

2D LDPC Codes and Joint Detection and Decoding for Two-Dimensional Magnetic Recording

Chaitanya Kumar Matcha¹, Shounak Roy¹, Mohsen Bahrami², Shayan Garani Srinivasa¹ and Bane Vasic².

¹Department of Electronics Systems Engineering, Indian Institute of Science, Bengaluru, 560012, India.

²Department of Electrical and Computer Engineering, University of Arizona, Tucson, 85721, USA.

Correspondence Email: shayan.gs@dese.iisc.ernet.in

Abstract—Two-dimensional magnetic recording (TDMR) is a promising technology for boosting areal densities using sophisticated signal processing algorithms within a systems framework. The read/write channel architectures have to effectively tackle 2D inter-symbol interference (ISI), 2D synchronization errors, media and electronic noise sources as well as thermal asperities resulting in burst erasures. 1D low-density parity check (LDPC) codes are well studied to correct large 1D burst errors/erasures. However, such 1D LDPC codes are not suitable for correcting 2D burst errors/erasures due to the 2D span of errors. In this paper, we propose construction of a native 2D LDPC code to effectively correct 2D burst erasures. We also propose a joint detection-decoding engine based on the generalized belief propagation (GBP) algorithm to simultaneously handle 2D ISI, as well as correct bit/burst errors for TDMR channels. Our work is novel in two aspects: (a) We propose the construction of native 2D LDPC codes to correct large 2D burst erasures, (b) We develop a 2D joint signal detection-decoder engine that incorporates 2D ISI constraints, modulation code constrains along with LDPC decoding. The native 2D LDPC code can correct > 20% more burst erasures compared to the 1D LDPC code over a 2D page of detected bits. Also, the proposed algorithm is observed to achieve a signal-to-noise ratio (SNR) gain of > 0.5 dB in bit error rate (BER) performance (translating to 10% increase in areal densities around the 1.8 Tb/in² regime with grain sizes of 9 nm) as compared to a decoupled detector-decoder system configuration. The efficacy of our proposed algorithm and system architecture is evaluated by assessing areal density (AD) gains via simulations for a TDMR configuration comprising of a 2D generalized partial response (GPR) over the Voronoi media model assuming perfect 2D synchronization.

I. INTRODUCTION

Two dimensional magnetic recording (TDMR) aims at improving areal densities beyond 1 Tb/in² by packing tracks closely, allowing for 2D inter-symbol interference (ISI) [1]. The effects of bit-size reduction is handled using sophisticated signal processing algorithms. Along with the random errors due to channel impairments, we also observe burst erasures in magnetic recording channels due to thermal asperities, inherent defects on the recording medium, scratches on the recording surface or due to external mechanical shocks/vibrations [2]. This requires sophisticated signal detection and strong error correcting codes (ECC), to meet stringent (sector failure rate (SFR) ~ 1e-15) reliability requirements.

A. Burst Erasure Correction for TDMR

Low density parity check (LDPC) codes have been widely studied and are successfully deployed in hard disk drives due

to excellent error correction abilities and amenable to Silicon implementation. Systematically constructed quasi-cyclic (QC) LDPC codes [3] using permutation matrices help in efficient implementation of the LDPC decoder architectures using variants of the belief propagation (BP) algorithm. QC LDPC codes are also observed to give good 1D burst erasure correction capability [2]. The LDPC decoders are aided by defect detectors in the read channel for estimating the locations of erasures which are flag to the LDPC decoder for burst erasure correction [4], [5].

Matcha and Srinivasa [6] have proposed a 2D defect detector using 1D QC LDPC codes with various interleaving schemes for 2D burst erasure correction in TDMR. The 1D LDPC code is populated into a 2D page in the raster scan order. Study of ECCs for 2D burst error/erasure correction has been restricted to algebraic codes that can correct a predefined pattern of errors with guaranteed error correction ability [7], [8]. These 2D codes are not suitable for correction of large burst errors/erasures.

1D burst erasure correction capability of LDPC codes can be studied using zero-spans in the parity check matrix. Fossorier [9] has proved that most LDPC codes can achieve burst erasure correction capability achieving the Roger bound i.e., a (n, k) code correcting up to $(n - k - 1)$ length of burst erasure. In this paper, we extend the idea of 1D QC LDPC code construction and propose a native 2D LDPC code with good 2D burst erasure correction capability. We also study the burst erasure correction capability of the proposed code using the zero-spans in the parity check matrix.

B. Efficient Detection and Decoding Architectures

Two-dimensional signal detection algorithms can be broadly classified into trellis based approaches and those derived using the generalized belief propagation (GBP) algorithm. Examples to the trellis class of algorithms include [10]–[12] etc. Trellis based approaches achieve maximum likelihood performance locally (i.e., over a multi-rows and columns). Near optimal 2D MAP performance can be achieved by coupling 2D soft equalizers with iterative multi-row/column detectors within an iterative set up using 2D generalized partial response (GPR) targets. On the other hand, the GBP algorithm operates over an entire 2D page using a region based hierarchy. This algorithm is empirically observed to provide marginal estimates, close to true marginals at the expense of huge computational

complexity. The state-of-the-art work on coding and detection for TDMR channels includes coupling soft detectors with 1D iterative error correcting codes such as LDPC codes to realize significant SNR gains [6]. It must be noted that the signal detectors and ECC decoders are optimized separately using different criterion even though they are coupled through a turbo loop.

Intuitively, improved performance can be accomplished by fusing a signal detector with an error correction decoder using message passing equations over the joint detection-decoder engine. The message passing equations are derived from the same cost function, unlike earlier approaches. Kurkoski, Siegel and Wolf [13] considered the idea of fusing the partial response channel with parity check constraints to form joint factor graphs and obtained message passing decoders that showed better performance than individually optimized detectors and decoders over the perpendicular magnetic recording channel. In this work, we extend the notion of joint 2D detection and decoding towards TDMR channels. It must be noted that there are several significant differences between the 1D joint detector-decoder engine [13] and the 2D case. First, we introduce a novel 2D LDPC code with parity check equations represented by a 3D tensor using a composition of tiles of permutations matrices in 2D. This code is inherently 2D and resilient to 2D burst erasures and random errors. Second, we introduce the parity check constraints directly into the formulation of the Gibbs free energy. Using a constrained optimization framework, we derive the message passing equations using the same cost function from first principles by tracking the ancestors of all bits within a region using ancestors corresponding to the ISI and the parity check constraints. We study the efficacy of our new algorithm and joint detector-decoder architecture over individual detector and decoder engines coupled via a turbo loop. This framework can be extended for fusing a broad class of signal detectors and code constraints towards a hybrid architecture.

C. Paper Organization

The paper is organized as follows: In Section II, we propose a novel 2D LDPC code construction for correcting burst erasures in TDMR. In Section III, we formulate GBP for joint detection and decoding by considering 2D ISI and LDPC code constraints. In Section IV, we present and discuss our simulation results using a Voronoi based TDMR channel model. Section V concludes the paper.

II. LDPC CODE FOR 2D BURST ERASURE CORRECTION

LDPC codes are observed to give near Shannon-capacity performance. These codes are successfully deployed in magnetic storage as well as in other storage technologies such as flash memories and optical storage devices due to amenable hardware architectures for realizing the algorithms in practice.

For magnetic recording channels, the error correcting code should be able to correct random errors as well as burst erasures that are often seen due to thermal asperities [14]. This requires us to construct LDPC codes that can correct large burst erasures along with a mix of random errors. Fossorier

[9] proved that LDPC parity check matrices can be written in burst correction form (BCF) that enables us to correct 1D burst erasures achieving the Roger bound i.e., a (n, k) -code that can correct erasures of length up to $n - k$. However, the BCF of the parity check matrix has a large number of parity check equations that makes it unsuitable for practical decoding using the BP algorithm. QC LDPC codes provide a regular construction of the LDPC parity check matrix, and are observed to give a good 1D burst erasure correction performance. Other techniques to improve burst erasure correction capability of LDPC codes include permuting the codeword using simulated annealing [15], or by studying trapping sets of the code [16], [17].

For TDMR, we model the burst erasures as a random 2D connected shape. Algebraic code construction for 2D burst error correction are studied by Yoon and Moon in [8] and by Roy and Srinivasa in [7]. However, these codes are designed for very small burst errors of predefined shapes. Cassuto and Shokrollahi [18] have proved existential results of 2D LDPC codes based on the ideas of array codes. However, LDPC code construction to correct 2D burst erasures is not explicitly provided. Matcha and Srinivasa [6] have used 1D LDPC codes for 2D burst erasure correction by rasterizing 1D codewords into a 2D page along with interleavers. However, these codes are not constructed and optimized to correct large 2D burst erasures. In this section, we extend the ideas of 1D QC-LDPC code construction towards a native 2D LDPC code that can effectively correct 2D burst erasures than without it.

A. 2D LDPC Code Construction

1D QC LDPC codes [3] are constructed using parity check matrices realized as a tiling of permutation matrices. If \mathbf{I} is an identity matrix of size $p \times p$ and \mathbf{P} is a permutation matrix of the same size, an example parity check matrix with row-weight r and column weight c is given by

$$\mathbf{H}_{1D} = \begin{bmatrix} \mathbf{I} & \mathbf{I} & \mathbf{I} & \dots & \mathbf{I} \\ \mathbf{P} & \mathbf{P}^2 & \mathbf{P} & \dots & \mathbf{P}^r \\ \mathbf{P}^2 & \mathbf{P}^4 & \mathbf{P}^6 & \dots & \mathbf{P}^{2r} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}^{c-1} & \mathbf{P}^{2c-2} & \mathbf{P}^{3c-3} & \dots & \mathbf{P}^{r(c-1)} \end{bmatrix}. \quad (1)$$

A typical choice of the permutation matrix is a unit-circular shift matrix given by

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ 1 & 0 & 0 & \dots & 0 \end{bmatrix}. \quad (2)$$

The powers of \mathbf{P} in the construction of \mathbf{H}_{1D} give tiles with different shifts. The choice of the permutation matrix and the shifts provide us with a variety of QC-LDPC codes.

The burst erasure correction capability of 1D QC-LDPC can be explained using the zero-span of the code: a code will be able to correct burst erasure of length l if for every position j in the codeword, there exists an i^{th} row in \mathbf{H}_{1D} such that

$h_{ij} = 1$ and $h_{i(j+1)} = h_{i(j+2)} = \dots = h_{i(j+l-1)} = 0$. The condition essentially ensures that the j^{th} bit erasure can be corrected using i^{th} parity check equation.

We extend these ideas to construct 2D LDPC codes. The codeword of a 2D LDPC code is a 2D page of bits. Therefore, the words orthogonal to the 2D LDPC codes are also 2D page of bits. The parity check equations will be represented by a 3D parity check tensor obtained by stacking these orthogonal words one below the other. Each layer of the parity-check tensor represents a single parity check equation.

We propose the construction of the parity check tensor by stacking 3D permutation tensors in a 3D fashion. We use the indexing notation such that the position (i, j, k) represents the (j, k) bit in the i^{th} layer of the parity check tensor. We consider an identity tensor \mathbf{I} of size $p \times p \times p$ given by

$$\mathbf{I} = [I_{i,j,k}]_{i,j,k=1}^p, \quad I_{i,j,k} = \begin{cases} 1, & i = j = k, \\ 0, & \text{otherwise.} \end{cases}$$

The tensor \mathbf{I} can be permuted in two directions. Let $P : \{1, 2, \dots, p\} \rightarrow \{1, 2, \dots, p\}$ and $Q : \{1, 2, \dots, p\} \rightarrow \{1, 2, \dots, p\}$ be the permutation operations defined on any tensor $\mathbf{T} = [T_{i,j,k}]_{i,j,k=1}^p$ as

$$P(\mathbf{T}) = [T_{i,P(j),k}]_{i,j,k=1}^p, \quad \text{and} \quad Q(\mathbf{T}) = [T_{i,j,Q(k)}]_{i,j,k=1}^p.$$

The following proposition proves the properties of row and column permutations described in this section.

Proposition 1. *If $P(\mathbf{T}) = [T_{i,P(j),k}]_{i,j,k=1}^p$ and $Q(\mathbf{T}) = [T_{i,j,Q(k)}]_{i,j,k=1}^p$ are row and column permutations operating on a tensor; the following properties hold true:*

- 1) $P \circ Q(\mathbf{T}) = Q \circ P(\mathbf{T})$.
- 2) *In general, any order of m permutations of P and n permutations of Q on \mathbf{T} can be written as $P^m \circ Q^n(\mathbf{T})$.*

Proof: From the definitions of P and Q , we have

$$P \circ Q(\mathbf{T}) = P \left([T_{i,j,Q(k)}]_{i,j,k=1}^p \right) = [T_{i,P(j),Q(k)}]_{i,j,k=1}^p.$$

Similarly,

$$Q \circ P(\mathbf{T}) = Q \left([T_{i,P(j),k}]_{i,j,k=1}^p \right) = [T_{i,P(j),Q(k)}]_{i,j,k=1}^p.$$

This proves the first property that the order of row and column permutations can be interchanged. The second property follows from the first by exchanging adjacent P and Q in such a way that all P s appear on the left and all Q s appear on the right. ■

Similar to the 1D QC-LDPC code construction, we can construct the 2D LDPC tensor \mathbf{H}_{2D} using various permutations of the identity tensor \mathbf{I} using the permutations P and Q . Each permutation of \mathbf{I} gives us a cube. A \mathbf{H}_{2D} can be obtained by stacking $c \times h \times w$ cubes along (i, j, k) directions respectively.

Similar to the permutation in (2), we choose the permutations P and Q to be circular shifts given by

$$P(i) = Q(i) = \begin{cases} 1, & i = p \\ i + 1, & \text{otherwise.} \end{cases}$$

The identity tensor along with its row column permutations are shown in Figure 1.

We can also construct \mathbf{H}_{2D} similar to \mathbf{H}_{1D} in (1), where the $(x, y, z)^{\text{th}}$ cube in the stack is given by $P^{x(z-1)} \circ Q^{x(y-1)}(\mathbf{I})$.

Figure 2 shows the pictorial representation of an example \mathbf{H}_{2D} constructed using the above idea.

B. Design of 2D LDPC Code for TDMR

The shifts/powers of the permutations used in the construction of \mathbf{H}_{2D} for our purpose are chosen as follows. The \mathbf{H}_{2D} is constructed by stacking $c \times h \times w$ cubes, each of size $p \times p \times p$ such that c is a multiple of p .

In our code construction, the $(i, j, k)^{\text{th}}$ cube is given by $P^{a(i,j,k)} \circ Q^{b(i,j,k)}(\mathbf{I})$, where the powers $a(i, j, k)$ and $b(i, j, k)$ are chosen as

$$a(i, j, k) = i \pmod{p}, \quad (3)$$

$$b(i, j, k) = \left\lfloor \frac{i-1}{p} \right\rfloor jk. \quad (4)$$

Notice that each cube in \mathbf{H}_{2D} contributes only to p bits for the column weight. Therefore, it is important to choose the shifts such that every bit position in the codeword has sufficient column weight for good error correction. The following lemma shows that the above choice of shifts give a uniform column weight of $\frac{c}{p}$.

Lemma 2. *Stacking $c \times h \times w$ cubes with the shifts given in (3) and (4) gives a parity check tensor with uniform column weight of $\frac{c}{p}$.*

Proof: We prove this by showing that the column weight contributed by the stack of cubes $\{(i, j, k) \mid (p-1)n < i \leq np\}$ is uniformly 1 at all codeword positions within the stack for all $(j, k) \in \{1, 2, \dots, p\}^2$ and $n = 1, 2, \dots, \frac{c}{p}$.

The set of positions contributing to column weight by identity cube is given by

$$P^{(i,j,k)} = \{(x, x) \mid 1 \leq x \leq p\}$$

The set of positions contributing to column weight by $(i, j, k)^{\text{th}}$ cube is given by

$$P^{(i,j,k)} = \left\{ \left(P^{i \pmod{p}}(x), Q^{\lfloor \frac{i-1}{p} \rfloor jk}(x) \right) \mid 1 \leq x \leq p \right\}.$$

The set of positions contributing to column weight by the

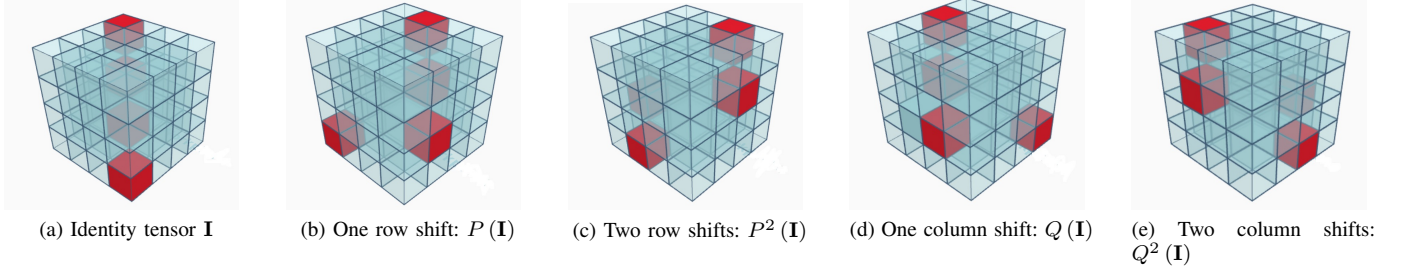
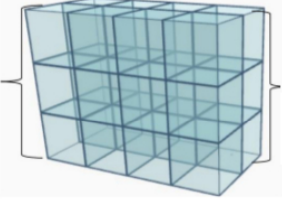


Figure 1: Identity tensor and the row and column permutations of the tensor.

$$\begin{bmatrix} \mathbf{I} & \mathbf{I} & \cdots & \mathbf{I} \\ P \circ Q(\mathbf{I}) & P^2 \circ Q(\mathbf{I}) & \cdots & P^w \circ Q(\mathbf{I}) \\ P^2 \circ Q^2(\mathbf{I}) & P^4 \circ Q^2(\mathbf{I}) & \cdots & P^{2w} \circ Q^2(\mathbf{I}) \\ \vdots & \vdots & \ddots & \vdots \\ P^{c-1} \circ Q^{c-1}(\mathbf{I}) & P^{2c-2} \circ Q^{c-1}(\mathbf{I}) & \cdots & P^{(c-1)} \circ Q^{c-1}(\mathbf{I}) \end{bmatrix}$$


$$\begin{bmatrix} \mathbf{I} & \mathbf{I} & \cdots & \mathbf{I} \\ P(\mathbf{I}) & P^2(\mathbf{I}) & \cdots & P^w(\mathbf{I}) \\ P^2(\mathbf{I}) & P^4(\mathbf{I}) & \cdots & P^{2w}(\mathbf{I}) \\ \vdots & \vdots & \ddots & \vdots \\ P^{c-1}(\mathbf{I}) & P^{2c-2}(\mathbf{I}) & \cdots & P^{w(c-1)}(\mathbf{I}) \end{bmatrix}$$

Figure 2: An example of a parity check tensor for a 2D LDPC code. The building blocks are the identity tensor and its row and column permutations.

stack $\{(i, j, k) \mid (p-1)n < i \leq np\}$ cube is given by

$$P_n^{(j,k)} = \bigcup_{i=(p-1)n+1}^{np} \left\{ \left(P^{i \bmod p}(x), P^{\lfloor \frac{i-1}{p} \rfloor jk}(x) \right) \mid 1 \leq x \leq p \right\} \quad (6)$$

$$= \bigcup_{i'=1}^p \left\{ \left(P^{i'}(x), Q^{njk}(x) \right) \mid 1 \leq x \leq p \right\} \quad (7)$$

$$= \left\{ \left(P^{i'}(x), Q^{njk}(x) \right) \mid 1 \leq x \leq p, 1 \leq i' \leq p \right\} \quad (8)$$

$$= \bigcup_{x=1}^p \left\{ \left(P^{i'}(x), Q^{njk}(x) \right) \mid 1 \leq i' \leq p \right\} \quad (10)$$

$$= \bigcup_{x=1}^p \left\{ (i', Q^{njk}(x)) \mid 1 \leq i' \leq p \right\} \quad (11)$$

$$= \bigcup_{x=1}^p \left\{ (i', x) \mid 1 \leq i' \leq p \right\} \quad (12)$$

$$= \left\{ (i', x) \mid 1 \leq i' \leq p, 1 \leq x \leq p \right\}. \quad (13)$$

Step (11) uses the fact that the row permutations are complete i.e.,

$$\{P^i(x) \mid 1 \leq i \leq p\} = \{1, 2, \dots, p\} \forall 1 \leq x \leq p.$$

Step (12) uses the fact that the column permutation is a one-to-one mapping i.e.,

$$\{Q^i(x) \mid 1 \leq x \leq p\} = \{1, 2, \dots, p\} \forall 1 \leq i \leq p.$$

Therefore, we have proven that each p -stack at position (j, k) given by $\{(i, j, k) \mid (p-1)n < i \leq np\}$ contributes to a column weight of 1 at every corresponding codeword position. Since there are $\frac{c}{p}$ such p -stacks, the constructed code has a uniform column weight of $\frac{c}{p}$.

Remarks:

- 1) Each cube contributes to a row weight of exactly 1. Therefore, the row weight of the code constructed by stacking $c \times h \times w$ cubes is hw .
- 2) If c is not a multiple of p , the column weights can be bounded as $\lfloor \frac{c}{p} \rfloor \leq r \leq \lceil \frac{c}{p} \rceil$.

C. Burst Erasure Correction Capability of the 2D LDPC Code

Similar to the 1D QC-LDPC code, we analyze the burst error correction capability of the code using the zero-spans on the parity check tensor \mathbf{H}_{2D} using the following lemma.

Lemma 3. A 2D error correcting code with parity check tensor $\mathbf{H}_{2D} = [h_{i,j,k}]$ can correct up to a 2D burst of size $B_h \times B_w$ if for every position (j, k) , there exists i^{th} layer in \mathbf{H}_{2D} such that

$$h_{i,a,b} := \begin{cases} 1, & (a, b) = (j, k), \\ 0, & (a, b) \in \{(j+x, k+y) \mid -B_h < x < B_h, \\ & \mid -B_w < y < B_w\} \setminus \{(j, k)\}. \end{cases} \quad (14)$$

Proof: The condition in (14) means that the i^{th} parity check equation uses only the $(j, k)^{\text{th}}$ bit within the erasure region.

Let the $B_h \times B_w$ burst erasure occur with the starting position of (s_h, s_w) i.e., the erasure locations are given by

$$R = \{(s_h + x, s_w + y) \mid 0 \leq x < B_h, 0 \leq y < B_w\}.$$

We prove the result by showing that each bit in the burst erasure can be corrected independently using the condition given in (14). For the position (s_h, s_w) , the condition in (14) ensures that there is a parity check equation such that the bits

in

$$R_{(s_h, s_w)} = \{(s_h + x, s_w + y) \mid -B_h \leq x < B_h, \\ -B_w \leq y < B_w\} \setminus \{(s_h, s_w)\}.$$

are not involved. Since $R \setminus \{(s_h, s_w)\} \subset R_{(s_h, s_w)}$, none of the bits in the erasure location except for $(s_h, s_w)^{\text{th}}$ bit is used in this parity check equation. This allows us to correct the $(s_h, s_w)^{\text{th}}$ bit using this parity check equation.

Similarly, for any location $(s_h + j, s_w + k)$, $0 \leq j < B_h$, $0 \leq k < B_w$, the condition in (14) ensures that there is a parity check equation such that the bits in

$$R_{(s_h+j, s_w+k)} \\ = \{(s_h + j + x, s_w + k + y) \mid -B_h \leq x < B_h, \\ -B_w \leq y < B_w\} \setminus \{(s_h + j, s_w + k)\} \\ = \{(s_h + x, s_w + y) \mid -B_h + j \leq x < B_h + j, \\ -B_w + k \leq y < B_w + k\} \setminus \{(s_h + j, s_w + k)\}$$

are not involved. For $0 \leq j < B_h$, $0 \leq k < B_w$, it is easy to verify that $R \subset R_{(s_h+j, s_w+k)} \setminus \{(s_h + j, s_w + k)\}$. Therefore, the parity check equation can be used to correct the bit at location $(s_h + j, s_w + k)$. This is illustrated in Figure 3c.

This proves that every bit in the erasure location can be independently corrected. ■

Theorem 4. *The 2D LDPC code constructed using the shifts in (3) and (4) can correct a 2D burst erasure of size at least $p \times p$.*

Proof: The first layer of cubes in the constructed \mathbf{H}_{2D} is a stack of identity tensors of size $p \times p \times p$. Therefore, the position of ones in the first parity check layer is given by

$$\{(j-1)p+1, (k-1)p+1 \mid 1 \leq j \leq h, 1 \leq k \leq w\}.$$

Similarly, the position of ones in i^{th} parity check layer is given by

$$\{(j-1)p+i, (k-1)p+i \mid 1 \leq j \leq h, 1 \leq k \leq w\}.$$

Therefore, the relative positions of ones in the first p layers of \mathbf{H}_{2D} is same as the first layer.

From (3) and (4), we have the next p layers ($p+1 \leq i \leq 2p$) obtained by a single row-shift of the top p layers. Similarly, layers from $p+1 \leq i \leq p^2$ are all obtained by row shifts of the first p layers. Therefore, the relative positions of ones in each of these layers $1 \leq i \leq p^2$ remain the same.

We can easily see that if a one is located at (j, k) in the first layer, then the following locations contain all zeros:

$$\{(j+a, k+b) \mid -p \leq j \leq p, -p \leq k \leq p\} \setminus \{(j, k)\}.$$

This is same as the condition (14) in Lemma 3 with $B_h = p$ and $B_w = p$.

Since the relative positions of ones remain the same for the first p^2 layers, the condition (14) is satisfied for every position where a one is present in the first p^2 layers. From Lemma 2, we have that the column weight contributed by the first p^2 layers is uniformly 1 at all positions. Therefore, the condition (14) is satisfied by every position (j, k) .

Therefore, using Lemma 3, the code constructed in Section II-B can correct burst erasure of at least $p \times p$ in size. ■

III. JOINT DETECTION AND DECODING USING GBP

Generalized belief propagation (GBP) algorithm [19] is a graph based decoding/detection algorithm that passes messages between regions instead of messages between nodes as in the traditional BP algorithm. GBP can be formulated as a convex optimization problem that minimizes the Gibbs free energy, and provides a method to approximate marginal distributions which makes it suitable for MAP detection with soft outputs.

The GBP algorithm is known to give exact marginals if and only if the region based graph has no loops [19]. When used for 2D ISI signal detection, the region based graphs always contain loops. However, the GBP algorithm provides a method to approximate the marginals that are empirically observed to be close to the actual marginals [20].

In order to achieve high fidelity rates (SFR $\sim 1e-15$) in magnetic recording, the detection engine is followed by an ECC decoder for error correction. The detector and the decoder often operate in a turbo loop to achieve significant gains in the performance. The BP algorithm that is used for LDPC decoding is shown to be a special case of the GBP algorithm [19]. Since the GBP based 2D ISI detector and the BP decoder are separately optimized, the overall performance is not guaranteed to be the best and depends on the number of turbo iterations.

In this section, we reformulate the GBP algorithm by incorporating the ECC parity-check constraints in addition to the ISI constraints towards joint detection and decoding of the readback samples. We achieve an improvement in the overall performance by incorporating ISI and LDPC constraints into a single instance of the algorithm instead of handling the constraints in two separate instances of the same GBP algorithm.

Let $x_{i,j} \in \{0, 1\}$, $(i, j) \in \mathbb{Z}^2$ represent bits written onto the medium post bipolar mapping and $\mathbf{x} = [x_{i,j}] \in \{0, 1\}^{n \times m}$ represent a $n \times m$ page of bits. Let $y_{i,j} \in \mathbb{R}$, $(i, j) \in \mathbb{Z}^2$ represent the read-back samples obtained from the channel model and $\mathbf{y} = [y_{i,j}] \in \mathbb{R}^{n \times m}$ represent the corresponding page of read-back samples. Let $\mathbf{x}_{i,j}$ represent the set of bits that contribute to the read-back sample $y_{i,j}$. For a channel with 3×3 ISI span, $\mathbf{x}_{i,j}$ contains 9 bits $\{x_{i+k, j+l} \mid k, l \in \{-1, 0, 1\}\}$. Let \mathbf{x}_k represent the set of bits that are involved in the k^{th} parity check constraint of the LDPC code for $k = 1, 2, \dots, n_p$.

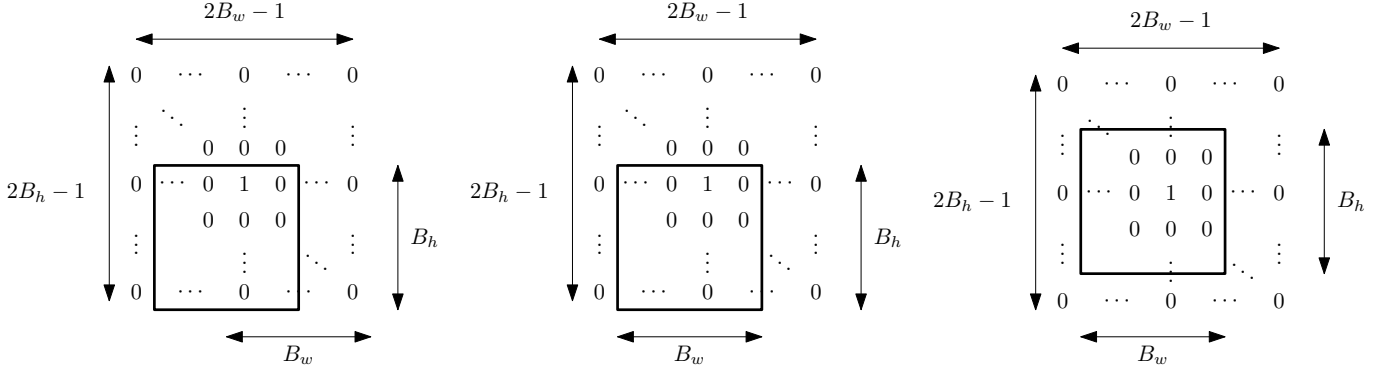
A. Gibbs Free Energy

Assuming uniform distribution of the input bits and additive white Gaussian noise (AWGN) in the channel model, the *a-posteriori* probability of \mathbf{x} given read-back samples \mathbf{y} is given by

$$p(\mathbf{x} \mid \mathbf{y}) = p(\mathbf{y} \mid \mathbf{x}) p(\mathbf{x}) p(\mathbf{y})^{-1} \propto p(\mathbf{y} \mid \mathbf{x}) \\ p(\mathbf{y} \mid \mathbf{x}) = \prod_{i,j} f_{i,j}(\mathbf{x}_{i,j}) \quad (15)$$

where $f_{i,j}(\mathbf{x}_{i,j}) = p(y_{i,j} \mid \mathbf{x}_{i,j})$ is a Gaussian function representing the distribution of noise sample at location (i, j) . Therefore, we have

$$p(\mathbf{x} \mid \mathbf{y}) = \frac{1}{Z} \prod_{i,j} f_{i,j}(\mathbf{x}_{i,j}), \quad (16)$$



(a) Correcting the starting position of the burst erasure. (b) Correcting a bit in the first row of the burst erasure. (c) Correcting an arbitrary bit within the burst erasure.

Figure 3: A span of $(2B_h - 1) \times (2B_w - 1)$ zeros around a 1 in each parity check layer can help in correcting a burst of size at least $B_h \times B_w$. The burst erasure region is indicated using a rectangle relative to various locations within erasure region. This condition ensures that every location within the burst erasure can be corrected using the parity check equation since the remaining bits are not involved in the corresponding parity check equation.

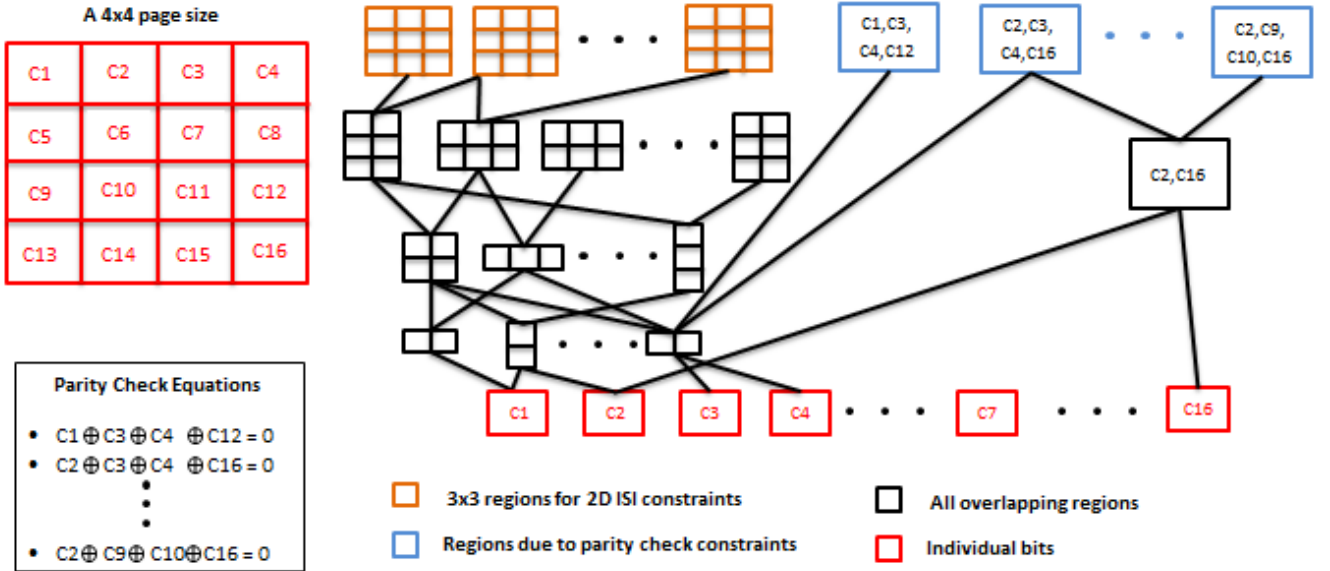


Figure 4: An example of a region graph used by the GBP algorithm for joint-detection decoding is shown. The regions are chosen based on 2D ISI as well as LDPC code parity check constraints.

for some $Z(\mathbf{y})$. Let $b(\mathbf{x})$ represent the belief of the *a-posteriori* probability of \mathbf{x} . Using the properties of KL-divergence, the belief $b(\mathbf{x}) = p(\mathbf{x} | \mathbf{y})$ can be achieved by minimizing the free energy given by

$$F = E - H = \mathcal{D}(b(\mathbf{x}) \| p(\mathbf{x} | \mathbf{y})) - \ln Z(\mathbf{y}), \quad (17)$$

$$\text{average energy } E = - \sum_{i,j} \sum_{\mathbf{x}_{i,j}} b(\mathbf{x}_{i,j}) \ln f_{i,j}(\mathbf{x}_{i,j}), \quad (18)$$

$$\text{entropy } H = \sum_{\mathbf{x}} b(\mathbf{x}) \ln b(\mathbf{x}). \quad (19)$$

Let a region $r \subset \mathbb{R}^2$ be defined as a set of positions within a page and \mathcal{R} represent a collection of such regions. We choose \mathcal{R} such that

1) each of $\mathbf{x}_{i,j}$ is included in at least one region,

2) each of \mathbf{x}_k is included in at least one region.

For each $r \in \mathcal{R}$, let \mathbf{x}_r be the vector of bits in the region r . Let $b(\mathbf{x}_r)$ and $p(\mathbf{x}_r)$ be the corresponding marginal beliefs and probabilities within the region r . The regions graph can be formed by partial ordering of regions based on the containment of one region inside another [19]. Figure 4 shows an example region graph containing regions corresponding to 2D ISI constraints as well as the LDPC constraints.

The free energy is approximated using the entropy of individual regions as

$$\hat{F} = - \sum_{i,j} \sum_{\mathbf{x}_{i,j}} b(\mathbf{x}_{i,j}) \ln f_{i,j}(\mathbf{x}_{i,j}) + \sum_{r \in \mathcal{R}} c_r \sum_{\mathbf{x}_r} b(\mathbf{x}_r) \ln b(\mathbf{x}_r), \quad (20)$$

where c_R are overcounting numbers given by $c_r = \sum_{p \in \mathcal{P}_r} 1 - c_p$, and \mathcal{P}_r are parents of region r in the region graph. This is called Kikuchi approximation, or region based approximation (RBA).

B. Joint Detection and Decoding

When GBP is used for 2D ISI detection as in [21], the Gibb's free energy in (20) is minimized under the edge-constraints given by

$$\sum_{u \in \mathbf{x}_p \setminus r} b(\mathbf{x}_p) = b(\mathbf{x}_r) \quad \forall p \in \mathcal{P}_r, \forall r \in \mathcal{R}, \quad (21)$$

and the normalization constraints given by

$$\sum_{\mathbf{x}_r} b(\mathbf{x}_r) = 1, \quad \forall r \in \mathcal{R}. \quad (22)$$

These constraints ensure that the belief of a sub-region can be obtained by marginalizing the beliefs of their parents [22]. The GBP algorithm is obtained from the constrained optimization of \hat{F} using Lagrange multipliers.

The regions and c_r are chosen [22] to 1) strict convexity of the free energy, 2) closely approximate the marginals 3) reduce computational complexity.

For the joint detection and decoding using GBP, we ensure that the parity checks are satisfied using the LDPC constraints given by

$$b(\mathbf{x}_k) = 0, \quad \text{if parity}(\mathbf{x}_k) = 1, \quad k = 1, 2, \dots, n_p. \quad (23)$$

C. Derivation of the Joint Detection Decoding Algorithm

In order to minimize the free energy in (20) under the constraints in (21), (22) and (23), we define the hierarchy of regions as follows:

- 1) \mathcal{A}_r : Ancestors of the region r

$$\mathcal{A}_R = \{r' \in \mathcal{R} \mid r \subset r'\}.$$

- 2) \mathcal{D}_r : Descendants of the region r

$$\mathcal{D}_R = \{r' \in \mathcal{R} \mid r \supset r'\}.$$

- 3) \mathcal{P}_r : Parents of the region r

$$\mathcal{P}_r = \{r' \in \mathcal{R} \mid r \subset r', \nexists r'' \in \mathcal{R}, \quad r \subset r'' \subset r'\}.$$

- 4) \mathcal{C}_r : Children of the region r

$$\mathcal{C}_r = \{r' \in \mathcal{R} \mid r \supset r', \nexists r'' \in \mathcal{R}, \quad r \supset r'' \supset r'\}.$$

- 5) $\mathcal{A}^{(\text{ISI})}$: All ancestors of \mathcal{R} arising out of ISI constraints.

$$\mathcal{A}^{(\text{ISI})} = \bigcup_{(i,j)} \{\mathbf{x}_{i,j}\}.$$

- 6) $\mathcal{A}^{(\text{LDPC})}$: All ancestors of \mathcal{R} arising out of LDPC constraints.

$$\mathcal{A}^{(\text{LDPC})} = \bigcup_p \{\mathbf{x}_p\}.$$

- 7) \mathcal{A} : All ancestors of \mathcal{R}

$$\mathcal{A} = \mathcal{A}^{(\text{ISI})} \cup \mathcal{A}^{(\text{LDPC})} = \{r \in \mathcal{R} \mid \nexists r' \in \mathcal{R}, \quad r' \supset r\}.$$

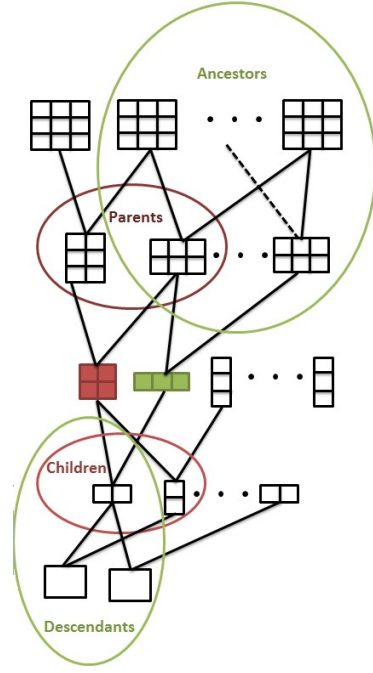


Figure 5: An example of a region graph is shown with the relations between different regions in terms of their hierarchy. Ancestors of a region r are the set of regions that contain r . Descendants of a region r are the set of regions that are contained in r . Children of a region r are its immediate descendants. Parents of a region r are its immediate ancestors. Ancestors and descendants of the region marked in green are shown. Parents and children for the region marked in red are also shown.

Figure 5 shows an example region graph illustrating the relations between different regions in terms of their hierarchy.

We assume that none of the regions defined by LDPC constraints and ISI constraints are inclusive of each other. This implies that the bits involved in a 2D ISI constraint are not all included in the same parity check equation. Similarly, all bits involved in a parity check equation are fully contained within the 2D ISI span. LDPC code constructions usually satisfy this assumption and is useful for a good burst erasure correction.

Enforcing the parity constraints, we rewrite the free energy in (20) using the above definitions as

$$\begin{aligned} \hat{F} = & - \sum_{r \in \mathcal{A}^{(\text{ISI})}} c_r \sum_{\mathbf{x}_r} b(\mathbf{x}_r) \ln f_r(\mathbf{x}_r) \\ & + \sum_{r \in \mathcal{R} \setminus \mathcal{A}^{(\text{LDPC})}} c_r \sum_{\mathbf{x}_r} b(\mathbf{x}_r) \ln b(\mathbf{x}_r) \\ & + \sum_{r \in \mathcal{A}^{(\text{LDPC})}} c_r \sum_{\mathbf{x}_r: \text{parity}(\mathbf{x}_r)=0} b(\mathbf{x}_r) \ln b(\mathbf{x}_r). \end{aligned} \quad (24)$$

The edge constraints between a region r and its parent p in (21) can be equivalently written [19] as

$$c_r b_r(\mathbf{x}_r) + \sum_{a \in \mathcal{A}_r \setminus (\{r\} \cup \mathcal{A}_p)} c_a \sum_{\mathbf{x}_a \setminus r} b_a(\mathbf{x}_a) = 0. \quad (25)$$

We solve the constrained minimization of free energy [19] using Lagrange multipliers $\lambda_{pr}(\mathbf{x}_r)$, and γ_r by defining the cost function as

$$\begin{aligned}
C = & - \sum_{r \in \mathcal{A}^{(ISI)}} c_r \sum_{\mathbf{x}_r} b(\mathbf{x}_r) \ln f_r(\mathbf{x}_r) \\
& + \sum_{r \in \mathcal{R} \setminus \mathcal{A}^{(LDPC)}} c_r \sum_{\mathbf{x}_r} b(\mathbf{x}_r) \ln b(\mathbf{x}_r) \\
& + \sum_{r \in \mathcal{A}^{(LDPC)}} c_r \sum_{\mathbf{x}_r: \text{parity}(\mathbf{x}_r)=0} b(\mathbf{x}_r) \ln b(\mathbf{x}_r) \\
& - \sum_{r \in \mathcal{R}} \sum_{p \in \mathcal{P}_r} \sum_{\mathbf{x}_r} \lambda_{pr}(\mathbf{x}_r) (c_r b_r(\mathbf{x}_r) \\
& \quad + \sum_{a \in \mathcal{A}_r \setminus (\{p\} \cup \mathcal{A}_p)} c_a \sum_{\mathbf{x}_a \setminus r} b_a(\mathbf{x}_a)) \\
& - \sum_{r \in \mathcal{R} \setminus \mathcal{A}^{(LDPC)}} c_r \gamma_r \left(\sum_{\mathbf{x}_r} b(\mathbf{x}_r) - 1 \right) \\
& - \sum_{r \in \mathcal{A}^{(LDPC)}} c_r \gamma_r \left(\sum_{\mathbf{x}_r: \text{parity}(\mathbf{x}_r)=0} b(\mathbf{x}_r) - 1 \right) \quad (26)
\end{aligned}$$

We have incorporated the parity check constraints in (23) by restricting the summation of \mathbf{x}_r to the words satisfying $\text{parity}(\mathbf{x}_r) = 1, \forall \mathbf{x}_r \in \mathcal{A}^{(LDPC)}$.

For $r \in \mathcal{R} \setminus \mathcal{A}^{(LDPC)}$, we get

$$\begin{aligned}
\frac{\partial C}{\partial b(\mathbf{x}_r)} = & -c_r \sum_{a \in \mathcal{A}_r} \ln f_a(\mathbf{x}_a) + c_r \ln b(\mathbf{x}_r) \\
& - c_r \sum_{p \in \mathcal{P}_r} \lambda_{pr}(\mathbf{x}_r) - c_r \sum_{d \in \mathcal{D}_r} \sum_{p' \in \mathcal{P}_d \setminus \{r\}} \lambda_{p'd}(\mathbf{x}_d) \\
& - c_r \gamma_r. \quad (27)
\end{aligned}$$

Setting $\frac{\partial C}{\partial b(\mathbf{x}_r)} = 0$, we have

$$\begin{aligned}
\ln b(\mathbf{x}_r) = & \sum_{a \in \mathcal{A}_r} \ln f_a(\mathbf{x}_a) + \sum_{p \in \mathcal{P}_r} \lambda_{pr}(\mathbf{x}_r) \\
& + c_r \sum_{d \in \mathcal{D}_r} \sum_{p' \in \mathcal{P}_d \setminus \{r\}} \lambda_{p'd}(\mathbf{x}_d) + \gamma_r. \quad (28)
\end{aligned}$$

Defining $m_{pr}(\mathbf{x}_r) = \exp(\lambda_{pr}(\mathbf{x}_r))$, we have

$$\begin{aligned}
b(\mathbf{x}_r) = & \exp(\gamma_r) \prod_{a \in \mathcal{A}_r} f_a(\mathbf{x}_a) \prod_{p \in \mathcal{P}_r} m_{pr}(\mathbf{x}_r) \\
& \times \prod_{d \in \mathcal{D}_r} \prod_{p' \in \mathcal{P}_d \setminus \{r\}} m_{p'd}(\mathbf{x}_d) \quad (29)
\end{aligned}$$

$$\Rightarrow b(\mathbf{x}_r) \propto \prod_{a \in \mathcal{A}_r} f_a(\mathbf{x}_a) \prod_{p \in \mathcal{P}_r} m_{pr}(\mathbf{x}_r) \quad (30)$$

$$\times \prod_{d \in \mathcal{D}_r} \prod_{p' \in \mathcal{P}_d \setminus \{r\}} m_{p'd}(\mathbf{x}_d). \quad (31)$$

For $r \in \mathcal{A}^{(LDPC)}$ and $\mathbf{x}_r: \text{parity}(\mathbf{x}_r) = 1$, we get

$$\begin{aligned}
\frac{\partial C}{\partial b(\mathbf{x}_r)} = & c_r \ln b(\mathbf{x}_r) - c_r \sum_{p \in \mathcal{P}_r} \lambda_{pr}(\mathbf{x}_r) \\
& - c_r \sum_{d \in \mathcal{D}_r} \sum_{p' \in \mathcal{P}_d \setminus \{r\}} \lambda_{p'd}(\mathbf{x}_d) - c_r \gamma_r. \quad (32)
\end{aligned}$$

Using $m_{pr}(\mathbf{x}_r) = \exp(\lambda_{pr}(\mathbf{x}_r))$, we have

$$\begin{aligned}
b(\mathbf{x}_r) = & \exp(\gamma_r) \prod_{p \in \mathcal{P}_r} m_{pr}(\mathbf{x}_r) \\
& \times \prod_{d \in \mathcal{D}_r} \prod_{p' \in \mathcal{P}_d \setminus \{r\}} m_{p'd}(\mathbf{x}_d), \quad (33)
\end{aligned}$$

$$\Rightarrow b(\mathbf{x}_r) \propto \prod_{p \in \mathcal{P}_r} m_{pr}(\mathbf{x}_r) \prod_{d \in \mathcal{D}_r} \prod_{p' \in \mathcal{P}_d \setminus \{r\}} m_{p'd}(\mathbf{x}_d). \quad (34)$$

Since $\mathcal{A}_r = \{\}$ for $r \in \mathcal{A}^{(LDPC)}$, the equation (34) is same as (30).

Equation (30) gives us the belief-update rules and the proportionality constant can be obtained by the normalization constraints in (22). The message update rules can be obtained using

$$\begin{aligned}
\sum_{\mathbf{x}_p \setminus r} b(\mathbf{x}_p) = & b(\mathbf{x}_r), \quad \forall r \in \mathcal{R} \setminus \mathcal{A}, \quad \forall p \in \mathcal{P}_r, \\
= & \prod_{a \in \mathcal{A}_r} f_a(\mathbf{x}_a) \prod_{p'' \in \mathcal{P}_r} m_{p''r}(\mathbf{x}_r) \\
& \times \prod_{d \in \mathcal{D}_r} \prod_{p' \in \mathcal{P}_d \setminus \{r\}} m_{p'd}(\mathbf{x}_d). \quad (35)
\end{aligned}$$

Writing

$$\prod_{p'' \in \mathcal{P}_r} m_{p''r}(\mathbf{x}_r) = m_{pr}(\mathbf{x}_r) \prod_{p'' \in \mathcal{P}_r \setminus \{p\}} m_{p''r}(\mathbf{x}_r), \quad (36)$$

we can obtain $m_{pr}(\mathbf{x}_r)$ as in (37)

Equations (30) and (37) provide the update rules for the joint-detection-decoding algorithm. Algorithm 1 summarized the joint detection and decoding engine derived in this section.

D. Remarks

It must be noted that in the formulation of the joint algorithm,

- 1) Regions corresponding to LDPC constraints are considered along with the 2D ISI constraints.
- 2) The beliefs of the regions corresponding to LDPC constraints are restricted to the words that satisfy the parity checks.

The implementation differs in using more regions, but ignoring the belief update equations in (30) for the words that do not satisfy parity check constraints. Similarly, if a modulation code is used for constraining 3×3 patterns, the algorithm is updated to ignore the belief update equations for the corresponding forbidden patterns.

The GBP algorithm for 2D ISI constraints is known to be computationally very hard and is not suitable for large page sizes. In the current formulation of the GBP algorithm for joint detection and decoding, the size of the regions corresponding to LDPC constraints is dictated by the row weight of the LDPC code. Since the number of beliefs for each region depend on its size, it is practically not feasible to use this formulation of the algorithm with very large native codeword lengths where the row weights are typically high. For our simulations, we have considered a small 2D page with multiple small LDPC codes interleaved for the to ensure the computational feasibility of the GBP algorithm.

$$m_{pr}(\mathbf{x}_r) = \frac{\sum_{\mathbf{x}_p \setminus r} b(\mathbf{x}_p)}{\prod_{a \in \mathcal{A}_r} f_a(\mathbf{x}_a) \prod_{p' \in \mathcal{P}_r} m_{p'r}(\mathbf{x}_r) \prod_{d \in \mathcal{D}_r} \prod_{p' \in \mathcal{P}_d \setminus \{r\}} m_{p'd}(\mathbf{x}_d)}. \quad (37)$$

Algorithm 1 GBP based Joint Detection and Decoding Algorithm.

Inputs: The input page size $H \times W$, the ISI span, LDPC code parity check matrix. Region graph formed based on the ISI and LDPC constraints. \mathcal{R} set of regions. $\mathcal{A}^{(LDPC)}$ set of regions corresponding to LDPC constraints.

Initialization: Initialize all parent to child messages m_{pc} to 1 and all beliefs to uniform distribution. Set the beliefs $b(\mathbf{x}_r) = 0$ for $r \in \mathcal{A}^{(LDPC)}$ when parity (\mathbf{x}_f) . Set iteration count $iter = 1$.

Processing:

Loop until $iter = threshold$

- **Message updates:** For each $r \in \mathcal{R}$, update messages as

$$m_{pr}(\mathbf{x}_r) = \frac{\sum_{\mathbf{x}_p \setminus r} b(\mathbf{x}_p)}{\prod_{a \in \mathcal{A}_r} f_a(\mathbf{x}_a) \prod_{p' \in \mathcal{P}_r} m_{p'r}(\mathbf{x}_r) \prod_{d \in \mathcal{D}_r} \prod_{p' \in \mathcal{P}_d \setminus \{r\}} m_{p'd}(\mathbf{x}_d)}$$

- **Belief updates:**

- For each $r \in \mathcal{R} \setminus \mathcal{A}^{(LDPC)}$, update all beliefs

$$b(\mathbf{x}_r) = \prod_{a \in \mathcal{A}_r} f_a(\mathbf{x}_a) \prod_{p \in \mathcal{P}_r} m_{pr}(\mathbf{x}_r) \prod_{d \in \mathcal{D}_r} \prod_{p' \in \mathcal{P}_d \setminus \{r\}} m_{p'd}(\mathbf{x}_d).$$

- For each $r \in \mathcal{A}^{(LDPC)}$, update beliefs only if parity $(\mathbf{x}_r) = 0$:

$$b(\mathbf{x}_r) = \prod_{p \in \mathcal{P}_r} m_{pr}(\mathbf{x}_r) \prod_{d \in \mathcal{D}_r} \prod_{p' \in \mathcal{P}_d \setminus \{r\}} m_{p'd}(\mathbf{x}_d).$$

- Normalize the beliefs to satisfy

$$\sum_{\mathbf{x}_r} b(\mathbf{x}_r) = 1$$

- Set

$$iter = iter + 1.$$

Outputs: The beliefs of 1×1 regions $b(x_{i,j})$.

IV. SIMULATIONS AND RESULTS

We have simulated the TDMR channel using a Voronoi based media model as proposed in [11]. The channel model parameters include grain size of CTC = 9 nm, bit-size = 18×18 nm, achieving a channel bit density of 1.8 Tb/in². 2D pages of size 128×256 data corresponding to 4KB sector are generated at a time. The readback signal is generated using the Voronoi based media model and is equalized using a 5×5 PR equalizer for a 3×3 PR target designed under monic constraint as in [11].

A. Burst Erasure Correction Performance of Proposed 2D LDPC Code

We use the media defect model in [6] to introduce rectangular defects. The defect detector proposed in [6] is used at the output of 2D SOVA to detect the location of the burst erasures and to indicate the LDPC decoder using the BP algorithm. The performance of the proposed 2D LDPC code is compared against 1D QC LDPC code in the read channel architecture as in [6].

The 2D LDPC code is constructed using a cube of size $p = 32$. For a 2D codeword size of 128×256 , we choose to construct parity check tensor \mathbf{H}_{2D} by filling $c \times h \times w =$

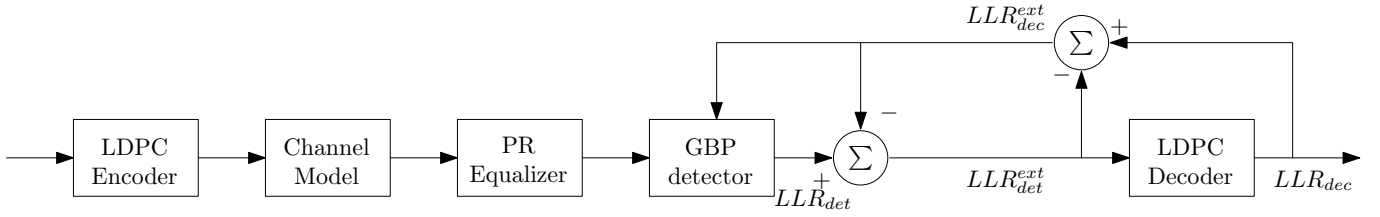
$128 \times 4 \times 8$. This gives us a code-rate of $1 - \frac{128 \times 32}{128 \times 256} = 0.875$, row weight of 32 and column weight of 4. The 1D LDPC code of length $128 \times 256 = 32768$ bits is designed with code rate of 0.87 and a circulant size of 1024. 1D codes are populated into the 2D page in the raster order. The code rates are chosen such that the detector achieves BER of $\sim 1e-4$. This detector performance could achieve sector failure rate (SFR) of $1e-15$ with a carefully designed LDPC codes.

Figure 7 compares the performance of the TDMR channel using 1D LDPC codes as against the proposed 2D LDPC codes. We can see that the proposed 2D LDPC codes is able to correct burst erasures of size 42×42 with the similar performance as 1D LDPC codes with erasures of size 38×38 . This is a significant ($> 20\%$) improvement in the burst erasure correction capability.

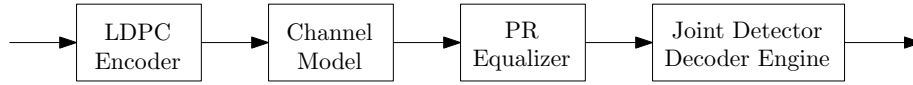
B. Performance of Joint Detection-Decoding Engine

To evaluate the performance of the joint detection and decoding engine proposed in Section III, we compare the performance of following architectures:

- 1) A GBP based detector followed by BP algorithm for LDPC decoding. The detector and the decoder operate in a turbo fashion [6] as shown in Figure 6a. The number of turbo iterations is limited to 6.



(a) A read channel architecture showing GBP based ISI detector operating in a turbo loop with LDPC decoder.



(b) A read channel architecture where the proposed joint detection and decoding engine is used.

Figure 6: The performance of joint detection and decoding engine is compared against the performance of an architecture with the detector operating in turbo loop with decoder. Same LDPC code and channel conditions are used.

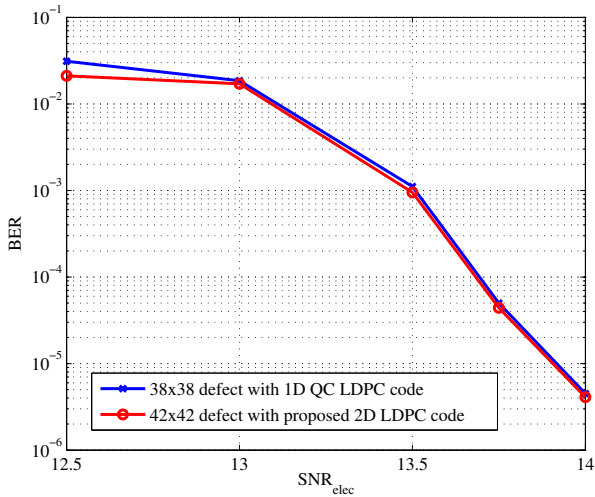


Figure 7: BER performance of the proposed 2D LDPC code is compared against a 1D QC LDPC code for burst erasure correction performance. The proposed 2D LDPC code is able to correct bursts of $> 20\%$ larger than the 1D QC LDPC code with a similar performance.

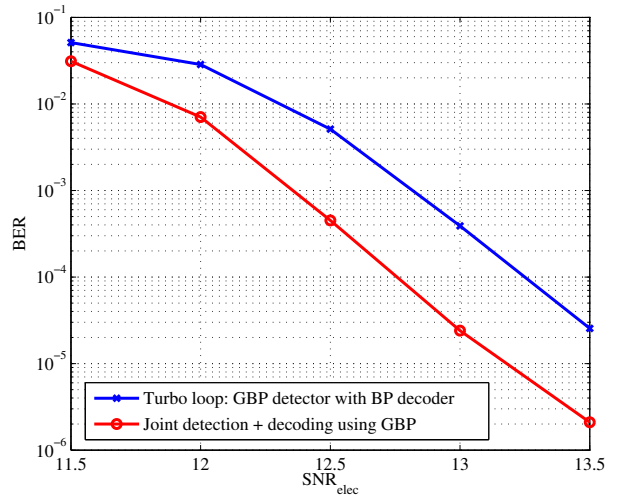


Figure 8: BER performance proposed joint detection-decoder engine is compared against a traditional architecture where the detector and the decoder operate in a turbo loop with 6 iterations. The proposed joint detection engine outperforms the turbo loop architecture by about 0.5 dB.

- 2) The PR equalized samples are jointly detected and decoded using the proposed algorithm in Section III as shown in Figure 6b.

Since the GBP algorithm is computationally not feasible for large page sizes, we have evaluated the performance on a small page size of 16×16 . We expect higher gains using the joint detection-decoding engine for larger page sizes. Since the joint detection engine is limited by the row-weight of the LDPC code, we have designed a 2D LDPC code of size 8×16 with $p = 4$, $c \times h \times w = 8 \times 2 \times 4$. The rows of two codewords are alternatively interleaved to obtain a 2D codeword of size 16×16 .

Figure 8 compares the performance of the two architectures for a TDMR channel operating at 1.8 Tb/in^2 . We observe that the proposed joint detection-decoder engine outperforms the turbo loop architecture by about 0.5 dB.

V. CONCLUSIONS

We have proposed the construction of a native 2D LDPC code suitable for correction of 2D burst erasures. We have looked into the zero-span of the codes and lower bounded the burst erasure size that the code can correct. The proposed code construction is observed to give a gain of $> 20\%$ improvement in the burst erasure correction capability. We have also derived the GBP algorithm for joint detection and decoding by taking care of 2D ISI constraints as well as the LDPC code constraints. The proposed joint detection-decoding engine is observed to give a gain of about 0.5 dB over an architecture where separate detector and decoder operate in turbo fashion. It would be interesting to develop hardware architectures using FPGA arrays and hardware acceleration engines towards performance assessment for large page sizes.

ACKNOWLEDGMENTS

We thank the Indo-US Science and Technology Forum, grant no. JC-Data Storage Research/16-2014, for funding the work presented in this paper.

REFERENCES

- [1] R. Wood, M. Williams, A. Kavčić, and J. Miles, "The Feasibility of Magnetic Recording at 10 Terabits per Square inch on Conventional Media," *IEEE Trans. Magn.*, vol. 45, pp. 917–923, Feb. 2009.
- [2] B. Vasić and M. E. Kurtas, *Coding and Signal Processing For Magnetic Recording Systems*. CRC Press LLC, 2004.
- [3] M. P. C. Fossorier, "Quasicyclic low-density Parity-Check Codes from Circulant Permutation Matrices," *IEEE Tran. on Inform. Theory*, vol. 50, Aug. 2004.
- [4] S. Garani and S. Parthasarathy, "Identifying a Defect in a Data-Storage Medium," *U.S. Patent 9,324,370*, Apr. 2016.
- [5] W. Tan, H. Xia, and J. Cruz, "Erasure Detection Algorithms for Magnetic Recording Channels," *IEEE Trans. Magn.*, vol. 40, pp. 3099–3101, Jul. 2004.
- [6] C. K. Matcha and S. G. Srinivasa, "Defect Detection and Burst Erasure Correction for TDMR," *IEEE Trans. on Magn.*, vol. 52, Nov. 2016.
- [7] S. Roy and S. G. Srinivasa, "Two Dimensional Error-Correcting Codes using Finite Field Fourier Transform," in *IEEE Inform. Theory Work.*, Oct. 2015.
- [8] S. W. Yoon and J. Moon, "Two-Dimensional Error-Pattern-Correcting Codes," *IEEE Trans. on Comm.*, vol. 63, Aug. 2015.
- [9] M. Fossorier, "Universal Burst Error Correction," in *IEEE Inter. Symp. on Inform. Theory*, pp. 1969–1973, Jul. 2006.
- [10] Y. Chen and S. G. Srinivasa, "Joint Self-Iterating Equalization and Detection for Two-Dimensional Intersymbol-Interference," *IEEE Trans. Comm.*, vol. 61, pp. 3219–3230, Aug. 2013.
- [11] C. K. Matcha and S. G. Srinivasa, "Generalized Partial Response Equalization and Data Dependent Noise Predictive Signal Detection over Media Models for TDMR," *IEEE Trans. Magn.*, vol. 51, Oct. 2015.
- [12] S. Datta and S. G. Srinivasa, "Design Architecture of a 2-d Separable Iterative Soft-Output Viterbi Detector," *IEEE Trans. on Magn.*, vol. 52, May 2016.
- [13] B. M. Kurkoski, P. H. Siegel, and J. K. Wolf, "Joint Message-Passing Decoding of LDPC Codes and Partial-Response Channels," *IEEE Trans. on Inform. Theory*, vol. 48, Jun. 2002.
- [14] W. Tan, S. Yang, K. Fitzpatrick, H. Zhong, L. Du, and Y. Lee, "Media Defect Recovery Using Full-Response Reequalization in Magnetic Recording Channels," in *IEEE Globecom*, pp. 1–5, Nov. 2008.
- [15] G. Sridharan, A. Kumarasubramanian, A. Thangaraj, and S. Bhashyam, "Optimizing Burst Erasure Correction of LDPC Codes by Interleaving," in *IEEE Inter. Symp. on Inform. Theory*, pp. 1143–1147, Jul. 2008.
- [16] K. Li, A. Kavčić, and M. Erden, "Construction of Burst-Erasure Efficient LDPC Codes for Use with Belief Propagation Decoding," in *IEEE Int. Conf. on Comm.*, pp. 1–5, May 2010.
- [17] E. Paolini and M. Chiani, "Construction of Near-Optimum Burst Erasure Correcting Low-Density Parity-Check Codes," *IEEE Trans. Comm.*, vol. 57, pp. 1320–1328, May 2009.
- [18] Y. Cassuto and A. Shokrollahi, "LDPC Codes for 2D Arrays," *IEEE Trans. on Inform. Theory*, vol. 60, Jun. 2014.
- [19] J. S. Yedidia, W. T. Freeman, and Y. Weiss, "Constructing Free-Energy Approximations and Generalized Belief Propagation Approximations," *Trans. Inform. Theory*, vol. 51, pp. 2282–2312, Jun. 2005.
- [20] C. K. Matcha, M. Bahrami, S. Roy, S. G. Srinivasa, and B. Vasić, "Generalized Belief Propagation based TDMR Detector and Decoder," in *IEEE Inter. Symp. on Inform. Theory*, pp. 210–214, Jul. 2016.
- [21] S. Khatami and B. Vasić, "Generalized Belief Propagation Detector for TDMR Microcell Model," *IEEE Trans. Magn.*, vol. 49, Jul. 2013.
- [22] P. Pakzad and V. Anantharam, "Estimation and Marginalization using Kikuchi Approximation Methods," *Neural Computation*, vol. 17, pp. 1836–1873, Aug. 2003.