Analysis of Absorbing Sets and Fully Absorbing Sets of Array-Based LDPC Codes

Lara Dolecek, Zhengya Zhang, Venkat Anantharam, Martin J. Wainwright, and Borivoje Nikolić*

dolecek@mit.edu; {zyzhang,ananth,wainwrig,bora}@eecs.berkeley.edu

January 28, 2008

Abstract

The class of low-density parity-check (LDPC) codes is attractive, since such codes can be decoded using practical message-passing algorithms, and their performance is known to approach the Shannon limits for suitably large blocklengths. For the intermediate blocklengths relevant in applications, however, many LDPC codes exhibit a so-called "error floor", corresponding to a significant flattening in the curve that relates signal-to-noise ratio (SNR) to the bit error rate (BER) level. Previous work has linked this behavior to combinatorial substructures within the Tanner graph associated with an LDPC code, known as (fully) absorbing sets. These fully absorbing sets correspond to a particular type of near-codewords or trapping sets that are stable under bit-flipping operations, and exert the dominant effect on the low BER behavior of structured LDPC codes. This paper provides a detailed theoretical analysis of these (fully) absorbing sets for the class of $C_{p,\gamma}$ array-based LDPC codes, including the characterization of all minimal (fully) absorbing sets for the array-based LDPC codes for $\gamma = 2, 3, 4$, and moreover, it provides the development of techniques to enumerate them exactly. Theoretical results of this type provide a foundation for predicting and extrapolating the error floor behavior of LDPC codes.

Keywords: LDPC codes; message passing decoding; bit-flipping; error floor; near-codeword; trapping set; absorbing set

1 Introduction

Low-density parity-check (LDPC) codes are a class of error-correcting codes based on sparse graphs. Their chief appeal is their excellent performance under practical decoding algorithms based on message passing, especially for moderate bit error rates (BER), say 10^{-6} and above [2, 18, 20]. As a consequence, LDPC codes have been adopted into several recent standards including Ethernet [25], digital video broadcasting [26], and broadband wireless [27].

[*] Work done while L. Dolecek was with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, Berkeley, CA, 94720. She is now with the Department of Electrical Engineering and Computer Sciences, Massachusetts Institute of Technology, Cambridge, MA, 02139. Z. Zhang, V. Anantharam, M. J. Wainwright and B. Nikolić are with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, Berkeley, CA, 94720.

However, several researchers [12, 16] have observed that LDPC codes often exhibit an *error floor*, meaning that beyond a certain signal-to-noise ratio (SNR), there is a significant change in slope in the plot of bit error rate (BER) versus SNR. For suitably designed codes, these error floors only occur at relatively low bit error rates (e.g. below 10^{-6}), and do not pose problems for applications requiring only moderately low BER, such as wireless communications. For other applications with low BER requirements, such as computer hard drives and optical channels, these error floors are extremely troublesome. An on-going line of research has shown that these error floors are closely related to the suboptimality of practical message-passing decoders. In early work, Mackay and Postol [12] recognized that certain classes of non-codewords, which they referred to as near-codewords, can cause the decoder to fail; in particular, an (a, b) *near-codeword* is a binary string of weight a whose syndrome has weight b. From simulation of a rate $1/2$ LDPC code with blocklength 2640 based on the Margulis construction, they found that $(12, 4)$ and $(14, 4)$ near-codewords are the main contributors to the error floor of this code when used for the transmission over an additive white Gaussian noise (AWGN) channel. They also postulated that the minimum distance of this code is significantly higher than the size of the observed near-codewords. Di et al. [1] defined a closely related concept of a *stopping set*, which governs the performance limits of iterative decoding for LDPC codes over the binary erasure channel (BEC). Subsequent work [14] has provided analytical characterization of the stopping set enumerator for different ensembles of LDPC codes. Although very useful for determining the performance over a BEC, stopping sets cannot be used directly to determine LDPC performance for other channels, such as AWGN channels, since the nature of errors in non-erasure channels is more subtle. For more general channels, pioneering work by Richardson [16] introduced the operationally-defined notion of a *trapping set* in order to address the error floor of LDPC codes, and developed a fast numerical method for estimating the error probability in the low BER region. Other researchers have studied closely related notions of elementary trapping sets [11], pseudocodewords for iterative decoding [8, 9], and pseudocodewords for linear-programming decoding [6].

In previous experimental work [23], we used a hardware emulator to explore the low BER regime of various classes of structured LDPC codes. On the basis of these experiments, we isolated a set of graph substructures, referred to as *absorbing sets*, that cause various message-passing decoders to fail by converging to these non-codeword states. Like stopping sets, these objects have a purely combinatorial definition in terms of the parity check matrix of a given LDPC code. They can be viewed as a special type of a near-codeword or a trapping set, in particular one that is guaranteed to be stable under a bit-flipping decoder. These absorbing sets represent the dominant contribution to the error floor for iterative sum-product decoding.

Under maximum likelihood decoding, it is well-known that the minimum distance and the weight enumerator of a code are key factors that determine its error-correcting performance. Given that absorbing sets (as opposed to neighboring codewords) are the dominant error event for iterative decoders, it is natural to consider the analogs of minimum distance and weight enumerator for absorbing sets. Herein lies the main contribution of this paper: in particular, we provide a detailed and systematic theoretical analysis of the absorbing sets of high rate array-based LDPC codes [5]. This class of codes is an exemplar of a structured LDPC code with excellent performance in the moderate BER region, but whose low BER performance is governed by the minimal absorbing sets. For this class of structured LDPC codes, we prove the non-existence of various possible candidate absorbing sets, and having thereby explicitly constructed minimal absorbing sets, we characterize their combinatorial structure and cardinalities. As described elsewhere [3], this explicit enumeration of absorbing sets is a key ingredient in an importance-sampling-based method for estimating error floor probabilities. In our concurrent work [24], the notion of absorbing sets has also been shown to be an extremely important component in the novel designs of practical implementations of high-throughput LDPC decoders. While the focus of this paper is on the theoretical description of the absorbing sets of high rate array-based LDPC codes, the complementary work [24] considers the AWGN transmission, and it contains numerous experimental hardware-based results which consistently support the claim that the absorbing sets dominate the error floor under a variety of iterative decoding algorithms.

The remainder of this paper is organized as follows. We begin in Section 2 with a brief overview of the class of array-based LDPC codes [5], and then formally introduce the definition of absorbing sets. In Section 3, we provide a detailed study of the absorbing sets for column weights $\gamma = 2$, 3 and 4 for the standard parity check matrices $H_{p,\gamma}$ of such codes, and enumerate all such sets of smallest size. All of the theoretical results are stated in this section, with some of the more technical proofs deferred to the Appendix. Section 4 concludes the paper.

2 Background

We begin with background on array-based LDPC codes, and then provide a precise definition of absorbing sets.

2.1 Array-based LDPC codes

Array-based LDPC codes [5] are regular LDPC codes parameterized by a pair of integers (p, γ), such that $\gamma \leq p$, and p is an odd prime. Given a $p \times p$ permutation matrix σ of the form

$$\sigma = \begin{bmatrix} 0 & 0 & \ldots & 0 & 1 \\ 1 & 0 & \ldots & 0 & 0 \\ 0 & 1 & \ldots & 0 & 0 \\ \vdots & \vdots & \ldots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & 0 \end{bmatrix}, \tag{1}$$

we form the $p\gamma \times p^2$ parity check matrix $H_{p,\gamma}$

$$H_{p,\gamma} = \begin{bmatrix} I & I & I & \dots & I \\ I & \sigma & \sigma^2 & \dots & \sigma^{p-1} \\ I & \sigma^2 & \sigma^4 & \dots & \sigma^{2(p-1)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ I & \sigma^{\gamma-1} & \sigma^{(\gamma-1)2} & \dots & \sigma^{(\gamma-1)(p-1)} \end{bmatrix}. \tag{2}$$

We use $C_{p,\gamma}$ to denote the binary linear code defined by this parity check matrix (2). The rate[1] of this code is $R = 1 - \frac{\gamma p - \gamma + 1}{p^2}$ [13]. Fan [5] first demonstrated that array-based LDPC codes have very good performance, and they have subsequently been proposed for a number of applications, including digital subscriber lines [4] and magnetic recording [21].

2.2 Absorbing Sets

A convenient representation of a $m \times n$ parity check matrix H of a binary linear code is in terms of its factor or Tanner graph [7, 10, 19]. In particular, let $G_H = (V, F, E)$ denote a bipartite graph, in which the set of vertices V is associated with n bits in the code (columns of H), and the set F is associated with m checks of the code (rows of H). The edge set E is defined by the structure of H: in particular, there exists an edge $e(i, j) \in E$ if and only if $i \in V$ and $j \in F$. Elements of V are called “bit nodes” and elements of F are called “check nodes”.

For the array-based LDPC codes defined previously, the factor graph associated with $H_{p,\gamma}$ does not have any cycles of length four, and thus the girth is at least six (see [5]). For any subset D of V we let N_D denote the subset of check nodes neighboring the elements of D. For any subset D of V, let $\mathcal{E}(D)$ (resp. $O(D)$) be the set of neighboring vertices of D in F in the graph G with even (resp. odd) degree with respect to D. With this set-up, we have the following:

Definition 1 *Given an integer pair (a, b), an (a, b)* absorbing set *is a subset $D \subseteq V$ of size a, with $O(D)$ of size b and with the property that each element of D has strictly fewer neighbors in $O(D)$ than in $F \backslash O(D)$. We say that an (a, b) absorbing set D is an (a, b)* fully absorbing set, *if in addition, all bit nodes in $V \backslash D$ have strictly more neighbors in $F \backslash O(D)$ than in $O(D)$.*

An example of a $(4, 4)$ absorbing set is given in Figure 1, where dark circles represent bits in the set D, dark squares constitute the set $O(D)$, white squares constitute the set $\mathcal{E}(D)$, $E(D, O(D))$ is given by solid lines, and $E(D, \mathcal{E}(D))$ is given by dashed lines. Observe that each element in D has more neighbors with even degree than odd degree. All check nodes not in the picture are denoted by empty squares. For this set

[1] Note that the parity check matrix $H_{p,\gamma}$ is not full rank, hence the slight increase in rate over $1 - \gamma/p$.

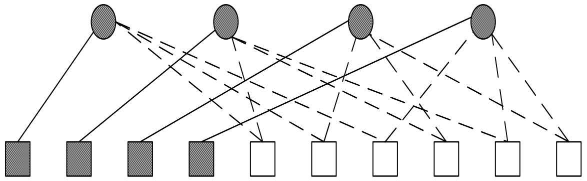

Figure 1: An example of a $(4, 4)$ absorbing set.

to be a fully absorbing set, every bit node not in the figure should also have strictly more empty squares than full squares as neighbors.

Note that $D \subseteq V$ is a fully absorbing set if and only if for all $v \in V$, $wt(Hx_{D\Delta v}) > wt(Hx_D) = b$, where $D\Delta v$ denotes the symmetric difference between D and $\{v\}$, $wt(y)$ is the Hamming weight of a binary string y, and x_D is a binary string with support D.

For the special case of a bit flipping algorithm [17], the configuration described as a fully absorbing set is stable, since each bit node receives strictly more messages from the neighboring checks that reinforce its value than messages that suggest the opposite bit value. However, as shown in previous hardware-based emulation [23], and concurrent work [24], absorbing sets also control the error floor behavior of more sophisticated message-passing decoders, such as the sum-product algorithm.

3 Theoretical Results

Our goal is to describe minimal absorbing sets and minimal fully absorbing sets (a, b) of the factor graph of the parity check matrix $H_{p,\gamma}$, for $\gamma = 2, 3, 4$, where the minimality refers to the smallest possible a, and where b is the smallest possible for the given a.

We use the following notation throughout the paper. Recall that $H_{p,\gamma}$ is a $\gamma p \times p^2$ matrix of 0's and 1's. It is convenient to view $H_{p,\gamma}$ as a two-dimensional array of component $p \times p$ submatrices with the rows i in the range $0 \leq i \leq \gamma - 1$ (also referred to as row groups) and the columns j in the range $0 \leq j \leq p - 1$ (also referred to as column groups). Each column of $H_{p,\gamma}$ is uniquely described by a pair (j, k) where j denotes the column index of the submatrix this column belongs to, and k, $0 \leq k \leq p - 1$, denotes the index of this column within the submatrix.

Let $G_{p,\gamma}$ be the factor graph associated with $H_{p,\gamma}$, so bit nodes and check nodes in $G_{p,\gamma}$ represent columns and rows in $H_{p,\gamma}$, respectively. In the graph $G_{p,\gamma}$, bit nodes have degree γ and check nodes have degree p. There is a total of p^2 bit nodes and γp check nodes. Each bit node in $G_{p,\gamma}$ receives the unique label (j, k) that describes the corresponding column of $H_{p,\gamma}$. Each check node in $G_{p,\gamma}$ receives a label i if the corresponding row of $H_{p,\gamma}$ belongs to the row group i. Multiple bit nodes can have the same j or k label, but not both. Multiple check nodes can have the same i label.

We note that the structure of the parity check matrix imposes the following conditions on the neighboring bit nodes and check nodes:

Bit Consistency: For a bit node, all its incident check nodes, labelled i_{s_1} through i_{s_γ}, must have distinct labels, i.e. these check nodes are in distinct row groups.

Check Consistency: All bit nodes, say (j_1, k_1) through (j_p, k_p), participating in the same check node must have distinct j_ℓ values, i.e. they are all in distinct column groups.

Both conditions follow from the fact that the parity check matrix $H_{p,\gamma}$ of $C_{p,\gamma}$ consists of a 2-dimensional array of permutation matrices of equal size. ■

We begin with elementary lemmas that play a central role throughout the paper.

Lemma 1 *(Pattern Consistency): The permutation submatrix σ has the following properties:*

(a) the (r, k) entry of σ^i is 1 if and only if $r - k \equiv i \mod p$.

(b) Let σ^{ij_1} and σ^{ij_2} be in the same row group of $H_{p,\gamma}$. If entry (r, k_1) of σ^{ij_1} is non-zero, then so is entry (r, k_2) of σ^{ij_2} where $k_1 + ij_1 \equiv k_2 + ij_2 \equiv r \mod p$.

We will refer to the constraints of the type described in Lemma 1 as *pattern consistency* constraints.

Lemma 2 *(Cycle consistency:) Consider a cycle in $G_{p,\gamma}$ of length $2t$, involving t bit nodes, with labels (j_1, k_1) through (j_t, k_t) and t check nodes, with labels i_1 through i_t, such that bit nodes (j_1, k_1) and (j_2, k_2) participate in the check labelled i_1, (j_2, k_2) and (j_3, k_3) participate in the check labelled i_2, and so on, until check labelled i_t in which (j_t, k_t) and (j_1, k_1) participate. Then*

$$i_1(j_2 - j_1) + i_2(j_3 - j_2) + \cdots + i_{t-2}(j_{t-1} - j_{t-2}) + i_{t-1}(j_t - j_{t-1}) + i_t(j_1 - j_t) \equiv 0 \mod p. \quad (3)$$

Proof: The pattern consistency constraints of Lemma 1(b) give:

$$\begin{array}{ccll} k_1 + i_1 j_1 & \equiv & k_2 + i_1 j_2 & \mod p, \\ k_2 + i_2 j_2 & \equiv & k_3 + i_2 j_3 & \mod p, \\ & \vdots & & \\ k_{t-1} + i_{t-1} j_{t-1} & \equiv & k_t + i_{t-1} j_t & \mod p, \\ k_t + i_t j_t & \equiv & k_1 + i_t j_1 & \mod p. \end{array} \quad (4)$$

Expand $k_1 - k_2$ into $(k_1 - k_t) - (k_{t-1} - k_t) - (k_{t-2} - k_{t-1}) - \cdots - (k_2 - k_3)$. Hence,

$$i_1(j_2 - j_1) \equiv i_t(j_t - j_1) - i_{t-1}(j_t - j_{t-1}) - i_{t-2}(j_{t-1} - j_{t-2}) - \cdots - i_2(j_3 - j_2) \mod p. \quad (5)$$

By rearranging the terms, relation (3) follows. ■

Constraints of the type (3) will subsequently be referred to as *cycle consistency* constraints. Note that the *cycle consistency* constraints are a consequence of the *pattern consistency* constraints.

Our main results can be summarized as follows: Let $G_{p,\gamma}$ be the factor graph associated with the parity check matrix $H_{p,\gamma}$ of the array-based LDPC code $C_{p,\gamma}$.

Theorem 1 (Minimality) *(a) For the $G_{p,2}$ family, all minimal absorbing sets are minimal fully absorbing sets, and are of size $(4,0)$.*

(b) For the $G_{p,3}$ family, the minimal absorbing sets are of size $(3,3)$, and the minimal fully absorbing sets are of size $(4,2)$.

(c) For the $G_{p,4}$ family, and for $p > 19$, the minimal absorbing sets and the minimal fully absorbing sets are of size $(6,4)$.

Our next result deals with the scaling behavior of the number of absorbing sets. Recall the standard asymptotic notation Θ: we say that some positive function $f(n)$ grows as $\Theta(n^{\ell})$ if there exist constants $0 < c \leq c' < +\infty$ such that $c\, n^{\ell} \leq f(n) \leq c'\, n^{\ell}$, for n sufficiently large.

Theorem 2 (Scaling) *Recalling that the blocklength $n = p^2$ of the $C_{p,\gamma}$ code corresponds to the number of columns in the parity check matrix $H_{p,\gamma}$, we have:*

(a) For $\gamma = 2$, the number of minimal (fully) absorbing sets in $G_{p,\gamma}$ grows as $\Theta(n^2)$.

(b) For $\gamma = 3$, the number of minimal absorbing sets as well as the number of minimal fully absorbing sets grows as $\Theta(n^{3/2})$.

(c) For $\gamma = 4$ and for all blocklengths $n > 19^2$ the number of minimal absorbing sets as well as the number of minimal fully absorbing sets grows as $\Theta(n^{3/2})$.

In the following three subsections, we provide proofs of these claims, where we treat each of the values of γ separately. Although Theorem 2 states the result in terms of the Θ-scaling behavior, our techniques in fact provide an exact count of the number of minimal (fully) absorbing sets. Note that Theorem 1(a) implies that for $\gamma = 2$, the smallest (fully) absorbing sets are codewords; in fact, for this code, these absorbing sets are the minimum distance codewords. This result should be contrasted with the assertions of Theorem 1(b) and (c), for $\gamma = 3$ and $\gamma = 4$ respectively, which establish the existence of (fully) absorbing sets *strictly smaller* than the minimum distance of the code. In particular, for $\gamma = 3$, the minimum distance is six [22, 13], whereas for $\gamma = 4$ and $p > 7$, the minimum distance is between eight and ten [22, 13]. Therefore, for both $\gamma = 3$ and $\gamma = 4$, the minimal absorbing sets and minimal fully absorbing sets are strictly smaller than the minimum distance of the code.

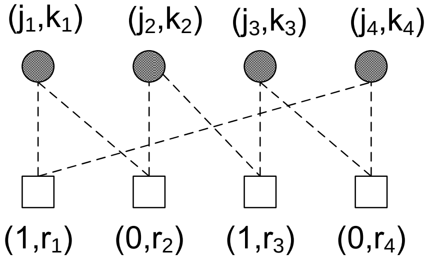

Figure 2: (Labelled) candidate $(4, 0)$ absorbing set.

3.1 Proof of Theorem 1(a) and 2(a)

We start by proving Theorem 1(a). The code $C_{\gamma,2}$ has uniform bit degree two, and is thus a cycle code. Even though such codes are known to be poor [15], we include the analysis for the sake of completeness.

Let $G_{p,2} = (V, F, E)$ denote the factor graph of $H_{p,2}$. Let D be an (a, b) absorbing set in $G_{p,2}$. Each bit node in D has degree 2 in $G_{p,2}$ and is required to have strictly more neighbors in $\mathcal{E}(D)$ than in $O(D)$. This implies that $O(D)$ is empty. The absorbing set is of type $(a, 0)$. It is thus a fully absorbing set, and is in fact a codeword.

Since the matrix $H_{p,2}$ has the top row consisting of identity matrices, the codewords of $C_{p,2}$ are of even weight. Moreover, since the bottom row of $H_{p,2}$ consists of distinct component submatrices, no two columns of $H_{p,2}$ sum to zero. Therefore $a > 2$ and even and there are no cycles of length 4 in this code.

We now consider $a = 4$. Let (j_1, k_1), (j_2, k_2), (j_3, k_3) and (j_4, k_4) be the bit nodes participating in a candidate $(4, 0)$ absorbing set. These nodes must necessarily be arranged as in Figure 2.

The following result proves Theorem 1(a).

Lemma 3 *There is a total of* $p^2(p-1)^2$ $(4, 0)$ *(fully) absorbing sets in the code described by* $H_{p,2}$.

Proof: The bit consistency conditions are automatically satisfied by the numbering of the row groups in Figure 2. The check consistency constraints give:

$$j_1 \neq j_4, \qquad j_1 \neq j_2, \qquad j_2 \neq j_3, \quad \text{and} \quad j_3 \neq j_4, \tag{6}$$

whereas the pattern consistency constraints of Lemma 1(b) give:

$$k_1 = k_2, \quad \text{and} \quad k_3 = k_4, \quad \text{and} \tag{7a}$$

$$k_2 + j_2 \equiv k_3 + j_3 \mod p, \quad \text{and} \quad k_4 + j_4 \equiv k_1 + j_1 \mod p. \tag{7b}$$

There are p ways of choosing k_2, which also determines k_1. Since $j_2 \neq j_3$, we must have $k_3 \neq k_2$, so we have $(p-1)$ ways of choosing k_3, which also determines k_4. We then have p ways of choosing j_2, which also determines j_3. Since $j_1 \neq j_2$, we have $(p-1)$ ways of choosing j_1, which also determines j_4. To verify

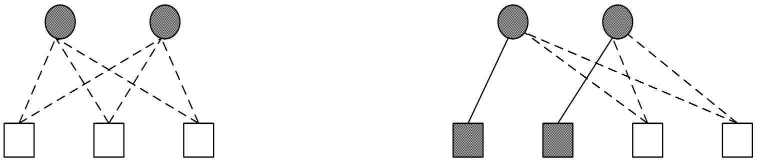

Figure 3: Candidate $(2, b)$ absorbing sets.

that every one of these choices satisfies all the equations it only remains to verify that $j_3 \neq j_4$. This holds because

$$j_3 - j_4 \equiv (k_2 - k_3 + j_2) - (k_1 - k_4 + j_1) \equiv j_2 - j_1 \neq 0 \mod p \, . \tag{8}$$

Now, for any choice of row group labels for the checks, and column labels for the bits that satisfy the bit and check consistency constraints and the pattern consistency constraints of Lemma 1(b), there is a unique way to choose the row index in the individual row groups so that the pattern consistency constraint of Lemma 1 are satisfied. This completes the proof of Lemma 3. ■

From Lemma 3 (and recalling that the blocklength $n = p^2$), we conclude that the number of $(4, 0)$ (fully) absorbing sets for the code described by $H_{p,2}$ is $\Theta(n^2)$, thereby establishing Theorem 2(a).

3.2 Proof of Theorem 1(b) and 2(b)

In our preceding analysis with $\gamma = 2$, note that $(4, 0)$ absorbing sets are actually codewords, so the performance of the cycle code under iterative decoding is dominated by low weight codewords. We now turn to the case $\gamma > 2$, which leads to more interesting results. In particular, our proof of Theorem 1(b) establishes the existence of minimal absorbing sets and minimal fully absorbing sets, for which the number of bit nodes a is *strictly smaller* than the minimum distance d_{min} of the code.

Let $G_{p,3} = (V, F, E)$ denote the factor graph of $H_{p,3}$. Let D be an (a, b) absorbing set in $G_{p,3}$. Each bit node in D has degree 3 in $G_{p,3}$ and is required to have strictly more neighbors in $\mathcal{E}(D)$ than in $O(D)$.

Suppose $a = 2$. In the graph $G_{p,3}$, an even number of edges from D terminates in $\mathcal{E}(D)$. Thus, either $b = 0$ or $b = 2$ corresponding to the situations in Figure 3. In either case there would be a cycle of length 4 in $G_{p,3}$, which cannot hold [5], implying that $a \geq 3$.

Suppose $a = 3$. In the graph $G_{p,3}$, an even number of edges from D terminates in $\mathcal{E}(D)$. Thus, either $b = 1$ or $b = 3$. Suppose $b = 1$. This must correspond to the left form in Figure 4, or the right form in Figure 4, which again involves a cycle of length 4 in $G_{p,3}$, a contradiction [5].

Still with $a = 3$, the remaining case to consider is $b = 3$. In this case, each bit node in D would then connect to exactly one check node in $O(D)$ implying the unlabelled form of Figure 5. Note that there is a cycle of length 6. Suppose that these 3 bit nodes are indexed as (j_1, k_1), (j_2, k_2) and (j_3, k_3), respectively, where j_1, j_2 and j_3 are distinct (by the check consistency) and $0 \leq j_1, j_2, j_3 \leq p - 1$. Without

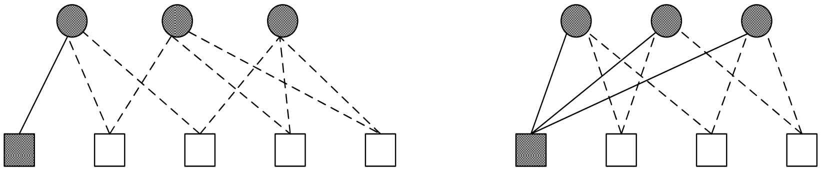

Figure 4: Candidate $(3,1)$ absorbing sets.

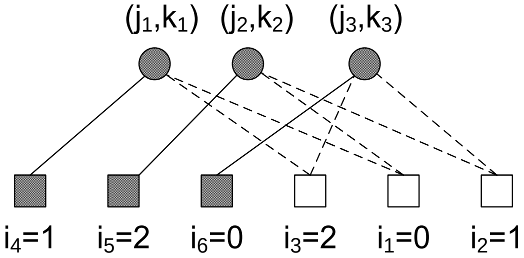

Figure 5: (Labelled) candidate $(3,3)$ absorbing set.

loss of generality, assume that (j_1,k_1) and (j_2,k_2) share a check in the row group i_1, (j_2,k_2) and (j_3,k_3) share a check in the row group i_2, and that (j_1,k_1) and (j_3,k_3) share a check in the row group i_3, where $i_1,i_2,i_3 \in \{0,1,2\}$ and are distinct by the bit consistency condition. We may assume without loss of generality that $i_1=0$, $i_2=1$ and $i_3=2$. Note that the bit consistency constraints force the values of i_4,i_5 and i_6 to be as given in Figure 5.

In the remainder of the discussion we first prove the existence of a $(3,3)$ absorbing set. We then show that these $(3,3)$ absorbing sets are not fully absorbing sets. This result will in turn imply the existence of $(4,2)$ fully absorbing sets, which are thus minimal fully absorbing sets for $\gamma=3$.

The bit consistency constraints are automatically satisfied by our labelling of the row groups in Figure 5. The check consistency constraints reduce to the distinctness of j_1, j_2 and j_3. The pattern consistency constraints of Lemma 1(b) give:

$$k_1+2j_1 \equiv k_3+2j_3 \mod p, \tag{9}$$

$$k_1 \equiv k_2 \mod p, \tag{10}$$

$$k_2+j_2 \equiv k_3+j_3 \mod p. \tag{11}$$

The existence of a solution and hence of a $(3,3)$ absorbing set is given in the proof of Lemma 4 below, which counts the number of such sets.

Even though a $(3,3)$ fully absorbing set seems plausible, care must be taken with respect to bit nodes *outside* a candidate fully absorbing set, that also participate in the unsatisfied checks. As we now show, a $(3,3)$ *fully* absorbing set cannot exist, though the existence of a $(3,3)$ absorbing set implies a $(4,2)$ fully absorbing set.

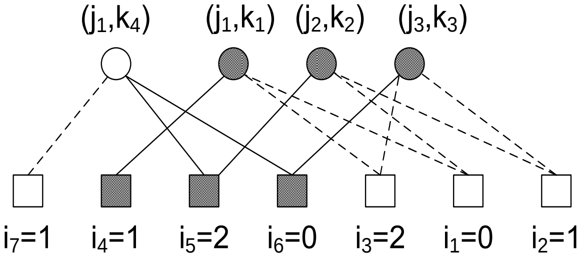

Figure 6: Candidate $(3,3)$ absorbing set (solid circles), with an adjacent bit node (empty circle).

Suppose first that a $(3,3)$ fully absorbing set were to exist. Since $\gamma = 3$, it is then necessary that no bit node outside of the absorbing set participates in more than one unsatisfied check adjacent to a $(3,3)$ absorbing set. Since (j_1, k_1) and (j_3, k_3) share a check, $j_1 \neq j_3$. Consider the bit node labelled (j_1, k_4) that connects to i_6, as in Figure 6. Since $i_6 = 0$, it follows from Lemma 1(b) that $k_3 = k_4$. Equations (9)-(11) imply that $k_3 + 2j_1 \equiv k_2 + 2j_2 \mod p$ so that (j_1, k_4) (= (j_1, k_3)) bit node also connects to the check labelled i_5, as shown in Figure 6. This eliminates the possibility of a $(3,3)$ fully absorbing set.

A $(4,0)$ absorbing set (i.e. a codeword of weight 4) cannot exist since the minimum distance of the code is 6 [22]. The next candidate size for the smallest fully absorbing set is $(4,2)$. Each of the unsatisfied checks in any such configuration would necessarily connect to only one of the bit nodes, else we would have a cycle of length 4, a contradiction [5]. Given this, no satisfied check node can connect to all four bit nodes, else we would have a cycle of length 4, a contradiction [5]. Since there are 10 edges from the bit nodes that go to satisfied checks we now see that there must be 5 satisfied checks in any candidate $(4,2)$ fully absorbing set. The two bit nodes that each have all their three edges going to satisfied check nodes must then share exactly one satisfied check (they have to share at least one, and cannot share more than one [5]). We have therefore concluded that any candidate $(4,2)$ fully absorbing set must look like (an unlabelled version of) Figure 6. The existence of such $(4,2)$ fully absorbing sets is proved in Lemma 4, which also counts the number of such sets.

Lemma 4 *The total number of* $(3,3)$ *absorbing sets and* $(4,2)$ *fully absorbing sets in the factor graph* $G_{p,3}$ *is* $p^2(p-1)$*, and* $3p^2(p-1)/2$*, respectively.*

Proof: Referring to Figure 5, the bit consistency and the check consistency constraints are satisfied for the given labels of row groups and since j_1, j_2 and j_3 are distinct. Then j_1 and k_1 can each be chosen in p ways, and then j_3 can be chosen in $p-1$ ways. This fixes k_3 by equation (9), k_2 by equation (10) and then j_2 by equation (11). There is then a unique way to choose the row indices in the individual row groups so that the pattern consistency conditions of Lemma 1 are satisfied. Thus the total number of $(3,3)$ absorbing sets is $p^2(p-1)$.

Turning to counting $(4,2)$ fully absorbing sets, every such set must look like an unlabelled version of Figure 6, and so it contains exactly two distinct $(3, 3)$ absorbing sets (corresponding respectively to removing one of the bit nodes that connects to an unsatisfied check). From Figure 5 one can see that every $(3, 3)$ absorbing set is contained in three distinct $(4, 2)$ fully absorbing sets (for each pair of unsatisfied checks in Figure 5 one can find a bit node that these checks connect to, which when appended to the $(3, 3)$ absorbing set gives a $(4, 2)$ fully absorbing set). The total number of $(4, 2)$ fully absorbing sets is therefore $3p^2(p - 1)/2$. ■

Observe that Lemma 4 immediately implies Theorem 1(b).

3.3 Proof of Theorem 1(c) and 2(c)

In order to establish that $(6, 4)$ (fully) absorbing sets are minimal for $H_{p,4}$ and $p > 19$, we will first show that (a, b) absorbing sets for $a < 6$ do not exist. This section contains the following auxiliary results on the non-existence of certain candidate absorbing sets, which hold for sufficiently large code parameter p (specifically $p > 19$ will be sufficient for all auxiliary results) . In particular:
-Lemma 5 proves that $(4, 4)$ absorbing sets do not exist,
-Lemma 6 proves that $(5, b)$ absorbing sets do not exist, and
-Lemma 7 proves that $(6, 2)$ absorbing sets do not exist.
Lastly, Lemma 8 provides an in-depth analysis of the $(6, 4)$ absorbing sets.

Let D denote an (a, b) absorbing set in $G_{p,4} = (V, F, E)$, the factor graph of $H_{p,4}$. If $a = 2$ (respectively 3) then at least 6 (respectively 9) edges from D in $G_{p,4}$ terminate in $\mathcal{E}(D)$, which implies the existence of a cycle of length 4 in $G_{p,4}$, which is false [5]. Thus, $a \geq 4$.

Suppose $a = 4$ and note that b must be even. We cannot have $b = 0$, since this would imply the existence of a codeword of weight 4, which is false [22]. If $b = 2$, one can conclude that there must be a cycle of length 4 in the code (whether the number of edges going into unsatisfied checks is 2 or 4), and this is false, [5]. Thus we must have $b = 4$ and, since each bit node must have at least three edges going to satisfied checks, the impossibility of a cycle of length 4 [5] implies that the absorbing set can be described as in Figure 7. In this figure, each vertex represents a distinct bit node of the candidate $(4, 4)$ absorbing set and each edge represents a satisfied check node that connects to the bit nodes in the absorbing set, that correspond to its end points in the figure. The following lemma establishes that such sets do not exist if the prime p is large enough.

Lemma 5 *For $p > 7$, the factor graph family $G_{p,4}$ does not contain any $(4, 4)$ absorbing sets.*

Proof: Without loss of generality we may let $i_1 = x$, $i_4 = y$ and $i_5 = z$, where $x, y, z \in \{0, 1, 2, 3\}$ and distinct by the vertex consistency conditions. Then, by propagating the vertex consistency conditions at each remaining vertex, and exploiting the symmetry, it suffices to consider $(i_1, i_2, i_3, i_4, i_5, i_6)$ either (x, y, x, y, z, z) or (x, y, x, y, z, w) where $x, y, z, w \in \{0, 1, 2, 3\}$ and are distinct.

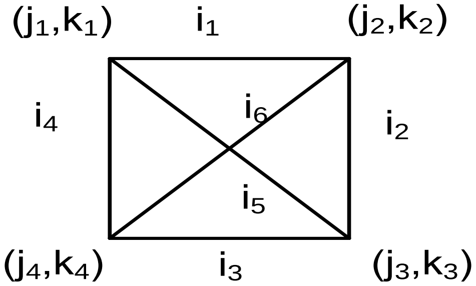

Figure 7: Depiction of the candidate $(4, 4)$ absorbing set.

For the case $(i_1, i_2, i_3, i_4, i_5, i_6)$ = (x, y, x, y, z, z), we establish the following cycle consistency conditions based on the cycles within the graph in Figure 7:

$$\begin{aligned} x(j_2 - j_1) + y(j_3 - j_2) + z(j_1 - j_3) &\equiv 0 \mod p, \\ x(j_2 - j_1) + z(j_4 - j_2) + y(j_1 - j_4) &\equiv 0 \mod p, \quad \text{and} \\ x(j_4 - j_3) + y(j_1 - j_4) + z(j_3 - j_1) &\equiv 0 \mod p. \end{aligned} \tag{12}$$

By adding and subtracting the conditions in (12), it follows that

$$\begin{aligned} (y - z)(j_3 + j_4 - j_1 - j_2) &\equiv 0 \mod p, \\ (x - z)(j_2 + j_3 - j_1 - j_4) &\equiv 0 \mod p, \quad \text{and} \\ (x - y)(j_2 + j_4 - j_1 - j_3) &\equiv 0 \mod p. \end{aligned} \tag{13}$$

Since x, y, z are distinct, relation (13) implies that j's would have to be all the same, which contradicts the check consistency constraint.

For the case $(i_1, i_2, i_3, i_4, i_5, i_6)$ = (x, y, x, y, z, w), again based on the cycle structure in Figure 7, we obtain the cycle consistency conditions

$$\begin{aligned} x(j_2 - j_1) + y(j_3 - j_2) + z(j_1 - j_3) &\equiv 0 \mod p, \\ x(j_2 - j_1) + w(j_4 - j_2) + y(j_1 - j_4) &\equiv 0 \mod p, \quad \text{and} \\ x(j_4 - j_3) + y(j_1 - j_4) + z(j_3 - j_1) &\equiv 0 \mod p. \end{aligned} \tag{14}$$

We let $u_1 := j_2 - j_1$, $u_2 := j_3 - j_1$, and $u_3 := j_4 - j_1$. By the check consistency condition, all of u_1, u_2, and u_3 are non-zero. Substituting u_1, u_2 and u_3 in (14) and then expressing u_2 and u_3 in terms of u_1, one arrives at the condition

$$(z - x)(w - y) + (z - y)(w - x) \equiv 0 \mod p. \tag{15}$$

It can be verified that this condition cannot hold for any choice of x, y, z, w, where $x, y, z, w \in \{0, 1, 2, 3\}$

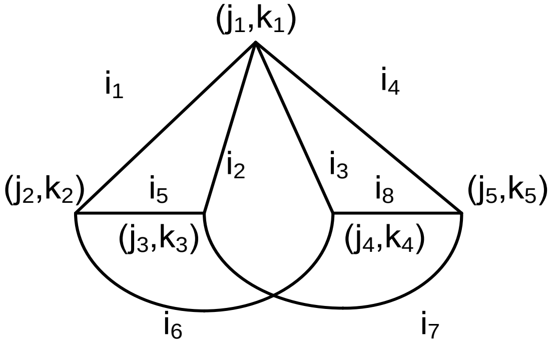

Figure 8: Depiction of the candidate $(5, 4)$ absorbing set.

and are distinct for $p > 7$. There are $4! = 24$ ways of assigning numerical values to (x, y, z, w). Substituting each numerical assignment (x, y, z, w) yields possible choices of prime p for which the expression in (15) becomes zero $\mod p$. The condition (15) holds for $p \in \{2, 5, 7\}$. Therefore, for $p > 7$, $G_{p,\gamma}$ does not contain $(4, 4)$ absorbing sets. ■

We next show that $(5, b)$ absorbing sets do not exist for the parameter p large enough. In particular we establish a congruential constraint involving the labels of the edges emanating from the bits in the absorbing set that cannot hold for p large enough.

Lemma 6 *For $p > 19$, the factor graph family $G_{p,4}$ does not contain any $(5, b)$ absorbing sets.*

Proof: Since each bit node in the absorbing set has at most one neighboring unsatisfied check node, it follows that $b \leq 5$. Observe that the number of bit nodes with 3 satisfied and 1 unsatisfied check nodes is even, and thus b is even. First $b > 0$ by the minimum distance, $d_{min} \geq 8$ of the code, [22]. If $b = 2$, since we have at most five edges going to unsatisfied checks, there are two cases: (a) either three of them go to one unsatisfied check and one to another, or (b) one edge goes to each unsatisfied check. In case (a), because the girth of the factor graph is bigger than 4 [5], none of the three bit nodes that share an unsatisfied check can share a satisfied check. Further, no two bit nodes can share a satisfied check for the same reason. By counting, this eliminates case (a). In case (b), if we drop one of the bit nodes that has an unsatisfied check we would have a $(4, 4)$ absorbing set which we have argued in Lemma 5 does not exist for $p > 7$.

Thus for $p > 7$ we are left with considering the case $b = 4$ since at most five edges go into unsatisfied checks. This means the candidate absorbing set contains 1 bit node with all checks satisfied and 4 bit nodes each with 3 satisfied and 1 unsatisfied check. The only way that such an absorbing set could exist is if one has the configuration shown in Figure 8, where the vertices represent bit nodes and edges represent their satisfied check nodes.

Since i_1, i_2, i_3 and i_4 are all distinct elements of the set $\{0, 1, 2, 3\}$, by the bit consistency condition, and by the symmetry of the candidate configuration in Figure 8, we may assume that $i_1 = 0$. We let $x := i_2$, $y := i_3$ and $z := i_4$, where $x, y, z \in \{1, 2, 3\}$ and distinct. By propagating possible values of the labels for remaining edges, while maintaining vertex consistency conditions, it follows that $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8)$ is either $(0, x, y, z, y, z, 0, x)$ or $(0, x, y, z, z, x, y, 0)$.

For $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8)$ = $(0, x, y, z, y, z, 0, x)$, and for each edge and its endpoints in Figure 8, we write the pattern consistency constraints of Lemma 1(b), in terms of x, y and z,

$$k_1 \equiv k_2 \mod p, \tag{16a}$$
$$k_3 \equiv k_5 \mod p, \tag{16b}$$
$$k_1 + xj_1 \equiv k_3 + xj_3 \mod p, \tag{16c}$$
$$k_1 + yj_1 \equiv k_4 + yj_4 \mod p, \tag{16d}$$
$$k_1 + zj_1 \equiv k_5 + zj_5 \mod p, \tag{16e}$$
$$k_2 + yj_2 \equiv k_3 + yj_3 \mod p, \tag{16f}$$
$$k_2 + zj_2 \equiv k_4 + zj_4 \mod p, \text{ and} \tag{16g}$$
$$k_4 + xj_4 \equiv k_5 + xj_5 \mod p. \tag{16h}$$

This last system simplifies to

$$\begin{array}{llllll} k_1 + xj_1 & \equiv & k_3 + xj_3 & \mod p, & & \text{(from (16c))} \\ k_1 + yj_1 & \equiv & k_4 + yj_4 & \mod p, & & \text{(from (16d))} \\ k_1 + zj_1 & \equiv & k_3 + zj_5 & \mod p, & & \text{(from (16b) and (16e))} \\ k_1 + yj_2 & \equiv & k_3 + yj_3 & \mod p, & & \text{(from (16a) and (16f))} \\ k_1 + zj_2 & \equiv & k_4 + zj_4 & \mod p, & \text{and} & \text{(from (16a) and (16g))} \\ k_4 + xj_4 & \equiv & k_3 + xj_5 & \mod p. & & \text{(from (16b) and (16h))} \end{array} \tag{17}$$

Thus

$$k_1 - k_3 \equiv x(j_3 - j_1) \equiv z(j_5 - j_1) \equiv y(j_3 - j_2) \mod p \tag{18a}$$
$$k_1 - k_4 \equiv y(j_4 - j_1) \equiv z(j_4 - j_2) \mod p \tag{18b}$$
$$k_3 - k_4 \equiv x(j_4 - j_5) \mod p. \tag{18c}$$

We let $u_1 := j_3 - j_1$, $u_2 := j_4 - j_1$, $u_3 := j_5 - j_1$, and $u_4 := j_3 - j_2$. Note that by the check consistency condition, all of u_1, u_2, u_3, and u_4 are non-zero.

We then obtain

$$\begin{array}{rclll} xu_1 & \equiv & zu_3 & \mod p, & \text{(from (18a))} \\ xu_1 & \equiv & yu_4 & \mod p, & \text{(from (18a))} \\ yu_2 & \equiv & z(u_2 - u_1 + u_4) & \mod p, & \text{(from (18b))} \\ x(u_2 - u_3) & \equiv & yu_2 - xu_1 & \mod p, & \text{from } k_3 - k_4 = (k_1 - k_4) - (k_1 - k_3) \text{ and} \\ & & & & \text{substituting from (18c), (18b), and (18a)), resp.} \end{array} \qquad (19)$$

This last system can be rewritten as

$$\begin{bmatrix} x & 0 & 0 & -y \\ x & 0 & -z & 0 \\ -z & z-y & 0 & z \\ -x & y-x & x & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \equiv \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \mod p\,. \qquad (20)$$

Therefore, the determinant of the matrix multiplying the (non-zero) vector $[u_1\ u_2\ u_3\ u_4]^T$ in (20) is itself zero, which simplifies to

$$xy(z-x)(z-y) - z^2(x-y)^2 \equiv 0 \mod p, \qquad (21)$$

Since $x, y, z \in \{1, 2, 3\}$ and distinct, we consider all $3! = 6$ assignments for (x, y, z), and for each we evaluate the left hand side expression in (21). Note that for distinct $x, y, z \in \{1, 2, 3\}$, this expression is at most 19 in absolute value, and therefore the constraint in (21) does not have a solution for $p > 19$ for distinct $x, y, z \in \{1, 2, 3\}$. (Solutions exist for $p = 5, 11$ and 19, which can be verified by direct numerical substitution).

For $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8)$ = $(0, x, y, z, z, z, y, 0)$ we likewise establish the constraints as in (16) and (17). We again let $u_1 := j_3 - j_1$, $u_2 := j_4 - j_1$, $u_3 := j_5 - j_1$, and $u_4 := j_3 - j_2$, and obtain

$$\begin{bmatrix} 0 & y & -z & 0 \\ x & y-x & 0 & -x \\ x & 0 & 0 & -z \\ y-x & y & -y & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \equiv \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \mod p\,. \qquad (22)$$

Since the entries in $[u_1\ u_2\ u_3\ u_4]^T$ are all non-zero, it follows that the determinant of the matrix in (22) is zero. Simplifying the expression for the determinant yields again the condition in (21). Therefore for $p > 19$, $(5, 4)$ absorbing sets do not exist. ■

We can now proceed with the analysis of $(6, b)$ absorbing sets. Since the number of bit nodes with 3

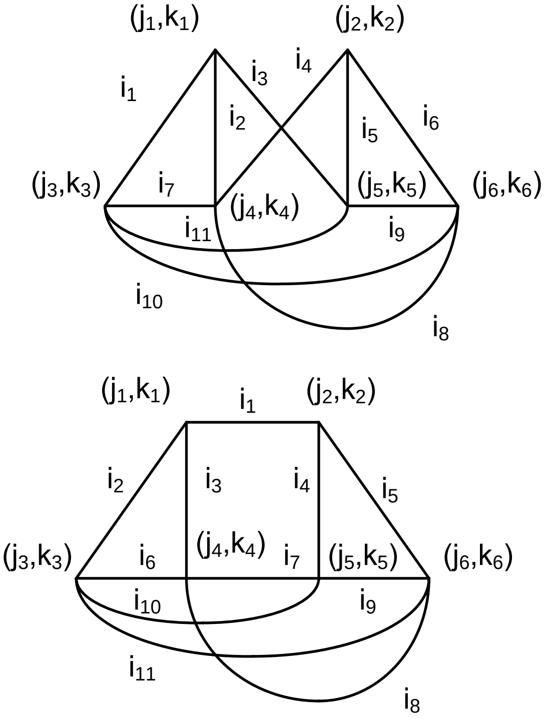

Figure 9: Depiction of the candidate $(6, 2)$ absorbing sets.

satisfied and 1 unsatisfied check node is even, b is even. First, $b = 0$ is not possible since $d_{min} \geq 8$ [22]. The following lemma addresses the case of $b = 2$.

Lemma 7 *For $p > 19$, the factor graph family $G_{p,4}$ does not contain any $(6, 2)$ absorbing sets.*

Proof: We first claim that there is no check node of degree at least 3 with respect to the bit nodes in the absorbing set. Let us first suppose that there exists one such check node and that it has an even degree with respect to the bit nodes in the absorbing set. Since we are considering an absorbing set with 6 bit nodes, such a check node would have degree either 4 or 6 with respect to the bit nodes in the absorbing set. If this satisfied check is of degree 6, there would exist 2 bit nodes in the absorbing set which would share an additional satisfied check. This situation would imply the existence of a cycle of length 4, which is impossible by the girth condition [5].

Suppose now that this satisfied check has degree 4. Each bit node that participates in this check has at least 2 more neighboring satisfied checks, which it then necessarily must share with the remaining two bit nodes in the absorbing set that themselves do not participate in this degree-4 check by the girth condition [5]. If there exists a bit node that participates in this degree-4 check and has all checks satisfied, it then shares its remaining neighboring check with one of the bit nodes with which it already shares a check. This situation violates the girth constraint [5]. If all bit nodes in the absorbing set that participate in this degree-4 check have 3 satisfied and 1 unsatisfied check, three of them would have to participate in the same unsatisfied check to make the total number of unsatisfied checks be 2. This again violates the girth condition [5].

Therefore, all satisfied checks with respect to the bit nodes in the absorbing set have degree 2. Suppose there exists a check node that is unsatisfied with respect to the bits in the absorbing set and that has degree bigger than 1. If such a check node has degree 5, there would necessarily exist 2 bit nodes in the absorbing set that share this degree-5 check and another satisfied check, which is impossible by the girth condition [5].

Suppose that there exist two degree-3 checks incident to the bit nodes in the absorbing set. First, these degree-3 checks do not have any neighboring bit nodes in common since we require that each bit node has at most 1 unsatisfied check. We can then group the bit nodes in the absorbing set into two disjoint groups, each of size 3, such that the bits in the same group share the same degree-3 check. Consider a bit node in, say, the first group. It shares its remaining 3 (satisfied) checks with each one of the bit nodes in the second group. The same is true with the other two bit nodes in the first group, namely they too share their remaining 3 (satisfied) checks with the bit nodes in the second group, and these satisfied checks connect to two bit nodes in the absorbing set. Therefore, there exist two bit nodes in the first group and two bit nodes in the second group such that any two share a distinct check. This configuration is not possible by Lemma 5 for $p > 7$.

Suppose now that there exists one unsatisfied check of degree 3 with respect to the bit nodes in the absorbing set. The remaining unsatisfied check then has degree 1 with respect to the bit nodes in the absorbing set, and the neighboring bit nodes in the absorbing set of these two unsatisfied checks are different. There are two bit nodes in the absorbing set that have all checks satisfied. Partition the bit nodes in the absorbing set into three groups: the first group contains the three bit nodes that share a degree-3 unsatisfied check, the second group contains the one bit node that has one unsatisfied check, and the third group contains the two bit nodes that have all four checks satisfied. Each of the three bit nodes in the first group has one unsatisfied and three satisfied checks and thus it shares a satisfied check with each of the bit nodes in the second and third group since it cannot share a satisfied check with another bit node in the first group by the girth condition [5]. The bit node in the second group also has one unsatisfied and three satisfied checks, and the latter are shared then with the bit nodes in the first group. The two bit nodes in the third group have all four checks satisfied, the three of which they each share with each of the bit nodes in the first group. Since all three satisfied checks of the bit node in the second group are used up with the checks it shares with the bit nodes in the first group, the two bit nodes in the third group share a satisfied check with each other. Therefore, there exist two bit nodes in the first group and two bit nodes in the third group such that any two share a distinct check. This configuration is not possible by Lemma 5 for $p > 7$.

We conclude that no check incident to the bit nodes in the absorbing set has degree larger than 2, namely that all neighboring satisfied (respectively unsatisfied) checks have degree 2 (respectively 1). By requiring that each vertex corresponding to a bit node in the absorbing set has either 3 or 4 outgoing edges, and that there are no parallel edges, it follows that there are 2 possible configurations, as shown in Figure 9, that relate bit nodes in the absorbing set (vertices) and their shared satisfied checks (edges).

Observe that the bottom configuration in Figure 9 contains a $(4,4)$ absorbing set which consists of (j_3,k_3), (j_4,k_4), (j_5,k_5), and (j_6,k_6). By Lemma 5 such configuration is not possible for $p > 7$. The analysis of the topmost configuration is considerably more involved and its technical details are deferred to Appendix 5.1, in which we derive a congruency constraint that cannot hold for prime $p > 19$ under all possible configuration labellings. With that result, the proof of Lemma 7 is complete. ■

Having eliminated smaller candidate absorbing sets, we now prove the following result.

Lemma 8 *For all $p > 5$, the factor graph family $G_{p,4}$ has $(6,4)$ (fully) absorbing sets.*

Proof: We will first show that all satisfied checks neighboring bit nodes in one such absorbing set must have degree 2. Note that there cannot be a degree-6 check with respect to the bits in the absorbing set as then some of these bits would have to share another satisfied check which is not possible by the girth condition [5]. Suppose that there exists a check node of degree 4 with respect to a $(6,4)$ absorbing set. Let t_1, t_2, t_3, t_4 be the bit nodes in the absorbing set participating in this degree-4 check node, and let t_5 and t_6 be the remaining two bit nodes in the absorbing set. By the girth condition there can be at most one degree-4 check incident to the bit nodes in the absorbing set. If at least one of t_1, t_2, t_3, t_4 had all check nodes satisfied, it would be necessary that such a bit node shares another distinct check node with some other bit node participating in the degree-4 check node, which is impossible by the girth constraint [5]. Thus, all of t_1, t_2, t_3, t_4 are each connected to 3 satisfied and 1 unsatisfied check node, and all unsatisfied checks are distinct. Then t_5 and t_6 are each connected to 4 satisfied check nodes each of degree 2 with respect to the bit nodes in the absorbing set. Since t_1 through t_4 have 3 satisfied neighboring checks (one of which is a degree-4 check by assumption), they each share a check with t_5 and with t_6. Therefore, t_5 and t_6 do not share a check. Let i_j for $1 \leq j \leq 4$ be the labels of the four check nodes connecting t_j and t_5. By the bit consistency condition at t_5, they are all different. By the bit consistency condition at each of t_j for $1 \leq j \leq 4$, the label of their shared degree-4 check node must be different from all i_j for $1 \leq j \leq 4$, which is impossible as there are only 4 distinct labels available. Therefore, all satisfied check nodes neighboring bit nodes in the absorbing set have degree 2.

We first consider the case where there exists an unsatisfied check of degree 3 with respect to the bit nodes in the absorbing set (an unsatisfied check of degree larger than 3 is not possible by the girth condition). Consider a candidate $(6,4)$ absorbing set in which three bit nodes, call them t_1, t_2, t_3 connect to the same unsatisfied check, and the remaining three bit nodes, call them t_4, t_5, t_6, each have a distinct unsatisfied check. Since there are no cycles of length 4, each of the t_1, t_2, t_3 shares a distinct satisfied check with each of t_4, t_5, t_6. Appendix 5.2 contains the proof that in fact for prime p, where $p > 13$, such a configuration is not possible.

We now continue with the analysis of the candidate configurations in which each satisfied check has degree 2 with respect to the bit nodes in the absorbing set, and each unsatisfied check has degree 1 with respect to the bits in the absorbing set. By separately considering the cases when the two bit nodes that have all neighboring checks satisfied also have a satisfied check in common, and the cases when they do not, one can show that there are 3 possible non isomorphic configurations, as shown in Figure 10, 11, and 12. By ensuring the bit consistency, it further follows that for each configuration there are 8 distinct edge labellings (as we show below). Let us consider the configuration in Figure 10 first. The other two configurations are analyzed subsequently.

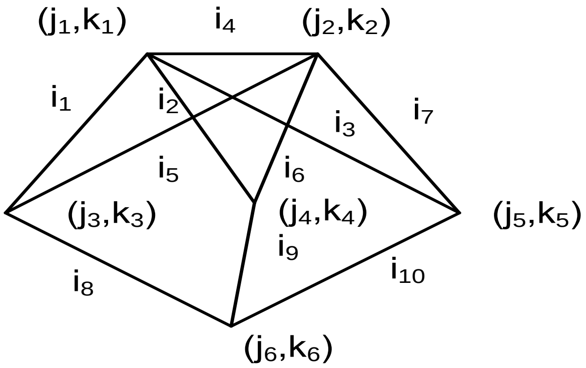

Figure 10: Depiction of the first candidate $(6,4)$ set.

(a) First candidate (6,4) configuration, given in Figure 10.

We first determine all possible edge labellings. For convenience, we assign $(i_1,i_2,i_3,i_4) := (x,y,z,w)$, where $x,y,z,w \in \{0,1,2,3\}$ and distinct by the bit consistency condition at (j_1,k_1). Then, by imposing the bit consistency conditions at remaining vertices, the possible assignments for the remaining edge labels are as follows,

$$(i_5,i_6,i_7,i_8,i_9,i_{10}) \in \{(y,z,x,z,x,y),(z,x,y,y,z,x),(y,z,x,z,w,y),(y,z,x,w,x,y),\\ (y,z,x,z,x,w),(z,x,y,y,z,w)(z,x,y,y,w,x),(z,x,y,w,z,x)\}. \tag{23}$$

We first observe that the assignments $(i_1,i_2,i_3,i_4,i_5,i_6,i_7,i_8,i_9,i_{10})$ =(x,y,z,w,y,z,x,z,x,y) and $(i_1,i_2,i_3,i_4,i_5,i_6,i_7,i_8,i_9,i_{10})$ =(x,y,z,w,z,x,y,y,z,x) are in fact symmetric (exchange y and z) and is thus sufficient to analyze only one of them. Likewise, by appealing to symmetry and after appropriate

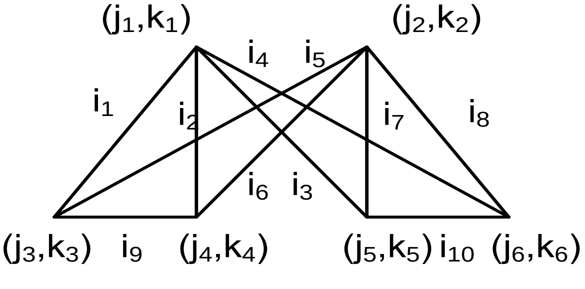

Figure 11: Depiction of the second candidate $(6,4)$ set.

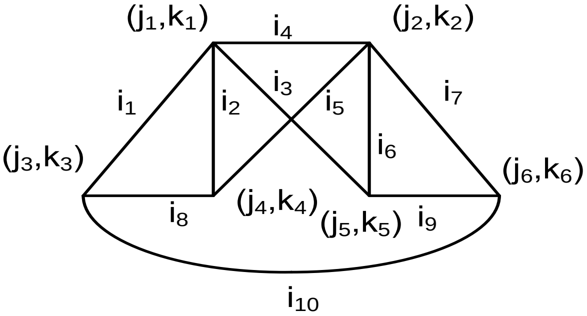

Figure 12: Depiction of the third candidate $(6,4)$ set.

renamings, the remaining six assignments also represent the same labelled configuration. In particular, third and sixth assignments in (23) are symmetric, as are fourth and seventh, and as are fifth and eighth assignments. Fourth assignment follows from the third by exchanging the labels x and z, and the fifth assignment follows from the third by exchanging the labels x and y. It is thus sufficient to consider only $(i_1,i_2,i_3,i_4,i_5,i_6,i_7,i_8,i_9,i_{10})$= (x,y,z,w,y,z,x,z,x,y) or (x,y,z,w,y,z,x,z,w,y).

I. Consider the labelling $(i_1,i_2,i_3,i_4,i_5,i_6,i_7,i_8,i_9,i_{10})$= (x,y,z,w,y,z,x,z,x,y).

By applying the pattern consistency for each edge and its end points in Figure 10 we obtain

$$\begin{aligned}
k_1+xj_1 &\equiv k_3+xj_3 \mod p,\\
k_1+yj_1 &\equiv k_4+yj_4 \mod p,\\
k_1+zj_1 &\equiv k_5+zj_5 \mod p,\\
k_1+wj_1 &\equiv k_2+wj_2 \mod p,\\
k_2+yj_2 &\equiv k_3+yj_3 \mod p,\\
k_2+zj_2 &\equiv k_4+zj_4 \mod p,\\
k_2+xj_2 &\equiv k_5+xj_5 \mod p,\\
k_3+zj_3 &\equiv k_6+zj_6 \mod p,\\
k_4+xj_4 &\equiv k_6+xj_6 \mod p, \quad \text{and}\\
k_5+yj_5 &\equiv k_6+yj_6 \mod p.
\end{aligned} \tag{24}$$

Using the cycle consistency conditions for each of five cycles that span the cycle space of the graph in Figure 10 we also write

$$\begin{aligned}
w(j_2-j_1)+y(j_3-j_2)+x(j_1-j_3) &\equiv 0 \mod p,\\
w(j_2-j_1)+z(j_4-j_2)+y(j_1-j_4) &\equiv 0 \mod p,\\
w(j_2-j_1)+x(j_5-j_2)+z(j_1-j_5) &\equiv 0 \mod p,\\
y(j_4-j_1)+x(j_6-j_4)+z(j_3-j_6)+x(j_1-j_3) &\equiv 0 \mod p, \quad \text{and}\\
x(j_5-j_2)+y(j_6-j_5)+x(j_4-j_6)+z(j_2-j_4) &\equiv 0 \mod p.
\end{aligned} \tag{25}$$

We will use the relationships in (25) to express j_3 through j_6 in terms of j_1 and $(j_2 - j_1)$, and then in turn use (24) to express k_2 through k_6 in terms of k_1, j_1 and $(j_2 - j_1)$.

By symmetry of the configuration (see Figure 10), for the current labelling it is sufficient to consider $x = 0$ and $w = 0$. Specifically, letting $y = 0$ or $z = 0$ reduces to the $x = 0$ case.

We let $a := j_2 - j_1$, $b := j_3 - j_1$, $c := j_4 - j_1$, $d := j_5 - j_1$, and $e := j_6 - j_1$. Note that in particular by the check consistency constraint, $a \neq 0$.

1. Case $x = 0$

The system in (25) reduces to

$$\begin{aligned} a(w-y) + by &\equiv 0 \mod p, \\ a(z-w) + c(y-z) &\equiv 0 \mod p, \\ aw - dz &\equiv 0 \mod p, \\ bz + yc - ze &\equiv 0 \mod p, \quad \text{and} \\ az - cz - dy + ey &\equiv 0 \mod p\ . \end{aligned} \tag{26}$$

Using (26) we express b, c , d and e in terms of a. In particular, the last constraint in (26) is redundant as it follows from the previous four, as we now show. Express b, c and d of a using top three equations in (26) so that

$$b \equiv a\frac{y-w}{y} \mod p, \quad c \equiv a\frac{w-z}{y-z} \mod p \quad \text{and} \quad d \equiv a\frac{w}{z} \mod p\ . \tag{27}$$

Substitute for b, c, d in terms of a in the fourth equation of the system (26) to obtain

$$e \equiv a\left(\frac{y-w}{y} + \frac{w-z}{y-z}\frac{y}{z}\right) \mod p\ . \tag{28}$$

Likewise, substitute for b, c, d in terms of a in the fifth equation of the system (26) to obtain

$$a\left(z - z\frac{w-z}{y-z} - \frac{wy}{z}\right) + ey \equiv 0 \mod p\ . \tag{29}$$

From (28) it follows that

$$(y-z)yze \equiv a\left(z(y-w)(y-z) + y^2(w-z)\right) \mod p, \tag{30}$$

and from (29) it follows that

$$a\left(z^2(y-z) - z^2(w-z) - wy(y-z)\right) + (y-z)yze \equiv 0 \mod p\ . \tag{31}$$

Rewrite (31) as

$$(y-z)yze \equiv a\left(-z^2(y-z)+z^2(w-z)+wy(y-z)\right) \mod p\ . \tag{32}$$

We expand the terms that multiply a in both (30) and (32). They both reduce to $(-wyz-yz^2+wz^2+y^2w)$, which makes the last equation in the system (26) redundant.

Therefore, for $q := j_1$ and $t := j_2 - j_1$, all of the remaining values of j_3, j_4, j_5, j_6 follow for each of the $3! = 6$ choices of (y, z, w).

From (24) we have that $k_1 = k_3$, $k_2 = k_5$, $k_4 = k_6$ as well as $k_4 \equiv k_1 - y(j_4 - j_1) \mod p$, and $k_5 \equiv k_1 - z(j_5 - j_1) \mod p$. We can thus express k_2 through k_6 in terms of $s := k_1$, q and t. The results for all choices of (y, z, w) are summarized in Table 1, where the indices are taken $\mod p$.

y,z,w	j_1	j_2	j_3	j_4	j_5	j_6	k_1	k_2	k_3	k_4	k_5	k_6
$3,2,1$	q	$q+t$	$q+2t/3$	$q-t$	$q+t/2$	$q-5t/6$	s	$s-t$	s	$s+3t$	$s-t$	$s+3t$
$3,1,2$	q	$q+t$	$q+t/3$	$q+t/2$	$q+2t$	$q+11t/6$	s	$s-2t$	s	$s-3t/2$	$s-2t$	$s-3t/2$
$2,3,1$	q	$q+t$	$q+t/2$	$q+2t$	$q+t/3$	$q+11t/6$	s	$s-t$	s	$s-4t$	$s-t$	$s-4t$
$2,1,3$	q	$q+t$	$q-t/2$	$q+2t$	$q+3t$	$q+7t/2$	s	$s-3t$	s	$s-4t$	$s-3t$	$s-8t$
$1,2,3$	q	$q+t$	$q-2t$	$q-t$	$q+3t/2$	$q-5t/2$	s	$s-3t$	s	$s+t$	$s-3t$	$s+t$
$1,3,2$	q	$q+t$	$q-t$	$q+t/2$	$q+2t/3$	$q-5t/6$	s	$s-2t$	s	$s-t/2$	$s-2t$	$s-t/2$

Table 1: Several solutions for a $(6,4)$ fully absorbing set.

Furthermore, under the current configuration, the bit nodes in one such $(6,4)$ absorbing set that have 3 satisfied and 1 unsatisfied check, all have unsatisfied checks in the row group labelled w. By the bit consistency condition, no bit node can connect to more than one such check. Therefore, this configuration is in fact a $(6,4)$ fully absorbing set. In particular, the solution set in row 1 holds for all $p > 5$ and t a multiple of 6.

We complete the analysis of this label assignment by considering $w = 0$.

2. Case $w = 0$

In this case the system in (25) reduces to:

$$\begin{aligned}
ay + b(x-y) &\equiv 0 \mod p, \\
az + c(y-z) &\equiv 0 \mod p, \\
ax + d(z-x) &\equiv 0 \mod p, \\
b(z-x) + c(y-x) + e(x-z) &\equiv 0 \mod p, \quad \text{and} \\
a(z-x) + c(x-z) + d(x-y) + e(y-x) &\equiv 0 \mod p\ .
\end{aligned} \tag{33}$$

Note that the last relation follows from the previous four. We again express b, c, d and e in terms of a, so that by setting $j_1 := q$ and $a := t$, all of j_2 through j_6 follow as a function of q and t. Then, by letting $k_1 := s$, the remaining k_2 through k_6 follow from q, t and s from (25). The solution set for various numerical assignments of (x, y, z) is given in Table 2, where the indices are taken $\mod p$.

x, y, z	j_1	j_2	j_3	j_4	j_5	j_6	k_1	k_2	k_3	k_4	k_5	k_6
$3, 2, 1$	q	$q+t$	$q-2t$	$q-t$	$q+3t/2$	$q-5t/2$	s	s	$s+6t$	$s+2t$	$s-3t/2$	$s+13t/2$
$3, 1, 2$	q	$q+t$	$q-t/2$	$q+2t$	$q+3t$	$q+7t/2$	s	s	$s+3t/2$	$s-2t$	$s-6t$	$s-13t/2$
$2, 3, 1$	q	$q+t$	$q+3t$	$q-t/2$	$q+2t$	$q+7t/2$	s	s	$s-6t$	$s+3t/2$	$s-2t$	$s-13t/2$
$2, 1, 3$	q	$q+t$	$q-t$	$q+3t/2$	$q-2t$	$q-5t/2$	s	s	$s+2t$	$s-3t/2$	$s+6t$	$s+13t/2$
$1, 2, 3$	q	$q+t$	$q+2t$	$q+3t$	$q-t/2$	$q+7t/2$	s	s	$s-2t$	$s-6t$	$s+3t/2$	$s-13t/2$
$1, 3, 2$	q	$q+t$	$q+3t/2$	$q-2t$	$q-t$	$q-5t/2$	s	s	$s-3t/2$	$s+6t$	$s+2t$	$s+13t/2$

Table 2: Several solutions for a $(6, 4)$ fully absorbing set.

As in the $x = 0$ case, the unsatisfied checks all belong in the row group labelled w. By the bit consistency condition, no bit node can connect to more than one such check. Therefore, this configuration is also in fact a $(6, 4)$ fully absorbing set. In particular, the solution set in row 1 of Table 2 holds for all $p > 5$ and t even.

We now consider the remaining labelled configuration of Figure 10.

II. Consider the labelling $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10})$ = $(x, y, z, w, y, z, x, z, w, y)$.

We again let $a := j_2 - j_1$, $b := j_3 - j_1$, $c := j_4 - j_1$, $d := j_5 - j_1$, and $e := j_6 - j_1$. Note that in particular by the check consistency constraint, $a \neq 0$.

Based on the cycle consistency condition for the five cycles in Figure 10 we establish

$$\begin{aligned}
a(w-y) + b(y-x) &\equiv 0 \quad \mod p, \\
a(w-z) + c(z-y) &\equiv 0 \quad \mod p, \\
a(w-x) + d(x-z) &\equiv 0 \quad \mod p, \\
c(y-w) + b(z-x) + e(w-z) &\equiv 0 \quad \mod p, \quad \text{and} \\
d(x-y) + a(z-x) + c(w-z) + e(y-w) &\equiv 0 \quad \mod p\,.
\end{aligned} \tag{34}$$

By expressing b, c and d in terms of a, from this system we obtain

$$a\left(\frac{(y-w)(w-z)}{y-z} + \frac{(z-x)(w-y)}{x-y}\right) + e(w-z) \equiv 0 \quad \mod p \tag{35}$$

$$a\left(\frac{(x-y)(w-x)}{z-x} + (z-x) + \frac{(w-z)^2}{y-z}\right) + e(y-w) \equiv 0 \quad \mod p, \tag{36}$$

where $\{x, y, z, w\} = \{0, 1, 2, 3\}$ and are distinct. For all $4! = 24$ distinct ways of assigning numerical values to x, y, z and w, the system (35)–(36) produces the unique solution $a = 0$, $e = 0$, provided that $p > 3$. Since $a \neq 0$ by the edge consistency condition, we conclude that this configuration is not possible.

We now analyze possible solutions for the next candidate $(6, 4)$ configuration, for which we show that there exist $(6, 4)$ absorbing sets which are not fully absorbing sets.

(b) Second candidate (6,4) configuration, given in Figure 11.

We first determine all possible edge labellings. For convenience, let $(i_1, i_2, i_3, i_4) := (x, y, z, w)$, where $x, y, z, w \in \{0, 1, 2, 3\}$ and are distinct by the bit consistency condition at (j_1, k_1). Then, by imposing the bit consistency conditions at remaining vertices, the assignments for the remaining edge labels are given by the following set

$$\begin{aligned}(i_5, i_6, i_7, i_8, i_9, i_{10}) \in \{&(y, x, w, z, z, x), (w, x, y, z, z, x), (y, x, w, z, z, y), (y, w, x, z, z, y), \\ &(y, x, w, z, w, x), (z, x, w, y, w, x), (y, z, w, x, w, y), (y, x, w, z, w, y)\}\ .\end{aligned} \tag{37}$$

Out of these 8 possible labelled configurations by appealing to symmetry and label renaming it is sufficient to consider only 2 of these as we now show. Note that the eighth labelling is the same as the first labelling after we exchange (j_3, k_3) and (j_4, k_4), (j_5, k_5) and (j_6, k_6), and labels y with x and w with z. Likewise, the second labelling is the same as the seventh labelling after we exchange (j_3, k_3) and (j_4, k_4), (j_5, k_5) and (j_6, k_6), and labels y with x and w with z. The sixth labelling is the same as the fourth labelling after we exchange labels z with x, y with w, and nodes (j_1, k_1) with (j_2, k_2), (j_3, k_3) with (j_4, k_4), and (j_5, k_5) with (j_6, k_6), and take the mirror image of the resulting configuration. Fifth labelling is the same as the third after we exchange labels z with x and y with w and take the mirror image of the whole configuration. Fourth (respectively first) labelling is the same as the second (respectively third) after we exchange (j_3, k_3) and (j_4, k_4) and labels x and y.

It is thus sufficient to consider only two different labellings, namely $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10})$= $(x, y, z, w, y, x, w, z, z, y)$ (third labelling) and $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10})$ = $(x, y, z, w, y, z, w, x, w, y)$ (seventh labelling). The analysis utilizes the same tools as the ones developed for the previous candidate configuration, and its technical details are deferred to Appendix 5.3. The outcome of the analysis gives the solution sets listed in Tables 3 and 4, again the entries are taken $\bmod p$, which are absorbing but not fully absorbing sets, as further argued in Appendix 5.3.

x, y, w	j_1	j_2	j_3	j_4	j_5	j_6	k_1	k_2	k_3	k_4	k_5	k_6
$1, 3, 2$	q	$q + 4t$	$q + 3t$	$q + t$	$q + t$	$q + 3t$	s	$s - 6t$	$s - 3t$	$s - 3t$	s	$s - 6t$

Table 3: A solution for a $(6, 4)$ absorbing set.

y, z, w	j_1	j_2	j_3	j_4	j_5	j_6	k_1	k_2	k_3	k_4	k_5	k_6
$2, 1, 3$	q	q	$q - t$	$q + t$	$q - t$	$q + t$	s	$s - 2t$	s	$s - 2t$	$s + t$	$s - 3t$

Table 4: A solution for a $(6, 4)$ absorbing set.

Lastly, we consider the third and final unlabelled candidate $(6, 4)$ absorbing set, for which we show that in fact does not yield $(6, 4)$ absorbing sets for the prime p large enough.

(c) Third candidate (6,4) configuration, given in Figure 12.

We first determine all possible edge labellings. As before we let $(i_1, i_2, i_3, i_4) := (x, y, z, w)$, where $x, y, z, w \in \{0, 1, 2, 3\}$ and distinct by the bit consistency condition at (j_1, k_1). Then, by propagating bit consistency conditions for remaining vertices, the assignments for the remaining edge labels are given by the following set

$$\begin{aligned}(i_5, i_6, i_7, i_8, i_9, i_{10}) \quad &\in \{(x, y, z, z, x, y), (x, y, z, z, x, w), (x, y, z, z, w, y), (x, y, z, w, x, y), \\ &(x, y, z, w, w, y), (z, x, y, w, w, z), (z, y, x, w, w, y), (z, y, x, w, w, z)\} .\end{aligned}$$

By exploiting the symmetry, one can show that after renaming the labelling $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10})$= $(x, y, z, w, x, y, z, z, w, y)$ and $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10})$= $(x, y, z, w, x, y, z, w, x, y)$ reduce to the same case (by exchanging z and x). We are thus left with analyzing the remaining seven cases. As before, we let $a := j_2 - j_1$, $b := j_3 - j_1$, $c := j_4 - j_1$, $d := j_5 - j_1$, and $e := j_6 - j_1$. Note that in particular by the check consistency constraint, $a \neq 0$.

Consider the labelling $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10})$ = $(x, y, z, w, x, y, z, z, x, y)$. We apply the cycle consistency conditions to five cycles spanning the cycle space of the graph in Figure 12 and obtain:

$$\begin{aligned} xb + z(c-b) - yc &\equiv 0 \mod p, \\ yc + x(a-c) - wa &\equiv 0 \mod p, \\ -wa + zd + y(a-d) &\equiv 0 \mod p, \\ y(d-a) + x(e-d) + z(a-e) &\equiv 0 \mod p, \quad \text{and} \\ xb + y(e-b) + x(d-e) - zd &\equiv 0 \mod p . \end{aligned} \tag{38}$$

By expressing b, c and d in terms of a, and substituting in the bottom two constraints of (38) we obtain

$$a\left(z - y + \frac{(y-w)(y-x)}{y-z}\right) + e(x-z) \equiv 0 \mod p, \quad \text{and} \tag{39}$$

$$a\left(\frac{(y-z)(x-w)}{x-z} + \frac{(y-w)(x-z)}{y-z}\right) + e(y-x) \equiv 0 \mod p, \tag{40}$$

where $\{x, y, z, w\} = \{0, 1, 2, 3\}$ and are distinct. For all $4! = 24$ distinct ways of assigning numerical values to x, y, z and w, the system (39) – (40) produces the unique solution $a = 0$, $e = 0$, provided that $p > 3$. Since $a \neq 0$ by the check consistency condition, we conclude that this configuration is not possible.

One can likewise establish the constraints of the (38) type for the remaining six cases, from which the two equations (as in (39) and (40)) relating a and e will follow. In all five cases, the unique solution for p large enough is $(a, e) = (0, 0)$. In particular, $p > 13$ is sufficient for all cases considered.

Having exhaustively considered all possible configurations of a $(6, 4)$ absorbing sets, the proof of the lemma is complete. ■

Using these results the proof of Theorem 1(c) now follows. We complete our analysis of $\gamma = 4$ by proving the claim in Theorem 2: The number of $(6, 4)$ (fully) absorbing sets scales as $\Theta(n^{3/2})$, where n is the codeword length.

Proof: Recall that for the configuration in Figure 10 we identified two sets of labellings given in Tables 1 and 2 that determine $(6, 4)$ fully absorbing sets. For each such assignment there are three parameters that determine all of j's and k's, and each parameter is chosen independently in at most p ways (to ensure the all j's and k's have integer values), yielding an upper bound which grows as $\Theta(p^3)$. A lower bound on the cardinality of the $(6, 4)$ fully absorbing sets is given by one solution set in Table 1, which also grows as $\Theta(p^3)$. Note that the number of solutions of absorbing sets in Table 3 and Table 4 grows as $\Theta(p^3)$ as well [2]. Since $n = p^2$, the result follows. ■

We have thus proven Theorem 2 for $\gamma = 4$.

4 Conclusion

Absorbing sets are a substructure of the factor graphs defining LDPC codes that cause error floors in iterative decoding. The main contribution of this paper was to develop algebraic techniques for analyzing and enumerating minimal fully absorbing sets for the class of array-based LDPC codes.We provided an explicit description of these minimal (fully) absorbing sets and showed the non-existence of certain candidate configurations. We also enumerated minimal (fully) absorbing sets and showed how their number scales with the codeword length. In concurrent work [3], we have used these theoretical results to develop a fast simulation method, based on importance sampling, for computing estimates of the error floor of LDPC codes. Although the current paper has focused on a particular subclass of LDPC codes, we suspect that the techniques and analysis performed in the current work can be fruitfully extended to a larger class of structured LDPC codes.

5 Appendix

5.1 Non-existence of $(6, 2)$ absorbing sets

By ensuring the bit consistency, it follows that the topmost configuration in Figure 9 has 2 distinct edge labellings. In particular, by the vertex consistency at (j_3, k_3) we may let $x := i_1$, $y := i_7$, $z := i_{11}$ and $w := i_{10}$, where $x, y, z, w \in \{0, 1, 2, 3\}$ and distinct. By propagating the labels while making sure that the bit consistency constraints are satisfied we conclude that either

- $x = i_1 = i_5 = i_8$, $y = i_7 = i_9$, $z = i_2 = i_6 = i_{11}$, $w = i_3 = i_4 = i_{10}$ or

[2] For $p = 37$, Remark 1 and Remark 2 in Appendix 5.3 also show that the number of additional solution sets also scales as 37^3.

- $x = i_1 = i_4 = i_9$, $y = i_3 = i_6 = i_7$, $z = i_8 = i_{11}$, $w = i_2 = i_5 = i_{10}$

where throughout x, y, z, w are distinct and belong to the set $\{0, 1, 2, 3\}$.

I. Consider the labelling $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10}, i_{11})$ = $(x, z, w, w, x, z, y, x, y, w, z)$.

Using the pattern consistency constraint (see Lemma 1(b)) for each edge in Figure 9 for the current labelling we obtain

$$k_1 + xj_1 \equiv k_3 + xj_3 \mod p, \tag{41a}$$

$$k_1 + zj_1 \equiv k_4 + zj_4 \mod p, \tag{41b}$$

$$k_1 + wj_1 \equiv k_5 + wj_5 \mod p, \tag{41c}$$

$$k_2 + wj_2 \equiv k_4 + wj_4 \mod p, \tag{41d}$$

$$k_2 + xj_2 \equiv k_5 + xj_5 \mod p, \tag{41e}$$

$$k_2 + zj_2 \equiv k_6 + zj_6 \mod p, \tag{41f}$$

$$k_3 + yj_3 \equiv k_4 + yj_4 \mod p, \tag{41g}$$

$$k_4 + xj_4 \equiv k_6 + xj_6 \mod p, \tag{41h}$$

$$k_5 + yj_5 \equiv k_6 + yj_6 \mod p, \tag{41i}$$

$$k_3 + wj_3 \equiv k_6 + wj_6 \mod p, \text{ and} \tag{41j}$$

$$k_3 + zj_3 \equiv k_5 + zj_5 \mod p\ . \tag{41k}$$

We now separately consider $x = 0$, $y = 0$, $z = 0$, and $w = 0$.

1. For $x = 0$, the set of constraints (41a)-(41k) reduces to

$$\begin{array}{lllllllll}
k_1 - k_3 & \equiv & 0 & & & & & \mod p & \text{(from (41a))} \\
k_2 - k_5 & \equiv & 0 & & & & & \mod p & \text{(from (41e))} \\
k_4 - k_6 & \equiv & 0 & & & & & \mod p & \text{(from (41h))} \\
k_1 - k_4 & \equiv & z(j_4 - j_1) & \equiv & y(j_4 - j_3) & \equiv & w(j_6 - j_3) & \mod p & \text{(from (41b),} \\
 & & & & & \text{(41a) and (41g),} & \text{and(41h),} & \text{(41a) and (41j)} & \text{respectively.)} \\
k_2 - k_4 & \equiv & w(j_4 - j_2) & \equiv & z(j_6 - j_2) & \equiv & y(j_6 - j_5) & \mod p & \text{(from (41d) ,} \\
 & & & & & \text{(41h) and (41f),} & \text{and (41e),} & \text{(41h) and (41i)} & \text{respectively.)} \\
k_1 - k_2 & \equiv & w(j_5 - j_1) & \equiv & z(j_5 - j_3) & & & \mod p\ . & \text{(from (41c) and (41e)} \\
 & & & & & & \text{and (41a),} & \text{(41e) and (41k)} & \text{respectively.)}
\end{array} \tag{42}$$

Since $j_1 \neq j_4$, $j_2 \neq j_4$ and $j_1 \neq j_5$ by the check consistency conditions, we have that $k_1 \neq k_4$, $k_2 \neq k_4$ and $k_1 \neq k_2$.

Since $\{y, z, w\} = \{1, 2, 3\}$ and $p > 19$ is prime, we may let

$$\begin{aligned} k_1 - k_4 &\equiv ywzt \mod p, \\ k_2 - k_4 &\equiv ywzu \mod p, \quad \text{and} \\ k_1 - k_2 &\equiv wzs \mod p, \end{aligned} \tag{43}$$

for some integers t, s and u which are themselves nonzero. From $k_1 - k_2 = (k_1 - k_4) - (k_2 - k_4)$, $j_5 - j_3 = -(j_6 - j_5) + (j_6 - j_3)$, and $j_5 - j_1 = -(j_6 - j_5) + (j_6 - j_2) - (j_4 - j_2) + (j_4 - j_1)$, respectively, it follows that

$$\begin{aligned} wzs \equiv yzwt - ywzu \mod p, \\ ws \equiv -wzu + yzt \mod p, \qquad \text{and} \\ zs \equiv -wzu + ywu - yzu + ywt \mod p\ . \end{aligned} \tag{44}$$

From (44), by equating the expressions for ws and zs, it follows that

$$\begin{aligned} wu(y - z) &\equiv yt(w - z) \mod p \quad \text{and} \\ wu(y - z) &\equiv yt(z - w) \mod p\ . \end{aligned} \tag{45}$$

The last set of constraints implies $w \equiv z \mod p$, which is a contradiction.

2. For $y = 0$ the set of constraints (41a)-(41k) reduces to

$$\begin{aligned} k_3 - k_4 &\equiv 0 && && && \mod p, \\ k_5 - k_6 &\equiv 0 && && && \mod p, \\ k_1 - k_3 &\equiv x(j_3 - j_1) &&\equiv z(j_4 - j_1) && && \mod p, \\ k_2 - k_5 &\equiv x(j_5 - j_2) &&\equiv z(j_6 - j_2) && && \mod p, \\ k_3 - k_5 &\equiv x(j_6 - j_4) &&\equiv w(j_6 - j_3) &&\equiv z(j_5 - j_3) && \mod p, \quad \text{and} \\ k_1 - k_5 &\equiv w(j_5 - j_1) && && && \mod p\ . \end{aligned} \tag{46}$$

Note that $j_1 \neq j_3$, $j_2 \neq j_5$, $j_4 \neq j_6$ and $j_1 \neq j_5$ by the check consistency conditions, so that $k_1 \neq k_3$, $k_2 \neq k_5$, $k_3 \neq k_5$ and $k_1 \neq k_5$. Since $\{x, z, w\} = \{1, 2, 3\}$, we may let

$$\begin{aligned} k_1 - k_3 &\equiv xzs \mod p, \\ k_1 - k_5 &\equiv wv \mod p, \\ k_2 - k_5 &\equiv xzu \mod p, \quad \text{and} \\ k_3 - k_5 &\equiv xwzt \mod p \end{aligned} \tag{47}$$

for some integers s, u, v and t, which are themselves nonzero. The identities $k_1 - k_3 = (k_1 - k_5) - (k_3 - k_5)$, $j_5 - j_1 = (j_5 - j_3) + (j_3 - j_1)$ and $j_4 - j_1 = -(j_6 - j_4) + (j_6 - j_3) + (j_3 - j_1)$ respectively, yield the following constraints,

$$\begin{aligned} xzs &\equiv wv - xwzt && \mod p, \\ v &\equiv xwt + zs && \mod p, \quad \text{and} \\ xs &\equiv -wzt + xzt + zs && \mod p\,. \end{aligned} \tag{48}$$

Eliminating v from the top two constraints in (48) implies $zs(x-w) \equiv xwt(w-z) \mod p$, which combined with the bottom constraint in (48) yields

$$z^2(x-w)^2 \equiv xw(w-z)(x-z) \quad \mod p\,. \tag{49}$$

Since $\{x, y, w\} = \{1, 2, 3\}$, this cannot hold for $p > 19$.

3. For $z = 0$ we obtain

$$\begin{aligned} k_1 - k_4 &\equiv 0 && && && \mod p, \\ k_2 - k_6 &\equiv 0 && && && \mod p, \\ k_3 - k_5 &\equiv 0 && && && \mod p, \\ k_1 - k_3 &\equiv x(j_3 - j_1) &&\equiv w(j_5 - j_1) &&\equiv y(j_3 - j_4) && \mod p, \\ k_2 - k_3 &\equiv x(j_5 - j_2) &&\equiv y(j_5 - j_6) &&\equiv w(j_3 - j_6) && \mod p, \quad \text{and} \\ k_1 - k_2 &\equiv w(j_2 - j_4) &&\equiv x(j_6 - j_4) && && \mod p\,. \end{aligned} \tag{50}$$

As before, some algebra yields $x \equiv w \mod p$, a contradiction.

4. For $w = 0$ we obtain

$$\begin{aligned} k_1 - k_5 &\equiv 0 && && && \mod p, \\ k_2 - k_4 &\equiv 0 && && && \mod p, \\ k_3 - k_6 &\equiv 0 && && && \mod p, \\ k_1 - k_3 &\equiv x(j_3 - j_1) &&\equiv y(j_6 - j_5) &&\equiv z(j_3 - j_5) && \mod p, \\ k_2 - k_3 &\equiv z(j_6 - j_2) &&\equiv y(j_3 - j_4) &&\equiv x(j_6 - j_4) && \mod p, \quad \text{and} \\ k_1 - k_2 &\equiv z(j_4 - j_1) &&\equiv x(j_2 - j_5) && && \mod p\,. \end{aligned} \tag{51}$$

After some algebra, we obtain the following condition

$$xz(z-y)(x-y) \equiv -y^2(x-z)^2 \quad \mod p, \tag{52}$$

which, because $\{x, y, z\} = \{1, 2, 3\}$ has no solution for $p > 19$.

II. For the labelling $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10}, i_{11})$ = $(x, w, y, x, w, y, y, z, x, w, z)$ we separately consider $x = 0$, $y = 0$, $z = 0$, and $w = 0$, and proceed along the lines of the previous case. For $x = 0$, resp. $y = 0$, it follows after some algebra that $y \equiv w \mod p$, resp. $x \equiv w \mod p$, a contradiction in each case. For $z = 0$, resp. $w = 0$, it follows similarly that $xw(w - y)(x - y) \equiv y^2(x - w)^2 \mod p$, resp. $xy(y - z)(x - z) \equiv -z^2(x - y)^2 \mod p$, neither of which can hold for $p > 19$.

This completes the proof of Lemma 7. ■

5.2 Non-existence of $(6, 4)$ absorbing sets with an unsatisfied check of degree 3

Recall that we are considering the case where there exists an unsatisfied check of degree 3 with respect to the bit nodes in a candidate $(6, 4)$ absorbing set. In this absorbing set bit nodes t_1, t_2, t_3 connect to the same unsatisfied check, and the remaining three bit nodes, t_4, t_5, t_6, each have a distinct unsatisfied check. Since there are no cycles of length 4, each of t_1, t_2, t_3 shares a distinct satisfied check with each of t_4, t_5, t_6.

Let the check incident to t_1, t_2, and t_3 have label x, where $x \in \{0, 1, 2, 3\}$. Using the bit consistency condition, we let y be the label of the satisfied check incident to t_1 and t_4, z be the label of the satisfied check incident to t_1 and t_5, and w be the label of the satisfied check incident to t_1 and t_6, where $y, z, w \in \{0, 1, 2, 3\}$ are distinct and are different from x.

By propagating remaining edge labels while ensuring that the bit consistency is satisfied, we obtain that the labels of the checks connecting t_2 with t_4, t_5 and t_6, respectively, are z, w and y and the labels of the checks connecting t_3 with t_4, t_5 and t_6, respectively, are w, y and z.

Let (j_l, k_l) for $1 \leq l \leq 6$ be the labels of the bit nodes t_l. Using the pattern consistency (see Lemma 1(b)) we write one equation for each pair of the bit nodes in the absorbing set that share a satisfied check as follows:

$$\begin{aligned}
k_1 + yj_1 &\equiv k_4 + yj_4 \mod p,\\
k_1 + zj_1 &\equiv k_5 + zj_5 \mod p,\\
k_1 + wj_1 &\equiv k_6 + wj_6 \mod p,\\
k_2 + zj_2 &\equiv k_4 + zj_4 \mod p,\\
k_2 + wj_2 &\equiv k_5 + wj_5 \mod p,\\
k_2 + yj_2 &\equiv k_6 + yj_6 \mod p,\\
k_3 + wj_3 &\equiv k_4 + wj_4 \mod p,\\
k_3 + yj_3 &\equiv k_5 + yj_5 \mod p, \quad \text{and}\\
k_3 + zj_3 &\equiv k_6 + zj_6 \mod p\ .
\end{aligned} \tag{53}$$

In addition we may also write

$$k_1 + xj_1 \equiv k_2 + xj_2 \equiv k_3 + xj_3 \mod p, \tag{54}$$

since the bit nodes (j_1, k_1), (j_2, k_2) and (j_3, k_3), all participate in the same (unsatisfied) check with label x.

Since $x, y, z, w \in \{0, 1, 2, 3\}$ and are distinct we now consider different numerical assignments of these labels. In particular, it is sufficient to consider $x = 0$ and $y = 0$, since by the symmetry of the configuration both $z = 0$ and $w = 0$ reduce to the $y = 0$ case.

1. Case $x = 0$

Equation (54) reduces to $k_1 = k_2 = k_3$ which combined with (53) gives

$$\begin{aligned} k_1 - k_4 &\equiv y(j_4 - j_1) \equiv z(j_4 - j_2) \equiv w(j_4 - j_3) \mod p, \\ k_1 - k_5 &\equiv z(j_5 - j_1) \equiv w(j_5 - j_2) \equiv y(j_5 - j_3) \mod p, \quad \text{and} \\ k_1 - k_6 &\equiv w(j_6 - j_1) \equiv y(j_6 - j_2) \equiv z(j_6 - j_3) \mod p \,. \end{aligned} \tag{55}$$

Since y, z, w do not have any non trivial factors and by the check consistency conditions, we may let $yzwt \equiv k_1 - k_4 \mod p$, $yzwv \equiv k_1 - k_5 \mod p$ and $yzws \equiv k_1 - k_6 \mod p$ for some non-zero integers t, v and s. Using the identity $j_5 - j_4 = (j_5 - j_1) - (j_4 - j_1) = (j_5 - j_2) - (j_4 - j_2) = (j_5 - j_3) - (j_4 - j_3)$ we obtain (using $(j_5 - j_1) \equiv ywv \mod p, (j_4 - j_1) \equiv zwt \mod p$, and so on),

$$ywv - zwt \equiv yzv - ywt \equiv zwv - yzt \mod p \,. \tag{56}$$

The last expression implies

$$y^2(w - z)^2 \equiv wz(z - y)(y - w) \mod p \,. \tag{57}$$

Likewise, expression (56) implies

$$z^2(y - w)^2 \equiv yw(z - y)(w - z) \mod p \,, \tag{58}$$

and

$$w^2(z - y)^2 \equiv zy(w - z)(y - w) \mod p \,. \tag{59}$$

Since $\{y, z, w\} = \{1, 2, 3\}$, the equations (57), (58) and (59) hold only for prime $p = 13$.

2. Case $y = 0$

In this case equation (53) implies $k_1 = k_4$, $k_3 = k_5$ and $k_2 = k_6$. Combined with (54), we further obtain

$$\begin{aligned} k_1 - k_3 &\equiv z(j_5 - j_1) \equiv w(j_3 - j_4) \equiv x(j_3 - j_1) \mod p, \\ k_1 - k_2 &\equiv w(j_6 - j_1) \equiv z(j_2 - j_4) \equiv x(j_2 - j_1) \mod p, \quad \text{and} \\ k_2 - k_3 &\equiv w(j_5 - j_2) \equiv z(j_3 - j_6) \equiv x(j_3 - j_2) \mod p \,. \end{aligned} \tag{60}$$

We let $xzwt \equiv k_1 - k_3 \mod p$, $xzwv \equiv k_1 - k_2 \mod p$, and $xzws \equiv k_2 - k_3 \mod p$, for some non-zero integers t, v and s. From $k_1 - k_3 = (k_1 - k_2) + (k_2 - k_3)$, we have

$$t \equiv v + s \mod p. \tag{61}$$

Substituting t, v and s in (60) and using the identities $j_6 - j_1 = -(j_3 - j_6) + (j_3 - j_1)$, $j_5 - j_1 = (j_5 - j_2) + (j_2 - j_1)$ and $j_3 - j_4 = (j_3 - j_2) + (j_2 - j_4)$, respectively, we obtain

$$zxv \equiv -wxs + zwt \mod p \tag{62}$$

$$xwt \equiv xzs + zwv \mod p, \quad \text{and} \tag{63}$$

$$xzt \equiv zws + xwv \mod p, \tag{64}$$

respectively. From (61) and (62) by equating the expressions for zwt we obtain

$$zv(x-w) \equiv ws(z-x) \mod p. \tag{65}$$

Likewise, from (61) and (63) by equating the expressions for xwt we obtain

$$wv(z-x) \equiv xs(w-z) \mod p, \tag{66}$$

and from (61) and (64) by equating the expressions for xzt we obtain

$$xv(z-w) \equiv zs(w-x) \mod p\ . \tag{67}$$

From (65), (66) and (67), it follows that

$$\begin{aligned} w^2(z-x)^2 &\equiv xz(w-z)(x-w) \mod p, \\ -z^2(x-w)^2 &\equiv xw(z-x)(z-w) \mod p, \quad \text{and} \\ -x^2(w-z)^2 &\equiv wz(w-x)(z-x) \mod p\ , \end{aligned} \tag{68}$$

Since the constraints in (68) also only hold for $p = 13$ we conclude that for prime $p, p > 13$ this candidate configuration does not exist.

5.3 Analysis of the candidate $(6,4)$ absorbing sets given in Figure 11

Recall that it is sufficient to consider only two different labellings, namely $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10})$= $(x, y, z, w, y, x, w, z, z, y)$ and $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10})$ = $(x, y, z, w, y, z, w, x, w, y)$. For the first case, by symmetry, it is sufficient to consider $x = 0$ and $z = 0$ as $w = 0$ and $y = 0$ reduce to the $x = 0$ and $z = 0$ case respectively. Likewise, for the second case it is sufficient to consider $x = 0$ and $y = 0$, as $z = 0$ and $w = 0$ each reduce to the $x = 0$ and $y = 0$ cases, respectively.

I. Consider $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10})$= $(x, y, z, w, y, x, w, z, z, y)$

We start with the $z = 0$ analysis.

1. Case $z = 0$

From Figure 11 and under the current edge label assignment using the pattern consistency constraints of Lemma 1(b) we write

$$\begin{aligned}
k_1 &\equiv k_5 && \bmod p, \\
k_2 &\equiv k_6 && \bmod p, \\
k_3 &\equiv k_4 && \bmod p, \\
k_1 + xj_1 &\equiv k_3 + xj_3 && \bmod p, \\
k_1 + yj_1 &\equiv k_4 + yj_4 && \bmod p, \\
k_1 + wj_1 &\equiv k_6 + wj_6 && \bmod p, \\
k_2 + yj_2 &\equiv k_3 + yj_3 && \bmod p, \\
k_2 + xj_2 &\equiv k_4 + xj_4 && \bmod p, \\
k_2 + wj_2 &\equiv k_5 + wj_5 && \bmod p, \quad \text{and} \\
k_5 + yj_5 &\equiv k_6 + yj_6 && \bmod p\ .
\end{aligned} \tag{69}$$

Let $a := j_2 - j_1$, $b := j_3 - j_1$, $c := j_4 - j_1$, $d := j_5 - j_1$ and $e := j_6 - j_1$. Using the cycle constraint for four cycles spanning the cycle space of the configuration in Figure 11 and under the current edge labelling we have

$$\begin{aligned}
xb + y(-c) &\equiv 0 \quad \bmod p, \\
y(b-a) + x(a-c) &\equiv 0 \quad \bmod p, \\
y(e-d) + w(-e) &\equiv 0 \quad \bmod p, \quad \text{and} \\
w(d-a) + y(e-d) &\equiv 0 \quad \bmod p.
\end{aligned} \tag{70}$$

From the systems (69) and (70) we write

$$\begin{aligned}
&k_1 - k_2 \equiv k_1 - k_6 \equiv w(j_6 - j_1) \equiv we && \bmod p, \\
&k_1 - k_3 \equiv k_1 - k_4 \equiv y(j_4 - j_1) \equiv yc && \bmod p, \ \text{and} \\
&k_2 - k_3 \equiv y(j_3 - j_2) \equiv y(b-a) && \bmod p.
\end{aligned} \tag{71}$$

Using the identity $(k_1 - k_2) = (k_1 - k_3) - (k_2 - k_3)$, and (71) we obtain

$$we \equiv y(c - b + a) \quad \bmod p. \tag{72}$$

There are six possible assignments for (x, y, w), as permutations of the set $\{1, 2, 3\}$. In the remainder we will show that in fact only $(x, y, w) = (1, 3, 2)$ gives rise to absorbing sets. In all other cases, we will reach a contradiction.

From (70) we have

$$\begin{aligned}
xb &\equiv yc \quad \bmod p, \quad \text{and} \\
yd &\equiv (y - w)e \quad \bmod p.
\end{aligned} \tag{73}$$

We also have

$$\begin{aligned} xa-(y+x)c &\equiv 0 \mod p, \quad \text{and} \\ (2y-w)e &\equiv ya \mod p, \end{aligned} \tag{74}$$

where the top expression in (74) follows from substituting top expression in (73) into the second expression of (70) and some algebra, and the bottom expression in (74) follows from substituting bottom expression in (73) into the fourth expression of (70).

For $(y,w,x)=(1,2,3)$, the bottom expression in (74) gives $a \equiv 0 \mod p$, which then implies $c \equiv 0 \mod p$, by the top expression in (74). Since $c=j_4-j_1$, and (j_1,k_1) and (j_4,k_4) share a check, c must be non-zero, implying a contradiction.

For $(y,w,x) \in \{(1,3,2),(2,1,3),(2,3,1),(3,1,2)\}$ we express b,c,d,e in terms of a using (73) and (74) and obtain,

– for $(y,w,x)=(1,3,2)$: $b \equiv a/3 \mod p, c \equiv 2a/3 \mod p, d \equiv 2a \mod p, e \equiv -a \mod p$,

– for $(y,w,x)=(2,1,3)$: $b \equiv 2a/5 \mod p, c \equiv 3a/5 \mod p, d \equiv a/3 \mod p, e \equiv 2a/3 \mod p$,

– for $(y,w,x)=(2,3,1)$: $b \equiv 2a/3 \mod p, c \equiv a/3 \mod p, d \equiv -a \mod p, e \equiv 2a \mod p$, and

– for $(y,w,x)=(3,1,2)$: $b \equiv 3a/5 \mod p, c \equiv 2a/5 \mod p, d \equiv 2a/5 \mod p, e \equiv 3a/5 \mod p$.

In all four cases, when b,c and e are substituted in (72) it follows that $a \equiv 0 \mod p$ (we get $-3a \equiv 4a/3 \mod p$, $2a/3 \equiv 12a/5 \mod p$, $6a \equiv 4a/3 \mod p$, and $3a/5 \equiv 12a/5 \mod p$, respectively). Since b is a multiple of a in all four cases, if $a \equiv 0 \mod p$, then $b \equiv 0 \mod p$ as well. Since $b=j_3-j_1$ and nodes (j_1,k_1) and (j_3,k_3) share a check, b must be non-zero, thus implying a contradiction.

For $(y,w,x)=(3,2,1)$ we obtain $b \equiv 3a/4 \mod p, c \equiv a/4 \mod p, d \equiv a/4 \mod p, e \equiv 3a/4 \mod p$. When b,c and e are substituted in (72), we obtain the identity $3a/2 \equiv 3a/2 \mod p$. Since $c \equiv d \mod p$, we have that $j_4=j_5$ and since $b \equiv e \mod p$, we have that $j_3=j_6$. Note that neither of these two conditions on j's violates the check consistency constraint since the respective bit nodes do not share a check in Figure 11. Let $q=j_1$ and $t=j_4-j_1$. Then $j_4=q+t \mod p$ and $j_5=q+t \mod p$. Since $b=3c$, and $b=j_3-j_1$ and $c=j_4-j_1$, we have that $j_3=q+3t \mod p$. Since $j_3=j_6$, $j_6=q+3t \mod p$ as well. Likewise, since $a=4c$, and $a=j_2-j_1$ and $c=j_4-j_1$, we have that $j_2=q+4t \mod p$. We have thus expressed all of j_1 through j_6 in terms of q and t. Now the system (69) reduces to

$$k_1=k_5, \qquad k_2=k_6, \qquad k_3=k_4, \tag{75a}$$

$$k_1-k_3 \equiv 3t \mod p, \qquad k_1-k_2 \equiv 6t \mod p, \qquad k_2-k_3 \equiv -3t \mod p. \tag{75b}$$

Thus, with $s=k_1$ and using (75a)–(75b) we can express all of k_1 through k_6 in terms of s and t. This solution set for j_1 through j_6 and k_1 through k_6 is listed in Table 3, where the entries are taken $\mod p$.

Note that the result in Table 3 establishes the existence of a $(6,4)$ absorbing set. Even though $j_3=j_6$ and $j_4=j_5$, the check consistency constraints are not violated as (j_3,k_3) and (j_6,k_6) do not share an edge, and neither do (j_4, k_4) and (j_5, k_5), see Figure 11.

We now discuss whether this set is also a $(6,4)$ fully absorbing set. Suppose there exists a bit node (j_7, k_7) outside this absorbing set that is incident to some of the unsatisfied checks. By the bit consistency constraint, both (j_3, k_3) and (j_4, k_4) each have a neighboring unsatisfied check whose label is w. These two checks must be distinct by the girth condition [5]. Likewise, both (j_5, k_5) and (j_6, k_6) each have a neighboring unsatisfied check whose label is x, and these are also distinct by the girth condition. By the bit consistency condition, the bit node (j_7, k_7) can then share at most 2 of these checks with the bit nodes (j_3, k_3) through (j_7, k_7). Suppose that the bit node (j_7, k_7) shares a check labelled w with (j_3, k_3) and a check labelled x with (j_5, k_5). From the cycles relating bit nodes (j_7, k_7), (j_3, k_3), (j_5, k_5), (j_1, k_1), and (j_2, k_2), we obtain

$$\begin{aligned} x(j_7 - j_5) + w(j_3 - j_7) + x(j_1 - j_3) &\equiv 0 \mod p, \quad \text{and} \\ w(j_5 - j_2) + x(j_7 - j_5) + w(j_3 - j_7) + y(j_2 - j_3) &\equiv 0 \mod p \,. \end{aligned}$$

For $(x, y, z, w) = (1, 3, 0, 2)$ of present interest, we obtain that $j_7 \equiv q + 2t \mod p$ using the result in Table 3. Since we further have

$$k_3 + 2j_3 \equiv k_7 + 2j_7 \mod p, \quad \text{and } k_5 + j_5 \equiv k_7 + j_7 \mod p,$$

it follows that $k_7 \equiv s - t \mod p$. Therefore by the existence of this bit node (j_7, k_7), the current $(6,4)$ absorbing set is not a $(6,4)$ fully absorbing set.

2. Case $x = 0$

As before, using the pattern consistency constraints we establish:

$$\begin{array}{rcll} k_1 & \equiv & k_3 & \mod p, \\ k_2 & \equiv & k_4 & \mod p, \\ k_1 + yj_1 & \equiv & k_4 + yj_4 & \mod p, \\ k_1 + zj_1 & \equiv & k_5 + zj_5 & \mod p, \\ k_1 + wj_1 & \equiv & k_6 + wj_6 & \mod p, \\ k_2 + yj_2 & \equiv & k_3 + yj_3 & \mod p, \\ k_2 + wj_2 & \equiv & k_5 + wj_5 & \mod p, \\ k_2 + zj_2 & \equiv & k_6 + zj_6 & \mod p, \\ k_3 + zj_3 & \equiv & k_4 + zj_4 & \mod p, \quad \text{and} \\ k_5 + yj_5 & \equiv & k_6 + yj_6 & \mod p, \ . \end{array} \tag{76}$$

Let $a := j_2 - j_1$, $b := j_3 - j_1$, $c := j_4 - j_1$, $d := j_5 - j_1$, and $e := j_6 - j_1$. Using the cycle constraints for four cycles spanning the cycle space of the configuration in Figure 11 we may also write

$$\begin{aligned} z(c-b)+y(-c) &\equiv 0 \mod p, \\ y(b-a)+z(c-b) &\equiv 0 \mod p, \\ zd+y(e-d)+w(-e) &\equiv 0 \mod p, \quad \text{and} \\ w(d-a)+y(e-d)+z(a-e) &\equiv 0 \mod p\,. \end{aligned} \tag{77}$$

There are 6 possible assignments for (y, z, w) as permutations of the set $\{1, 2, 3\}$. We will show that in fact the only possible assignment is $(y, z, w) = (2, 1, 3)$, whereas a contradiction will be reached in all other cases.

Consider first the assignment $(y, z, w) = (1, 2, 3)$. Using (77) we express a, b, c and d in terms of e so that

$$\begin{aligned} a &\equiv 3e \mod p, & b &\equiv e \mod p, \\ c &\equiv 2e \mod p, & d &\equiv 2e \mod p\,. \end{aligned} \tag{78}$$

Note that since $c \equiv d \mod p$ and $b \equiv e \mod p$ the above implies that $j_4 = j_5$ and $j_3 = j_6$. Even though now some vertices have the same j components, the check consistency condition is not violated as (j_4, k_4) and (j_5, k_5) do not share an edge, and neither do (j_3, k_3) and (j_6, k_6) (see Figure 11).

From (76) and by substituting for a, c and d in terms of e using (78) we note that

$$\begin{aligned} k_1 - k_2 &\equiv 1(2e) \mod p, \\ k_1 - k_5 &\equiv 2(2e) \mod p, \quad \text{and} \\ k_2 - k_5 &\equiv 3(2e-3e) \mod p\,. \end{aligned} \tag{79}$$

The system (79) implies that $e \equiv 0 \mod p$ for $p > 5$. Since $e = j_6 - j_1$ and (j_1, k_1) and (j_6, k_6) do share an edge, the condition $e \equiv 0 \mod p$ violates the check consistency constraint. We thus conclude that the current numerical assignment for (y, z, w) is not possible.

By expressing a, b, c and d in terms of e as in (78) and then using (76) to express the differences $k_1 - k_2$, $k_1 - k_5$, and $k_2 - k_5$ as in (79) we conclude that $e \equiv 0 \mod p$ for all primes $p > 13$ when (y, z, w) = $(1, 3, 2)$, $(3, 1, 2)$ or $(3, 2, 1)$.

Consider now the assignment $(y, z, w) = (2, 3, 1)$. Using (77) it follows after substituting for d in terms of e in the last expression that $a \equiv 0 \mod p$. By substituting for b in terms of c in the second expression in (77) it also follows that $c \equiv 0 \mod p$, which violates the check consistency constraint for the edge connecting bit nodes (j_1, k_1) and (j_4, k_4). (The condition $a \equiv 0 \mod p$ by itself does not yield a contradiction as the nodes (j_1, k_1) and (j_2, k_2) do not have any edges in common.)

Finally, we consider the remaining assignment $(y, z, w) = (2, 1, 3)$. First, by substituting for d in terms of e in the last expression of (77), it follows that $a \equiv 0 \mod p$, which implies $j_1 = j_2$. By substituting for

b in terms of c in the second expression in (77), it follows that $1a \equiv (2-2)c \mod p$, which unfortunately does not tell us anything about the actual value of c. We express b, c and d in terms of e, using (76) and (77), and obtain

$$b \equiv d \equiv -e \mod p, \quad \text{and} \quad c \equiv e \mod p\,. \tag{80}$$

Since $b \equiv d \mod p$, $j_3 = j_5$, and since $c \equiv e \mod p$, $j_4 = j_6$. Neither of these conditions on j's violates the check consistency constraints as the respective bit nodes do not share edges (see Figure 11). Thus, with $q := j_1$ and $t := e$ we can express all of j_1 through j_6 in terms of q and t. Having verified that all constraints given by (76) are in fact consistent for $s := k_1$ we obtain the solution set given in Table 4, in terms of q, t and s.

From Figure 11 and under current labelling, note that the bit nodes (j_3, k_3) and (j_4, k_4) both have an unsatisfied check whose label is w, and that likewise the bit nodes (j_5, k_5) and (j_6, k_6) both have an unsatisfied check whose label is x. Therefore there could exist a bit node that connects to 2 satisfied and 2 unsatisfied check nodes. Consider a bit node (j_7, k_7) that shares a check labelled w with (j_3, k_3) and a check labelled x with (j_5, k_5). By the parity check constraint

$$k_7 + wj_7 \equiv k_3 + wj_3 \mod p, \quad \text{and} \quad k_7 + xj_7 \equiv k_5 + xj_5 \mod p\,,$$

for $(x, y, z, w) = (0, 2, 1, 3)$, it follows that $k_7 = k_5 \equiv s + t \mod p$ and $j_7 \equiv q - 4t/3 \mod p$. Thus, the existence of this (j_7, k_7) bit node for t a multiple of 3, makes the candidate configuration be a $(6, 4)$ absorbing set but not a $(6, 4)$ fully absorbing set. We will now show that in fact the remaining labelling is not possible for p large enough.

II. Consider the labelling $(i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10})$= $(x, y, z, w, y, z, w, x, w, y)$.

Applying the cycle consistency condition to the four cycles in Figure 11 for $a := j_2 - j_1$, $b := j_3 - j_1$, $c := j_4 - j_1$, $d := j_5 - j_1$, and $e := j_6 - j_1$ we obtain

$$\begin{aligned} b(x-w) + c(w-y) &\equiv 0 \mod p, \\ e(w-y) + d(y-z) &\equiv 0 \mod p, \\ a(x-w) + d(w-y) + e(y-x) &\equiv 0 \mod p, \quad \text{and} \\ a(z-y) + b(y-w) + c(w-z) &\equiv 0 \mod p\,. \end{aligned} \tag{81}$$

Using the pattern consistency conditions we may also write:

$$\begin{aligned} k_1 + xj_1 &\equiv k_3 + xj_3 \mod p, \\ k_1 + yj_1 &\equiv k_4 + yj_4 \mod p, \\ k_1 + zj_1 &\equiv k_5 + zj_5 \mod p, \\ k_1 + wj_1 &\equiv k_6 + wj_6 \mod p, \\ k_2 + yj_2 &\equiv k_3 + yj_3 \mod p, \\ k_2 + zj_2 &\equiv k_4 + zj_4 \mod p, \\ k_2 + wj_2 &\equiv k_5 + wj_5 \mod p, \\ k_2 + xj_2 &\equiv k_6 + xj_6 \mod p, \\ k_3 + wj_3 &\equiv k_4 + wj_4 \mod p, \quad \text{and} \\ k_5 + yj_5 &\equiv k_6 + yj_6 \mod p\ . \end{aligned} \tag{82}$$

Recall that it is sufficient to only consider $x = 0$ and $y = 0$.

1. Case $x = 0$

With $x = 0$, (82) yields $k_1 \equiv k_3 \mod p$ and $k_2 \equiv k_6 \mod p$ so that

$$k_1 - k_2 \equiv we \equiv y(a - b) \mod p\ . \tag{83}$$

From (81) we then have

$$\begin{aligned} a(z-y)(y-w) + b[(-w)(w-z) + (y-w)^2] &\equiv 0 \qquad \mod p, \quad \text{and} \\ aw(y-z) + e[(w-y)^2 + y(z-y)] &\equiv 0 \qquad \mod p\ . \end{aligned} \tag{84}$$

From (83), and (84) it follows that $a \equiv 0 \mod p$ for all $3! = 6$ numerical assignments of y, z and w, for $p \notin \{2, 3, 5, 7, 37\}$ and consequently $b \equiv 0 \mod p$. Since (j_1, k_1) and (j_3, k_3) share an edge in Figure 11, the $b \equiv 0 \mod p$ condition violates the check consistency constraint for all but a small finite number of values of p.

Remark 1 *Since Theorem 2(c) is concerned with counting* $(6, 4)$ *absorbing sets for* $p > 19$*, note that for* $p = 37$ *and the assignment* (x, y, z, w) *either* $(0, 2, 1, 3)$ *or* $(0, 3, 1, 2)$*, from the equations* (83) *and* (84) *we may express* b, c, d *and* e *in terms of* a *(itself non-zero). Combined with* (82)*, we may then express all of* (j_l, k_l)*,* $1 \leq l \leq 6$ *indices of bits in this absorbing set in terms of three independent parameters:* $q := j_1$*,* $t := j_2 - j - 1$ *and* $s := k_1$.

2. Case $y = 0$

We now have $k_1 \equiv k_4 \mod p$, $k_2 \equiv k_3 \mod p$ and $k_5 \equiv k_6 \mod p$ and

$$xb \equiv zd - w(d-a) \mod p \ , \tag{85}$$

which follows from $k_1 - k_2 = (k_1 - k_5) - (k_2 - k_5)$ and $k_2 = k_3$. From (81) we also have

$$\begin{aligned} a(-wz) + b[w^2 + (w-z)(x-w)] &\equiv 0 \qquad \mod p, \\ a(x-w)w + d(w^2 - xz) &\equiv 0 \qquad \mod p \ . \end{aligned} \tag{86}$$

Combining (85) and (86) it again follows that $a \equiv 0 \mod p$ for all $3! = 6$ numerical assignments of x, z and w for $p \notin \{2, 3, 5, 7, 37\}$. This in turn implies that $b \equiv 0 \mod p$, which violates the check consistency condition.

Remark 2 *Since Theorem 2(c) is concerned with counting $(6, 4)$ absorbing sets for $p > 19$, note that for $p = 37$ and the assignment (x, y, z, w) either $(3, 0, 2, 1)$ or $(2, 0, 3, 1)$, from the equations* (85) *and* (86) *we may express b, c, d and e in terms of a (itself non-zero). Combined with* (82)*, we may then express all of (j_l, k_l), $1 \leq l \leq 6$ indices of bits in this absorbing set in terms of three independent parameters: $q := j_1$, $t := j_2 - j - 1$ and $s := k_1$.*

Acknowledgment

This research was supported in part by NSF grant CCF-0635372, and by Marvell Semiconductor and Intel Corp. through the UC MICRO program. The work of the first author was also supported by the University of California Dissertation Year Fellowship. Portions of this work were presented at the IEEE International Conference on Communications in Glasgow, Scotland in July 2007.

References

[1] C. Di, D. Proietti, T. Richardson, E. Telatar, and R. Urbanke, “Finite length analysis of low-density parity-check codes on the binary erasure channel,” *IEEE Transactions on Information Theory*, vol. 48 (6), pp. 1570–1579, June 2002.

[2] I. Djurdjevic, J. Xu, K. Abdel-Ghaffar, and S. Lin, “A class of low-density parity-check codes constructed based on Reed-Solomon codes with two information symbols,” *IEEE Communications Letters*, vol. 7, pp. 317–319, July 2003.

[3] L. Dolecek, Z. Zhang, M. Wainwright, V. Anantharam, and B. Nikolic, “Evaluation of the low frame error rate performance of LDPC codes using importance sampling,” *Proceedings of IEEE Information Theory Workshop, ITW07*, Lake Tahoe, CA, Sept. 2007, pp. 202–207.

[4] E. Eleftheriou and S. Ölçer, "Low density parity check codes for digital subscriber lines," *Proceedings of the IEEE International Conference on Communications*, New York, NY, April-May 2002, pp. 1752–1757.

[5] J. L. Fan, "Array-codes as low-density parity-check codes," *Second International Symposium on Turbo Codes*, Brest, France, Sept. 2000, pp. 543–546.

[6] J. Feldman, M. J. Wainwright and D. R. Karger, "Using linear programming to decode binary linear codes," *IEEE Transactions on Information Theory*, vol. 51 (3), pp. 954–972, March 2005.

[7] G. D. Forney, "Codes on graphs: normal realizations," *IEEE Transactions on Information Theory*, vol. 47 (2), pp. 520–548, Feb. 2001.

[8] G. D. Forney, Jr., R. Koetter, F. R. Kschischang and A. Reznick, "On the effective weights of pseudocodewords for codes defined on graphs with cycles," *Codes, Systems and Graphical Models*, Springer, pp.101–112, 2001.

[9] R. Koetter and P. Vontobel, "Graph covers and iterative decoding of finite-length codes," *Proceedings of the 3rd International Conference on Turbo Codes and Related Topics*, Brest, France, Sept. 2003, pp. 75–82.

[10] F. R. Kschischang, B. J. Frey and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 42 (2), pp. 498–519, Feb. 2001.

[11] S. Laendner and O. Milenkovic, "Algorithmic and combinatorial analysis of trapping sets in structured LDPC codes," *Wireless Comm*, Hawaii, USA, June 2005, pp. 630–635.

[12] D. MacKay and M. Postol, "Weaknesses of Margulis and Ramanujan-Margulis low-density parity-check codes," *Electronic Notes in Theoretical Computer Science*, vol. 74, 2003.

[13] T. Mittelholzer, "Efficient encoding and minimum distance bounds of Reed-Solomon-type array codes," *International Symposium on Information Theory*, Lausanne, Switzerland, July 2002, p. 282.

[14] A. Orlitsky, K. Viswanathan, J. Zhang, "Stopping set distribution of LDPC code ensembles," *IEEE Transactions on Information Theory*, vol. 51 (3), pp. 929–953, March 2005.

[15] W. W. Peterson and E. J. Weldon, *Error Correcting Codes*, MIT Press 1972.

[16] T. Richardson, "Error-floors of LDPC codes," *Proceedings of the 41st Annual Allerton Conference*, Monticello, Ill., Oct. 2003, pp. 1426–1435.

[17] M. Sipser and D. A. Spielman, "Expander codes," *IEEE Transactions on Information Theory*, vol. 42 (6) pp. 1710–22, Nov. 1996.

[18] Y. Y. Tai, L. Lan, L. Zeng, S. Lin, and K. Abdel-Ghaffar, "Algebraic construction of quasi-cyclic LDPC codes for the AWGN and erasure channels," *IEEE Transactions on Information Theory*, vol. 54 (10), pp. 1765 – 1774, Oct. 2006.

[19] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Transactions on Information Theory*, vol. 27, pp. 533–547, Sept. 1980.

[20] R. M. Tanner, D. Sridhara, A. Sridharan, T. E. Fuja, D. J. Costello, "LDPC block and convolutional codes based on circulant matrices," *IEEE Transactions on Information Theory*, vol. 50 (12), pp. 2966–2984, Dec. 2004.

[21] B. Vasic and E. Kurtas, *Coding and Signal Processing for Magnetic Recording Systems*, CRC press, 2005.

[22] K. Yang and T. Helleseth, “On the minimum distance of array codes as LDPC codes,” *IEEE Transactions on Information Theory*, vol. 49 (12), pp. 3268–3271, Dec. 2003.

[23] Z. Zhang, L. Dolecek, B. Nikolic, V. Anantharam and M. J. Wainwright, “Investigation of error floors of structured low-density parity-check codes via hardware simulation,” *Procedings of GLOBECOM 2006*, San Francisco, CA, Oct.-Nov. 2006, pp. 1–6.

[24] Z. Zhang, L. Dolecek, B. Nikolic, V. Anantharam, and M. J. Wainwright, “Design of LDPC decoders for low bit error rate performance,” preprint, 2008.

[25] IEEE Standard 802.3an-2006, Sept. 2006. Available at http://ieeexplore.ieee.org/servlet/opac?punumber=4039890

[26] Digital Video Broadcasting Project. Available at http://www.dvb.org

[27] The IEEE 802.16 Working Group on Broadband Wireless Access Standards. Available at http://www.ieee802.org/16/