Linear Programming Decoding of Spatially Coupled Codes

Louay Bazzi
Dep. of Electrical & Computer Eng.
American University of Beirut
Beirut, Lebanon
Email: lb13@aub.edu.lb

Badih Ghazi
Dep. of Electrical & Computer Eng.
American University of Beirut
Beirut, Lebanon
Email: bbg01@mail.aub.edu

Rüdiger Urbanke
ECE Department
EPFL
Lausanne, Switzerland
Email: ruediger.urbanke@epfl.ch

***Abstract*—For a given family of spatially coupled codes, we prove that the LP threshold on the BSC of the tail-biting graph cover ensemble is the same as the LP threshold on the BSC of the derived spatially coupled ensemble. This result is in contrast with the fact that the BP threshold of the derived spatially coupled ensemble is believed to be larger than the BP threshold of the tail-biting graph cover ensemble [1], [2].**

I. Introduction

A. Binary linear codes

A binary linear code ζ of block length n is a subspace of the $\mathbb{F}_2$-vector space $\mathbb{F}_2^n$. The ϵ-BSC (Binary Symmetric Channel) with input $X \in \mathbb{F}_2^n$ and output $Y \in \mathbb{F}_2^n$ flips each input bit independently with probability ϵ. Let γ be the log-likelihood ratio vector which is given by $\gamma_i = \log\left(\frac{p_{Y_i|X_i}(y_i|0)}{p_{Y_i|X_i}(y_i|1)}\right) = (-1)^{y_i} \log \frac{1-\epsilon}{\epsilon}$ for any $i \in \{1, \ldots, n\}$. The optimal decoder is the Maximum Likelihood (ML) decoder which is given by $\hat{x}_{\mathrm{ML}} = \underset{x\in\zeta}{\operatorname{argmax}}\, p_{Y|X}(y|x) = \underset{x\in\zeta}{\operatorname{argmin}} \sum_{i=1}^{n} \gamma_i x_i$ where the second equality follows from the fact that the channel is memoryless. Since the objective function is linear in x, replacing ζ by the convex hull $\mathrm{conv}(\zeta)$ of ζ does not change the value of the minimal solution. Hence, we get

$$\hat{x}_{\mathrm{ML}} = \underset{x\in\mathrm{conv}(\zeta)}{\operatorname{argmin}} \sum_{i=1}^{n} \gamma_i x_i \tag{1}$$

ML decoding is known to be NP-hard for general binary linear codes [3]. This motivates the study of suboptimal decoding algorithms that have small running times.

B. Linear Programming Decoding

One such algorithm is the LP (Linear Programming) decoder that was introduced by [4]. It is based on the idea of replacing $\mathrm{conv}(\zeta)$ in (1) with a larger subset of $\mathbb{R}^n$, with the goal of reducing the running time while maintaining a good error correction performance. First, note that $\mathrm{conv}(\zeta) = \mathrm{conv}(\bigcap_{j\in C} \zeta_j)$ where $\zeta_j = \{z \in \{0,1\}^n : w(z|_{N(j)}) \text{ is even}\}$[1] for all j in the set C of check nodes corresponding to a fixed Tanner graph of ζ and where $N(j)$ is the set of all neighbors of check node j. Then, LP decoding is given by "relaxing" $\mathrm{conv}(\bigcap_{j\in C} \zeta_j)$ to $\bigcap_{j\in C} \mathrm{conv}(\zeta_j)$:

$$\hat{x}_{\mathrm{LP}} = \underset{x\in P}{\operatorname{argmin}} \sum_{i=1}^{n} \gamma_i x_i \tag{2}$$

where $P = \bigcap_{j\in C} \mathrm{conv}(\zeta_j)$ is the so-called "fundamental polytope" that was introduced by [5]. A central property of P is that, in the case of LDPC codes, it can be described by a number of inequalities that is linear in n, which implies that the linear program (2) can be solved in time polynomial in n using the ellipsoid algorithm or interior point methods. When analyzing the operation of LP decoding, one can assume that the zero codeword was transmitted [4]. By normalizing the expression of the log-likelihood ratio γ given in Section I-A by the positive constant $\log(\frac{1-\epsilon}{\epsilon})$, we can also assume that the log-likelihood ratio is given by $\gamma_i = 1$ if $y_i = x_i$ and $\gamma_i = -1$ if $y_i \neq x_i$ for all $i \in \{1, \ldots, n\}$. As in previous work, we make the conservative assumption that LP decoding fails whenever there are multiple optimal solutions to the linear program (2). So assuming that the zero codeword is transmitted, LP decoding succeeds if and only if the zero codeword is the unique optimal solution to the linear program (2).

C. Spatially coupled codes

Spatially coupled codes (or convolutional LDPC codes) were introduced in [6]. Recently, [1] showed that the BP threshold of spatially coupled codes is the same as the MAP (Maximum Aposteriori Probability) threshold of the base LDPC code in the case of the Binary Erasure Channel. Moreover, [2] showed that spatially coupled codes achieve capacity under belief

[1]For $x \in \{0,1\}^n$ and $S \subseteq \{1, \ldots, n\}$, $x|_S \in \{0,1\}^n$ denotes the restriction of x to S, i.e., $(x|_S)_i = x_i$ if $i \in S$ and $(x|_S)_i = 0$ otherwise, and $w(x)$ denotes the Hamming weight of x.

propagation. The intuition behind the improvement in performance due to spatial coupling is that the check nodes located at the boundaries have low degrees, which enables the BP algorithm to initially recover the transmitted bits at the boundaries. Then, the other transmitted bits are progressively recovered from the boundaries to the center of the code.

D. The conjecture

Numerical simulations don't show an improvement in the performance of LP decoding under spatial coupling [7]. This lead to the conjecture that the LP threshold of a spatially coupled ensemble on the BSC is the same as that of the base ensemble. A natural approach to prove this claim is twofold:

1) Show that the LP threshold of the spatially coupled ensemble on the BSC is the same as that of the tail-biting graph cover ensemble.
2) Show that the LP threshold of the tail-biting graph cover ensemble on the BSC is the same as that of the base ensemble.

E. Contributions and Outline

We prove the first part of the conjecture. To do so, we prove some general results about LP decoding of LDPC codes that may be of independent interest (Theorems IV.4, V.2, V.3, V.4 and VI.1). Due to space limitations, we present in this extended abstract the main steps of the proof and we sketch some arguments. We give the complete proofs in the full version of the paper [8]. We leave the second part of the conjecture open.

The paper is organized as follows. In Section III, we formally state the main result of the paper. In Section IV, we present a condition that was previously known to be sufficient for LP decoding success and that we prove in [8] to be also necessary and to be equivalent to the existence of certain weighted directed acyclic graphs. In Section V, we present sublinear (in the block length) upper bounds on the weight of any edge in such graphs, for both regular and spatially coupled codes. In Section VI, we show how to trade crossover probability for "LP excess" on all the variable nodes, for any binary linear code. The results of Sections V and VI are finally used in Section VII where we prove the main result of the paper.

II. Notation and terminology

For any integers n, a, b with $n \geq 1$, we denote by $[n]$ the set $\{1, \ldots, n\}$ and by $[a : b]$ the set $\{a, \ldots, b\}$. For any event A, let $\overline{A}$ be the complement of A. For any vertex v of a graph G, we let $N(v)$ denote the set of all neighbors of v in G. A binary linear code ζ can be fully described as the nullspace of a matrix $H \in \mathbb{F}_2^{r \times n}$, called the parity check matrix of ζ. For a fixed H, ζ can be graphically represented by a Tanner graph (V, C, E) which is a bipartite graph where $V = \{v_1, \ldots, v_n\}$ is the set of variable nodes, $C = \{c_1, \ldots, c_r\}$ is the set of check nodes and for any $i \in [n]$ and any $j \in [r]$, $(v_i, c_j) \in E$ if and only if $H_{j,i} = 1$. If H is sparse, then ζ is called a Low Density Parity Check (LDPC) code. If the number of ones in each column of H is d_{v} and the number of ones in each row of H is d_{c}, ζ is called a $(d_{\mathrm{v}}, d_{\mathrm{c}})$-regular code. Let $\hat{d}_{\mathrm{v}} = (d_{\mathrm{v}} - 1)/2$. We assume that $n, d_{\mathrm{c}}, d_{\mathrm{v}} > 2$.

III. Main result

First, we define the spatially coupled codes under consideration.

Definition III.1. (Spatially coupled code)
A $(d_{\mathrm{v}}, d_{\mathrm{c}} = kd_{\mathrm{v}}, L, M)$ spatially coupled code, with d_{v} an odd integer and M divisible by k, is constructed by considering the index set $[-L - \hat{d}_{\mathrm{v}} : L + \hat{d}_{\mathrm{v}}]$ and satisfying the following conditions:[2]

1) M variable nodes are placed at each position in $[-L : L]$ and $M\frac{d_{\mathrm{v}}}{d_{\mathrm{c}}}$ check nodes are placed at each position in $[-L - \hat{d}_{\mathrm{v}} : L + \hat{d}_{\mathrm{v}}]$.
2) For any $j \in [-L + \hat{d}_{\mathrm{v}} : L - \hat{d}_{\mathrm{v}}]$, a check node at position j is connected to k variable nodes at position $j + i$ for all $i \in [-\hat{d}_{\mathrm{v}} : \hat{d}_{\mathrm{v}}]$.
3) For any $j \in [-L - \hat{d}_{\mathrm{v}} : -L + \hat{d}_{\mathrm{v}} - 1]$, a check node at position j is connected to k variable nodes at position i for all $i \in [-L : j + \hat{d}_{\mathrm{v}}]$.
4) For any $j \in [L - \hat{d}_{\mathrm{v}} + 1 : L + \hat{d}_{\mathrm{v}}]$, a check node at position j is connected to k variable nodes at position i for all $i \in [j - \hat{d}_{\mathrm{v}} : L]$.
5) No two check nodes at the same position are connected to the same variable node.

With the exception of the non-degeneracy condition 5, Definition III.1 above is the same as that given in Section II-A of [1]. We next define the tail-biting graph cover codes under consideration which are similar to the tail-biting LDPC convolutional codes introduced by [9].

Definition III.2. (Tail-biting graph cover code)
A $(d_{\mathrm{v}}, d_{\mathrm{c}} = kd_{\mathrm{v}}, L, M)$ tail-biting graph cover code, with d_{v} an odd integer and M divisible by k, is constructed by considering the index set $[-L : L]$ and satisfying the following conditions:

1) M variable nodes and $M\frac{d_{\mathrm{v}}}{d_{\mathrm{c}}}$ check nodes are placed at each position in $[-L : L]$.
2) For any $j \in [-L : L]$, a check node at position j is connected to k variable nodes at position $(j + i) \bmod [-L : L]$ for all $i \in [-\hat{d}_{\mathrm{v}} : \hat{d}_{\mathrm{v}}]$.
3) No two check nodes at the same position are connected to the same variable node.

[2]Informally, $2L + 1$ is the number of "layers" and M is the number of variable nodes per "layer".

2

Note that "cutting" a tail-biting graph cover code at any position $i \in [-L : L]$ yields a spatially coupled code. This motivates the following definition.

Definition III.3. (Derived spatially coupled codes)
Let ζ be a $(d_{\mathrm{v}}, d_{\mathrm{c}} = kd_{\mathrm{v}}, L, M)$ tail-biting graph cover code. For each $i \in [-L : L]$, the $(d_{\mathrm{v}}, d_{\mathrm{c}} = kd_{\mathrm{v}}, L - \hat{d}_{\mathrm{v}}, M)$ spatially coupled code ζ_i' is obtained from ζ by removing all M variable nodes and their adjacent edges at each position $i + j \mod [-L : L]$ for every $j \in [0 : 2\hat{d}_{\mathrm{v}} - 1]$. Then, $\mathcal{D}(\zeta) = \{\zeta_{-L}', \ldots, \zeta_L'\}$ is the set of all $2L + 1$ derived spatially coupled codes of ζ.

Definition III.4. (Ensembles and Thresholds)
Let Γ be an ensemble, i.e, a probability distribution over codes. The LP threshold ξ of Γ on the BSC is $\xi := \sup\{\epsilon > 0 \mid \Pr_{\substack{\zeta\sim\Gamma \\ \epsilon\text{-BSC}}}[\text{LP error on } \zeta] = o(1)\}$.

We are can now state the main result of this paper.

Theorem III.5. (Main result: $\xi_{\mathrm{GC}} = \xi_{\mathrm{SC}}$)
Let Γ_{GC} be a $(d_{\mathrm{v}}, d_{\mathrm{c}} = kd_{\mathrm{v}}, L, M)$ tail-biting graph cover ensemble with d_{v} an odd integer and M divisible by k. Let Γ_{SC} be the $(d_{\mathrm{v}}, d_{\mathrm{c}} = kd_{\mathrm{v}}, L - \hat{d}_{\mathrm{v}}, M)$ spatially coupled ensemble which is sampled by choosing a tail-biting graph cover code $\zeta \sim \Gamma_{\mathrm{GC}}$ and returning a element of $\mathcal{D}(\zeta)$ chosen uniformly at random[3]. Denote by ξ_{GC} and ξ_{SC} the respective LP threholds of Γ_{GC} and Γ_{SC} on the BSC. Then, there exists $\nu > 0$ depending only on d_{v} and d_{c} s.t. if $M = o(L^{\nu})$ and Γ_{SC} satisfies the property that for any constant $\Delta > 0$,

$$\Pr_{\substack{\zeta'\sim\Gamma_{\mathrm{SC}} \\ (\xi_{\mathrm{SC}}-\Delta)\text{-BSC}}}[\text{LP error on } \zeta'] = o(\frac{1}{L^2}) \tag{3}$$

then, $\xi_{\mathrm{GC}} = \xi_{\mathrm{SC}}$.

Note that for $M = \omega(\log L)$, condition (3) above is expected to hold for Γ_{SC} since under typical decoding algorithms, the error probability on the $(\xi_{\mathrm{SC}} - \Delta)$-BSC is expected to decay to zero as $O(Le^{-\Omega(\Delta^2 M)})$. Note also that in the regime $M = \Theta(L^{\delta})$ (for any constant $\delta > 0$), spatial coupling provides empirical improvements under iterative decoding [10].

IV. LP DECODING, DUAL WITNESSES, HYPERFLOWS AND WDAGS

The dual of the LP decoder was first examined in [11] (and further studied in [12] and [13]). The next definition is based on Definition 1 of [11].

Definition IV.1. (Dual witness)
For a given Tanner graph $\mathcal{T} = (V, C, E)$ and a (possibly scaled) log-likelihood ratio function $\gamma : V \to \mathbb{R}$, a dual witness w is a function $w : E \to \mathbb{R}$ that satisfies the following two properties:

$$\forall v \in V : \sum_{c\in N(v)} w(v, c) < \gamma(v) \tag{4}$$

$$\forall c \in C, \forall v, v' \in N(c) : \; w(v, c) + w(v', c) \geq 0 \tag{5}$$

It was proved in [11] that the existence of a dual witness is sufficient for LP decoding success. By a careful consideration of the fundamental polytope, we prove in [8] that it is also necessary. The following definition is based on Definition 1 of [14].

Definition IV.2. (Hyperflow)
For a given Tanner graph $\mathcal{T} = (V, C, E)$ and a (possibly scaled) log-likelihood ratio function $\gamma : V \to \mathbb{R}$, a hyperflow w is a function $w : E \to \mathbb{R}$ that satisfies property (4) above as well as the following property:

$$\begin{aligned} &\forall c \in C, \exists P_c \geq 0, \exists v \in N(c) \text{ s.t. } w(v, c) = -P_c \\ &\quad \text{and } \forall v' \in N(c) \text{ s.t. } v' \neq v, w(v', c) = P_c \end{aligned} \tag{6}$$

By Proposition 1 of [14], the existence of a hyperflow is equivalent to that of a dual witness. Note that any dual witness or hyperflow can be viewed as a weighted directed graph (WDG) where for any $v \in V$ and any $c \in C$, an arrow is directed from v to c if $w(v, c) > 0$, an arrow is directed from c to v if $w(v, c) < 0$ and v and c are not connected by an arrow if $w(v, c) = 0$. In [8], we give an algorithm that transforms a WDG satisfying (4) and (5) into an *acyclic WDG* (denoted by *WDAG*) satisfying (4) and (6). Hence, we get the following theorem:

Theorem IV.3. (Existence of an acyclic WDG)
Let $\mathcal{T} = (V, C, E)$ be a Tanner graph of a binary linear code with block length n and let $\eta \in \{0, 1\}^n$ be any error pattern. If $G = (V, C, E, w, \gamma)$ is a WDG corresponding to a dual witness for η on $\mathcal{T}$, then there is an acyclic WDG $G'' = (V, C, E, w', \gamma)$ corresponding to a hyperflow for η on $\mathcal{T}$.

The following theorem summarizes the different characterizations of LP decoding success.

Theorem IV.4. Let $\mathcal{T} = (V, C, E)$ be a Tanner graph of a binary linear code with block length n and let $\eta \in \{0, 1\}^n$ be any error pattern. Then, the following are equivalent:

1) There is LP decoding success for η on $\mathcal{T}$.
2) There is a dual witness for η on $\mathcal{T}$.
3) There is a hyperflow for η on $\mathcal{T}$.
4) There is a WDAG for η on $\mathcal{T}$.

V. MAXIMUM WEIGHT OF AN EDGE IN A WDAG ON THE BSC

In this section, we present sublinear (in the block length n) upper bounds on the weight of an edge in a WDAG. The main idea of the proofs is the following. Consider a $(d_{\mathrm{v}}, d_{\mathrm{c}})$-regular WDAG G corresponding to a hyperflow. Note that each variable node has a log-likelihood ratio of ± 1. Thus, the total amount of flow available in G is most n. For a substantial weight to get "concentrated" on an edge in G, the $+1$'s should

[3]Here, $\mathcal{D}(\zeta)$ refers to Definition III.3.

3

"move" from variable nodes accross G toward that edge. By the hyperflow equation (6), each check node cuts its incoming flow by a factor of $d_c - 1$. Thus, it can be seen that the maximum weight that can get concentrated on an edge is $o(n)$. Formalizing this idea is easier if G is acyclic when viewed as an undirected graph. Since this is not necessarily true, we give in [8] an algorithm that transforms a WDAG corresponding to a hyperflow into a directed weighted forest (which is by definition a directed graph that is acyclic even when viewed as an undirected graph) that preserves the properties of the original WDAG.

Theorem V.1. A WDAG $G = (V, C, E, w, \gamma)$ can be transformed into a directed weighted forest $T = (V', C', E', w', \gamma')$ that has the following properties:

1) $\{V'_v \mid v \in V\}$ is a partition of V'.
2) For all $v \in V$, $\sum_{v' \in V'_v} \gamma'(v') = \gamma(v)$.
3) For all $v \in V$ and all $v' \in V_v$, $\gamma'(v') \times \gamma(v) \geq 0$.
4) T satisfies the hyperflow equations (4) and (6).
5) The directed paths of G are in a bijective correspondence with the directed paths of T.
6) If G has a single sink node with a single incoming edge weighted by α, then T has a single sink node with a single incoming edge also weighted by α.

Theorem V.1 is used in the proofs, given in [8], of Theorems V.2 and V.4 below.

Theorem V.2. Let $G = (V, C, E, w, \gamma)$ be a WDAG corresponding to LP decoding of a (d_v, d_c)-regular LDPC code on the BSC. Let $n = |V|$ and $\alpha_{\max} = \max_{e \in E} |w(e)|$. Then, $\alpha_{\max} \leq c n^{\frac{\ln(d_v - 1)}{\ln(d_v - 1) + \ln(d_c - 1)}} = o(n)$ for some constant $c > 0$ depending only on d_v.

It is worth noting that the RHS of the above bound on $\alpha_{\max}$ also appears in [15] as a bound on the minimum AWGNC pseudoweight. In [8], we show that the bound of Theorem V.2 is asymptotically tight for (d_v, d_c)-regular LDPC codes.

Theorem V.3. There is an infinite family of (d_v, d_c)-regular Tanner graphs $\{(V_n, C_n, E_n)\}_n$, an infinite family of error patterns $\{\gamma_n\}_n$ and a positive constant c s.t. there exists a hyperflow for γ_n on (V_n, C_n, E_n) and any WDAG $(V_n, C_n, E_n, w, \gamma_n)$ corresponding to a hyperflow for γ_n on (V_n, C_n, E_n) must have

$$\max_{e \in E_n} |w(e)| \geq c n^{\frac{\ln(d_v - 1)}{\ln(d_v - 1) + \ln(d_c - 1)}}$$

Theorem V.4. Let $G = (V, C, E, w, \gamma)$ be a WDAG corresponding to LP decoding of any $(d_v, d_c = k d_v, L, M)$ spatially coupled code on the BSC. Let $n = (2L+1)M = |V|$ and $\alpha_{\max} = \max_{e \in E} |w(e)|$. Then, $\alpha_{\max} \leq c n^{1-\epsilon} = o(n)$ for some constants $c > 0$ depending only on d_v and $0 < \epsilon < 1$ depending only on d_v and d_c.

VI. Interplay between crossover probability and LP excess

We now show that if the probability of LP decoding success is large on some BSC, then if we slightly decrease the crossover probability, we can find a dual witness with a non-negligible "gap" in the inequalities (4) w.h.p.. This result holds for any binary linear code.

Theorem VI.1. Consider a binary linear code with Tanner graph (V, C, E). Let $0 < \epsilon, \delta < 1$, $\epsilon' = \epsilon + (1 - \epsilon)\delta$ and $q_{\epsilon'}$ be the probability of LP decoding error on the ϵ'-BSC. For any error pattern $x \in \{0, 1\}^n$, if $G = (V, C, E, w, \gamma)$ is a dual witness for x, let the "LP excess" on variable node $v \in V$ be given by $\gamma(v) - \sum_{c \in N(v)} w(v, c)$. Then, the probability, over the ϵ-BSC, that there exists a dual witness with LP excess at least $\delta/2$ on all variable nodes is at least $1 - \frac{2 q_{\epsilon'}}{\delta}$.

Idea of proof of Theorem VI.1. For every $x \in \{0, 1\}^n$, let w^x be any dual witness for x if x has one and w^x be the zero vector otherwise, and let $c^x = \frac{(1 + \frac{\delta}{2})}{(1 - \frac{\delta}{2})} \mathbb{E}_{y \sim \delta\text{-BSC}} \{w^{x \vee y}\}$.[4] Note that for any $x \in \{0, 1\}^n$, c^x satisfies Eq. (5). We show that with probability at least $1 - \frac{2 q_{\epsilon'}}{\delta}$ over $x \sim \epsilon$-BSC, c^x has an LP excess of at least $\delta/2$ on all variable nodes. □

VII. Proof of the main result

Definition VII.1. (Special variable nodes, extra flow) Let ζ be a tail-biting graph cover code, $\zeta' \in \mathcal{D}(\zeta)$ and η an error pattern on ζ. The "special variable nodes" of ζ are those variable nodes that are in ζ but not in ζ'. A dual witness for η on ζ with "extra flow" f is a dual witness with f added to the RHS of Eq. (4).

Using Theorem V.4, we can derive the following relation between LP decoding on a tail-biting graph cover code and on a derived spatially coupled code.

Theorem VII.2. Let ζ be a $(d_v, d_c = k d_v, L, M)$ tail-biting graph cover code, $\zeta' \in \mathcal{D}(\zeta)$, η an error pattern on ζ and η' the restriction of η to ζ'. The existence of a dual witness for η' on ζ' is equivalent to the existence of a dual witness for η on ζ with the special variable nodes having an extra flow of $d_v c n^{1-\epsilon} + 1$ for $c > 0$ and $0 < \epsilon < 1$ given in Theorem V.4.

Lemma VII.3. Assume that Γ_{SC} satisfies Eq. (3). Then, for all constants $\Delta_1, \Delta_2, \alpha, \beta > 0$, there is a tail-biting graph cover code $\zeta \in \Gamma_{GC}$ satisfying the following two properties for sufficiently large L:

1) The probability, over the $(\xi_{GC} + \Delta_2)$-BSC, of LP decoding success on ζ is at most α.
2) For all $\zeta' \in \mathcal{D}(\zeta)$, the probability, over the $(\xi_{SC} - \Delta_1)$-BSC, of LP error on ζ' is at most $\beta/(2L+1)$.

[4] $x \vee y$ is the bitwise OR of the binary vectors x and y.

4

Idea of proof of Lemma VII.3. A random $C \sim \Gamma_{\mathrm{GC}}$ satisfies both properties w.h.p.. The $o(1/L^2)$ decay in Eq. (3) is used to get 2) via a union bound. $\square$

Proof of Theorem III.5. We show that $\xi_{\mathrm{GC}} \geq \xi_{\mathrm{SC}}$ and refer the reader to the full version [8] for the proof of $\xi_{\mathrm{GC}} \leq \xi_{\mathrm{SC}}$. We proceed by contradiction. Assume that $\xi_{\mathrm{GC}} < \xi_{\mathrm{SC}}$. Let $\delta = (\xi_{\mathrm{SC}} - \xi_{\mathrm{GC}})/2$, $\eta = \xi_{\mathrm{SC}} - \delta$ and $\lambda = \eta - \delta/2 = \xi_{\mathrm{GC}} + \delta/2$. Note that $\eta > \lambda + (1-\lambda)\delta/2$. Let ζ be one of the tail-biting graph cover codes whose existence is guaranteed by Lemma VII.3 with $\Delta_1 = \delta$, $\Delta_2 = \delta/2$ and $\alpha, \beta > 0$ with $\alpha < 1 - 2\beta/\delta$ and let $\{\zeta'_{-L}, \ldots, \zeta'_L\} = \mathcal{D}(\zeta)$. Let μ be an error pattern on ζ and for any $i \in [-L : L]$, let μ_i be the restriction of μ to ζ'_i and E_i the event that there exists a dual witness for μ_i on $\zeta_i^{'}$ with excess $\delta/2$ on all variable nodes. Let $E = \bigwedge_{i\in[-L:L]} E_i$. Then,

$$\begin{aligned}
\Pr_{\lambda\text{-BSC}}\{\overline{E}\} &\leq \sum_{i=-L}^{L} \Pr_{\lambda\text{-BSC}}\{\overline{E_i}\} \\
&\leq \sum_{i=-L}^{L} \frac{2}{\delta} \Pr_{\eta\text{-BSC}}\{\text{LP decoding error on } \zeta_i^{'}\} \\
&\leq \sum_{i=-L}^{L} \frac{2}{\delta} \times \frac{\beta}{2L+1} = \frac{2\beta}{\delta}
\end{aligned}$$

where the second inequality above follows from Theorem VI.1. If event E occurs, then by Corollary VII.2, for every $l \in [-L : L]$, there exists a dual witness $\{\tau_{ij}^l\}_{i\in V, j\in C}$ for μ on ζ with the special variable nodes being at positions $[l : l + 2\hat{d}_{\mathrm{v}} - 1]$ and having an extra flow of $d_{\mathrm{v}} c n^{1-\epsilon} + 1$ with $c > 0$ and $0 < \epsilon < 1$ given in Theorem V.4 and with the non-special variable nodes having excess $\delta/2$. Then, we can construct a dual witness for μ on ζ by averaging the above $2L+1$ dual witnesses as follows. For all $i \in V$ and $j \in C$, let $\tau_{ij}^{\mathrm{avg}} = \frac{1}{2L+1} \sum_{l=-L}^{L} \tau_{ij}^l$. We claim that $\{\tau_{ij}^{\mathrm{avg}}\}_{i,j}$ forms a dual witness for μ on ζ. In fact, for each $i \in V$, $j \in C$ and $l \in [-L : L]$, $\tau_{ij}^l + \tau_{i'j}^l \geq 0$ which implies that $\tau_{ij}^{\mathrm{avg}} + \tau_{i'j}^{\mathrm{avg}} = \frac{1}{2L+1} \sum_{l=-L}^{L} (\tau_{ij}^l + \tau_{i'j}^l) \geq 0$. Moreover, for all $i \in V$, $\sum_{j\in N(i)} \tau_{ij}^{\mathrm{avg}}$ is equal to

$$\begin{aligned}
&\frac{1}{2L+1} \sum_{l=-L}^{L} \Big(\sum_{j\in N(i)} \tau_{ij}^l \Big) \\
&< \frac{1}{2L+1} \big((d_{\mathrm{v}} - 1)(d_{\mathrm{v}} c (M(2L+1))^{1-\epsilon} + 1 + \gamma_i) \\
&+ (2L + 1 - (d_{\mathrm{v}} - 1))(\gamma_i - \frac{\delta}{2}) \big) \\
&= \gamma_i + (d_{\mathrm{v}} - 1) d_{\mathrm{v}} c \frac{(M(2L+1))^{1-\epsilon}}{2L+1} - \frac{\delta}{2} + O(\frac{1}{L}) \\
&< \gamma_i \text{ if } M = o(L^{\nu}),\ L \text{ sufficiently large and } \nu = \frac{\epsilon}{1-\epsilon}
\end{aligned}$$

Letting $p_\lambda = \Pr_{\lambda\text{-BSC}}\{\text{LP decoding success on } \zeta\}$, we get that $p_\lambda \geq \Pr_{\lambda\text{-BSC}}\{E\} \geq 1 - \frac{2\beta}{\delta}$, which contradicts the fact that $p_\lambda = p_{\xi_{\mathrm{GC}}+\Delta_2} \leq \alpha < 1 - \frac{2\beta}{\delta}$. $\square$

VIII. Conclusion

We proved Part 1 of the conjecture in Section I-D. We leave Part 2 open. Since the performance of min-sum is believed to be generally similar to that of LP decoding, an interesting related question is whether there is an improvement in the performance of min-sum under spatial coupling on the BSC, and if not why do min-sum and BP differ so significantly ?

Acknowledgments

The authors would like to thank Pascal Vontobel and the anonymous reviewer for very helpful comments.

References

[1] S. Kudekar, T. Richardson, and R. Urbanke, “Threshold saturation via spatial coupling: Why convolutional LDPC ensembles perform so well over the BEC,” *Information Theory, IEEE Transactions on*, vol. 57, no. 2, pp. 803–834, 2011.

[2] ——, “Spatially coupled ensembles universally achieve capacity under belief propagation,” in *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*. Cambridge, MA, USA: IEEE, 2012, pp. 453–457.

[3] E. Berlekamp, R. McEliece, and H. Van Tilborg, “On the inherent intractability of certain coding problems (corresp.),” *Information Theory, IEEE Transactions on*, vol. 24, no. 3, pp. 384–386, 1978.

[4] J. Feldman, M. Wainwright, and D. Karger, “Using linear programming to decode binary linear codes,” *Information Theory, IEEE Transactions on*, vol. 51, no. 3, pp. 954–972, 2005.

[5] R. Koetter and P. Vontobel, “Graph-covers and iterative decoding of finite length codes,” in *Proceedings of the IEEE International Symposium on Turbo Codes and Applications*, 2003, pp. 75–82.

[6] A. Jimenez Felstrom and K. Zigangirov, “Time-varying periodic convolutional codes with low-density parity-check matrix,” *Information Theory, IEEE Transactions on*, vol. 45, no. 6, pp. 2181–2191, 1999.

[7] D. Burshtein, 2011, private communication.

[8] L. Bazzi, B. Ghazi, and R. Urbanke, “Linear programming decoding of spatially coupled codes,” *arXiv preprint arXiv:1301.6410*, 2013.

[9] M. Tavares, K. Zigangirov, and G. Fettweis, “Tail-biting LDPC convolutional codes,” in *Information Theory, 2007. ISIT 2007. IEEE International Symposium on*. Nice, France: IEEE, 2007, pp. 2341–2345.

[10] P. Olmos and R. Urbanke, “Scaling behavior of convolutional LDPC ensembles over the BEC,” in *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*. Saint Petersburg, Russia: IEEE, 2011, pp. 1816–1820.

[11] J. Feldman, T. Malkin, R. Servedio, C. Stein, and M. Wainwright, “LP decoding corrects a constant fraction of errors,” *Information Theory, IEEE Transactions on*, vol. 53, no. 1, pp. 82–89, 2007.

[12] P. O. Vontobel and R. Koetter, “Towards low-complexity linear-programming decoding,” in *Turbo Codes&Related Topics; 6th International ITG-Conference on Source and Channel Coding (TURBOCODING), 2006 4th International Symposium on*. VDE, 2006, pp. 1–9.

[13] D. Burshtein, “Iterative approximate linear programming decoding of ldpc codes with linear complexity,” *Information Theory, IEEE Transactions on*, vol. 55, no. 11, pp. 4835–4859, 2009.

[14] C. Daskalakis, A. Dimakis, R. Karp, and M. Wainwright, “Probabilistic analysis of linear programming decoding,” *Information Theory, IEEE Transactions on*, vol. 54, no. 8, pp. 3565–3578, 2008.

[15] P. O. Vontobel and R. Koetter, “Graph-cover decoding and finite-length analysis of message-passing iterative decoding of ldpc codes,” *arXiv preprint cs/0512078*, 2005.

5