}$ $\theta(x) = 0.5455x^2 + 0.3709x^3 + 0.0400x^{10} + 0.0436x^{11}$	69	0.5528	0.5599	0.5529	0.5599

dB over the AWGN channels. For the fast fading channels, the $(4, 12)$ MIM-LQMS decoder achieves a performance gain of about 1.5 dB at FER $= 10^{-2}$ compared to the 4-bit LNMS decoder, and it approaches the performance of the LBP(∞) decoder within 0.1 dB at FER $= 10^{-4}$.

Example 3: Fig. 10 and Fig. 11 illustrate the FERs of the proposed MIM-LQMS decoders for the length-1296 IEEE 802.11n LDPC codes with different code rates over the AWGN channels and the fast fading channels, respectively. The scaling factor of the LNMS decoder is set to 0.8. We observe that the $(3, 12)$ and $(4, 12)$ MIM-LQMS decoders have nearly the same FER performance compared to the MIM-QMS decoders with the same precision. More importantly, the performance gaps between the MIM-LQMS decoders and the LBP(∞) decoder reduces with the increase of the code rates. For the AWGN channels, the FER performance of the $(3, 12)$ MIM-LQMS decoder is about 0.4 dB away from that of the LBP(∞) decoder for code rate 1/2, while it approaches the FER of the LBP(∞) decoder within 0.3 and 0.2 dB for the code rates of 2/3 and 5/6, respectively. Moreover, the performance gap between the $(4, 12)$ MIM-LQMS decoder and the LBP(∞) decoder is less than 0.2 and 0.1 dB compared with that of the LBP(∞) decoder for code rates 1/2 and 2/3, respectively. For code rate 5/6, the $(4, 12)$ MIM-LQMS decoder has almost the same FER performance compared to the LBP(∞) decoder. In addition, compared to the 4-bit LNMS decoder, the $(4, 12)$ MIM-LQMS decoder has a performance gain of about 0.4 dB, 0.3 dB and 0.2 dB at FER $= 10^{-2}$, for the code rates of 1/2, 2/3, and 5/6, respectively. The $(3, 12)$ MIM-LQMS decoder can still obtain 0.1 dB gain at FER $= 10^{-2}$ for the code rates of 1/2 and 2/3 when compared with the 4-bit LNMS decoder.

With regard to the fast fading channels, the $(3, 12)$ MIM-LQMS decoder approaches the FER performance of the LBP(∞) decoder by about 1 dB, 0.7 dB, and 0.6 dB for the code rates of 1/2, 2/3 and 5/6, respectively. Furthermore, the $(4, 12)$ MIM-LQMS decoder has the FER performance of less than 0.3 dB and 0.15 dB away compared to the LBP(∞) decoder for code rates 1/2 and 2/3, respectively. For code rate 5/6, the $(4, 12)$ MIM-LQMS decoder achieves nearly the same FER performance as the LBP(∞) decoder at FER $= 10^{-2}$, and outperforms the LBP(∞) decoder by 0.25 dB at FER $= 10^{-4}$. We also observe that the $(4, 12)$ MIM-LQMS decoder outperforms the 4-bit LNMS decoder by about 0.6 dB

Fig. 10: FER of the length-1296 IEEE 802.11n LDPC codes [42] with code rates 1/2, 2/3, and 5/6 over the AWGN channels.

Fig. 11: FER of the length-1296 IEEE 802.11n LDPC codes [42] with code rates 1/2, 2/3, and 5/6 over the fast fading channels.

for all code rates, and even the $(3, 12)$ MIM-LQMS decoder can perform slightly better than the 4-bit LNMS decoder for all code rates in the low-to-moderate SNR region.

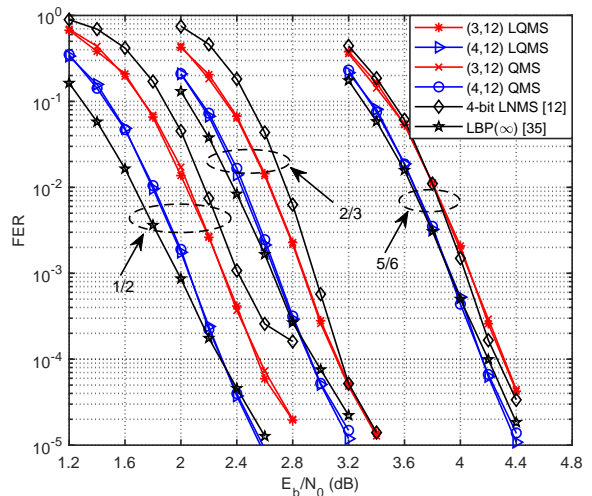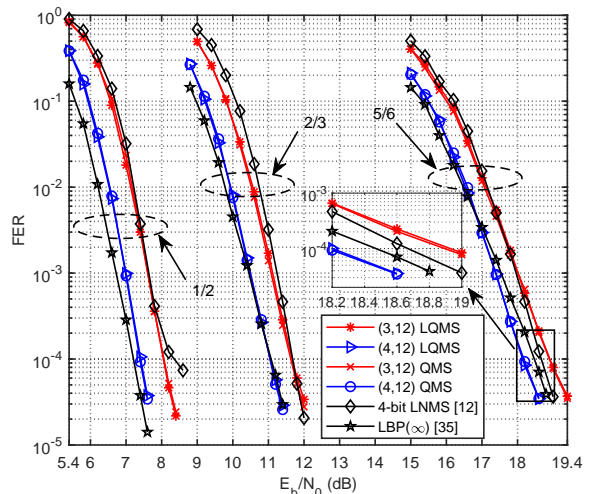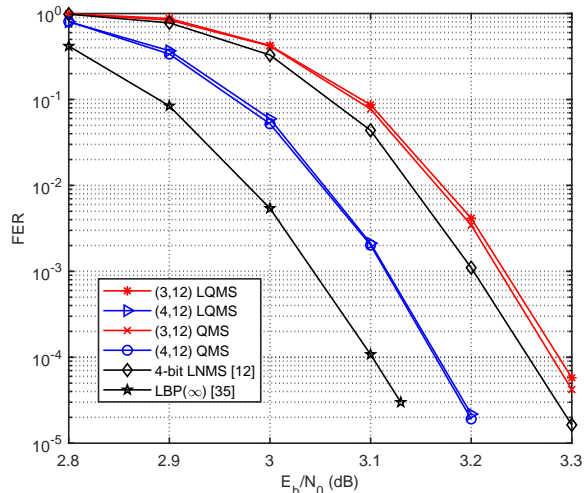Fig. 12: FER of the length-17664 IEEE 802.3ca LDPC code [47] with code rate 0.826 over the AWGN channels.

Fig. 13: FER of the length-17664 IEEE 802.3ca LDPC code [47] with code rate 0.826 over the fast fading channels.

Example 4: In Fig. 12 and Fig. 13, we compare the FER performance of the proposed MIM-LQMS decoders to that of various decoders over the AWGN channels and the fast fading channels. Note that we optimize the scaling factor of the LNMS decoder as 0.65. It is shown that the $(3, 12)$ and $(4, 12)$ MIM-LQMS decoders have almost the same FER performance as the $(3, 12)$ and $(4, 12)$ MIM-QMS decoders, respectively. The performance gap between the $(3, 12)$ MIM-LQMS decoder and the LBP(∞) decoder is about 0.2 dB over the AWGN channels, and about 0.8 dB over the fast fading channels. Furthermore, the $(4, 12)$ MIM-LQMS decoder can approach the FER performance of the LBP(∞) decoder within 0.1 and 0.3 dB over the AWGN channels and the fast fading channels, respectively. Compared to the 4-bit LNMS decoder, the $(3, 12)$ MIM-LQMS decoder shows negligible performance loss over the AWGN channel, and it achieves the performance gain of 0.1 dB for the fast fading channels. In addition, the $(4, 12)$ MIM-LQMS decoder also performs better than the 4-bit LNMS decoder by about 0.5 dB over the fast fading channel.

VI. CONCLUSION

In this paper, we have proposed a generalized framework of the MIM quantized decoding for the LDPC codes to improve the error performance of the MIM-LUT decoding. The proposed decoding method can be designed based on either the BP or the MS algorithm, and implemented by only simple mappings and fixed-point additions. Particularly, we have proposed the MIM-DE to construct the LUTs of the reconstruction and quantization mappings for the node updates. For practical concerns, we discussed the computational complexity and the memory requirements of the MIM quantized decoders. To speed up the convergence of the decoder for QC-LDPC codes, we developed the LMIM-DE and adopted it to design the LUTs for the MIM-LQMS decoder. We also presented an optimization method to reduce the number of LUTs required for the VN update of the MIM-LQMS decoder, which achieves less memory requirement. Simulation results
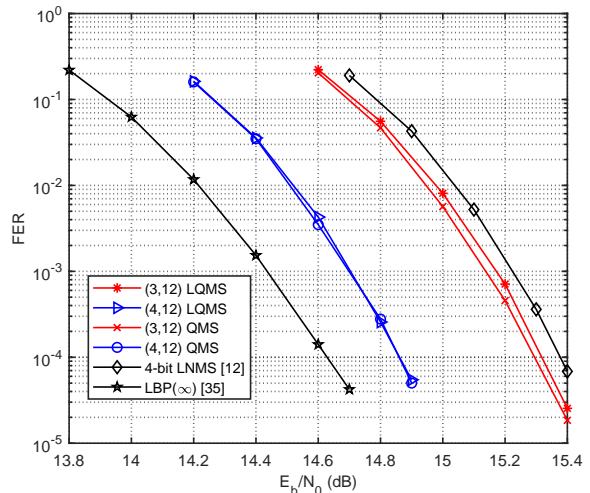
show that the MIM quantized decoders with 3-bit and 4-bit precisions outperform the state-of-the-art LUT decoders with the same message precision with respect to the error performance in the low-to-moderate SNR regions and the convergence speed for small decoding iterations. Moreover, the 4-bit MIM-LQMS decoder achieves the FER performance of less than 0.3 dB away from the floating-point LBP decoder in the moderate-to-high SNR regions.

REFERENCES

[1] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.

[2] Y. Kou, S. Lin, and M. P. C. Fossorier, "Low-density parity-check codes based on finite geometries: a rediscovery and new results," *IEEE Trans. Inf. Theory*, vol. 47, no. 7, pp. 2711–2736, Nov. 2001.

[3] D. G. M. Mitchell, M. Lentmaier, and D. J. Costello, "Spatially coupled LDPC codes constructed from protographs," *IEEE Trans. Inf. Theory*, vol. 61, no. 9, pp. 4866–4889, Sep. 2015.

[4] Y. Xie, L. Yang, P. Kang, and J. Yuan, "Euclidean geometry-based spatially coupled LDPC codes for storage," *IEEE J. Select. Areas Commun.*, vol. 34, no. 9, pp. 2498–2509, Sep. 2016.

[5] P. Chen, K. Cai, and S. Zheng, "Rate-adaptive protograph LDPC codes for multi-level-cell NAND flash memory," *IEEE Commun. Lett.*, vol. 22, no. 6, pp. 1112–1115, Jun. 2018.

[6] M. Nasseri, X. Xiao, B. Vasić, and S. Lin, "Globally coupled finite geometry and finite field LDPC coding schemes," *IEEE Trans. Veh. Technol.*, vol. 70, no. 9, pp. 9207–9216, Aug. 2021.

[7] T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.

[8] B. Kurkoski, P. Siegel, and J. Wolf, "Joint message-passing decoding of LDPC codes and partial-response channels," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1410–1422, Jun. 2002.

[9] Y. Fang, Y. L. Guan, G. Bi, L. Wang, and F. C. M. Lau, "Rate-compatible root-protograph LDPC codes for quasi-static fading relay channels," *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 2741–2747, Apr. 2016.

[10] M. Shakiba-Herfeh, A. K. Tanc, and T. M. Duman, "LDPC code design for fading interference channels," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2374–2385, Mar. 2019.

[11] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inf. Theory*, vol. 45, no. 2, pp. 399–431, Mar. 1999.

[12] J. Chen, A. Dholakia, E. Eleftheriou, M. P. C. Fossorier, and Xiao-Yu Hu, "Reduced-complexity decoding of LDPC codes," *IEEE Trans. Commun.*, vol. 53, no. 8, pp. 1288–1299, Aug. 2005.

[13] V. Savin, "Self-corrected min-sum decoding of LDPC codes," in *Proc. IEEE Int. Symp. on Inf. Theory*, Jul. 2008, pp. 146–150.

[14] M. P. C. Fossorier, "Iterative reliability-based decoding of low-density parity check codes," *IEEE J. Select. Areas Commun.*, vol. 19, no. 5, pp. 908–917, May. 2001.

[15] N. Varnica, M. P. C. Fossorier, and A. Kavcic, "Augmented belief propagation decoding of low-density parity check codes," *IEEE Trans. Commun.*, vol. 55, no. 7, pp. 1308–1317, Jul. 2007.

[16] P. Kang, Y. Xie, L. Yang, and J. Yuan, "Reliability-based windowed decoding for spatially coupled LDPC codes," *IEEE Commun. Lett.*, vol. 22, no. 7, pp. 1322–1325, Jul. 2018.

[17] P. Kang, Y. Xie, L. Yang, and J. Yuan, "Enhanced quasi-maximum likelihood decoding based on 2D modified min-sum algorithm for 5G LDPC codes," *IEEE Trans. Commun.*, vol. 68, no. 11, pp. 6669–6682, Nov. 2020.

[18] T. T. Nguyen-Ly, V. Savin, K. Le, D. Declercq, F. Ghaffari, and O. Boncalo, "Analysis and design of cost-effective, high-throughput LDPC decoders," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 3, pp. 508–521, Mar. 2018.

[19] F. Ghaffari, K. Le, and D. Declercq, "The probabilistic finite alphabet iterative decoder for low-density parity-check codes," in *2019 17th IEEE Int. New Circuits Syst. Conf.*, Jun. 2019, pp. 1–4.

[20] F. J. C. Romero and B. M. Kurkoski, "LDPC decoding mappings that maximize mutual information," *IEEE J. Select. Areas Commun.*, vol. 34, no. 9, pp. 2391–2401, Sep. 2016.

[21] J. Lewandowsky and G. Bauch, "Information-optimum LDPC decoders based on the information bottleneck method," *IEEE Access*, vol. 6, pp. 4054–4071, Jan. 2018.

[22] M. Stark, J. Lewandowsky, and G. Bauch, "Information-bottleneck decoding of high-rate irregular LDPC codes for optical communication using message alignment," *Appl. Sci.*, vol. 8, no. 10, Oct. 2018.

[23] M. Stark, L. Wang, G. Bauch, and R. D. Wesel, "Decoding rate-compatible 5G-LDPC codes with coarse quantization using the information bottleneck method," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 646–660, May. 2020.

[24] M. Meidlinger, G. Matz, and A. Burg, "Design and decoding of irregular LDPC codes based on discrete message passing," *IEEE Trans. Commun.*, vol. 68, no. 3, pp. 1329–1343, Mar. 2020.

[25] B. M. Kurkoski and H. Yagi, "Quantization of binary-input discrete memoryless channels," *IEEE Trans. Inf. Theory*, vol. 60, no. 8, pp. 4544–4552, Aug. 2014.

[26] T. Cover and J. Thomas, *Elements of Information Theory*. Wiley, 2012.

[27] X. He, K. Cai, and Z. Mei, "On mutual information-maximizing quantized belief propagation decoding of LDPC codes," in *Proc. IEEE Global Commun. Conf.*, Dec. 2019, pp. 1–6.

[28] X. He, K. Cai, and Z. Mei, "Mutual information-maximizing quantized belief propagation decoding of regular LDPC codes," *arXiv*, Jul. 2020. [Online]. Available: https://arxiv.org/abs/1904.06666

[29] X. He, K. Cai, W. Song, and Z. Mei, "Dynamic programming for sequential deterministic quantization of discrete memoryless channels," *IEEE Trans. Commun.*, doi:10.1109/TCOMM.2021.3062838.

[30] L. Wang, R. D. Wesel, M. Stark, and G. Bauch, "A reconstruction-computation-quantization (RCQ) approach to node operations in LDPC decoding," in *Proc. IEEE Global Commun. Conf.*, Dec. 2020, pp. 1–6.

[31] F. R. Kschischang and B. J. Frey, "Iterative decoding of compound codes by probability propagation in graphical models," *IEEE J. Select. Areas Commun.*, vol. 16, no. 2, pp. 219–230, Feb. 1998.

[32] R. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inf. Theory*, vol. 27, no. 5, pp. 533–547, Sep. 1981.

[33] M. M. Mansour and N. R. Shanbhag, "High-throughput LDPC decoders," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 11, no. 6, pp. 976–996, Dec. 2003.

[34] K. Zhang, X. Huang, and Z. Wang, "High-throughput layered decoder implementation for quasi-cyclic LDPC codes," *IEEE J. Select. Areas Commun.*, vol. 27, no. 6, pp. 985–994, Aug. 2009.

[35] J. Zhang and M. P. C. Fossorier, "Shuffled iterative decoding," *IEEE Trans. Commun.*, vol. 53, no. 2, pp. 209–213, Feb. 2005.

[36] Z. Cui, Z. Wang, and X. Zhang, "Reduced-complexity column-layered decoding and implementation for LDPC codes," *IET Commun.*, vol. 5, pp. 2177–2186(9), Oct. 2011.

[37] C. A. Aslam, Y. L. Guan, K. Cai, and G. Han, "Informed fixed scheduling for faster convergence of shuffled belief-propagation decoding," *IEEE Commun. Lett.*, vol. 21, no. 1, pp. 32–35, Jan. 2017.

[38] P. Mohr, G. Bauch, F. Yu, and M. Li, "Coarsely quantized layered decoding using the information bottleneck method," in *Proc. IEEE Int. Commun. Conf.*, Jun. 2021, pp. 1–6.

[39] C. Kameni Ngassa, V. Savin, E. Dupraz, and D. Declercq, "Density evolution and functional threshold for the noisy min-sum decoder," *IEEE Trans. Commun.*, vol. 63, no. 5, pp. 1497–1509, May. 2015.

[40] J.-S. Lee and J. Thorpe, "Memory-efficient decoding of LDPC codes," in *Proc. IEEE Int. Symp. on Inf. Theory*, Sep. 2005, pp. 459–463.

[41] Y. Lee, B. Kim, J. Jung, and I. Park, "Low-complexity tree architecture for finding the first two minima," *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 62, no. 1, pp. 61–64, Jan. 2015.

[42] *IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks-Specific Requirements— Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Std. 802.11n, Oct. 2009.

[43] P. Kang, K. Cai, X. He, and J. Yuan, "Generalized mutual information-maximizing quantized decoding of LDPC codes with layered scheduling," *arXiv*, Jun. 2021. [Online]. Available: https://arxiv.org/abs/2011.13147

[44] T. Richardson, "Error floor of LDPC codes," in *Proc. 41st Annu. Allerton Conf. Commun., Control, Comput.*, Monticello, IL, Oct. 2003, pp. 1426–1435.

[45] W. Ryan and S. Lin, *Channel codes: classical and modern*. Cambridge University Press, 2009.

[46] 3GPP, "NR; Multiplexing and channel coding," Release 15, Technical Specification (TS) 38.212, 2017.

[47] *IEEE Standard for Ethernet Amendment 9: Physical Layer Specifications and Management Parameters for 25 Gb/s and 50 Gb/s Passive Optical Networks*, IEEE Std. 802.3ca, Jul. 2020.

APPENDIX A
CONSTRUCTED LUTS FOR MIM QUANTIZED DECODERS
WITH FLOODING SCHEDULE

A. The 3-bit MIM-QBP decoder in Example 1

TABLE VI: Reconstruction Functions ϕ_c, ϕ_v, and ϕ_{ch}

Iteration	$\phi_c(r)$, $r \in \mathcal{R}$							
	0	1	2	3	4	5	6	7
1	4	26	85	255	-255	-85	-26	-4
2	3	24	81	244	-255	-83	-24	-3
3	3	23	83	255	-252	-80	-22	-3
4	2	21	78	243	-255	-81	-22	-2
5	2	18	71	237	-255	-77	-19	-2
6	1	14	60	255	-102	-34	-9	-1
7	1	16	71	255	-196	-59	-14	-1
8	1	7	29	91	-255	-51	-9	-1
9	1	13	67	255	-210	-66	-14	-1
10	1	2	14	255	-16	-4	-1	-1
11	1	19	81	255	-246	-76	-18	-1
12	1	10	55	255	-121	-34	-7	-1
13	1	11	65	255	-182	-41	-7	-1
14	1	9	44	169	-255	-57	-9	-1
15	1	8	53	255	-160	-37	-6	-1
16	1	9	60	255	-255	-60	-9	-1
17	1	9	56	255	-166	-41	-8	-1
18	1	8	51	255	-156	-40	-7	-1
19	1	7	39	160	-255	-51	-8	-1
20	1	9	60	255	-255	-60	-9	-1
21	1	9	57	255	-174	-35	-5	-1
22	1	9	58	255	-255	-58	-9	-1
23	1	5	32	161	-255	-51	-8	-1
24	1	8	55	255	-233	-55	-8	-1
25	1	8	51	255	-160	-31	-4	-1
26	1	8	54	234	-255	-54	-8	-1
27	1	9	58	255	-255	-58	-9	-1
28	1	9	58	255	-255	-58	-9	-1
29	1	9	58	255	-255	-58	-9	-1
30	1	9	58	254	-255	-58	-9	-1
31	1	9	60	254	-255	-60	-9	-1
32	1	9	59	255	-255	-59	-9	-1
33	1	9	58	255	-255	-58	-9	-1
34	1	9	59	255	-255	-59	-9	-1
35	1	9	59	254	-255	-59	-9	-1
36	1	8	59	255	-255	-59	-8	-1
37	1	8	59	255	-255	-59	-8	-1
38	1	8	60	255	-255	-60	-8	-1
39	1	8	60	255	-255	-60	-8	-1
40	1	8	60	255	-255	-60	-8	-1
41	1	8	58	255	-249	-58	-8	-1
42	1	8	58	255	-251	-58	-8	-1
43	1	7	58	255	-255	-58	-7	-1
44	1	7	58	255	-255	-58	-7	-1
45	1	6	56	255	-255	-56	-6	-1
46	1	6	54	255	-255	-54	-6	-1
47	1	6	51	246	-255	-51	-6	-1
48	1	5	50	255	-255	-50	-5	-1
49	1	5	49	251	-255	-49	-5	-1
50	1	4	43	255	-255	-43	-4	-1

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$							
	0	1	2	3	4	5	6	7
1	62	35	21	8	0	-9	-28	-57
2	68	34	11	1	-10	-25	-42	-74
3	75	38	13	1	-11	-27	-46	-81
4	85	46	28	10	-5	-19	-45	-85
5	94	52	32	13	-1	-15	-44	-87
6	106	70	44	23	0	-23	-49	-93
7	102	54	27	10	-10	-27	-54	-102
8	106	57	29	0	-27	-50	-80	-123
9	162	95	47	12	-12	-47	-95	-162
10	127	82	37	1	-22	-42	-82	-127
11	166	97	52	15	-15	-52	-97	-166
12	124	65	32	9	-9	-32	-65	-124
13	128	67	33	14	-14	-33	-67	-128
14	128	68	35	15	-15	-35	-68	-128
15	136	72	35	14	-14	-35	-72	-136
16	170	99	51	15	-15	-51	-99	-170
17	170	98	53	16	-16	-53	-98	-170
18	170	98	52	15	-15	-52	-98	-170
19	170	99	53	16	-16	-53	-99	-170
20	170	97	50	14	-14	-50	-97	-170
21	170	101	55	16	-16	-55	-101	-170
22	170	97	51	15	-15	-51	-97	-170
23	170	102	55	16	-16	-55	-102	-170
24	170	97	50	15	-15	-50	-97	-170
25	170	102	55	16	-16	-55	-102	-170
26	170	98	51	15	-15	-51	-98	-170
27	170	99	51	15	-15	-51	-99	-170
28	170	99	51	15	-15	-51	-99	-170
29	170	99	51	16	-16	-51	-99	-170
30	170	99	51	16	-16	-51	-99	-170
31	170	97	50	16	-16	-50	-97	-170
32	170	98	50	16	-16	-50	-98	-170
33	170	98	51	16	-16	-51	-98	-170
34	170	97	51	16	-16	-51	-97	-170
35	170	97	50	16	-16	-50	-97	-170
36	170	97	50	16	-16	-50	-97	-170
37	170	97	50	16	-16	-50	-97	-170
38	170	97	49	16	-16	-49	-97	-170
39	170	96	49	16	-16	-49	-96	-170
40	170	96	50	16	-16	-50	-96	-170
41	170	96	50	16	-16	-50	-96	-170
42	170	96	49	16	-16	-49	-96	-170
43	170	96	49	16	-16	-49	-96	-170
44	170	96	49	16	-16	-49	-96	-170
45	170	96	47	14	-14	-47	-96	-170
46	170	96	47	14	-14	-47	-96	-170
47	170	96	47	13	-13	-47	-96	-170
48	170	96	47	13	-13	-47	-96	-170
49	170	96	47	13	-13	-47	-96	-170
50	170	98	49	13	-13	-49	-98	-170

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$							
	0	1	2	3	4	5	6	7
1-15	170	99	54	17	-17	-54	-99	-170
16	136	79	43	14	-14	-43	-79	-136
17	134	78	42	14	-14	-42	-78	-134
18	131	76	41	13	-13	-41	-76	-131
19	131	76	41	13	-13	-41	-76	-131
20	130	76	41	13	-13	-41	-76	-130
21	128	74	41	13	-13	-41	-74	-128
22	126	74	40	13	-13	-40	-74	-126
23	124	72	39	13	-13	-39	-72	-124
24	124	72	39	12	-12	-39	-72	-124
25	122	71	39	12	-12	-39	-71	-122
26	123	71	39	12	-12	-39	-71	-123
27	124	72	39	13	-13	-39	-72	-124
28	123	72	39	12	-12	-39	-72	-123
29	123	71	39	12	-12	-39	-71	-123
30	122	71	39	12	-12	-39	-71	-122
31	121	70	38	12	-12	-38	-70	-121
32	121	70	38	12	-12	-38	-70	-121
33	120	70	38	12	-12	-38	-70	-120
34	119	69	38	12	-12	-38	-69	-119
35	118	69	37	12	-12	-37	-69	-118
36	117	68	37	12	-12	-37	-68	-117
37	116	68	37	12	-12	-37	-68	-116
38	115	67	36	12	-12	-36	-67	-115
39	114	66	36	12	-12	-36	-66	-114
40	113	66	36	11	-11	-36	-66	-113
41	112	65	35	11	-11	-35	-65	-112
42	110	64	35	11	-11	-35	-64	-110
43	109	64	35	11	-11	-35	-64	-109
44	108	63	34	11	-11	-34	-63	-108
45	106	62	34	11	-11	-34	-62	-106
46	105	61	33	11	-11	-33	-61	-105
47	104	61	33	11	-11	-33	-61	-104
48	103	60	33	10	-10	-33	-60	-103
49	101	59	32	10	-10	-32	-59	-101
50	100	58	32	10	-10	-32	-58	-100

TABLE VII: Threshold sets Γ_c, Γ_v, and Γ_{ch}

Iteration	Γ_c						
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
1	94	175	279	485	-470	-252	-127
2	84	204	420	-472	-254	-159	-81
3	81	201	420	-479	-249	-152	-75
4	54	134	232	564	-318	-164	-69
5	48	123	222	448	-401	-182	-76
6	23	51	97	205	-203	-91	-38
7	37	118	239	1785	-240	-119	-45
8	29	77	188	-198	-86	-44	-20
9	7	46	152	1695	-160	-54	-18
10	7	13	39	-54	-27	-14	-8
11	7	43	167	1785	-168	-57	-23
12	28	85	190	1585	-192	-87	-30
13	33	103	220	1785	-221	-104	-34
14	31	101	213	1699	-214	-102	-38
15	27	95	209	1785	-210	-96	-28
16	7	39	149	1785	-156	-46	-14
17	7	35	127	1785	-129	-36	-13
18	7	33	123	1785	-124	-34	-12
19	7	31	121	-1785	-122	-32	-12
20	7	39	149	1785	-156	-46	-14
21	7	27	119	1785	-120	-30	-10
22	7	39	145	1785	-152	-46	-14
23	7	28	114	1691	-115	-29	-10
24	7	35	136	1741	-142	-41	-13
25	7	28	111	1595	-112	-29	-9
26	7	35	134	1764	-140	-41	-13
27	7	31	145	1785	-152	-38	-14
28	7	31	145	1785	-152	-38	-14
29	7	31	145	1785	-152	-38	-14
30	7	31	145	1782	-152	-38	-14
31	7	31	141	1782	-148	-38	-14
32	7	31	139	1785	-146	-38	-14
33	7	31	137	1785	-144	-38	-14
34	7	31	139	1785	-146	-38	-14
35	7	31	139	1587	-146	-38	-14
36	7	28	137	1589	-143	-34	-13
37	7	28	137	1589	-143	-34	-13
38	7	28	139	1590	-145	-34	-13
39	7	28	139	1590	-145	-34	-13
40	7	28	139	1590	-145	-34	-13
41	7	28	135	1525	-141	-34	-13
42	7	28	135	1527	-141	-34	-13
43	7	25	133	1531	-138	-30	-12
44	7	25	133	1531	-138	-30	-12
45	7	27	127	1531	-131	-31	-11
46	7	27	128	1531	-132	-31	-11
47	7	27	122	1318	-126	-31	-11
48	7	23	117	1326	-120	-26	-10
49	7	23	119	1317	-123	-26	-10
50	7	19	106	1319	-132	-21	-9

Iteration	Γ_v						
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
1	134	78	38	2	-33	-74	-131
2	138	77	35	-1	-38	-81	-143
3	146	81	38	2	-35	-78	-141
4	152	86	40	1	-38	-84	-152
5	145	75	28	-12	-50	-95	-164
6	158	86	38	-1	-43	-89	-159
7	177	105	57	15	-31	-85	-165
8	165	88	37	-2	-39	-85	-161
9	167	82	25	-22	-70	-127	-210
10	162	85	40	1	-39	-85	-161
11	177	93	37	-11	-56	-109	-191
12	185	97	42	-4	-54	-112	-197
13	194	109	56	7	-44	-101	-191
14	193	105	45	-7	-58	-117	-205
15	197	108	48	1	-47	-107	-196
16	156	83	36	-5	-44	-89	-160
17	158	84	36	-6	-44	-89	-158
18	156	88	45	6	-34	-82	-153
19	153	82	37	1	-36	-81	-152
20	151	80	35	-4	-45	-93	-163
21	154	83	37	1	-36	-82	-153
22	165	95	47	6	-34	-80	-149
23	152	81	37	-1	-37	-80	-151
24	148	78	35	-5	-44	-92	-161
25	149	80	36	1	-35	-79	-148
26	149	80	37	0	-36	-79	-148
27	150	81	37	1	-36	-80	-149
28	150	81	37	0	-36	-80	-149
29	149	80	38	2	-37	-79	-148
30	149	78	38	2	-37	-77	-148
31	148	78	36	0	-35	-77	-147
32	149	79	38	1	-37	-78	-148
33	147	78	37	1	-36	-77	-146
34	147	77	36	1	-35	-76	-146
35	146	77	36	1	-35	-76	-145
36	145	77	35	1	-34	-76	-144
37	145	76	35	0	-34	-75	-144
38	144	77	35	0	-34	-76	-143
39	143	76	35	0	-34	-75	-142
40	144	75	35	0	-34	-74	-143
41	144	75	34	0	-33	-74	-143
42	142	74	34	1	-33	-73	-141
43	142	74	34	1	-33	-73	-141
44	142	73	32	0	-31	-72	-141
45	141	72	31	1	-30	-71	-140
46	140	73	31	1	-30	-72	-139
47	140	72	31	0	-30	-71	-139
48	140	72	31	1	-30	-71	-139
49	141	74	31	0	-30	-73	-140
50	141	74	31	1	-30	-73	-140

Iteration	Γ_e
	γ_1
1	2
2	-1
3	2
4	-1
5	0
6	1
7	2
8	0
9	0
10	1
11	0
12	1
13	1
14	0
15	1
16	1
17	0
18	1
19	1
20	1
21	1
22	0
23	-1
24	0
25	1
26	0
27	1
28	1
29	2
30	2
31	0
32	0
33	1
34	0
35	1
36	0
37	1
38-42	0
43	1
44	0
45	1
46-50	0

Γ_{ch} (in LLR format)						
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
3.36	1.94	0.90	0	-0.90	-1.94	-3.36

B. The 4-bit MIM-QBP decoder in Example 1

TABLE VIII: Reconstruction Functions ϕ_c, ϕ_v, and ϕ_{ch}

Iteration	$\phi_c(r), r \in \mathcal{R}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	5	12	24	43	74	128	255	-235	-122	-72	-42	-24	-12	-5	-1
2	1	5	12	24	43	74	125	239	-255	-127	-75	-44	-24	-11	-4	-1
3	1	4	11	24	45	77	131	255	-245	-125	-73	-42	-23	-10	-4	-1
4	1	4	11	23	42	74	127	255	-233	-121	-71	-41	-22	-10	-3	-1
5	1	1	6	14	31	62	117	255	-240	-118	-64	-32	-15	-6	-2	-1
6	1	1	4	9	18	34	62	116	-255	-91	-46	-22	-10	-4	-1	-1
7	1	1	5	12	26	51	98	208	-255	-122	-66	-33	-14	-5	-1	-1
8	1	1	5	14	30	59	112	235	-255	-120	-66	-35	-16	-6	-1	-1
9	1	1	5	12	27	54	105	255	-199	-95	-51	-27	-12	-5	-1	-1
10	1	1	5	15	33	64	119	255	-241	-114	-62	-32	-14	-5	-1	-1
11	1	1	5	14	31	61	115	255	-242	-112	-59	-30	-14	-5	-1	-1
12	1	1	5	14	31	58	109	229	-255	-114	-59	-30	-14	-5	-1	-1
13	1	1	5	13	29	59	110	231	-255	-110	-57	-29	-13	-5	-1	-1
14	1	1	5	13	29	58	111	238	-255	-109	-54	-27	-13	-5	-1	-1
15	1	1	5	13	30	60	112	233	-255	-114	-60	-30	-13	-5	-1	-1
16	1	1	4	13	28	55	105	222	-255	-109	-55	-27	-12	-4	-1	-1
17	1	1	5	14	30	58	110	233	-255	-115	-61	-31	-14	-5	-1	-1
18	1	1	4	13	28	57	109	229	-255	-107	-55	-27	-12	-4	-1	-1
19	1	1	4	11	25	51	103	219	-255	-113	-59	-29	-13	-4	-1	-1
20	1	1	4	12	26	53	108	233	-255	-110	-59	-30	-14	-5	-1	-1
21	1	1	4	12	28	55	107	218	-255	-105	-54	-27	-12	-4	-1	-1
22	1	1	3	10	25	55	111	241	-255	-107	-53	-24	-10	-3	-1	-1
23	1	1	4	11	27	54	101	208	-255	-106	-53	-26	-11	-4	-1	-1
24	1	1	3	9	24	52	108	243	-255	-111	-57	-28	-13	-4	-1	-1
25	1	1	4	12	29	58	107	220	-255	-110	-57	-28	-12	-4	-1	-1
26	1	1	3	9	22	49	102	221	-255	-103	-47	-21	-9	-3	-1	-1
27	1	1	1	3	6	14	28	60	-255	-58	-26	-12	-5	-1	-1	-1
28	1	1	2	8	20	43	87	187	-255	-95	-43	-20	-8	-2	-1	-1
29	1	1	1	3	7	14	29	63	-255	-60	-27	-13	-5	-2	-1	-1
30	1	1	2	8	21	46	92	206	-255	-105	-49	-23	-10	-3	-1	-1
31	1	1	1	3	7	15	29	63	-255	-60	-27	-12	-5	-2	-1	-1
32	1	1	2	7	18	46	103	255	-179	-83	-42	-18	-7	-2	-1	-1
33	1	1	1	2	6	14	29	62	-255	-60	-28	-13	-5	-2	-1	-1
34	1	1	3	12	28	54	106	255	-244	-102	-51	-26	-11	-3	-1	-1
35	1	1	2	6	16	35	74	255	-84	-40	-20	-10	-4	-1	-1	-1
36	1	1	2	7	18	46	102	255	-178	-81	-41	-18	-7	-2	-1	-1
37	1	1	1	2	6	14	28	60	-255	-58	-28	-13	-5	-2	-1	-1
38	1	1	3	10	25	50	97	218	-255	-109	-52	-24	-9	-3	-1	-1
39	1	1	1	3	7	14	26	56	255	-56	-25	-10	-3	-1	-1	-1
40	1	1	2	7	20	45	90	199	-255	-103	-48	-20	-7	-2	-1	-1
41	1	1	2	6	16	40	94	255	-226	-91	-39	-16	-6	-2	-1	-1
42	1	1	2	8	19	40	82	185	-255	-97	-44	-19	-8	-2	-1	-1
43	1	1	2	6	16	41	95	255	-217	-89	-40	-16	-6	-2	-1	-1
44	1	1	1	5	15	39	94	255	-209	-90	-38	-15	-5	-1	-1	-1
45	1	1	2	7	19	41	98	255	-232	-95	-40	-18	-7	-2	-1	-1
46	1	1	2	7	20	47	101	255	-243	-98	-46	-19	-7	-1	-1	-1
47	1	1	2	7	18	44	102	255	-241	-99	-43	-17	-7	-2	-1	-1
48	1	1	1	6	19	47	102	255	-243	-99	-46	-18	-6	-2	-1	-1
49	1	1	2	6	16	40	94	255	-197	-86	-38	-15	-6	-1	-1	-1
50	1	1	1	6	18	44	99	255	-244	-96	-43	-17	-5	-1	-1	-1

Iteration	$\phi_v(s), s \in \mathcal{S}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	63	45	34	25	18	12	7	3	0	-4	-9	-15	-22	-31	-42	-60
2	77	56	43	32	24	16	10	4	0	-5	-11	-18	-28	-40	-54	-76
3	86	61	46	35	26	18	11	4	0	-5	-12	-20	-30	-43	-58	-83
4	89	62	45	32	21	13	5	0	-5	-12	-19	-28	-36	-47	-63	-89
5	102	74	55	40	30	21	14	6	-1	-7	-16	-25	-39	-55	-74	-102
6	111	80	61	47	36	26	16	8	0	-10	-18	-29	-42	-58	-79	-111
7	116	81	60	47	35	25	16	6	-1	-7	-18	-29	-42	-59	-81	-116
8	118	81	58	40	26	17	7	0	-7	-17	-25	-38	-49	-62	-83	-118
9	122	86	63	45	32	23	12	4	-4	-12	-23	-32	-45	-63	-86	-122
10	147	106	81	59	41	26	17	6	-6	-17	-26	-41	-59	-81	-106	-147
11	151	108	83	61	43	27	18	6	-6	-18	-27	-43	-61	-83	-108	-151
12	154	109	82	61	43	28	18	6	-6	-18	-28	-43	-61	-82	-109	-154
13	158	112	85	62	44	28	19	7	-7	-19	-28	-44	-62	-85	-112	-158
14	158	118	90	65	45	28	19	7	-7	-19	-28	-45	-65	-90	-118	-158
15	162	112	84	62	43	26	17	6	-6	-17	-26	-43	-62	-84	-112	-162
16	165	115	88	65	46	29	20	7	-7	-20	-29	-46	-65	-88	-115	-165
17	165	113	85	62	43	27	18	7	-7	-18	-27	-43	-62	-85	-113	-165
18	169	117	88	65	46	28	19	7	-7	-19	-28	-46	-65	-88	-117	-169
19	170	117	88	66	46	36	24	7	-7	-24	-36	-46	-66	-88	-117	-170
20	170	121	91	66	46	29	20	7	-7	-20	-29	-46	-66	-91	-121	-170
21	170	116	86	63	44	27	18	7	-7	-18	-27	-44	-63	-86	-116	-170
22	170	119	91	66	45	36	24	7	-7	-24	-36	-45	-66	-91	-119	-170
23	170	116	87	63	44	27	19	7	-7	-19	-27	-44	-63	-87	-116	-170
24	170	120	98	84	64	43	24	7	-7	-24	-43	-64	-84	-98	-120	-170
25	170	119	101	85	62	43	25	8	-8	-25	-43	-62	-85	-101	-119	-170
26	170	119	90	66	46	37	24	7	-7	-24	-37	-46	-66	-90	-119	-170
27	155	110	89	64	43	34	19	0	-19	-34	-43	-60	-69	-90	-110	-155
28	170	124	91	66	45	27	13	6	-6	-13	-27	-45	-66	-91	-124	-170
29	170	137	110	85	61	42	33	19	0	-19	-40	-61	-85	-110	-137	-170
30	170	135	114	88	64	44	26	8	-8	-26	-44	-64	-88	-114	-135	-170
31	170	138	109	93	80	63	40	19	0	-19	-40	-63	-87	-109	-138	-170
32	170	124	91	66	43	25	12	5	-5	-12	-25	-43	-66	-91	-124	-170
33	170	127	105	83	59	38	18	0	-18	-38	-51	-60	-83	-105	-127	-170
34	170	119	102	85	63	45	28	9	-9	-28	-45	-63	-85	-102	-119	-170
35	170	138	106	89	76	59	38	19	0	-19	-38	-59	-83	-106	-138	-170
36	170	124	89	65	43	25	12	5	-5	-12	-25	-43	-65	-89	-124	-170
37	170	126	104	81	57	37	18	0	-18	-37	-49	-58	-81	-104	-126	-170
38	170	118	102	84	61	42	25	8	-8	-25	-42	-61	-84	-102	-118	-170
39	170	104	82	70	61	43	21	0	-21	-34	-43	-61	-70	-82	-104	-170
40	170	129	111	88	63	42	25	8	-8	-25	-42	-63	-88	-111	-129	-170
41	170	130	112	92	68	47	27	9	-9	-27	-47	-68	-92	-112	-130	-170
42	170	126	105	83	61	42	24	7	-7	-24	-42	-61	-83	-105	-126	-170
43	170	130	112	91	68	46	27	9	-9	-27	-46	-68	-91	-112	-130	-170
44	170	105	88	76	51	43	28	8	-8	-28	-43	-51	-76	-88	-105	-170
45	170	129	110	85	63	44	25	8	-8	-25	-44	-63	-85	-110	-129	-170
46	170	145	124	95	68	47	28	9	-9	-28	-47	-68	-95	-124	-145	-170
47	170	129	109	86	63	43	24	7	-7	-24	-43	-63	-86	-109	-129	-170
48	170	146	123	96	69	46	28	9	-9	-28	-46	-69	-96	-123	-146	-170
49	170	145	121	94	70	47	26	8	-8	-26	-47	-70	-94	-121	-145	-170
50	170	102	85	73	66	49	29	9	-9	-29	-49	-66	-73	-85	-102	-170

Iteration	$\phi_{ch}(l), l \in \mathcal{L}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1-19	169	121	92	70	52	36	21	6	-8	-22	-37	-53	-71	-93	-122	-170
20	169	121	92	70	52	35	20	6	-8	-22	-36	-52	-71	-92	-122	-169
21	165	119	90	68	51	35	20	6	-7	-21	-36	-51	-69	-91	-119	-166
22	159	114	86	66	49	33	19	6	-7	-20	-34	-49	-66	-87	-114	-159
23	163	117	88	67	50	34	20	6	-7	-21	-35	-50	-68	-89	-117	-163
24	159	114	86	66	49	33	19	6	-7	-20	-34	-49	-66	-87	-115	-160
25	162	116	88	67	50	34	20	6	-7	-21	-35	-50	-68	-89	-117	-162
26	152	109	82	63	46	32	18	6	-7	-19	-33	-47	-63	-83	-109	-152
27	169	121	92	70	52	36	21	6	-8	-22	-37	-53	-71	-93	-122	-170
28	149	106	81	61	46	31	18	6	-7	-19	-32	-46	-62	-81	-107	-149
29	164	117	89	68	50	34	20	6	-7	-21	-35	-51	-68	-90	-118	-164
30	147	105	80	61	45	31	18	5	-7	-19	-32	-46	-61	-80	-106	-147
31	163	117	89	67	50	34	20	6	-7	-21	-35	-51	-68	-89	-117	-163
32	142	102	77	59	44	30	17	5	-6	-18	-31	-44	-59	-78	-102	-143
33	159	114	86	66	49	33	19	6	-7	-20	-34	-49	-66	-87	-114	-159
34	146	105	79	60	45	31	18	5	-7	-19	-32	-45	-61	-80	-105	-147
35	157	113	85	65	48	33	19	6	-7	-20	-34	-49	-66	-86	-113	-158
36	140	100	76	58	43	29	17	5	-6	-18	-30	-44	-59	-77	-101	-141
37	154	111	84	64	47	32	19	6	-7	-20	-33	-48	-65	-85	-111	-155
38	141	101	76	58	43	30	17	5	-6	-18	-30	-44	-59	-77	-101	-141
39	164	118	89	68	50	35	20	6	-7	-21	-35	-51	-69	-90	-118	-165
40	136	98	74	56	42	29	17	5	-6	-17	-29	-42	-57	-75	-98	-137
41	129	92	70	53	39	27	16	5	-6	-16	-28	-40	-54	-70	-93	-129
42	136	97	74	56	42	29	16	5	-6	-17	-29	-42	-57	-74	-98	-136
43	128	92	69	53	39	27	16	5	-6	-16	-28	-40	-53	-70	-92	-128
44	146	105	80	61	45	31	18	5	-7	-19	-32	-45	-61	-80	-105	-147
45	127	91	69	53	39	27	15	5	-6	-16	-27	-40	-53	-70	-92	-128
46	136	97	74	56	42	29	16	5	-6	-17	-29	-42	-57	-74	-98	-136
47	126	90	68	52	39	26	15	5	-6	-16	-27	-39	-53	-69	-91	-126
48	134	96	73	55	41	28	16	5	-6	-17	-29	-42	-56	-73	-96	-134
49	136	97	74	56	42	29	16	5	-6	-17	-29	-42	-57	-74	-98	-136
50	140	100	76	58	43	29	17	5	-6	-18	-30	-44	-59	-77	-101	-141

TABLE IX: Threshold sets Γ_c, Γ_v, and Γ_{ch}

Iteration	Γ_c														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
1	41	65	91	122	160	209	283	436	-376	-252	-185	-139	-103	-73	-46
2	28	48	71	99	133	177	245	388	-373	-236	-166	-119	-81	-53	-30
3	23	43	66	92	126	171	240	384	-365	-227	-158	-112	-76	-48	-26
4	22	43	69	104	148	215	349	-371	-229	-162	-118	-87	-63	-42	-23
5	18	35	58	88	120	164	235	401	-358	-217	-145	-97	-60	-36	-19
6	10	19	31	45	64	91	131	215	-203	-122	-83	-56	-36	-21	-12
7	11	26	43	64	93	132	197	351	-312	-183	-117	-77	-48	-27	-14
8	12	30	56	93	140	210	361	-368	-214	-146	-100	-67	-47	-28	-14
9	11	23	45	72	105	154	242	1569	-243	-155	-106	-73	-47	-25	-14
10	7	15	29	57	94	143	221	1649	-222	-144	-95	-58	-30	-18	-10
11	7	15	28	50	88	138	215	1619	-216	-139	-89	-51	-30	-18	-10
12	7	15	28	50	86	133	208	1566	-209	-134	-87	-51	-30	-18	-10
13	7	15	27	51	87	133	206	-1568	-207	-134	-88	-54	-30	-18	-10
14	7	11	23	49	86	133	208	1533	-209	-134	-87	-50	-26	-14	-10
15	7	15	27	48	86	136	208	1521	-209	-137	-87	-50	-30	-18	-10
16	7	13	25	46	79	127	199	1552	-200	-128	-80	-47	-26	-15	-9
17	7	15	28	50	88	134	204	1522	-205	-135	-89	-51	-30	-18	-10
18	7	13	25	46	83	130	199	-1585	-200	-131	-84	-47	-26	-15	-9
19	7	13	23	44	69	99	172	1531	-173	-100	-70	-45	-24	-15	-9
20	7	11	24	47	84	132	201	1530	-202	-133	-85	-48	-25	-12	-9
21	7	13	24	45	79	127	194	1457	-195	-128	-80	-46	-25	-15	-9
22	7	11	20	42	70	100	176	1411	-177	-101	-71	-43	-21	-12	-8
23	7	13	23	43	78	123	186	-1390	-187	-124	-79	-44	-24	-15	-9
24	7	10	15	23	45	86	173	1519	-174	-87	-46	-24	-16	-11	-8
25	7	10	13	24	46	84	163	-1461	-164	-85	-47	-25	-15	-12	-9
26	7	11	19	38	63	91	165	1429	-166	-92	-64	-39	-20	-12	-8
27	7	9	13	23	35	51	102	-103	-52	-36	-25	-19	-13	-10	-8
28	6	8	16	34	69	131	219	-1462	-220	-132	-70	-35	-17	-9	-7
29	6	7	9	14	24	37	53	105	-105	-48	-26	-15	-10	-8	-7
30	6	7	10	18	38	77	156	1432	-157	-78	-39	-19	-11	-8	-7
31	6	7	9	11	15	25	48	105	-106	-49	-26	-14	-10	-8	-7
32	6	8	15	33	71	133	217	1303	-218	-134	-72	-34	-16	-9	-7
33	6	7	9	13	25	47	102	-103	-48	-27	-21	-14	-10	-8	-7
34	7	9	11	22	44	82	167	1366	-168	-83	-45	-23	-12	-10	-8
35	6	7	10	13	18	31	61	132	-134	-62	-32	-18	-11	-8	-7
36	6	8	15	32	69	129	214	1276	-215	-130	-70	-33	-16	-9	-7
37	6	7	9	13	25	46	99	-100	-47	-27	-22	-14	-10	-8	-7
38	7	9	11	20	41	79	158	1277	-159	-80	-42	-21	-12	-10	-8
39	7	9	13	16	22	43	96	-97	-48	-35	-23	-17	-14	-10	-8
40	6	7	9	15	34	73	149	1277	-150	-74	-35	-16	-10	-8	-7
41	6	7	9	14	30	67	153	1277	-154	-68	-31	-15	-10	-8	-7
42	6	7	9	16	33	68	141	1207	-142	-69	-34	-17	-10	-8	-7
43	6	7	9	14	30	67	151	1277	-152	-68	-31	-15	-10	-8	-7
44	7	11	15	29	49	73	148	1231	-149	-74	-50	-30	-18	-14	-10
45	6	7	9	15	32	68	154	1253	-155	-69	-33	-16	-10	-8	-7
46	6	7	9	16	35	74	160	1264	-161	-75	-36	-17	-10	-8	-7
47	6	7	9	15	32	72	159	1110	-160	-73	-33	-16	-10	-8	-7
48	6	7	9	15	33	73	159	1252	-160	-74	-34	-16	-10	-8	-7
49	6	7	9	14	29	65	143	1028	-144	-66	-30	-15	-10	-8	-7
50	7	12	17	24	34	71	156	1098	-157	-72	-35	-25	-18	-14	-10

Iteration	Γ_v														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
1	146	108	82	62	46	30	16	2	-13	-28	-43	-61	-81	-108	-146
2	152	109	82	61	43	28	14	0	-14	-29	-45	-63	-85	-113	-155
3	159	113	85	63	45	29	14	0	-14	-29	-45	-63	-85	-114	-159
4	191	138	103	76	54	35	17	-1	-18	-35	-54	-77	-104	-138	-191
5	193	139	105	80	59	40	23	6	-11	-30	-50	-74	-101	-136	-191
6	199	143	107	80	58	38	20	2	-14	-31	-50	-74	-101	-137	-194
7	200	142	106	79	57	37	19	1	-17	-34	-52	-74	-102	-139	-197
8	204	142	107	80	57	37	17	-2	-20	-39	-58	-80	-107	-143	-203
9	205	143	106	78	55	35	18	0	-18	-36	-55	-78	-106	-143	-205
10	208	144	107	79	56	37	18	0	-18	-37	-56	-79	-106	-144	-208
11	211	145	106	78	56	38	19	1	-17	-35	-56	-79	-106	-145	-211
12	215	147	109	81	57	38	20	1	-18	-38	-57	-81	-109	-147	-215
13	216	147	110	81	58	38	19	1	-18	-38	-60	-82	-110	-147	-216
14	219	148	107	79	56	36	18	1	-16	-35	-55	-78	-107	-148	-219
15	220	150	110	82	59	39	21	2	-18	-38	-59	-82	-111	-150	-220
16	220	148	107	79	57	37	19	1	-17	-35	-55	-78	-107	-148	-220
17	223	151	110	81	58	38	19	2	-18	-38	-58	-82	-111	-152	-223
18	225	154	113	84	61	39	19	2	-16	-34	-55	-78	-107	-148	-220
19	224	152	112	84	61	40	20	1	-18	-37	-57	-79	-108	-150	-223
20	226	152	110	80	57	37	19	2	-17	-37	-57	-80	-110	-152	-226
21	227	157	117	86	60	38	19	1	-18	-38	-60	-85	-117	-157	-226
22	214	145	105	77	54	36	19	2	-15	-34	-54	-76	-105	-144	-213
23	229	158	117	86	60	38	18	1	-17	-35	-55	-77	-105	-145	-216
24	215	145	105	76	53	35	19	2	-16	-34	-53	-76	-103	-143	-213
25	228	158	117	87	61	39	19	2	-17	-39	-61	-87	-117	-158	-228
26	225	159	120	91	68	48	29	9	-9	-29	-50	-74	-103	-142	-209
27	242	169	126	94	67	44	24	3	-19	-42	-66	-92	-124	-169	-242
28	224	158	119	90	67	47	28	9	-9	-28	-48	-70	-99	-140	-207
29	242	168	124	91	64	43	23	3	-18	-39	-61	-84	-115	-158	-232
30	223	156	116	88	65	47	28	9	-9	-28	-50	-73	-101	-138	-204
31	241	167	125	92	63	38	17	-2	-23	-42	-64	-91	-125	-167	-241
32	217	154	116	88	65	45	27	9	-9	-26	-45	-66	-93	-132	-197
33	228	156	114	83	61	41	22	1	-21	-41	-61	-83	-113	-155	-227
34	208	141	101	72	49	29	11	-7	-25	-44	-62	-84	-114	-154	-221
35	232	161	119	88	61	37	17	-3	-22	-41	-62	-88	-119	-161	-231
36	215	152	115	87	64	44	26	9	-9	-26	-44	-65	-92	-131	-196
37	226	156	112	81	57	39	21	2	-17	-37	-57	-81	-113	-155	-226
38	210	146	108	82	62	46	28	10	-9	-28	-48	-74	-105	-146	-210
39	246	171	127	93	66	43	23	3	-17	-39	-63	-91	-127	-173	-248
40	215	151	114	86	62	40	20	0	-20	-40	-62	-86	-114	-150	-214
41	191	129	94	70	51	34	18	3	-13	-30	-49	-69	-94	-129	-190
42	214	150	113	86	61	39	19	-1	-20	-40	-61	-85	-113	-149	-213
43	212	149	109	80	56	35	16	-1	-17	-34	-56	-79	-109	-149	-211
44	228	159	117	86	62	40	19	0	-19	-40	-62	-86	-117	-159	-227
45	207	146	106	77	54	36	18	0	-18	-35	-54	-76	-106	-145	-207
46	214	149	109	81	58	36	17	0	-17	-36	-58	-81	-109	-148	-214
47	206	146	107	78	54	35	18	0	-18	-35	-54	-77	-106	-142	-203
48	215	149	109	81	57	36	16	-1	-18	-36	-57	-81	-109	-149	-215
49	224	158	117	85	60	39	19	0	-18	-38	-59	-85	-116	-158	-224
50	224	158	118	86	61	38	18	0	-18	-38	-61	-86	-118	-158	-224

Iteration	Γ_e
	γ_1
1	1
2	0
3	0
4	-1
5	0
6	0
7	1
8	0
9	0
10	0
11	1
12	1
13	1
14	1
15	0
16	1
17	1
18	1
19	0
20	1
21	1
22	1
23	0
24	0
25	0
26	0
27	0
28	1
29	0
30	1
31	0
32	1
33	1
34	1
35	0
36	1
37	0
38	0
39	0
40	1
41	0
42	1
43	1
44	1
45	1
46	0
47	1
48	1
49	0
50	0

Γ_{ch} (in LLR format)														
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
4.50	3.34	2.54	1.92	1.38	0.88	0.42	-0.02	-0.46	-0.92	-1.40	-1.94	-2.56	-3.36	-4.52

C. The 3-bit MIM-QMS decoder in Example 1

TABLE X: Reconstruction Functions ϕ_v and ϕ_{ch}

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$							
	0	1	2	3	4	5	6	7
1	103	58	26	5	-5	-26	-58	-103
2	111	65	31	7	-7	-31	-65	-111
3	125	73	35	8	-8	-35	-73	-125
4	136	76	37	10	-10	-37	-76	-136
5	146	83	42	11	-11	-42	-83	-146
6	155	87	43	11	-11	-43	-87	-155
7	169	99	50	13	-13	-50	-99	-169
8	170	99	51	16	-16	-51	-99	-170
9	170	96	45	11	-11	-45	-96	-170
10	170	99	51	15	-15	-51	-99	-170
11	170	95	44	12	-12	-44	-95	-170
12	170	100	50	16	-16	-50	-100	-170
13	170	97	51	16	-16	-51	-97	-170
14	170	98	52	16	-16	-52	-98	-170
15	170	99	53	16	-16	-53	-99	-170
16	170	98	50	15	-15	-50	-98	-170
17	170	102	55	17	-17	-55	-102	-170
18	170	98	51	15	-15	-51	-98	-170
19	170	99	51	16	-16	-51	-99	-170
20-23	170	97	50	16	-16	-50	-97	-170
24	170	97	49	16	-16	-49	-97	-170
25	170	96	50	16	-16	-50	-96	-170
26	170	96	49	16	-16	-49	-96	-170
27	170	96	49	16	-16	-49	-96	-170
28	170	96	47	14	-14	-47	-96	-170
29	170	96	47	13	-13	-47	-96	-170
30	170	96	47	13	-13	-47	-96	-170
31	170	96	48	13	-13	-48	-96	-170
32	170	98	49	13	-13	-49	-98	-170
33	170	99	51	12	-12	-51	-99	-170
34	170	99	50	12	-12	-50	-99	-170
35	170	98	51	13	-13	-51	-98	-170
36	170	98	50	13	-13	-50	-98	-170
37	170	97	50	13	-13	-50	-97	-170
38	170	97	50	13	-13	-50	-97	-170
39	170	97	50	12	-12	-50	-97	-170
40	170	93	46	12	-12	-46	-93	-170
41	170	96	48	12	-12	-48	-96	-170
42	170	93	45	12	-12	-45	-93	-170
43	170	92	44	12	-12	-44	-92	-170
44	170	100	51	12	-12	-51	-100	-170
45	170	99	51	18	-18	-51	-99	-170
46	170	104	58	18	-18	-58	-104	-170
47	170	100	55	20	-20	-55	-100	-170
48	170	99	53	18	-18	-53	-99	-170
49	170	99	52	18	-18	-52	-99	-170
50	170	99	52	18	-18	-52	-99	-170

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$							
	0	1	2	3	4	5	6	7
1-7	170	99	54	17	-17	-54	-99	-170
8	156	91	50	16	-16	-50	-91	-156
9	153	89	49	16	-16	-49	-89	-153
10	147	85	47	15	-15	-47	-85	-147
11	143	83	45	15	-15	-45	-83	-143
12	136	79	43	14	-14	-43	-79	-136
13	135	78	43	14	-14	-43	-78	-135
14	132	77	42	13	-13	-42	-77	-132
15	129	75	41	13	-13	-41	-75	-129
16	128	74	41	13	-13	-41	-74	-128
17	124	72	39	13	-13	-39	-72	-124
18	121	71	39	12	-12	-39	-71	-121
19	122	71	39	12	-12	-39	-71	-122
20	120	70	38	12	-12	-38	-70	-120
21	119	69	38	12	-12	-38	-69	-119
22	118	68	37	12	-12	-37	-68	-118
23	116	67	37	12	-12	-37	-67	-116
24	114	66	36	12	-12	-36	-66	-114
25	112	65	36	11	-11	-36	-65	-112
26	110	64	35	11	-11	-35	-64	-110
27	108	63	34	11	-11	-34	-63	-108
28	105	61	33	11	-11	-33	-61	-105
29	103	60	33	10	-10	-33	-60	-103
30	101	59	32	10	-10	-32	-59	-101
31	101	59	32	10	-10	-32	-59	-101
32	97	57	31	10	-10	-31	-57	-97
33	97	56	31	10	-10	-31	-56	-97
34	93	54	29	9	-9	-29	-54	-93
35	92	54	29	9	-9	-29	-54	-92
36	91	53	29	9	-9	-29	-53	-91
37	90	52	29	9	-9	-29	-52	-90
38	89	52	28	9	-9	-28	-52	-89
39	87	51	28	9	-9	-28	-51	-87
40	86	50	27	9	-9	-27	-50	-86
41	84	49	27	9	-9	-27	-49	-84
42	83	48	26	8	-8	-26	-48	-83
43	83	48	26	8	-8	-26	-48	-83
44	81	47	26	8	-8	-26	-47	-81
45	79	46	25	8	-8	-25	-46	-79
46	77	45	24	8	-8	-24	-45	-77
47	75	44	24	8	-8	-24	-44	-75
48	73	43	23	7	-7	-23	-43	-73
49	72	42	23	7	-7	-23	-42	-72
50	71	41	23	7	-7	-23	-41	-71

TABLE XI: Threshold sets Γ_v and Γ_{ch}

Iteration	Γ_v						
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
1	135	77	37	0	-36	-76	-134
2	143	80	37	0	-36	-79	-142
3	147	80	37	0	-36	-79	-146
4	152	83	39	1	-38	-82	-151
5	156	84	39	1	-38	-83	-155
6	168	92	44	3	-38	-90	-167
7	209	128	74	25	-24	-78	-160
8	168	94	48	11	-27	-78	-158
9	192	116	66	22	-22	-73	-151
10	166	91	44	10	-26	-75	-153
11	192	115	64	20	-23	-72	-149
12	154	85	40	1	-39	-84	-153
13	158	88	45	6	-35	-82	-153
14	154	83	36	-4	-43	-87	-157
15	153	82	37	1	-36	-81	-152
16	151	81	35	-4	-46	-94	-163
17	151	81	37	0	-36	-80	-150
18	148	80	36	1	-35	-79	-147
19	149	78	37	-1	-36	-77	-148
20	146	77	36	1	-35	-76	-145
21	146	77	35	1	-34	-76	-145
22	146	77	35	1	-34	-76	-145
23	145	76	35	1	-34	-75	-144
24	145	77	36	1	-35	-76	-144
25	142	75	35	1	-33	-74	-141
26	142	74	34	0	-33	-73	-141
27	142	73	32	1	-31	-72	-141
28	140	72	31	0	-30	-71	-139
29	139	72	31	1	-30	-71	-137
30	137	71	31	1	-30	-70	-135
31	142	75	32	2	-31	-73	-141
32	137	74	33	2	-29	-72	-136
33	143	75	32	2	-31	-73	-142
34	137	74	31	1	-30	-72	-132
35	135	73	31	3	-30	-71	-133
36	135	71	30	3	-29	-70	-133
37	135	71	31	2	-30	-70	-132
38	135	71	31	2	-30	-69	-132
39	131	68	31	1	-29	-66	-130
40	135	71	30	3	-28	-70	-133
41	129	69	26	-5	-27	-65	-125
42	126	65	28	2	-27	-64	-125
43	135	74	30	1	-28	-72	-134
44	140	84	45	11	-22	-64	-130
45	139	79	36	2	-35	-78	-137
46	133	75	36	1	-35	-74	-132
47	134	74	35	1	-34	-73	-133
48	131	72	35	2	-33	-71	-129
49	130	72	35	1	-32	-71	-129
50	127	71	35	1	-33	-70	-126

Iteration	Γ_e
	γ_1
1	1
2	0
3	1
4	2
5	0
6	-1
7-12	1
13	0
14	1
15	1
16	1
17	0
18	0
19	-1
20	1
21	0
22	1
23	0
24	1
25	1
26	1
27	0
28-34	1
35	0
36	0
37	1
38	1
39	1
40	3
41	1
42	0
43	0
44	1
45	2
46	1
47	1
48	2
49	1
50	1

Γ_{ch} (in LLR format)						
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
3.46	2.00	0.94	0	-0.94	-2.00	-3.46

D. The 4-bit MIM-QMS decoder in Example 1

TABLE XII: Reconstruction Functions ϕ_v and ϕ_{ch}

Iteration	$\phi_v(s), s \in \mathcal{S}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	113	81	58	41	27	15	7	2	-2	-7	-15	-27	-41	-58	-81	-113
2	118	85	63	44	29	18	9	2	-2	-9	-18	-29	-44	-63	-85	-118
3	128	93	68	49	34	21	10	3	-3	-10	-21	-34	-49	-68	-93	-128
4	145	107	79	57	39	24	11	3	-3	-11	-24	-39	-57	-79	-107	-145
5	153	110	81	58	39	24	13	3	-3	-13	-24	-39	-58	-81	-110	-153
6	170	123	89	65	45	28	15	4	-4	-15	-28	-45	-65	-89	-123	-170
7	170	121	86	61	41	25	13	4	-4	-13	-25	-41	-61	-86	-121	-170
8	170	119	86	61	43	28	15	4	-4	-15	-28	-43	-61	-86	-119	-170
9	170	115	81	58	40	25	13	3	-3	-13	-25	-40	-58	-81	-115	-170
10	170	116	83	59	41	26	13	3	-3	-13	-26	-41	-59	-83	-116	-170
11	170	116	84	61	41	25	13	4	-4	-13	-25	-41	-61	-84	-116	-170
12	170	116	83	60	41	26	13	3	-3	-13	-26	-41	-60	-83	-116	-170
13	170	114	81	57	40	27	15	4	-4	-15	-27	-40	-57	-81	-114	-170
14	170	113	80	57	40	27	15	4	-4	-15	-27	-40	-57	-80	-113	-170
15	170	113	81	57	39	26	14	5	-5	-14	-26	-39	-57	-81	-113	-170
16	170	113	81	59	41	27	15	4	-4	-15	-27	-41	-59	-81	-113	-170
17	170	113	80	57	39	26	15	4	-4	-15	-26	-39	-57	-80	-113	-170
18	170	116	84	61	43	27	15	4	-4	-15	-27	-43	-61	-84	-116	-170
19	170	114	82	61	44	30	17	5	-5	-17	-30	-44	-61	-82	-114	-170
20	170	113	81	59	42	28	16	5	-5	-16	-28	-42	-59	-81	-113	-170
21	170	113	81	59	42	28	16	5	-5	-16	-28	-42	-59	-81	-113	-170
22	170	112	80	59	42	28	17	5	-5	-17	-28	-42	-59	-80	-112	-170
23	170	113	81	59	42	28	17	5	-5	-17	-28	-42	-59	-81	-113	-170
24	170	115	83	61	44	29	17	5	-5	-17	-29	-44	-61	-83	-115	-170
25	170	117	85	62	44	29	16	5	-5	-16	-29	-44	-62	-85	-117	-170
26	170	114	81	59	42	28	16	5	-5	-16	-28	-42	-59	-81	-114	-170
27	170	117	85	62	44	29	17	5	-5	-17	-29	-44	-62	-85	-117	-170
28	170	114	81	60	42	28	16	5	-5	-16	-28	-42	-60	-81	-114	-170
29	170	116	83	61	44	29	16	5	-5	-16	-29	-44	-61	-83	-116	-170
30	170	114	81	59	42	27	15	4	-4	-15	-27	-42	-59	-81	-114	-170
31	170	116	83	61	44	29	17	5	-5	-17	-29	-44	-61	-83	-116	-170
32	170	115	81	59	42	28	16	5	-5	-16	-28	-42	-59	-81	-115	-170
33	170	116	84	61	44	30	16	5	-5	-16	-30	-44	-61	-84	-116	-170
34	170	117	85	63	47	33	18	6	-6	-18	-33	-47	-63	-85	-117	-170
35	170	117	85	63	45	31	18	6	-6	-18	-31	-45	-63	-85	-117	-170
36	170	117	86	65	47	31	18	6	-6	-18	-31	-47	-65	-86	-117	-170
37	170	115	83	61	45	30	17	6	-6	-17	-30	-45	-61	-83	-115	-170
38	170	117	85	64	47	31	18	6	-6	-18	-31	-47	-64	-85	-117	-170
39	170	117	85	64	46	31	17	5	-5	-17	-31	-46	-64	-85	-117	-170
40	170	116	83	61	46	31	18	6	-6	-18	-31	-46	-61	-83	-116	-170
41	170	117	85	63	46	30	18	6	-6	-18	-30	-46	-63	-85	-117	-170
42	170	118	86	65	47	31	18	6	-6	-18	-31	-47	-65	-86	-118	-170
43	170	116	83	61	45	31	18	6	-6	-18	-31	-45	-61	-83	-116	-170
44	170	116	84	63	45	30	18	6	-6	-18	-30	-45	-63	-84	-116	-170
45	170	115	83	61	45	30	17	6	-6	-17	-30	-45	-61	-83	-115	-170
46	170	119	87	64	45	30	18	6	-6	-18	-30	-45	-64	-87	-119	-170
47	170	120	89	67	50	34	20	7	-7	-20	-34	-50	-67	-89	-120	-170
48	170	118	86	64	47	32	18	6	-6	-18	-32	-47	-64	-86	-118	-170
49	170	118	87	63	46	32	19	6	-6	-19	-32	-46	-63	-87	-118	-170
50	170	118	87	64	46	32	20	7	-7	-20	-32	-46	-64	-87	-118	-170

Iteration	$\phi_{ch}(l), l \in \mathcal{L}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1-5	170	122	93	71	52	36	22	8	-6	-21	-36	-52	-70	-92	-121	-169
6	166	119	91	69	51	36	21	7	-6	-20	-35	-51	-68	-90	-118	-165
7	170	122	93	71	52	36	22	8	-6	-21	-36	-52	-70	-92	-121	-169
8	152	109	83	63	47	33	19	7	-6	-18	-32	-46	-63	-82	-108	-151
9	152	109	83	63	47	33	19	7	-6	-18	-32	-46	-63	-83	-108	-151
10	142	102	78	59	44	30	18	6	-5	-17	-30	-43	-59	-77	-101	-141
11	141	102	77	59	44	30	18	6	-5	-17	-30	-43	-58	-77	-101	-140
12	135	97	74	56	42	29	17	6	-5	-16	-28	-41	-56	-73	-96	-134
13	136	98	74	57	42	29	17	6	-5	-16	-28	-41	-56	-74	-97	-135
14	132	95	72	55	41	28	17	6	-5	-16	-28	-40	-55	-72	-94	-131
15	130	94	71	54	40	28	17	6	-5	-16	-27	-40	-54	-71	-93	-129
16	128	92	70	53	39	27	16	6	-5	-15	-27	-39	-53	-69	-91	-127
17	126	91	69	53	39	27	16	6	-5	-15	-26	-38	-52	-68	-90	-125
18	124	89	68	52	38	27	16	6	-5	-15	-26	-38	-51	-67	-88	-123
19	122	88	67	51	38	26	16	5	-5	-15	-25	-37	-50	-66	-87	-121
20	121	87	66	51	37	26	15	5	-4	-15	-25	-37	-50	-66	-86	-120
21	122	88	67	51	38	26	16	5	-5	-15	-25	-37	-50	-66	-87	-121
22	122	88	67	51	38	26	16	5	-5	-15	-26	-37	-50	-66	-87	-121
23	122	87	67	51	38	26	16	5	-5	-15	-25	-37	-50	-66	-87	-121
24	117	84	64	49	36	25	15	5	-4	-14	-24	-36	-48	-63	-83	-116
25	111	80	61	46	34	24	14	5	-4	-13	-23	-34	-46	-60	-79	-110
26	116	84	64	48	36	25	15	5	-4	-14	-24	-35	-48	-63	-83	-116
27	112	80	61	46	34	24	14	5	-4	-13	-23	-34	-46	-60	-79	-111
28	116	83	63	48	36	25	15	5	-4	-14	-24	-35	-48	-63	-82	-115
29	109	78	59	45	34	23	14	5	-4	-13	-23	-33	-45	-59	-77	-108
30	115	82	63	48	35	25	15	5	-4	-14	-24	-35	-47	-62	-82	-114
31	108	77	59	45	33	23	14	5	-4	-13	-22	-33	-44	-58	-77	-107
32	114	82	62	47	35	24	14	5	-4	-14	-24	-35	-47	-62	-81	-113
33	106	76	58	44	33	23	14	5	-4	-13	-22	-32	-44	-58	-76	-106
34	109	78	59	45	34	23	14	5	-4	-13	-23	-33	-45	-59	-77	-108
35	107	77	58	44	33	23	14	5	-4	-13	-22	-32	-44	-58	-76	-106
36	108	78	59	45	33	23	14	5	-4	-13	-23	-33	-45	-59	-77	-107
37	106	76	58	44	33	23	14	5	-4	-13	-22	-32	-44	-58	-76	-106
38	107	77	58	44	33	23	14	5	-4	-13	-22	-32	-44	-58	-76	-106
39	104	75	57	43	32	22	13	5	-4	-13	-22	-32	-43	-56	-74	-103
40	106	76	58	44	33	23	14	5	-4	-13	-22	-32	-44	-58	-76	-105
41	104	75	57	43	32	22	13	5	-4	-13	-22	-32	-43	-56	-74	-103
42	103	74	56	43	32	22	13	5	-4	-12	-21	-31	-42	-56	-73	-102
43	104	75	57	43	32	22	13	5	-4	-13	-22	-32	-43	-56	-74	-103
44	101	72	55	42	31	22	13	4	-4	-12	-21	-31	-42	-55	-72	-100
45	103	74	56	43	32	22	13	5	-4	-12	-22	-31	-43	-56	-73	-102
46	99	71	54	41	31	21	13	4	-4	-12	-21	-30	-41	-54	-71	-99
47	97	70	53	40	30	21	12	4	-4	-12	-20	-30	-40	-53	-69	-96
48	95	69	52	40	29	20	12	4	-4	-12	-20	-29	-39	-52	-68	-95
49	95	68	52	40	29	20	12	4	-4	-11	-20	-29	-39	-51	-68	-94
50	94	68	51	39	29	20	12	4	-3	-11	-20	-29	-39	-51	-67	-93

TABLE XIII: Threshold sets Γ_v and Γ_{ch}

Iteration	Γ_v														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
1	146	108	82	61	43	27	13	-1	-16	-30	-45	-62	-81	-106	-145
2	153	112	86	65	47	32	16	1	-14	-30	-46	-64	-84	-111	-151
3	160	116	87	65	46	27	10	-6	-21	-37	-55	-76	-101	-134	-182
4	163	116	86	63	44	27	12	-5	-21	-37	-55	-76	-102	-136	-186
5	190	138	103	76	54	35	17	0	-18	-36	-55	-77	-102	-136	-188
6	179	127	93	69	48	31	15	1	-14	-29	-46	-66	-91	-125	-177
7	198	140	103	76	55	36	18	0	-18	-36	-54	-75	-102	-138	-196
8	173	119	87	64	46	30	15	2	-12	-26	-43	-62	-85	-117	-171
9	183	127	94	69	49	31	15	0	-13	-29	-47	-67	-91	-125	-181
10	164	113	84	61	42	27	14	1	-12	-27	-44	-66	-90	-122	-173
11	175	122	90	66	46	29	13	1	-12	-27	-44	-63	-86	-118	-172
12	159	107	76	54	37	22	8	-7	-21	-35	-48	-65	-87	-116	-168
13	173	120	89	67	50	35	21	7	-7	-22	-37	-54	-75	-106	-160
14	159	106	75	53	35	20	7	-6	-20	-33	-47	-64	-87	-118	-169
15	169	117	87	65	49	34	20	7	-7	-21	-37	-55	-76	-105	-158
16	158	104	74	52	34	20	7	-5	-18	-32	-47	-63	-84	-114	-165
17	164	112	82	60	40	23	8	-6	-18	-32	-46	-62	-83	-113	-163
18	166	116	87	64	46	30	16	1	-14	-28	-43	-58	-77	-104	-156
19	159	108	80	59	42	28	14	0	-13	-27	-40	-57	-78	-106	-157
20	156	106	78	57	41	27	13	0	-13	-26	-40	-56	-76	-104	-155
21	157	106	78	58	41	27	13	0	-14	-27	-40	-56	-76	-104	-155
22	158	106	78	58	41	27	14	0	-13	-27	-40	-57	-77	-105	-156
23	157	106	78	58	41	27	13	0	-14	-29	-45	-64	-87	-118	-170
24	166	115	85	62	44	28	14	0	-13	-27	-42	-60	-83	-113	-164
25	150	101	74	55	38	24	12	0	-12	-24	-37	-53	-72	-99	-147
26	164	113	83	61	43	28	14	0	-13	-27	-42	-60	-82	-112	-163
27	150	102	74	55	38	24	12	-1	-12	-24	-37	-53	-72	-100	-148
28	166	114	83	62	44	28	13	0	-13	-26	-42	-60	-81	-112	-164
29	147	99	72	52	37	24	13	3	-9	-22	-36	-51	-70	-97	-145
30	166	113	83	61	43	27	13	-1	-14	-27	-42	-59	-81	-111	-163
31	147	99	72	52	37	24	13	1	-11	-23	-35	-50	-70	-97	-145
32	165	113	83	61	44	28	13	0	-13	-26	-42	-60	-81	-112	-164
33	147	99	71	52	36	21	7	-7	-20	-34	-49	-65	-85	-112	-158
34	151	102	75	55	37	22	7	-6	-20	-33	-48	-64	-84	-114	-165
35	148	101	74	55	37	21	7	-6	-19	-33	-47	-64	-84	-110	-157
36	149	98	69	49	33	19	6	-5	-18	-32	-47	-65	-86	-115	-165
37	148	100	73	54	37	21	7	-5	-19	-33	-47	-64	-84	-112	-161
38	151	102	75	56	38	22	8	-5	-18	-31	-46	-64	-85	-113	-163
39	144	95	67	47	33	20	7	-5	-18	-32	-46	-63	-84	-110	-157
40	150	101	74	53	36	21	7	-6	-19	-33	-46	-64	-85	-113	-162
41	152	104	76	55	37	22	8	-5	-17	-31	-46	-63	-83	-110	-157
42	145	95	68	48	33	20	7	-5	-18	-31	-46	-63	-83	-110	-157
43	150	100	72	52	35	21	7	-7	-20	-32	-46	-63	-84	-112	-160
44	142	93	66	47	31	17	5	-7	-19	-31	-45	-62	-82	-109	-154
45	155	108	78	55	36	21	7	-6	-19	-32	-46	-62	-83	-111	-160
46	157	110	83	62	45	30	16	2	-13	-28	-44	-61	-81	-109	-155
47	153	106	79	58	42	27	13	1	-12	-25	-40	-57	-78	-105	-152
48	152	106	77	56	41	27	14	0	-13	-26	-39	-55	-76	-105	-150
49	152	106	78	57	41	28	15	1	-14	-26	-39	-55	-77	-105	-150
50	152	105	78	56	40	27	15	2	-10	-23	-38	-55	-77	-104	-150

Iteration	Γ_e
	γ_1
1	1
2	0
3	0
4	1
5-9	0
10-13	1
14	0
15	0
16-19	1
20	0
21	0
22	1
23	0
24	1
25	1
26	0
27	0
28	0
29	1
30	1
31	1
32	0
33	1
34	1
35	1
36-40	0
41	1
42	1
43	0
44	1
45	0
46	0
47	1
48	0
49	1
50	0

Γ_{ch} (in LLR format)														
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
4.54	3.38	2.58	1.94	1.40	0.92	0.46	0.02	-0.42	-0.88	-1.38	-1.92	-2.5600	-3.34	-4.50

E. The 3-bit MIM-QMS decoder in Example 2

TABLE XIV: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-560 5G LDPC code with code rate $1/2$

Iteration	$\phi_v(s), s \in \mathcal{S}$							
	0	1	2	3	4	5	6	7
1	138	74	29	5	-5	-29	-74	-138
2	144	79	33	6	-6	-33	-79	-144
3	153	84	35	7	-7	-35	-84	-153
4	160	86	37	7	-7	-37	-86	-160
5	167	90	39	8	-8	-39	-90	-167
6	169	91	40	9	-9	-40	-91	-169
7	173	93	41	9	-9	-41	-93	-173
8	176	94	42	9	-9	-42	-94	-176
9	180	96	43	10	-10	-43	-96	-180
10	186	100	46	11	-11	-46	-100	-186
11	190	103	48	11	-11	-48	-103	-190
12	194	105	48	12	-12	-48	-105	-194
13	201	107	49	12	-12	-49	-107	-201
14	204	116	59	16	-16	-59	-116	-204
15	204	118	57	15	-15	-57	-118	-204
16	204	117	56	15	-15	-56	-117	-204
17	204	116	56	15	-15	-56	-116	-204
18	204	115	55	15	-15	-55	-115	-204
19	204	114	55	15	-15	-55	-114	-204
20	204	113	54	16	-16	-54	-113	-204
21	204	112	54	16	-16	-54	-112	-204
22	204	110	53	16	-16	-53	-110	-204
23	204	108	51	15	-15	-51	-108	-204
24	204	107	51	16	-16	-51	-107	-204
25	204	106	50	15	-15	-50	-106	-204
26	204	105	50	15	-15	-50	-105	-204
27	204	104	49	15	-15	-49	-104	-204
28	204	103	49	15	-15	-49	-103	-204
29	204	103	49	15	-15	-49	-103	-204
30	204	103	49	15	-15	-49	-103	-204

Iteration	$\phi_{ch}(l), l \in \mathcal{L}$							
	0	1	2	3	4	5	6	7
1-13	204	120	66	21	-21	-66	-120	-204
14	185	109	60	19	-19	-60	-109	-185
15	157	92	51	16	-16	-51	-92	-157
16	152	89	49	16	-16	-49	-89	-152
17	146	86	47	15	-15	-47	-86	-146
18	142	83	46	15	-15	-46	-83	-142
19	137	81	44	14	-14	-44	-81	-137
20	131	77	42	14	-14	-42	-77	-131
21	126	74	41	13	-13	-41	-74	-126
22	121	71	39	13	-13	-39	-71	-121
23	117	69	38	12	-12	-38	-69	-117
24	115	68	37	12	-12	-37	-68	-115
25	113	66	36	12	-12	-36	-66	-113
26	111	65	36	12	-12	-36	-65	-111
27	110	65	35	11	-11	-35	-65	-110
28	109	64	35	11	-11	-35	-64	-109
29	109	64	35	11	-11	-35	-64	-109
30	108	64	35	11	-11	-35	-64	-108

TABLE XV: Threshold sets Γ_v and Γ_{ch} for the length-560 5G LDPC code with code rate $1/2$

Iteration	Γ_v						
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
1	162	93	45	1	-44	-92	-161
2	169	96	44	0	-43	-95	-168
3	173	97	45	1	-44	-96	-172
4	177	98	46	0	-45	-97	-176
5	178	99	46	1	-45	-98	-177
6	179	99	47	1	-46	-98	-178
7	181	100	46	0	-45	-99	-180
8	184	100	46	1	-45	-99	-183
9	188	103	47	0	-46	-102	-187
10	190	104	49	1	-48	-103	-189
11	192	104	48	1	-47	-103	-191
12	196	104	47	-1	-47	-103	-195
13	251	148	75	13	-42	-102	-195
14	215	117	48	-10	-69	-136	-232
15	190	103	43	-7	-57	-114	-197
16	186	99	42	-6	-54	-112	-195
17	182	96	40	-7	-52	-108	-192
18	189	108	53	8	-37	-92	-178
19	187	104	52	8	-37	-91	-177
20	182	103	50	7	-35	-88	-173
21	177	97	49	6	-34	-85	-169
22	169	84	32	-5	-46	-95	-173
23	168	93	47	6	-30	-77	-159
24	158	78	31	-6	-44	-90	-167
25	164	89	44	7	-29	-76	-155
26	155	76	30	-6	-42	-87	-162
27	160	87	43	6	-29	-74	-152
28	152	74	30	-5	-41	-86	-159
29	159	86	42	7	-29	-73	-151
30	152	74	30	-6	-41	-84	-158

Γ_{ch} (in LLR format)						
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
3.06	1.78	0.84	0	-0.84	-1.78	-3.06

Iteration	$\dfrac{\Gamma_e}{\gamma_1}$
1	1
2	-1
3	1
4	-2
5	1
6	0
7	0
8	1
9	0
10	1
11	0
12	0
13	1
14	-1
15	1
16	-1
17	4
18	3
19-22	0
23	1
24	0
25	1
26	0
27	1
28	0
29	1
30	0

F. The 4-bit MIM-QMS decoder in Example 2

TABLE XVI: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-560 5G LDPC code with code rate $1/2$

Iteration	$\phi_v(s), s \in \mathcal{S}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	154	111	79	54	33	18	8	2	-2	-8	-18	-33	-54	-79	-111	-154
2	161	115	82	56	36	20	9	2	-2	-9	-20	-36	-56	-82	-115	-161
3	169	120	84	58	37	21	10	2	-2	-10	-21	-37	-58	-84	-120	-169
4	175	123	86	60	39	22	10	3	-3	-10	-22	-39	-60	-86	-123	-175
5	180	125	88	61	40	23	11	3	-3	-11	-23	-40	-61	-88	-125	-180
6	197	141	101	72	48	30	15	4	-4	-15	-30	-48	-72	-101	-140	-196
7	199	141	102	72	49	30	15	4	-4	-15	-30	-49	-72	-102	-141	-198
8	204	147	107	77	53	33	15	3	-3	-15	-33	-53	-77	-107	-147	-204
9	204	146	105	74	50	28	12	3	-3	-12	-28	-50	-74	-105	-146	-204
10	204	145	105	75	52	33	16	4	-4	-16	-33	-52	-75	-105	-145	-204
11	204	145	104	74	50	30	14	3	-3	-14	-30	-50	-74	-104	-145	-204
12	204	144	104	74	51	30	14	3	-3	-14	-30	-51	-74	-104	-144	-204
13	204	143	102	73	50	30	14	3	-3	-14	-30	-50	-73	-102	-143	-204
14	204	142	102	73	51	33	16	4	-4	-16	-33	-51	-73	-102	-142	-204
15	204	141	101	71	47	29	15	4	-4	-15	-29	-47	-71	-101	-141	-204
16	204	140	100	72	51	33	16	4	-4	-16	-33	-51	-72	-100	-140	-204
17	204	139	99	70	46	29	15	4	-4	-15	-29	-46	-70	-99	-139	-204
18	204	138	98	70	50	33	16	4	-4	-16	-33	-50	-70	-98	-138	-204
19	204	138	97	70	49	31	16	4	-4	-16	-31	-49	-70	-97	-138	-204
20	204	136	96	69	49	31	17	4	-4	-17	-31	-49	-69	-96	-136	-204
21	204	134	94	68	49	34	19	6	-6	-19	-34	-49	-68	-94	-134	-204
22	204	132	92	67	49	34	19	6	-6	-19	-34	-49	-67	-92	-132	-204
23	204	130	90	66	48	34	19	6	-6	-19	-34	-48	-66	-90	-130	-204
24	204	128	88	65	48	33	19	6	-6	-19	-33	-48	-65	-88	-128	-204
25	204	126	86	63	47	33	19	6	-6	-19	-33	-47	-63	-86	-126	-204
26	204	124	85	62	46	33	19	6	-6	-19	-33	-46	-62	-85	-124	-204
27	204	122	83	61	45	32	18	6	-6	-18	-32	-45	-61	-83	-122	-204
28	204	120	82	60	45	32	18	6	-6	-18	-32	-45	-60	-82	-120	-204
29	204	119	81	60	44	31	16	4	-4	-16	-31	-44	-60	-81	-119	-204
30	204	119	81	59	44	31	16	4	-4	-16	-31	-44	-59	-81	-119	-204

Iteration	$\phi_{ch}(l), l \in \mathcal{L}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1-7	204	148	113	86	64	44	26	9	-8	-26	-44	-64	-86	-113	-148	-204
8	182	132	101	77	57	39	24	8	-7	-23	-39	-57	-77	-101	-132	-182
9	179	130	99	76	56	39	23	8	-7	-22	-39	-56	-76	-99	-130	-179
10	178	129	99	75	56	39	23	8	-7	-22	-38	-56	-75	-99	-129	-178
11	174	126	96	74	54	38	22	8	-7	-22	-37	-54	-74	-96	-126	-174
12	171	124	94	72	53	37	22	8	-7	-21	-37	-53	-72	-94	-124	-171
13	168	122	93	71	53	36	22	8	-7	-21	-36	-53	-71	-93	-122	-168
14	164	119	91	70	52	36	21	7	-6	-21	-35	-52	-70	-91	-119	-164
15	162	118	90	69	51	35	21	7	-6	-20	-35	-51	-69	-90	-118	-162
16	159	115	88	67	50	34	21	7	-6	-20	-34	-50	-67	-88	-115	-159
17	156	113	86	66	49	34	20	7	-6	-19	-34	-49	-66	-86	-113	-156
18	153	111	85	65	48	33	20	7	-6	-19	-33	-48	-65	-85	-111	-153
19	149	108	82	63	47	32	19	7	-6	-19	-32	-47	-63	-82	-108	-149
20	145	105	80	61	45	31	19	7	-6	-18	-31	-45	-61	-80	-105	-145
21	140	102	78	59	44	30	18	6	-5	-18	-30	-44	-59	-78	-102	-140
22	136	98	75	58	43	29	18	6	-5	-17	-29	-43	-58	-75	-98	-136
23	131	95	73	56	41	29	17	6	-5	-16	-28	-41	-56	-73	-95	-131
24	128	92	71	54	40	28	16	6	-5	-16	-27	-40	-54	-71	-92	-128
25	123	89	68	52	39	27	16	6	-5	-15	-27	-39	-52	-68	-89	-123
26	120	87	66	51	38	26	16	5	-5	-15	-26	-38	-51	-66	-87	-120
27	118	85	65	50	37	26	15	5	-5	-15	-25	-37	-50	-65	-85	-118
28	116	84	64	49	36	25	15	5	-5	-14	-25	-36	-49	-64	-84	-116
29	114	83	63	48	36	25	15	5	-4	-14	-25	-36	-48	-63	-83	-114
30	114	82	63	48	36	25	15	5	-4	-14	-24	-36	-48	-63	-82	-114

TABLE XVII: Threshold sets Γ_v and Γ_{ch} for the length-560 5G LDPC code with code rate $1/2$

Iteration	Γ_v														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
1	179	132	101	76	55	36	18	1	-17	-35	-54	-75	-100	-131	-178
2	183	134	101	76	55	36	18	0	-17	-35	-54	-76	-101	-133	-183
3	187	135	102	77	55	36	18	1	-17	-35	-54	-76	-102	-135	-187
4	191	137	103	77	55	36	18	1	-17	-35	-55	-76	-102	-136	-190
5	229	170	130	99	74	51	29	7	-14	-34	-54	-76	-103	-138	-194
6	229	170	130	100	74	51	29	6	-15	-34	-54	-76	-102	-137	-194
7	233	171	130	100	73	47	23	1	-22	-46	-72	-99	-130	-171	-232
8	210	153	115	87	61	38	19	1	-16	-36	-59	-85	-114	-152	-209
9	207	150	114	87	63	41	20	1	-20	-42	-64	-86	-113	-150	-206
10	210	153	115	87	64	42	21	2	-16	-35	-58	-83	-112	-149	-207
11	206	149	113	86	63	41	20	2	-16	-35	-58	-82	-111	-148	-205
12	204	146	110	83	61	40	20	2	-15	-34	-56	-80	-108	-145	-203
13	204	146	110	83	61	40	20	1	-18	-38	-59	-81	-108	-144	-203
14	200	142	105	75	50	29	11	-9	-26	-43	-61	-81	-107	-143	-201
15	201	142	106	80	59	38	18	1	-17	-37	-58	-79	-105	-141	-200
16	198	138	102	73	48	28	10	-8	-25	-41	-58	-78	-104	-140	-199
17	196	136	101	76	56	37	18	1	-16	-35	-55	-75	-100	-135	-195
18	197	136	101	75	51	30	10	-8	-24	-40	-57	-75	-100	-135	-196
19	193	132	97	72	50	29	10	-8	-24	-39	-55	-73	-97	-131	-192
20	192	130	96	72	53	35	17	-2	-20	-37	-53	-71	-95	-129	-191
21	188	126	92	70	51	34	16	-2	-19	-36	-51	-69	-91	-125	-187
22	185	122	89	67	50	33	15	-2	-19	-34	-49	-66	-88	-121	-184
23	181	118	86	65	49	32	15	-2	-18	-33	-48	-64	-85	-117	-180
24	178	115	83	63	47	31	14	-2	-18	-32	-46	-62	-83	-114	-177
25	174	112	81	61	46	30	14	-1	-17	-31	-45	-60	-79	-110	-173
26	170	108	78	59	44	30	13	-2	-17	-31	-44	-59	-78	-107	-169
27	167	106	76	58	43	28	13	-2	-16	-30	-42	-57	-76	-105	-166
28	165	104	75	57	43	28	13	1	-12	-27	-41	-56	-74	-103	-164
29	163	103	74	56	42	28	13	1	-12	-26	-41	-55	-73	-102	-162
30	162	102	74	56	42	28	13	1	-11	-26	-41	-55	-73	-101	-161

Γ_{ch} (in LLR format)														
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
4.12	3.08	2.36	1.78	1.28	0.84	0.42	0.02	-0.40	-0.82	-1.28	-1.78	-2.36	-3.08	-4.12

Iteration	Γ_e
	γ_1
1	1
2	0
3	1
4-7	0
8	1
9	1
10	0
11	0
12-18	1
19-23	0
24-30	1

G. The 3-bit MIM-QMS decoders in Example 3

TABLE XVIII: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $1/2$

Iteration	$\phi_v(s),\ s \in \mathcal{S}$							
	0	1	2	3	4	5	6	7
1	103	58	27	6	-6	-27	-58	-103
2	113	66	31	7	-7	-31	-66	-113
3	127	74	36	9	-9	-36	-74	-127
4	140	82	41	11	-11	-41	-82	-140
5	152	86	42	11	-11	-42	-86	-152
6	168	98	48	12	-12	-48	-98	-168
7	170	100	50	15	-15	-50	-100	-170
8	170	97	46	12	-12	-46	-97	-170
9	170	99	51	16	-16	-51	-99	-170
10	170	97	45	12	-12	-45	-97	-170
11	170	100	51	16	-16	-51	-100	-170
12	170	100	53	16	-16	-53	-100	-170
13	170	98	50	15	-15	-50	-98	-170
14	170	102	55	17	-17	-55	-102	-170
15	170	98	51	15	-15	-51	-98	-170
16	170	99	51	16	-16	-51	-99	-170
17	170	98	51	16	-16	-51	-98	-170
18	170	98	51	16	-16	-51	-98	-170
19	170	97	50	16	-16	-50	-97	-170
20	170	97	50	16	-16	-50	-97	-170
21	170	97	50	16	-16	-50	-97	-170
22	170	96	49	16	-16	-49	-96	-170
23	170	96	47	14	-14	-47	-96	-170
24	170	96	47	14	-14	-47	-96	-170
25	170	96	47	13	-13	-47	-96	-170
26	170	96	48	13	-13	-48	-96	-170
27	170	98	49	13	-13	-49	-98	-170
28	170	98	50	12	-12	-50	-98	-170
29	170	97	50	12	-12	-50	-97	-170
30	170	97	50	12	-12	-50	-97	-170

Iteration	$\phi_{ch}(l),\ l \in \mathcal{L}$							
	0	1	2	3	4	5	6	7
1-6	170	99	54	17	-17	-54	-99	-170
7	152	88	48	15	-15	-48	-88	-152
8	151	88	48	15	-15	-48	-88	-151
9	145	84	46	15	-15	-46	-84	-145
10	144	83	45	14	-14	-45	-83	-144
11	139	81	44	14	-14	-44	-81	-139
12	131	76	41	13	-13	-41	-76	-131
13	129	75	41	13	-13	-41	-75	-129
14	125	72	39	13	-13	-39	-72	-125
15	122	71	38	12	-12	-38	-71	-122
16	122	71	38	12	-12	-38	-71	-122
17	120	70	38	12	-12	-38	-70	-120
18	118	69	37	12	-12	-37	-69	-118
19	116	67	37	12	-12	-37	-67	-116
20	114	66	36	11	-11	-36	-66	-114
21	111	65	35	11	-11	-35	-65	-111
22	109	63	34	11	-11	-34	-63	-109
23	107	62	34	11	-11	-34	-62	-107
24	104	60	33	10	-10	-33	-60	-104
25	102	59	32	10	-10	-32	-59	-102
26	101	59	32	10	-10	-32	-59	-101
27	97	56	31	10	-10	-31	-56	-97
28	97	56	30	10	-10	-30	-56	-97
29	95	55	30	10	-10	-30	-55	-95
30	94	54	30	9	-9	-30	-54	-94

TABLE XIX: Threshold sets Γ_v and Γ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $1/2$

Iteration	Γ_v						
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
1	136	78	36	0	-35	-77	-135
2	142	79	38	1	-37	-78	-141
3	150	83	38	0	-38	-82	-149
4	155	84	39	0	-38	-83	-154
5	166	90	40	2	-38	-89	-165
6	207	126	71	23	-25	-81	-166
7	157	78	28	-10	-45	-90	-163
8	192	117	66	22	-21	-72	-150
9	151	75	26	-9	-44	-89	-163
10	176	109	63	20	-22	-73	-151
11	162	90	42	1	-41	-89	-161
12	153	82	37	1	-36	-81	-152
13	164	95	47	5	-34	-79	-150
14	151	82	37	0	-36	-81	-150
15	148	80	36	1	-35	-79	-147
16	147	78	37	-1	-36	-77	-146
17	147	78	36	0	-35	-77	-146
18	146	79	36	0	-35	-78	-145
19	144	76	35	1	-34	-75	-143
20	144	75	35	1	-34	-74	-143
21	143	74	34	0	-33	-73	-142
22	141	73	32	1	-31	-72	-140
23	141	72	31	1	-30	-71	-140
24	139	72	31	-1	-31	-71	-138
25	137	71	31	1	-30	-70	-136
26	142	75	31	2	-30	-73	-141
27	137	73	30	1	-29	-72	-136
28	136	72	30	1	-28	-71	-135
29	136	75	31	2	-30	-72	-134
30	137	71	27	-9	-50	-94	-156

Γ_{ch} (in LLR format)						
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
3.50	2.02	0.94	0	-0.94	-2.02	-3.50

Iteration	Γ_e
	γ_1
1	2
2	1
3	1
4	0
5	0
6	1
7	2
8-12	1
13	0
14	0
15	1
16	-1
17	0
18	1
19	1
20	1
21	0
22	1
23	0
24	2
25-29	1
30	2

TABLE XX: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $2/3$

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$							
	0	1	2	3	4	5	6	7
1	132	74	34	7	-7	-34	-74	-132
2	137	81	39	9	-9	-39	-81	-137
3	139	83	41	10	-10	-41	-83	-139
4	145	86	43	11	-11	-43	-86	-145
5	160	96	48	12	-12	-48	-96	-160
6	170	103	53	14	-14	-53	-103	-170
7	169	102	52	14	-14	-52	-102	-169
8	179	108	56	15	-15	-56	-108	-179
9	179	109	54	14	-14	-54	-109	-179
10	192	115	59	16	-16	-59	-115	-192
11	197	120	62	17	-17	-62	-120	-197
12	201	123	64	16	-16	-64	-123	-201
13	211	128	66	16	-16	-66	-128	-211
14	214	131	68	19	-19	-68	-131	-214
15	215	128	62	16	-16	-62	-128	-215
16	227	140	73	22	-22	-73	-140	-227
17	227	139	69	19	-19	-69	-139	-227
18	227	138	73	21	-21	-73	-138	-227
19	227	138	69	19	-19	-69	-138	-227
20	227	135	66	18	-18	-66	-135	-227
21	227	138	72	22	-22	-72	-138	-227
22	227	137	72	23	-23	-72	-137	-227
23	227	136	72	24	-24	-72	-136	-227
24	227	136	71	23	-23	-71	-136	-227
25	227	134	70	23	-23	-70	-134	-227
26	227	132	68	23	-23	-68	-132	-227
27	227	131	64	19	-19	-64	-131	-227
28	227	129	63	18	-18	-63	-129	-227
29	227	131	67	17	-17	-67	-131	-227
30	227	131	68	15	-15	-68	-131	-227

TABLE XXI: Threshold sets Γ_v and Γ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $2/3$

Iteration	Γ_v						
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
1	173	99	46	1	-45	-98	-172
2	172	98	45	0	-45	-97	-171
3	175	98	46	0	-45	-97	-174
4	186	105	46	-2	-51	-104	-185
5	194	110	51	1	-50	-109	-193
6	192	105	49	2	-48	-104	-191
7	197	109	51	1	-50	-108	-196
8	195	106	48	1	-47	-105	-194
9	204	112	51	1	-50	-111	-203
10	199	106	44	-8	-59	-119	-213
11	200	107	42	-12	-62	-122	-216
12	199	98	30	-26	-77	-139	-232
13	201	99	32	-25	-79	-141	-235
14	224	121	58	12	-37	-100	-202
15	254	157	90	29	-33	-106	-216
16	230	128	62	12	-39	-104	-207
17	246	147	83	27	-31	-100	-207
18	206	105	37	-13	-62	-125	-224
19	224	124	61	13	-36	-104	-213
20	205	100	35	-16	-70	-137	-234
21	205	111	50	0	-49	-110	-204
22	205	110	51	1	-50	-109	-204
23	207	111	51	1	-50	-110	-206
24	200	107	50	2	-48	-105	-199
25	195	103	47	1	-46	-102	-194
26	191	98	43	0	-42	-97	-190
27	186	97	43	1	-39	-95	-185
28	184	98	41	4	-40	-95	-181
29	185	101	40	-3	-40	-97	-181
30	211	126	68	23	-20	-73	-161

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$							
	0	1	2	3	4	5	6	7
1-15	227	129	69	22	-22	-69	-129	-227
16	216	123	66	21	-21	-66	-123	-216
17	218	123	67	21	-21	-67	-123	-218
18	210	119	64	20	-20	-64	-119	-210
19	209	118	64	20	-20	-64	-118	-209
20	197	112	60	19	-19	-60	-112	-197
21	187	106	57	18	-18	-57	-106	-187
22	183	104	56	18	-18	-56	-104	-183
23	178	101	54	17	-17	-54	-101	-178
24	167	95	51	16	-16	-51	-95	-167
25	158	89	48	15	-15	-48	-89	-158
26	150	85	46	15	-15	-46	-85	-150
27	143	81	44	14	-14	-44	-81	-143
28	136	77	41	13	-13	-41	-77	-136
29	131	74	40	13	-13	-40	-74	-131
30	126	71	38	12	-12	-38	-71	-126

Γ_{ch} (in LLR format)						
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
4.08	2.32	1.08	0	-1.08	-2.32	-4.08

TABLE XXII: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $5/6$

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$							
	0	1	2	3	4	5	6	7
1	229	129	62	13	-13	-62	-129	-229
2	227	138	68	16	-16	-68	-138	-227
3	235	144	74	18	-18	-74	-144	-235
4	244	150	77	19	-19	-77	-150	-244
5	248	154	80	20	-20	-80	-154	-248
6	252	157	80	20	-20	-80	-157	-252
7	256	159	82	21	-21	-82	-159	-256
8	260	162	84	22	-22	-84	-162	-260
9	263	165	87	24	-24	-87	-165	-263
10	266	168	89	24	-24	-89	-168	-266
11	280	172	91	25	-25	-91	-172	-280
12	283	177	94	26	-26	-94	-177	-283
13	281	175	93	26	-26	-93	-175	-281
14	284	178	94	26	-26	-94	-178	-284
15	288	180	96	27	-27	-96	-180	-288
16	294	183	97	27	-27	-97	-183	-294
17	299	188	100	28	-28	-100	-188	-299
18	305	194	103	29	-29	-103	-194	-305
19	318	201	102	27	-27	-102	-201	-318
20	328	208	105	28	-28	-105	-208	-328
21	339	211	107	29	-29	-107	-211	-339
22	359	227	122	34	-34	-122	-227	-359
23	361	235	124	36	-36	-124	-235	-361
24	400	254	130	39	-39	-130	-254	-400
25	409	255	126	36	-36	-126	-255	-409
26	409	242	123	42	-42	-123	-242	-409
27	409	229	118	46	-46	-118	-229	-409
28	409	223	112	43	-43	-112	-223	-409
29	409	216	111	31	-31	-111	-216	-409
30	409	211	106	23	-23	-106	-211	-409

Iteration	Γ_e
	γ_1
1	0
2	0
3	1
4	3
5	1
6	2
7	1
8	1
9	0
10	0
11	1
12	1
13	0
14	1
15	2
16	0
17	1
18	1
19	0
20	0
21	1
22	1
23	0
24	1
25	1
26	0
27	1
28	3
29	3
30	0

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$							
	0	1	2	3	4	5	6	7
1-24	409	222	117	37	-37	-117	-222	-409
25	372	202	107	34	-34	-107	-202	-372
26	329	178	94	30	-30	-94	-178	-329
27	290	157	83	26	-26	-83	-157	-290
28	248	135	71	22	-22	-71	-135	-248
29	225	122	65	20	-20	-65	-122	-225
30	205	111	59	19	-19	-59	-111	-205

TABLE XXIII: Threshold sets Γ_v and Γ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate 5/6

Iteration	Γ_v						
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
1	295	165	76	1	-75	-164	-293
2	300	166	78	3	-75	-165	-297
3	308	177	84	7	-71	-163	-301
4	312	169	74	-5	-82	-173	-305
5	309	169	73	-3	-81	-171	-309
6	312	175	83	10	-72	-169	-310
7	314	174	84	5	-71	-168	-313
8	315	175	80	1	-79	-173	-314
9	320	178	80	0	-81	-176	-317
10	327	179	83	1	-78	-176	-326
11	331	183	85	1	-83	-181	-330
12	326	180	81	3	-79	-178	-320
13	327	179	82	2	-78	-178	-326
14	328	180	83	3	-79	-179	-325
15	333	180	81	2	-79	-175	-330
16	335	183	84	1	-82	-179	-333
17	337	188	89	1	-83	-183	-336
18	333	166	55	-29	-106	-203	-354
19	340	171	60	-27	-106	-201	-356
20	368	209	109	29	-53	-162	-335
21	398	240	135	46	-50	-164	-334
22	337	167	54	-39	-129	-232	-396
23	404	233	130	39	-53	-175	-362
24	421	233	115	26	-65	-192	-393
25	390	207	101	9	-83	-191	-383
26	364	187	93	1	-90	-183	-358
27	352	183	88	0	-85	-180	-348
28	319	172	72	2	-71	-165	-318
29	307	159	63	3	-60	-157	-305
30	270	122	42	-19	-64	-139	-275

Iteration	Γ_e
	γ_1
1-5	1
6	2
7	1
8	1
9	0
10	1
11	1
12	0
13	0
14	2
15	0
16	1
17	1
18	1
19	3
20	1
21	1
22	0
23	0
24	0
25	1
26	1
27	0
28	2
29	3
30	7

Γ_{ch} (in LLR format)						
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
4.88	2.72	1.24	0	-1.24	-2.72	-4.88

H. The 4-bit MIM-QMS decoders in Example 3

TABLE XXIV: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $1/2$

Iteration	$\phi_v(s), s \in \mathcal{S}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	114	82	60	42	28	16	8	2	-2	-8	-16	-28	-42	-60	-82	-114
2	121	88	65	46	31	19	10	2	-2	-10	-19	-31	-46	-65	-88	-121
3	134	98	72	52	36	22	11	3	-3	-11	-22	-36	-52	-72	-98	-134
4	154	112	82	60	42	26	13	3	-3	-13	-26	-42	-60	-82	-112	-154
5	170	122	88	62	42	25	13	4	-4	-13	-25	-42	-62	-88	-122	-170
6	170	120	88	63	44	27	14	4	-4	-14	-27	-44	-63	-88	-120	-170
7	170	116	82	58	40	25	13	4	-4	-13	-25	-40	-58	-82	-116	-170
8	170	117	84	60	41	26	13	4	-4	-13	-26	-41	-60	-84	-117	-170
9	170	115	83	61	42	28	16	5	-5	-16	-28	-42	-61	-83	-115	-170
10	170	115	81	58	40	27	15	4	-4	-15	-27	-40	-58	-81	-115	-170
11	170	114	82	60	42	28	15	4	-4	-15	-27	-42	-60	-82	-114	-170
12	170	114	82	59	41	27	15	4	-4	-15	-27	-41	-59	-82	-114	-170
13	170	113	81	58	41	27	15	4	-4	-15	-27	-41	-58	-81	-113	-170
14	170	118	87	63	45	31	18	6	-6	-18	-31	-45	-63	-87	-118	-170
15	170	114	81	59	43	29	17	5	-5	-17	-29	-43	-59	-81	-114	-170
16	170	114	82	61	43	29	17	5	-5	-17	-29	-43	-61	-82	-114	-170
17	170	116	85	63	45	31	18	5	-5	-18	-31	-45	-63	-85	-116	-170
18	170	116	84	62	44	29	16	5	-5	-16	-29	-44	-62	-84	-116	-170
19	170	119	87	64	46	30	16	5	-5	-16	-30	-46	-64	-87	-119	-170
20	170	115	81	59	42	28	16	5	-5	-16	-28	-42	-59	-81	-115	-170
21	170	117	84	61	44	29	17	5	-5	-17	-29	-44	-61	-84	-117	-170
22	170	115	82	60	43	28	16	5	-5	-16	-28	-43	-60	-82	-115	-170
23	170	117	85	62	45	30	16	5	-5	-16	-30	-45	-62	-85	-117	-170
24	170	117	85	64	47	32	18	6	-6	-18	-32	-47	-64	-85	-117	-170
25	170	117	85	64	46	31	18	6	-6	-18	-31	-46	-64	-85	-117	-170
26	170	116	84	61	46	31	18	6	-6	-18	-31	-46	-61	-84	-116	-170
27	170	117	85	64	46	31	18	6	-6	-18	-31	-46	-64	-85	-117	-170
28	170	115	83	61	45	32	18	7	-7	-18	-32	-45	-61	-83	-115	-170
29	170	115	83	61	44	31	18	6	-6	-18	-31	-44	-61	-83	-115	-170
30	170	117	84	62	45	31	17	6	-6	-17	-31	-45	-62	-84	-117	-170

Iteration	$\phi_{ch}(l), l \in \mathcal{L}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1-4	170	122	93	71	52	36	21	7	-7	-21	-36	-52	-71	-93	-122	-170
5	161	115	88	67	50	34	20	7	-7	-20	-34	-50	-67	-88	-115	-161
6	153	110	83	64	47	33	19	6	-6	-19	-33	-47	-64	-83	-110	-153
7	152	108	83	63	47	32	19	6	-6	-19	-32	-47	-63	-83	-108	-152
8	138	98	75	57	42	29	17	6	-6	-17	-29	-42	-57	-75	-98	-138
9	134	96	73	56	41	29	17	6	-6	-17	-29	-41	-56	-73	-96	-134
10	132	94	72	55	41	28	17	6	-6	-17	-28	-41	-55	-72	-94	-132
11	128	92	70	53	39	27	16	5	-5	-16	-27	-39	-53	-70	-92	-128
12	124	89	68	52	38	26	16	5	-5	-16	-26	-38	-52	-68	-89	-124
13	122	87	66	51	37	26	15	5	-5	-15	-26	-37	-51	-66	-87	-122
14	120	86	65	50	37	25	15	5	-5	-15	-25	-37	-50	-65	-86	-120
15	118	85	64	49	36	25	15	5	-5	-15	-25	-36	-49	-64	-85	-118
16	120	86	65	50	37	25	15	5	-5	-15	-25	-37	-50	-65	-86	-120
17	118	84	64	49	36	25	15	5	-5	-15	-25	-36	-49	-64	-84	-118
18	113	81	62	47	35	24	14	5	-5	-14	-24	-35	-47	-62	-81	-113
19	110	79	60	46	34	23	14	5	-5	-14	-23	-34	-46	-60	-79	-110
20	113	81	62	47	35	24	14	5	-5	-14	-24	-35	-47	-62	-81	-113
21	106	76	58	44	33	23	13	4	-4	-13	-23	-33	-44	-58	-76	-106
22	111	80	61	46	34	24	14	5	-5	-14	-24	-34	-46	-61	-80	-111
23	104	74	56	43	32	22	13	4	-4	-13	-22	-32	-43	-56	-74	-104
24	106	76	58	44	33	22	13	4	-4	-13	-22	-33	-44	-58	-76	-106
25	103	73	56	43	32	22	13	4	-4	-13	-22	-32	-43	-56	-73	-103
26	104	75	57	43	32	22	13	4	-4	-13	-22	-32	-43	-57	-75	-104
27	101	72	55	42	31	22	13	4	-4	-13	-22	-31	-42	-55	-72	-101
28	102	73	56	42	31	22	13	4	-4	-13	-22	-31	-42	-56	-73	-102
29	98	70	54	41	30	21	12	4	-4	-12	-21	-30	-41	-54	-70	-98
30	98	70	54	41	30	21	12	4	-4	-12	-21	-30	-41	-54	-70	-98

TABLE XXV: Threshold sets Γ_v and Γ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $1/2$

Iteration	Γ_v														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
1	148	109	84	64	47	32	17	2	-12	-26	-41	-59	-81	-107	-147
2	157	114	87	66	47	30	16	1	-14	-29	-46	-65	-86	-113	-156
3	186	136	103	78	58	40	24	8	-8	-26	-44	-64	-87	-116	-163
4	191	137	101	73	51	32	16	1	-15	-31	-50	-72	-100	-136	-190
5	185	132	98	73	52	32	16	1	-15	-31	-51	-72	-97	-131	-184
6	172	118	86	63	44	28	14	1	-13	-27	-43	-62	-85	-117	-171
7	184	128	93	68	48	30	15	1	-14	-29	-47	-67	-92	-127	-183
8	174	122	91	68	51	37	22	7	-8	-23	-40	-58	-80	-110	-162
9	170	118	87	66	50	35	21	7	-6	-20	-35	-53	-76	-107	-160
10	170	119	89	66	49	34	21	7	-6	-22	-37	-56	-77	-106	-159
11	169	119	89	67	50	34	20	7	-6	-19	-34	-52	-74	-105	-158
12	169	118	89	67	50	34	19	7	-6	-18	-33	-50	-73	-103	-154
13	161	113	84	62	45	30	17	3	-12	-28	-43	-61	-83	-112	-160
14	158	107	78	58	42	28	15	2	-12	-26	-41	-57	-77	-106	-157
15	154	105	77	57	41	27	14	1	-13	-26	-40	-56	-76	-104	-153
16	156	107	78	58	41	27	13	0	-14	-29	-45	-62	-83	-112	-161
17	157	106	77	57	40	25	12	-1	-14	-28	-44	-62	-85	-116	-167
18	162	112	83	61	43	27	13	0	-13	-26	-42	-60	-82	-111	-161
19	149	100	72	53	38	25	13	2	-10	-23	-37	-52	-71	-99	-148
20	164	113	82	60	42	27	14	1	-12	-26	-41	-59	-81	-112	-163
21	147	99	73	53	37	24	13	2	-10	-22	-35	-50	-69	-96	-144
22	165	113	83	61	43	27	13	1	-12	-26	-42	-60	-82	-112	-164
23	157	110	84	64	48	33	20	7	-6	-19	-34	-50	-69	-96	-144
24	165	115	86	66	48	32	18	5	-7	-20	-35	-53	-73	-102	-150
25	157	111	84	64	47	33	20	7	-6	-19	-32	-46	-66	-93	-142
26	163	114	86	64	47	32	19	7	-7	-19	-34	-52	-73	-100	-148
27	156	110	83	63	47	32	20	7	-6	-19	-31	-46	-65	-93	-142
28	159	110	82	63	47	32	17	5	-8	-20	-34	-51	-70	-99	-149
29	155	108	80	61	45	30	17	5	-6	-19	-33	-49	-69	-96	-144
30	157	109	82	62	45	30	18	6	-5	-17	-31	-48	-68	-97	-145

Γ_{ch} (in LLR format)														
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
4.62	3.42	2.62	1.98	1.42	0.92	0.46	0	-0.46	-0.92	-1.42	-1.98	-2.62	-3.42	-4.62

Iteration	Γ_e
	γ_1
1	1
2	1
3	0
4	1
5	1
6	0
7	1
8	0
9-17	1
18	0
19	1
20	0
21	1
22	0
23	0
24-27	1
28	0
29	0
30	1

TABLE XXVI: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $2/3$

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	148	105	77	54	36	21	10	2	-2	-10	-21	-36	-54	-77	-105	-148
2	149	108	81	58	39	24	12	3	-3	-12	-24	-39	-58	-81	-108	-149
3	156	114	86	63	43	27	13	3	-3	-13	-27	-43	-63	-86	-114	-156
4	160	117	89	65	45	28	14	4	-4	-14	-28	-45	-65	-89	-117	-160
5	168	123	92	67	46	28	14	3	-3	-14	-28	-46	-67	-92	-123	-168
6	174	128	95	68	47	28	14	4	-4	-14	-28	-47	-68	-95	-128	-174
7	181	135	101	74	52	33	17	4	-4	-17	-33	-52	-74	-101	-135	-181
8	176	129	96	69	47	29	15	4	-4	-15	-29	-47	-69	-96	-129	-176
9	188	139	104	77	54	35	18	5	-5	-18	-35	-54	-77	-104	-139	-188
10	185	135	100	72	50	31	16	5	-5	-16	-31	-50	-72	-100	-135	-185
11	195	144	107	79	56	36	20	6	-6	-20	-36	-56	-79	-107	-144	-195
12	193	141	103	73	50	32	17	5	-5	-17	-32	-50	-73	-103	-141	-193
13	204	150	112	83	59	38	21	6	-6	-21	-38	-59	-83	-112	-150	-204
14	202	145	106	76	53	35	19	5	-5	-19	-35	-53	-76	-106	-145	-202
15	219	159	120	88	60	37	19	5	-5	-19	-37	-60	-88	-120	-159	-219
16	218	159	117	85	59	37	19	6	-6	-19	-37	-59	-85	-117	-159	-218
17	220	160	117	85	60	38	20	6	-6	-20	-38	-60	-85	-117	-160	-220
18	227	164	120	87	61	39	21	6	-6	-21	-39	-61	-87	-120	-164	-227
19	227	163	119	86	60	39	21	6	-6	-21	-39	-60	-86	-119	-163	-227
20	227	163	118	86	61	40	22	6	-6	-22	-40	-61	-86	-118	-163	-227
21	227	164	121	89	64	43	25	8	-8	-25	-43	-64	-89	-121	-164	-227
22	227	163	119	86	62	41	23	7	-7	-23	-41	-62	-86	-119	-163	-227
23	227	164	120	87	62	41	24	7	-7	-24	-41	-62	-87	-120	-164	-227
24	227	162	119	87	62	41	24	8	-8	-24	-41	-62	-87	-119	-162	-227
25	227	160	116	85	61	40	22	8	-8	-22	-40	-61	-85	-116	-160	-227
26	227	160	119	90	67	46	26	8	-8	-26	-46	-67	-90	-119	-160	-227
27	227	160	116	84	60	41	23	7	-7	-23	-41	-60	-84	-116	-160	-227
28	227	159	116	86	62	42	25	8	-8	-25	-42	-62	-86	-116	-159	-227
29	227	164	122	91	67	45	26	8	-8	-26	-45	-67	-91	-122	-164	-227
30	227	160	116	86	61	40	22	7	-7	-22	-40	-61	-86	-116	-160	-227

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1-17	227	159	120	91	67	46	27	9	-9	-27	-46	-67	-91	-120	-159	-227
18	226	159	120	91	67	46	27	9	-9	-27	-46	-67	-91	-120	-159	-226
19	221	155	117	89	65	45	26	9	-9	-26	-45	-65	-89	-117	-155	-221
20	217	152	115	87	64	44	26	9	-9	-26	-44	-64	-87	-115	-152	-217
21	205	144	109	82	61	42	25	8	-8	-25	-42	-61	-82	-109	-144	-205
22	203	142	107	81	60	41	24	8	-8	-24	-41	-60	-81	-107	-142	-203
23	197	138	104	79	58	40	24	8	-8	-24	-40	-58	-79	-104	-138	-197
24	194	137	103	78	58	40	23	8	-8	-23	-40	-58	-78	-103	-137	-194
25	187	131	99	75	55	38	22	7	-7	-22	-38	-55	-75	-99	-131	-187
26	178	125	94	71	53	36	21	7	-7	-21	-36	-53	-71	-94	-125	-178
27	176	124	93	71	52	36	21	7	-7	-21	-36	-52	-71	-93	-124	-176
28	170	119	90	68	50	35	20	7	-7	-20	-35	-50	-68	-90	-119	-170
29	161	113	85	65	48	33	19	6	-6	-19	-33	-48	-65	-85	-113	-161
30	155	109	82	62	46	32	19	6	-6	-19	-32	-46	-62	-82	-109	-155

TABLE XXVII: Threshold sets Γ_v and Γ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $2/3$

Iteration	Γ_v														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
1	188	138	105	79	57	38	19	1	-17	-35	-55	-77	-104	-137	-187
2	193	141	108	81	58	37	18	0	-20	-39	-60	-83	-109	-142	-193
3	195	142	109	82	59	38	19	0	-18	-37	-58	-81	-108	-141	-194
4	202	147	111	83	59	37	18	1	-17	-36	-58	-82	-110	-146	-201
5	207	150	112	83	59	39	21	3	-15	-34	-57	-82	-111	-149	-206
6	225	168	129	98	73	51	31	11	-10	-32	-54	-79	-108	-146	-204
7	207	148	111	82	58	37	19	1	-18	-36	-57	-81	-110	-147	-206
8	230	170	130	99	74	51	31	10	-11	-32	-55	-80	-109	-147	-208
9	213	152	113	83	59	38	19	1	-18	-37	-58	-82	-112	-151	-212
10	235	173	131	100	74	51	29	7	-14	-34	-55	-80	-111	-150	-213
11	218	155	113	82	58	38	19	1	-18	-37	-57	-81	-112	-154	-217
12	239	174	131	99	72	48	26	5	-16	-37	-60	-85	-115	-156	-220
13	223	157	114	84	60	40	20	1	-18	-38	-59	-83	-113	-156	-222
14	239	172	129	94	64	40	19	1	-18	-39	-63	-93	-128	-171	-238
15	237	168	124	90	63	39	19	1	-18	-38	-62	-89	-123	-167	-236
16	236	166	122	90	63	40	20	1	-19	-39	-62	-89	-121	-165	-235
17	241	169	124	91	64	40	20	0	-19	-39	-63	-90	-123	-168	-240
18	244	170	124	90	63	40	19	0	-19	-40	-63	-89	-123	-169	-243
19	239	166	121	89	63	41	20	1	-18	-39	-62	-88	-120	-165	-238
20	257	184	140	106	79	56	34	12	-11	-33	-57	-83	-117	-163	-237
21	227	157	113	79	53	31	10	-9	-29	-48	-70	-94	-123	-165	-236
22	247	176	131	98	72	50	30	10	-9	-29	-50	-76	-109	-153	-224
23	220	151	109	78	52	31	12	-7	-26	-46	-67	-91	-122	-165	-235
24	225	152	108	77	52	30	12	-6	-24	-44	-66	-91	-121	-163	-233
25	228	158	118	89	66	42	21	-1	-20	-41	-65	-88	-117	-157	-227
26	217	148	107	78	55	36	18	0	-17	-35	-54	-77	-106	-147	-216
27	220	150	111	81	58	38	19	1	-18	-37	-57	-80	-110	-149	-219
28	225	157	117	87	62	40	20	1	-18	-39	-61	-86	-116	-156	-224
29	215	147	108	79	54	34	17	1	-16	-33	-53	-78	-107	-146	-214
30	206	139	101	73	51	33	18	3	-13	-31	-50	-72	-100	-138	-205

Γ_{ch} (in LLR format)														
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
5.44	4	3.04	2.28	1.64	1.06	0.52	0	-0.52	-1.06	-1.64	-2.28	-3.04	-4.00	-5.44

Iteration	Γ_e
	γ_1
1	0
2	0
3	1
4-8	0
9-15	1
16	0
17	1
18	1
19	1
20	0
21	1
22	0
23	1
24	0
25	1
26	0
27	1
28	1
29	0
30	0

TABLE XXIX: Threshold sets Γ_v and Γ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate 5/6

Iteration	Γ_v														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
1	331	242	185	140	102	66	32	1	-31	-65	-101	-139	-184	-241	-330
2	340	247	188	141	99	62	30	1	-29	-61	-98	-140	-187	-246	-339
3	341	248	188	141	98	59	26	-4	-34	-64	-99	-140	-187	-247	-340
4	345	251	190	141	98	63	31	1	-30	-62	-97	-140	-189	-250	-344
5	344	248	186	137	98	62	31	1	-29	-61	-97	-136	-185	-247	-343
6	346	249	186	138	98	62	31	1	-28	-59	-93	-134	-185	-248	-345
7	349	252	189	141	99	62	30	0	-29	-61	-98	-140	-188	-251	-348
8	348	250	187	137	95	61	30	1	-28	-59	-93	-135	-185	-249	-347
9	351	251	187	138	97	61	30	1	-28	-60	-96	-137	-186	-250	-350
10	350	251	187	138	96	60	30	1	-29	-59	-95	-137	-186	-250	-349
11	354	253	188	138	95	60	29	1	-28	-59	-94	-137	-187	-252	-353
12	356	255	189	138	95	60	29	0	-28	-59	-94	-137	-188	-254	-355
13	358	255	189	138	95	60	29	1	-28	-59	-94	-137	-188	-254	-357
14	358	255	188	136	95	60	29	0	-28	-59	-94	-135	-187	-254	-357
15	361	257	189	138	97	62	31	1	-28	-59	-94	-137	-188	-256	-360
16	361	257	190	138	96	61	30	1	-29	-60	-95	-137	-189	-256	-360
17	362	257	189	137	95	60	30	0	-29	-59	-94	-136	-188	-256	-361
18	363	258	189	137	95	60	30	0	-29	-59	-94	-136	-188	-257	-362
19	363	257	189	138	98	63	32	1	-30	-62	-97	-137	-188	-256	-362
20	366	258	191	140	98	63	31	1	-30	-62	-97	-139	-190	-257	-365
21	368	261	192	140	98	62	31	1	-30	-61	-97	-139	-191	-260	-367
22	374	263	192	140	99	64	32	1	-31	-63	-98	-139	-191	-262	-373
23	370	260	189	137	98	65	32	1	-31	-64	-97	-136	-188	-259	-369
24	395	282	212	157	110	70	34	1	-33	-69	-109	-156	-211	-281	-394
25	415	303	232	178	132	91	53	17	-20	-60	-100	-145	-201	-273	-391
26	403	284	206	147	98	56	19	-18	-55	-93	-134	-178	-232	-303	-419
27	407	283	206	147	98	56	18	-17	-54	-93	-135	-182	-238	-311	-431
28	427	296	212	150	101	58	20	-15	-50	-87	-129	-177	-233	-309	-435
29	444	312	234	178	131	89	50	15	-19	-55	-96	-147	-211	-297	-433
30	441	304	224	166	117	77	39	1	-38	-76	-116	-165	-223	-303	-440

TABLE XXVIII: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate 5/6

Iteration	$\phi_v(s), s \in \mathcal{S}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	259	183	136	97	66	40	19	4	-4	-19	-40	-66	-97	-136	-183	-259
2	257	191	145	108	76	48	23	6	-6	-23	-48	-76	-108	-145	-191	-257
3	268	198	152	113	78	47	23	6	-6	-23	-47	-78	-113	-152	-198	-268
4	272	201	154	115	80	49	24	6	-6	-24	-49	-80	-115	-154	-201	-272
5	276	206	158	117	81	50	25	7	-7	-25	-50	-81	-117	-158	-206	-276
6	278	206	156	114	80	51	26	7	-7	-26	-51	-80	-114	-156	-206	-278
7	281	208	158	116	80	50	26	7	-7	-26	-50	-80	-116	-158	-208	-281
8	284	211	161	120	84	52	26	7	-7	-26	-52	-84	-120	-161	-211	-284
9	284	211	160	117	81	50	26	7	-7	-26	-50	-81	-117	-160	-211	-284
10	288	213	162	119	83	53	27	7	-7	-27	-53	-83	-119	-162	-213	-288
11	288	214	162	120	83	52	27	8	-8	-27	-52	-83	-120	-162	-214	-288
12	292	218	164	120	83	52	27	8	-8	-27	-52	-83	-120	-164	-218	-292
13	295	220	166	121	83	52	27	8	-8	-27	-52	-83	-121	-166	-220	-295
14	298	221	167	122	84	53	28	8	-8	-28	-53	-84	-122	-167	-221	-298
15	300	222	167	122	84	53	28	8	-8	-28	-53	-84	-122	-167	-222	-300
16	302	224	169	124	86	54	29	8	-8	-29	-54	-86	-124	-169	-224	-302
17	304	226	170	124	86	55	29	8	-8	-29	-55	-86	-124	-170	-226	-304
18	306	227	171	125	86	55	30	9	-9	-30	-55	-86	-125	-171	-227	-306
19	308	228	172	125	87	56	30	9	-9	-30	-56	-87	-125	-172	-228	-308
20	310	230	173	127	89	58	32	9	-9	-32	-58	-89	-127	-173	-230	-310
21	314	233	176	130	91	59	32	10	-10	-32	-59	-91	-130	-176	-233	-314
22	317	237	178	131	93	60	33	10	-10	-33	-60	-93	-131	-178	-237	-317
23	326	240	180	133	94	62	35	11	-11	-35	-62	-94	-133	-180	-240	-326
24	324	242	180	131	93	63	36	11	-11	-36	-63	-93	-131	-180	-242	-324
25	348	263	202	152	109	72	39	12	-12	-39	-72	-109	-152	-202	-263	-348
26	361	271	209	158	116	79	44	13	-13	-44	-79	-116	-158	-209	-271	-361
27	374	282	216	162	118	80	46	14	-14	-46	-80	-118	-162	-216	-282	-374
28	388	292	224	168	122	83	47	14	-14	-47	-83	-122	-168	-224	-292	-388
29	409	308	232	172	125	84	47	14	-14	-47	-84	-125	-172	-232	-308	-409
30	409	307	227	167	121	82	46	16	-16	-46	-82	-121	-167	-227	-307	-409

	Γ_{ch} (in LLR format)														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
	6.62	4.78	3.60	2.66	1.90	1.22	0.60	0	-0.60	-1.22	-1.90	-2.66	-3.60	-4.78	-6.62

Iteration	$\phi_{ch}(l), l \in \mathcal{L}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1-28	409	279	207	155	113	78	45	15	-15	-45	-78	-113	-155	-207	-279	-409
29	406	277	206	154	113	77	45	15	-15	-45	-77	-113	-154	-206	-277	-406
30	382	260	194	145	106	73	42	14	-14	-42	-73	-106	-145	-194	-260	-382

Iteration	Γ_e
	γ_1
1-6	0
7	1
8	1
9	0
10	1
11	0
12	0
13	0
14-22	1
23	0
24	1
25	1
26	0
27	0
28	0
29	1
30	1

I. The 3-bit MIM-QMS decoder in Example 4

TABLE XXX: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-17664 IEEE 802.3ca LDPC code with code rate 0.826

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$							
	0	1	2	3	4	5	6	7
1	85	47	22	4	-4	-22	-47	-85
2	85	50	25	5	-5	-25	-50	-85
3	85	51	25	5	-5	-25	-51	-85
4	87	52	25	6	-6	-25	-52	-87
5	88	53	27	6	-6	-27	-53	-88
6	90	54	27	7	-7	-27	-54	-90
7	91	55	27	7	-7	-27	-55	-91
8	91	55	28	7	-7	-28	-55	-91
9	93	58	30	7	-7	-30	-58	-93
10	97	60	31	8	-8	-31	-60	-97
11	102	66	35	9	-9	-35	-66	-102
12	100	63	32	8	-8	-32	-63	-100
13	105	67	36	9	-9	-36	-67	-105
14	101	65	34	9	-9	-34	-65	-101
15	103	68	36	10	-10	-36	-68	-103
16	103	66	35	9	-9	-35	-66	-103
17	104	69	38	10	-10	-38	-69	-104
18	105	67	34	9	-9	-34	-67	-105
19	114	72	38	10	-10	-38	-72	-114
20	107	69	35	9	-9	-35	-69	-107
21	117	74	40	11	-11	-40	-74	-117
22	115	72	37	9	-9	-37	-72	-115
23	122	75	39	11	-11	-39	-75	-122
24	119	74	38	10	-10	-38	-74	-119
25	128	79	41	11	-11	-41	-79	-128
26	124	78	40	10	-10	-40	-78	-124
27	137	83	41	11	-11	-41	-83	-137
28	145	87	42	11	-11	-42	-87	-145
29	148	90	44	13	-13	-44	-90	-148
30	157	96	48	14	-14	-48	-96	-157

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$							
	0	1	2	3	4	5	6	7
1-29	157	86	45	14	-14	-45	-86	-157
30	155	85	45	14	-14	-45	-85	-155

TABLE XXXI: Threshold sets Γ_v and Γ_{ch} for the length-17664 IEEE 802.3ca LDPC code with code rate 0.826

Iteration	Γ_v						
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
1	114	65	31	1	-30	-64	-113
2	113	62	28	-1	-28	-61	-112
3	114	64	28	-1	-28	-63	-113
4	114	63	30	1	-29	-62	-113
5	116	63	30	-1	-29	-62	-115
6	115	64	31	4	-26	-62	-114
7	115	64	29	-1	-29	-63	-114
8	116	62	27	-3	-32	-68	-117
9	117	63	27	-2	-34	-72	-126
10	128	73	35	1	-34	-72	-127
11	122	66	27	-4	-35	-72	-126
12	128	72	35	1	-34	-71	-127
13	125	71	36	3	-29	-69	-124
14	127	73	34	1	-33	-72	-126
15	125	71	35	3	-29	-68	-124
16	129	74	35	0	-34	-73	-128
17	127	70	32	-1	-31	-69	-126
18	134	76	34	1	-33	-75	-133
19	126	69	31	0	-30	-68	-125
20	135	75	35	0	-34	-74	-134
21	132	72	32	1	-31	-71	-131
22	135	75	36	3	-29	-70	-134
23	132	70	31	0	-30	-69	-131
24	137	71	30	-3	-34	-77	-139
25	132	70	30	1	-29	-69	-131
26	140	72	30	1	-29	-71	-139
27	144	73	31	0	-30	-72	-143
28	146	76	35	5	-26	-66	-135
29	146	75	34	3	-30	-72	-144
30	148	73	31	0	-30	-72	-147

Γ_{ch} (in LLR format)						
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
4.80	2.68	1.24	0	-1.24	-2.68	-4.80

Iteration	Γ_e
	γ_1
1	1
2	0
3	2
4	1
5	2
6	1
7	0
8	1
9	0
10	0
11	0
12	1
13	1
14	0
15	1
16	0
17	1
18	1
19	0
20	1
21	1
22	0
23	1
24	0
25	0
26	1
27	0
28	1
29	1
30	1

J. The 4-bit MIM-QMS decoder in Example 4

TABLE XXXII: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-17664 IEEE 802.3ca LDPC code with code rate 0.826

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	97	68	50	35	23	14	6	1	-1	-6	-14	-23	-35	-50	-68	-97
2	96	69	52	37	25	15	7	2	-2	-7	-15	-25	-37	-52	-69	-96
3	99	72	54	40	27	17	8	2	-2	-8	-17	-27	-40	-54	-72	-99
4	101	74	55	40	27	17	9	2	-2	-9	-17	-27	-40	-55	-74	-101
5	103	75	56	42	29	18	9	2	-2	-9	-18	-29	-42	-56	-75	-103
6	104	78	59	43	30	19	9	2	-2	-9	-19	-30	-43	-59	-78	-104
7	106	79	60	45	31	20	10	3	-3	-10	-20	-31	-45	-60	-79	-106
8	107	78	59	43	29	19	10	3	-3	-10	-19	-29	-43	-59	-78	-107
9	108	80	61	45	32	20	11	3	-3	-11	-20	-32	-45	-61	-80	-108
10	109	81	60	44	30	19	10	3	-3	-10	-19	-30	-44	-60	-81	-109
11	112	84	63	47	33	21	11	3	-3	-11	-21	-33	-47	-63	-84	-112
12	112	82	62	45	31	20	10	3	-3	-10	-20	-31	-45	-62	-82	-112
13	114	84	64	47	34	22	12	3	-3	-12	-22	-34	-47	-64	-84	-114
14	115	85	65	48	34	22	11	3	-3	-11	-22	-34	-48	-65	-85	-115
15	116	87	66	48	34	21	11	3	-3	-11	-21	-34	-48	-66	-87	-116
16	117	87	66	49	34	22	12	3	-3	-12	-22	-34	-49	-66	-87	-117
17	120	88	67	50	34	21	11	3	-3	-11	-21	-34	-50	-67	-88	-120
18	120	90	68	50	35	22	11	3	-3	-11	-22	-35	-50	-68	-90	-120
19	122	90	68	50	35	22	12	3	-3	-12	-22	-35	-50	-68	-90	-122
20	123	90	69	51	36	23	12	3	-3	-12	-23	-36	-51	-69	-90	-123
21	124	91	69	51	36	23	12	4	-4	-12	-23	-36	-51	-69	-91	-124
22	126	92	69	51	36	23	13	4	-4	-13	-23	-36	-51	-69	-92	-126
23	129	94	70	52	37	24	13	4	-4	-13	-24	-37	-52	-70	-94	-129
24	134	97	72	53	37	25	14	4	-4	-14	-25	-37	-53	-72	-97	-134
25	138	100	74	54	38	25	15	5	-5	-15	-25	-38	-54	-74	-100	-138
26	146	107	80	60	42	28	15	5	-5	-15	-28	-42	-60	-80	-107	-146
27	156	116	86	64	47	32	19	6	-6	-19	-32	-47	-64	-86	-116	-156
28	157	116	86	64	45	29	16	5	-5	-16	-29	-45	-64	-86	-116	-157
29	157	117	88	66	47	31	19	6	-6	-19	-31	-47	-66	-88	-117	-157
30	157	115	83	61	44	30	18	6	-6	-18	-30	-44	-61	-83	-115	-157

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1-27	157	107	80	60	44	30	17	6	-6	-17	-30	-44	-60	-80	-107	-157
28	151	103	76	57	42	29	17	6	-6	-17	-29	-42	-57	-76	-103	-151
29	137	93	69	52	38	26	15	5	-5	-15	-26	-38	-52	-69	-93	-137
30	124	84	63	47	34	23	14	5	-5	-14	-23	-34	-47	-63	-84	-124

TABLE XXXIII: Threshold sets Γ_v and Γ_{ch} for the length-17664 IEEE 802.3ca LDPC code with code rate 0.826

Iteration	Γ_v														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
1	127	92	69	51	37	24	12	1	-11	-23	-36	-50	-68	-91	-126
2	129	94	72	54	39	26	13	2	-10	-23	-36	-51	-68	-91	-127
3	131	96	75	57	42	29	16	4	-8	-20	-32	-47	-66	-91	-128
4	133	97	74	57	42	28	16	4	-8	-20	-34	-49	-67	-91	-129
5	133	97	73	55	39	25	12	1	-11	-24	-38	-54	-72	-96	-132
6	134	98	74	55	39	26	13	0	-13	-26	-40	-56	-74	-97	-133
7	135	99	75	57	41	28	15	3	-9	-20	-33	-49	-67	-93	-131
8	135	98	74	56	40	26	13	1	-12	-25	-39	-55	-73	-97	-134
9	134	95	69	50	34	22	10	-2	-14	-27	-40	-56	-75	-99	-137
10	139	101	76	56	41	26	13	0	-12	-25	-40	-55	-75	-100	-138
11	139	101	76	57	40	27	14	2	-9	-21	-34	-50	-70	-95	-135
12	140	101	76	56	41	27	14	1	-12	-25	-40	-55	-75	-100	-139
13	140	101	76	57	41	26	12	0	-12	-25	-40	-56	-75	-100	-139
14	141	102	76	56	39	25	13	1	-11	-24	-38	-55	-75	-101	-140
15	142	102	76	57	41	27	15	2	-10	-24	-38	-55	-75	-101	-141
16	144	103	77	56	39	25	12	0	-11	-24	-38	-55	-76	-102	-143
17	144	104	78	57	39	24	11	0	-12	-24	-38	-56	-77	-103	-143
18	144	103	77	56	39	25	12	1	-11	-24	-38	-55	-76	-102	-143
19	144	102	77	56	40	25	12	0	-11	-24	-39	-55	-76	-101	-143
20	145	103	76	56	40	25	12	0	-11	-24	-39	-55	-75	-102	-144
21	145	102	75	56	39	25	13	1	-12	-24	-38	-55	-74	-101	-144
22	146	102	75	55	39	25	13	2	-11	-24	-38	-54	-74	-101	-145
23	150	104	76	55	39	25	12	0	-12	-24	-38	-54	-75	-103	-149
24	151	104	75	55	38	25	13	0	-12	-24	-37	-54	-74	-103	-150
25	163	115	85	62	43	27	14	1	-11	-24	-38	-54	-75	-104	-152
26	171	121	91	69	51	35	19	4	-10	-23	-38	-55	-77	-109	-160
27	167	116	84	61	42	25	12	0	-13	-25	-41	-60	-83	-115	-166
28	178	125	94	70	50	35	21	7	-8	-23	-39	-59	-82	-112	-165
29	170	118	86	65	49	34	21	7	-6	-20	-35	-52	-75	-105	-157
30	160	107	77	56	40	26	13	1	-11	-25	-39	-55	-76	-106	-159

Γ_{ch} (in LLR format)														
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
6.54	4.72	3.54	2.64	1.88	1.20	0.60	0	-0.60	-1.20	-1.88	-2.64	-3.54	-4.72	-6.54

Iteration	Γ_e
	γ_1
1	0
2	0
3	1
4	1
5	1
6	0
7	0
8	0
9-12	1
13	0
14	0
15	1
16	0
17	0
18-28	1
29	0
30	0

APPENDIX B
CONSTRUCTED LUTs FOR MIM-LQMS DECODERS

A. The 3-bit MIM-LQMS decoder in Example 2

TABLE XXXIV: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-560 5G LDPC code with code rate $1/2$

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$							
	0	1	2	3	4	5	6	7
1	144	77	31	6	-6	-31	-77	-144
2	156	85	36	7	-7	-36	-85	-156
3	167	90	39	8	-8	-39	-90	-167
4	175	93	41	9	-9	-41	-93	-175
5	184	98	44	10	-10	-44	-98	-184
6	195	105	48	12	-12	-48	-105	-195
7	204	113	55	14	-14	-55	-113	-204
8	204	113	56	15	-15	-56	-113	-204
9	204	115	55	15	-15	-55	-115	-204
10	204	113	54	15	-15	-54	-113	-204
11	204	111	53	16	-16	-53	-111	-204
12	204	107	51	15	-15	-51	-107	-204
13	204	105	50	15	-15	-50	-105	-204
14	204	103	49	15	-15	-49	-103	-204
15	204	103	49	15	-15	-49	-103	-204

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$							
	0	1	2	3	4	5	6	7
1-6	204	120	66	21	-21	-66	-120	-204
7	193	114	62	20	-20	-62	-114	-193
8	157	92	51	16	-16	-51	-92	-157
9	142	83	46	15	-15	-46	-83	-142
10	132	78	43	14	-14	-43	-78	-132
11	122	72	39	13	-13	-39	-72	-122
12	116	68	37	12	-12	-37	-68	-116
13	112	66	36	12	-12	-36	-66	-112
14	110	64	35	11	-11	-35	-64	-110
15	109	64	35	11	-11	-35	-64	-109

TABLE XXXV: Threshold sets Γ_v and Γ_{ch} for the length-560 5G LDPC code with code rate $1/2$

Iteration	Γ_v						
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
1	165	95	44	0	-45	-95	-164
2	174	98	46	0	-45	-97	-173
3	179	98	45	1	-44	-97	-178
4	182	99	45	0	-45	-98	-181
5	189	104	48	1	-47	-103	-188
6	195	105	48	3	-45	-102	-192
7	239	140	71	13	-40	-103	-202
8	192	103	43	-8	-57	-115	-200
9	190	107	53	8	-37	-92	-177
10	183	103	50	6	-35	-88	-174
11	169	84	33	-6	-46	-95	-173
12	168	92	46	6	-30	-76	-158
13	155	76	30	-6	-42	-87	-162
14	160	87	42	6	-29	-73	-151
15	152	74	30	-6	-41	-85	-158

Γ_{ch} (in LLR format)						
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
3.06	1.78	0.84	0	-0.84	-1.78	-3.06

Iteration	Γ_e
	γ_1
1	-1
2	2
3	1
4	0
5	1
6	3
7	1
8	0
9	3
10	0
11	0
12	1
13	0
14	1
15	0

B. The 4-bit MIM-LQMS decoder in Example 2

TABLE XXXVI: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-560 5G LDPC code with code rate $1/2$

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	161	115	81	56	36	20	9	2	-2	-9	-20	-36	-56	-81	-115	-161
2	172	121	86	60	39	22	10	2	-2	-10	-22	-39	-60	-86	-121	-172
3	188	133	96	68	45	27	13	3	-3	-13	-27	-45	-68	-96	-133	-188
4	204	146	106	76	52	32	15	4	-4	-15	-32	-52	-76	-106	-146	-204
5	204	145	104	75	51	30	14	3	-3	-14	-30	-51	-75	-104	-145	-204
6	204	143	102	73	50	30	14	4	-4	-14	-30	-50	-73	-102	-143	-204
7	204	141	101	72	50	31	16	4	-4	-16	-31	-50	-72	-101	-141	-204
8	204	138	98	71	50	32	17	5	-5	-17	-32	-50	-71	-98	-138	-204
9	204	135	95	69	50	32	18	5	-5	-18	-32	-50	-69	-95	-135	-204
10	204	131	92	67	49	34	18	5	-5	-18	-34	-49	-67	-92	-131	-204
11	204	127	87	64	47	33	17	4	-4	-17	-33	-47	-64	-87	-127	-204
12	204	123	84	62	46	32	17	4	-4	-17	-32	-46	-62	-84	-123	-204
13	204	120	82	60	45	32	17	4	-4	-17	-32	-45	-60	-82	-120	-204
14	204	118	80	59	44	32	17	5	-5	-17	-32	-44	-59	-80	-118	-204
15	204	117	80	59	44	32	18	6	-6	-18	-32	-44	-59	-80	-117	-204

Γ_{ch} (in LLR format)														
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
4.12	3.08	2.36	1.78	1.28	0.82	0.40	-0.02	-0.44	-0.86	-1.32	-1.82	-2.40	-3.12	-4.16

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	203	147	112	86	63	44	25	8	-9	-27	-45	-65	-87	-114	-148	-204
2	203	147	112	86	63	44	25	8	-9	-27	-45	-65	-87	-114	-148	-204
3	203	147	112	86	63	44	25	8	-9	-27	-45	-65	-87	-114	-148	-204
4	195	141	108	82	61	42	24	8	-9	-26	-44	-63	-84	-109	-143	-196
5	174	126	96	73	54	37	22	7	-8	-23	-39	-56	-75	-97	-127	-175
6	167	121	92	70	52	36	21	6	-8	-22	-37	-54	-72	-93	-122	-168
7	159	115	88	67	50	34	20	6	-7	-21	-36	-51	-69	-89	-116	-160
8	152	110	84	64	47	33	19	6	-7	-20	-34	-49	-65	-85	-111	-153
9	143	104	79	61	45	31	18	6	-7	-19	-32	-46	-62	-80	-105	-144
10	134	97	74	57	42	29	17	5	-6	-18	-30	-43	-58	-75	-98	-135
11	125	90	69	53	39	27	16	5	-6	-17	-28	-40	-54	-70	-91	-126
12	119	86	66	50	37	25	15	5	-6	-16	-26	-38	-51	-66	-87	-119
13	115	83	63	48	36	25	14	4	-5	-15	-26	-37	-49	-64	-84	-115
14	113	82	63	48	35	24	14	4	-5	-15	-25	-36	-49	-63	-83	-114
15	112	81	62	47	35	24	14	4	-5	-15	-25	-36	-48	-63	-82	-113

TABLE XXXVII: Threshold sets Γ_v and Γ_{ch} for the length-560 5G LDPC code with code rate $1/2$

Iteration	Γ_v														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
1	181	132	101	76	54	36	18	1	-17	-35	-55	-76	-101	-132	-181
2	190	136	103	77	55	35	17	0	-17	-36	-56	-78	-104	-137	-190
3	229	169	130	100	75	53	33	11	-11	-33	-54	-76	-102	-138	-195
4	225	165	125	95	70	45	22	1	-19	-40	-65	-92	-123	-163	-223
5	206	149	112	85	62	40	20	1	-17	-36	-58	-83	-111	-148	-206
6	202	144	108	82	60	39	19	1	-16	-34	-56	-80	-108	-144	-202
7	200	141	105	79	59	41	23	4	-14	-33	-54	-77	-103	-139	-198
8	195	135	100	76	57	40	22	4	-13	-31	-51	-73	-99	-134	-194
9	190	128	94	71	52	34	16	0	-16	-34	-52	-71	-94	-128	-190
10	184	121	88	66	49	32	15	0	-15	-32	-49	-66	-88	-121	-184
11	175	113	81	61	46	30	14	0	-14	-30	-46	-61	-81	-113	-175
12	168	107	77	58	43	28	13	0	-13	-28	-43	-58	-77	-107	-168
13	164	103	74	56	42	28	13	0	-13	-28	-42	-56	-74	-103	-163
14	161	101	73	56	42	30	17	3	-12	-27	-41	-55	-73	-101	-161
15	160	100	73	55	41	27	12	0	-12	-27	-41	-55	-73	-100	-160

Iteration	Γ_e
	γ_1
1	1
2	1
3-6	0
7	1
8	0
9	2
10	1
11	0
12	1
13	0
14	1
15	0

C. The 3-bit MIM-LQMS decoders in Example 3

TABLE XXXVIII: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $1/2$

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$							
	0	1	2	3	4	5	6	7
1	113	65	30	7	-7	-30	-65	-113
2	136	79	38	10	-10	-38	-79	-136
3	161	94	48	13	-13	-48	-94	-161
4	170	100	51	14	-14	-51	-100	-170
5	170	99	49	13	-13	-49	-99	-170
6	170	99	51	15	-15	-51	-99	-170
7	170	98	52	16	-16	-52	-98	-170
8	170	98	51	16	-16	-51	-98	-170
9	170	98	50	16	-16	-50	-98	-170
10	170	97	50	16	-16	-50	-97	-170
11	170	96	49	16	-16	-49	-96	-170
12	170	96	47	14	-14	-47	-96	-170
13	170	96	47	13	-13	-47	-96	-170
14	170	98	50	12	-12	-50	-98	-170
15	170	97	50	13	-13	-50	-97	-170

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$							
	0	1	2	3	4	5	6	7
1	170	98	54	17	-17	-54	-98	-170
2	170	98	54	17	-17	-54	-98	-170
3	170	98	54	17	-17	-54	-98	-170
4	152	88	48	15	-15	-48	-88	-152
5	140	81	44	14	-14	-44	-81	-140
6	134	78	42	14	-14	-42	-78	-134
7	128	74	40	13	-13	-40	-74	-128
8	123	71	39	12	-12	-39	-71	-123
9	119	69	38	12	-12	-38	-69	-119
10	114	66	36	12	-12	-36	-66	-114
11	109	63	34	11	-11	-34	-63	-109
12	105	61	33	11	-11	-33	-61	-105
13	100	58	32	10	-10	-32	-58	-100
14	95	55	30	10	-10	-30	-55	-95
15	93	54	29	9	-9	-29	-54	-93

TABLE XXXIX: Threshold sets Γ_v and Γ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $1/2$

Iteration	Γ_v						
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
1	142	81	37	0	-36	-80	-141
2	154	85	39	0	-38	-84	-153
3	179	106	58	13	-35	-91	-173
4	182	108	59	18	-25	-77	-157
5	149	74	25	-11	-51	-101	-171
6	158	90	46	6	-35	-81	-152
7	151	81	37	1	-36	-80	-150
8	149	80	36	1	-35	-79	-148
9	146	78	36	0	-35	-77	-145
10	144	76	35	0	-34	-75	-143
11	141	73	32	1	-31	-72	-140
12	139	71	32	2	-30	-70	-138
13	140	74	31	2	-30	-73	-139
14	136	73	31	2	-30	-72	-133
15	135	72	30	3	-29	-71	-132

	Γ_{ch} (in LLR format)					
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
3.50	2.02	0.94	0	-0.94	-2.02	-3.50

Iteration	Γ_e
	γ_1
1	1
2	0
3	1
4	0
5	1
6	1
7	0
8	1
9	0
10	0
11	0
12	2
13	0
14	0
15	1

TABLE XL: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $2/3$

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$							
	0	1	2	3	4	5	6	7
1	135	78	37	8	-8	-37	-78	-135
2	144	86	42	10	-10	-42	-86	-144
3	161	98	50	13	-13	-50	-98	-161
4	178	108	55	14	-14	-55	-108	-178
5	189	113	57	15	-15	-57	-113	-189
6	206	123	62	17	-17	-62	-123	-206
7	217	133	69	18	-18	-69	-133	-217
8	227	138	71	20	-20	-71	-138	-227
9	227	136	68	19	-19	-68	-136	-227
10	227	136	69	21	-21	-69	-136	-227
11	227	138	75	24	-24	-75	-138	-227
12	227	134	69	23	-23	-69	-134	-227
13	227	131	65	20	-20	-65	-131	-227
14	227	128	64	17	-17	-64	-128	-227
15	227	126	64	15	-15	-64	-126	-227

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$							
	0	1	2	3	4	5	6	7
1-7	227	129	69	22	-22	-69	-129	-227
8	224	127	68	22	-22	-68	-127	-224
9	209	118	64	20	-20	-64	-118	-209
10	191	108	58	18	-18	-58	-108	-191
11	171	97	52	17	-17	-52	-97	-171
12	157	89	48	15	-15	-48	-89	-157
13	144	81	44	14	-14	-44	-81	-144
14	132	75	40	13	-13	-40	-75	-132
15	123	70	37	12	-12	-37	-70	-123

Γ_{ch} (in LLR format)						
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
4.08	2.32	1.08	0	-1.08	-2.32	-4.08

Iteration	Γ_e
	γ_1
1	1
2	2
3	1
4	0
5	2
6	1
7	0
8	0
9-13	1
14	4
15	3

TABLE XLI: Threshold sets Γ_v and Γ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $2/3$

Iteration	Γ_v						
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
1	171	97	45	0	-44	-96	-170
2	178	101	47	2	-46	-100	-177
3	196	111	51	1	-50	-110	-195
4	196	106	47	-2	-49	-107	-195
5	204	108	48	2	-47	-107	-203
6	234	139	78	27	-30	-96	-198
7	252	151	84	27	-32	-100	-204
8	210	105	40	-12	-63	-129	-234
9	224	124	61	12	-37	-105	-212
10	203	105	46	-6	-63	-132	-230
11	204	109	51	0	-50	-108	-203
12	194	104	49	4	-41	-99	-193
13	187	97	41	1	-36	-96	-186
14	179	95	40	4	-36	-94	-176
15	178	93	37	3	-31	-91	-175

TABLE XLII: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $5/6$

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$							
	0	1	2	3	4	5	6	7
1	230	135	66	15	-15	-66	-135	-230
2	238	147	75	19	-19	-75	-147	-238
3	250	155	80	20	-20	-80	-155	-250
4	259	161	84	22	-22	-84	-161	-259
5	269	169	89	24	-24	-89	-169	-269
6	283	176	93	26	-26	-93	-176	-283
7	286	179	95	27	-27	-95	-179	-286
8	297	187	99	28	-28	-99	-187	-297
9	315	200	104	29	-29	-104	-200	-315
10	343	217	113	31	-31	-113	-217	-343
11	380	242	127	37	-37	-127	-242	-380
12	409	250	126	37	-37	-126	-250	-409
13	409	230	117	41	-41	-117	-230	-409
14	409	213	107	38	-38	-107	-213	-409
15	409	200	97	24	-24	-97	-200	-409

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$							
	0	1	2	3	4	5	6	7
1-11	409	222	117	37	-37	-117	-222	-409
12	373	202	107	34	-34	-107	-202	-373
13	296	160	85	27	-27	-85	-160	-296
14	236	128	68	21	-21	-68	-128	-236
15	204	110	58	18	-18	-58	-110	-204

TABLE XLIII: Threshold sets Γ_v and Γ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $5/6$

Iteration	Γ_v						
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
1	297	167	80	3	-75	-164	-296
2	310	177	86	7	-68	-159	-302
3	312	174	82	7	-72	-168	-309
4	317	176	80	0	-78	-172	-314
5	329	180	84	1	-78	-176	-326
6	326	179	82	2	-78	-178	-325
7	328	180	83	1	-79	-177	-325
8	337	185	84	1	-81	-183	-334
9	337	168	59	-27	-107	-204	-357
10	347	170	54	-38	-129	-235	-390
11	367	180	61	-31	-118	-226	-401
12	390	194	83	-6	-93	-207	-379
13	353	183	88	2	-81	-175	-343
14	304	157	66	1	-61	-154	-303
15	275	134	58	13	-45	-131	-267

Γ_{ch} (in LLR format)						
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
4.90	2.72	1.24	0	-1.24	-2.72	-4.90

Iteration	Γ_e
	γ_1
1	0
2	3
3-8	1
9	0
10	3
11	1
12	1
13	0
14	1
15	5

D. The 4-bit MIM-LQMS decoders in Example 3

TABLE XLIV: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $1/2$

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	122	88	64	46	31	19	9	2	-2	-9	-19	-31	-46	-64	-88	-122
2	140	103	77	57	39	24	12	3	-3	-12	-24	-39	-57	-77	-103	-140
3	170	119	86	61	42	27	14	4	-4	-14	-27	-42	-61	-86	-119	-170
4	170	116	83	60	41	26	13	4	-4	-13	-26	-41	-60	-83	-116	-170
5	170	115	83	60	42	27	14	4	-4	-14	-27	-42	-60	-83	-115	-170
6	170	113	81	59	41	27	15	4	-4	-15	-27	-41	-59	-81	-113	-170
7	170	114	82	60	43	28	16	5	-5	-16	-28	-43	-60	-82	-114	-170
8	170	116	84	62	45	30	17	5	-5	-17	-30	-45	-62	-84	-116	-170
9	170	116	84	62	45	30	17	5	-5	-17	-30	-45	-62	-84	-116	-170
10	170	117	84	62	45	30	16	5	-5	-16	-30	-45	-62	-84	-117	-170
11	170	117	84	62	44	29	16	5	-5	-16	-29	-44	-62	-84	-117	-170
12	170	117	85	63	45	30	17	6	-6	-17	-30	-45	-63	-85	-117	-170
13	170	117	85	63	46	31	18	6	-6	-18	-31	-46	-63	-85	-117	-170
14	170	118	85	62	46	31	18	6	-6	-18	-31	-46	-62	-85	-118	-170
15	170	118	85	63	46	31	18	6	-6	-18	-31	-46	-63	-85	-118	-170

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	169	121	92	70	52	35	21	6	-8	-22	-37	-53	-71	-93	-122	-170
2	169	121	92	70	52	35	21	6	-8	-22	-37	-53	-71	-93	-122	-170
3	155	111	84	64	47	33	19	6	-7	-20	-34	-49	-65	-85	-112	-156
4	144	103	78	59	44	30	17	5	-7	-19	-31	-45	-60	-79	-103	-144
5	129	93	70	54	40	27	16	5	-6	-17	-28	-40	-54	-71	-93	-130
6	124	89	68	51	38	26	15	5	-6	-16	-27	-39	-52	-68	-90	-125
7	120	86	65	50	37	25	15	5	-5	-15	-26	-37	-50	-66	-86	-120
8	118	84	64	49	36	25	14	4	-5	-15	-25	-37	-49	-65	-85	-118
9	114	82	62	47	35	24	14	4	-5	-15	-25	-36	-48	-63	-82	-115
10	112	80	61	46	34	23	14	4	-5	-14	-24	-35	-47	-62	-81	-113
11	110	79	60	45	34	23	13	4	-5	-14	-24	-34	-46	-60	-79	-110
12	107	77	58	44	33	22	13	4	-5	-14	-23	-34	-45	-59	-77	-108
13	104	74	56	43	32	22	13	4	-5	-13	-22	-32	-44	-57	-74	-104
14	102	73	55	42	31	21	12	4	-5	-13	-22	-32	-43	-56	-73	-102
15	98	70	53	41	30	21	12	4	-4	-13	-21	-31	-41	-54	-71	-98

TABLE XLV: Threshold sets Γ_v and Γ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $1/2$

Iteration	Γ_v														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
1	150	110	85	64	46	29	14	-1	-15	-30	-47	-65	-85	-110	-150
2	189	137	102	76	54	35	17	1	-16	-34	-54	-76	-102	-137	-189
3	175	120	88	64	45	28	13	0	-14	-28	-45	-64	-88	-120	-175
4	178	125	93	67	46	28	12	-1	-14	-29	-46	-67	-93	-124	-178
5	168	117	87	65	49	34	19	6	-7	-22	-38	-56	-77	-106	-158
6	169	117	88	67	50	33	19	6	-6	-18	-33	-50	-72	-103	-155
7	155	105	76	56	41	27	13	-1	-15	-28	-44	-61	-83	-112	-159
8	161	113	84	62	44	28	14	1	-12	-26	-41	-56	-76	-105	-155
9	163	113	83	61	43	27	13	1	-12	-25	-39	-56	-75	-103	-153
10	163	113	83	61	43	27	13	1	-11	-24	-38	-54	-74	-102	-152
11	159	111	82	60	42	27	13	1	-11	-23	-36	-52	-72	-100	-149
12	165	115	87	65	48	34	20	7	-7	-21	-36	-54	-74	-101	-150
13	158	111	84	64	47	32	19	6	-5	-18	-32	-48	-69	-98	-147
14	157	110	83	63	47	32	18	5	-8	-20	-35	-51	-71	-99	-147
15	157	109	79	57	40	26	13	0	-13	-26	-40	-56	-78	-109	-156

Γ_{ch} (in LLR format)														
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
4.60	3.42	2.6000	1.96	1.40	0.90	0.44	-0.02	-0.48	-0.94	-1.44	-2.00	-2.64	-3.44	-4.62

Iteration	Γ_e
	γ_1
1	1
2	1
3	0
4	0
5	1
6	0
7	1
8	1
9	0
10	1
11	1
12	0
13	1
14	0
15	0

TABLE XLVI: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate $2/3$

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	151	109	80	57	38	23	11	3	-3	-11	-23	-38	-57	-80	-109	-151
2	159	116	87	63	43	27	13	3	-3	-13	-27	-43	-63	-87	-116	-159
3	168	124	93	68	46	29	14	4	-4	-14	-29	-46	-68	-93	-124	-168
4	177	130	98	71	49	30	16	4	-4	-16	-30	-49	-71	-98	-130	-177
5	185	135	100	73	51	32	16	4	-4	-16	-32	-51	-73	-100	-135	-185
6	193	139	103	75	52	33	17	5	-5	-17	-33	-52	-75	-103	-139	-193
7	200	144	107	78	54	34	18	5	-5	-18	-34	-54	-78	-107	-144	-200
8	211	153	112	82	57	36	19	6	-6	-19	-36	-57	-82	-112	-153	-211
9	227	165	122	89	62	39	21	6	-6	-21	-39	-62	-89	-122	-165	-227
10	227	165	122	89	63	42	23	7	-7	-23	-42	-63	-89	-122	-165	-227
11	227	164	121	88	62	42	24	7	-7	-24	-42	-62	-88	-121	-164	-227
12	227	162	119	87	62	42	24	8	-8	-24	-42	-62	-87	-119	-162	-227
13	227	161	118	88	64	44	24	8	-8	-24	-44	-64	-88	-118	-161	-227
14	227	161	117	87	63	43	25	8	-8	-25	-43	-63	-87	-117	-161	-227
15	227	160	115	85	62	42	24	7	-7	-24	-42	-62	-85	-115	-160	-227

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1-8	227	159	120	91	67	46	27	9	-9	-27	-46	-67	-91	-120	-159	-227
9	224	157	119	90	66	46	27	9	-9	-27	-46	-66	-90	-119	-157	-224
10	216	151	114	87	64	44	26	9	-9	-26	-44	-64	-87	-114	-151	-216
11	201	142	107	81	60	41	24	8	-8	-24	-41	-60	-81	-107	-142	-201
12	193	136	102	78	57	39	23	8	-8	-23	-39	-57	-78	-102	-136	-193
13	179	126	95	72	53	37	21	7	-7	-21	-37	-53	-72	-95	-126	-179
14	170	119	90	68	50	35	20	7	-7	-20	-35	-50	-68	-90	-119	-170
15	156	110	83	63	46	32	19	6	-6	-19	-32	-46	-63	-83	-110	-156

TABLE XLVII: Threshold sets Γ_v and Γ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate 2/3

Iteration	Γ_v														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
1	191	139	104	77	56	37	18	0	-19	-38	-59	-81	-107	-141	-192
2	198	145	110	82	59	39	19	1	-18	-38	-58	-81	-109	-144	-197
3	204	148	112	83	58	37	19	0	-18	-36	-57	-82	-111	-147	-203
4	209	150	113	84	60	39	20	2	-16	-36	-58	-83	-112	-149	-208
5	216	154	115	85	61	39	20	2	-17	-38	-60	-84	-114	-153	-215
6	216	152	112	82	58	37	18	0	-18	-36	-57	-81	-111	-151	-215
7	224	156	114	83	59	37	18	1	-17	-36	-58	-82	-113	-155	-223
8	245	175	130	94	64	40	20	0	-19	-39	-63	-93	-129	-174	-244
9	238	167	124	91	64	41	20	0	-19	-40	-63	-90	-123	-166	-237
10	255	182	137	102	76	53	32	11	-10	-32	-55	-83	-117	-163	-235
11	241	170	127	96	71	50	30	10	-9	-29	-51	-77	-109	-152	-223
12	228	156	114	84	59	35	14	-6	-24	-44	-66	-91	-121	-162	-233
13	221	153	113	85	61	40	20	0	-19	-39	-60	-84	-112	-152	-220
14	218	147	106	77	55	35	16	-1	-19	-39	-61	-86	-117	-157	-224
15	208	140	101	73	52	34	17	1	-16	-33	-51	-72	-100	-139	-207

Γ_{ch} (in LLR format)														
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
5.44	4	3.04	2.28	1.64	1.06	0.52	0	-0.52	-1.06	-1.64	-2.28	-3.04	-4	-5.44

Iteration	Γ_e
	γ_1
1	1
2	0
3	0
4	0
5	1
6	0
7	1
8	1
9	1
10	0
11	0
12	0
13	1
14	0
15	1

Iteration	$\phi_{ch}(l), l \in \mathcal{L}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1-12	409	278	207	155	114	78	45	15	-15	-45	-78	-114	-155	-207	-278	-409
13	399	271	202	152	111	76	44	14	-14	-44	-76	-111	-152	-202	-271	-399
14	337	230	171	128	94	64	37	12	-12	-37	-64	-94	-128	-171	-230	-337
15	280	190	141	106	78	53	31	10	-10	-31	-53	-78	-106	-141	-190	-280

TABLE XLIX: Threshold sets Γ_v and Γ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate 5/6

Iteration	Γ_v														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
1	335	247	189	143	102	66	32	0	-31	-65	-101	-142	-188	-246	-334
2	344	250	188	141	98	63	31	1	-30	-62	-97	-140	-187	-249	-343
3	347	248	186	138	98	62	30	0	-30	-61	-97	-137	-185	-247	-346
4	348	249	186	136	95	60	29	0	-29	-59	-94	-135	-185	-248	-347
5	353	253	188	137	95	60	29	0	-28	-59	-94	-136	-187	-252	-352
6	359	256	189	137	95	60	29	1	-28	-59	-94	-136	-188	-255	-358
7	360	257	190	139	98	62	31	0	-30	-61	-97	-138	-189	-256	-359
8	364	257	190	138	97	62	31	0	-30	-61	-96	-137	-189	-256	-363
9	366	260	191	140	99	63	31	1	-30	-62	-98	-139	-190	-259	-365
10	374	262	192	140	99	64	32	1	-31	-63	-98	-139	-191	-261	-373
11	394	281	209	154	109	69	33	-1	-35	-70	-109	-153	-208	-280	-393
12	413	290	210	148	98	55	18	-17	-54	-94	-136	-183	-238	-310	-429
13	437	307	230	172	123	82	42	2	-37	-74	-115	-164	-223	-302	-434
14	404	270	193	140	98	63	31	1	-30	-62	-97	-139	-192	-269	-403
15	374	256	190	144	107	75	46	15	-17	-46	-75	-112	-161	-233	-362

Γ_{ch} (in LLR format)														
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
6.62	4.78	3.58	2.68	1.90	1.22	0.60	0	-0.60	-1.22	-1.90	-2.68	-3.58	-4.78	-6.62

TABLE XLVIII: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-1296 IEEE 802.11n LDPC code with code rate 5/6

Iteration	$\phi_v(s), s \in \mathcal{S}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	261	190	142	104	71	44	21	5	-5	-21	-44	-71	-104	-142	-190	-261
2	269	201	154	114	80	50	25	7	-7	-25	-50	-80	-114	-154	-201	-269
3	277	206	157	116	81	50	26	7	-7	-26	-50	-81	-116	-157	-206	-277
4	283	209	159	117	81	51	26	7	-7	-26	-51	-81	-117	-159	-209	-283
5	287	213	162	118	82	51	27	7	-7	-27	-51	-82	-118	-162	-213	-287
6	294	220	166	121	83	52	27	8	-8	-27	-52	-83	-121	-166	-220	-294
7	301	224	168	123	85	54	28	8	-8	-28	-54	-85	-123	-168	-224	-301
8	306	227	171	126	88	57	31	9	-9	-31	-57	-88	-126	-171	-227	-306
9	311	231	174	128	89	58	32	9	-9	-32	-58	-89	-128	-174	-231	-311
10	319	238	179	132	93	61	34	10	-10	-34	-61	-93	-132	-179	-238	-319
11	339	252	190	141	100	66	37	11	-11	-37	-66	-100	-141	-190	-252	-339
12	372	280	214	160	116	78	44	13	-13	-44	-78	-116	-160	-214	-280	-372
13	409	305	230	171	124	84	49	15	-15	-49	-84	-124	-171	-230	-305	-409
14	409	295	216	161	117	80	48	16	-16	-48	-80	-117	-161	-216	-295	-409
15	409	276	194	140	102	69	39	12	-12	-39	-69	-102	-140	-194	-276	-409

Iteration	Γ_e
	γ_1
1	1
2	1
3-6	0
7	1
8-11	0
12	1
13	0
14	0
15	1

E. The 3-bit MIM-LQMS decoder in Example 4

TABLE L: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-17664 IEEE 802.3ca LDPC code with code rate 0.826

Iteration	$\phi_v(s), s \in \mathcal{S}$							
	0	1	2	3	4	5	6	7
1	85	49	23	5	-5	-23	-49	-85
2	85	51	25	6	-6	-25	-51	-85
3	89	53	27	6	-6	-27	-53	-89
4	91	56	29	7	-7	-29	-56	-91
5	95	59	30	8	-8	-30	-59	-95
6	100	64	34	9	-9	-34	-64	-100
7	101	66	35	9	-9	-35	-66	-101
8	103	67	36	9	-9	-36	-67	-103
9	105	68	36	9	-9	-36	-68	-105
10	110	70	36	10	-10	-36	-70	-110
11	111	71	37	10	-10	-37	-71	-111
12	116	74	38	10	-10	-38	-74	-116
13	129	78	39	10	-10	-39	-78	-129
14	139	83	41	11	-11	-41	-83	-139
15	152	91	44	13	-13	-44	-91	-152

Iteration	$\phi_{ch}(l), l \in \mathcal{L}$							
	0	1	2	3	4	5	6	7
1-15	157	86	45	14	-14	-45	-86	-157

TABLE LI: Threshold sets Γ_v and Γ_{ch} for the length-17664 IEEE 802.3ca LDPC code with code rate 0.826

Iteration	Γ_v						
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
1	112	62	29	0	-28	-61	-111
2	114	63	29	1	-28	-62	-113
3	116	63	30	2	-29	-62	-115
4	117	62	26	-4	-33	-68	-118
5	126	73	35	3	-27	-64	-116
6	126	73	34	0	-33	-72	-125
7	127	74	35	1	-34	-73	-126
8	126	70	30	-2	-33	-69	-125
9	134	75	35	3	-29	-69	-126
10	136	76	36	4	-29	-68	-126
11	129	71	31	1	-30	-70	-128
12	137	73	32	1	-31	-72	-136
13	138	71	31	0	-30	-70	-137
14	143	72	31	1	-30	-71	-142
15	146	73	32	0	-31	-72	-145

Γ_{ch} (in LLR format)						
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7
4.82	2.68	1.24	0	-1.24	-2.68	-4.82

Iteration	Γ_e
	γ_1
1	0
2	1
3	2
4	1
5	2
6	0
7	0
8-12	1
13	0
14	1
15	1

F. The 4-bit MIM-LQMS decoder in Example 4

TABLE LII: Reconstruction Functions ϕ_v and ϕ_{ch} for the length-17664 IEEE 802.3ca LDPC code with code rate 0.826

Iteration	$\phi_v(s)$, $s \in \mathcal{S}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	97	69	51	36	24	15	7	2	-2	-7	-15	-24	-36	-51	-69	-97
2	100	73	54	39	27	17	8	2	-2	-8	-17	-27	-39	-54	-73	-100
3	103	76	57	42	29	18	9	2	-2	-9	-18	-29	-42	-57	-76	-103
4	106	79	60	44	31	20	10	3	-3	-10	-20	-31	-44	-60	-79	-106
5	109	80	60	44	31	20	10	3	-3	-10	-20	-31	-44	-60	-80	-109
6	112	83	63	47	33	21	11	3	-3	-11	-21	-33	-47	-63	-83	-112
7	113	83	63	46	32	21	11	3	-3	-11	-21	-32	-46	-63	-83	-113
8	117	87	66	49	34	22	11	3	-3	-11	-22	-34	-49	-66	-87	-117
9	119	88	67	50	35	22	11	3	-3	-11	-22	-35	-50	-67	-88	-119
10	122	90	68	50	35	22	12	3	-3	-12	-22	-35	-50	-68	-90	-122
11	125	92	69	51	36	23	12	4	-4	-12	-23	-36	-51	-69	-92	-125
12	130	95	71	52	37	24	13	4	-4	-13	-24	-37	-52	-71	-95	-130
13	141	102	76	56	39	26	15	5	-5	-15	-26	-39	-56	-76	-102	-141
14	157	116	87	65	46	30	17	5	-5	-17	-30	-46	-65	-87	-116	-157
15	157	113	83	62	45	30	16	5	-5	-16	-30	-45	-62	-83	-113	-157

Iteration	$\phi_{ch}(l)$, $l \in \mathcal{L}$															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1-13	157	107	80	60	44	30	17	6	-6	-17	-30	-44	-60	-80	-107	-157
14	151	103	76	57	42	29	17	6	-6	-17	-29	-42	-57	-76	-103	-151
15	132	90	67	50	37	25	15	5	-5	-15	-25	-37	-50	-67	-90	-132

TABLE LIII: Threshold sets Γ_v and Γ_{ch} for the length-17664 IEEE 802.3ca LDPC code with code rate 0.826

Iteration	Γ_v														
	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
1	127	92	69	51	37	24	11	-1	-12	-23	-36	-50	-68	-91	-126
2	129	92	68	50	35	22	10	-2	-14	-26	-40	-55	-72	-95	-130
3	133	97	75	57	41	27	14	1	-11	-24	-38	-54	-72	-96	-132
4	132	94	69	50	34	21	10	-2	-13	-26	-40	-56	-74	-99	-135
5	138	100	75	57	41	27	14	1	-12	-25	-39	-55	-74	-99	-137
6	139	101	76	57	41	27	14	2	-9	-21	-34	-50	-70	-95	-134
7	142	103	77	57	40	25	11	-1	-13	-26	-40	-56	-76	-102	-141
8	142	102	77	57	40	26	13	1	-11	-24	-38	-55	-75	-101	-141
9	144	103	77	56	39	25	12	1	-11	-24	-38	-55	-76	-102	-143
10	144	102	77	56	40	25	12	1	-11	-24	-39	-55	-76	-101	-143
11	146	102	75	56	39	25	13	1	-12	-24	-38	-55	-74	-101	-145
12	150	104	76	55	39	25	13	0	-12	-24	-38	-54	-75	-103	-149
13	165	116	86	63	43	27	12	-1	-14	-27	-44	-62	-85	-115	-164
14	165	114	84	62	44	27	12	0	-11	-26	-43	-61	-83	-113	-164
15	162	110	80	60	45	31	18	6	-7	-19	-32	-48	-69	-99	-152

Γ_{ch} (in LLR format)														
γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}	γ_{11}	γ_{12}	γ_{13}	γ_{14}	γ_{15}
6.54	4.72	3.54	2.64	1.88	1.20	0.60	0	-0.60	-1.20	-1.88	-2.64	-3.54	-4.72	-6.54

Iteration	Γ_e
	γ_1
1	0
2-6	1
7	0
8	1
9	1
10	0
11	1
12	1
13	1
14	0
15	0