Probabilistic Analysis of Linear Programming Decoding

Constantinos Daskalakis* Alexandros G. Dimakis*

Richard M. Karp* Martin J. Wainwright*,†

*Department of Electrical Engineering and Computer Sciences
†Department of Statistics,
UC Berkeley, Berkeley, CA 94720.

Abstract

We initiate the probabilistic analysis of linear programming (LP) decoding of low-density parity-check (LDPC) codes. Specifically, we show that for a random LDPC code ensemble, the linear programming decoder of Feldman et al. succeeds in correcting a constant fraction of errors with high probability. The fraction of correctable errors guaranteed by our analysis surpasses all prior non-asymptotic results for LDPC codes, and in particular exceeds the best previous finite-length result on LP decoding by a factor greater than ten. This improvement stems in part from our analysis of probabilistic bit-flipping channels, as opposed to adversarial channels. At the core of our analysis is a novel combinatorial characterization of LP decoding success, based on the notion of a generalized matching. An interesting by-product of our analysis is to establish the existence of "almost expansion" in random bipartite graphs, in which one requires only that almost every (as opposed to every) set of a certain size expands, with expansion coefficients much larger than the classical case.

1 Introduction

Low-density parity-check (LDPC) codes are a class of sparse binary linear codes, first introduced by Gallager [13], and subsequently studied extensively by various researchers [19, 20, 18]. When decoded with efficient iterative algorithms (e.g., the sum-product algorithm [17]), suitably designed classes of LDPC codes yield error-correcting performance extremely close to the Shannon capacity of noisy channels for very large codes [4]. Most extant methods for analyzing the performance of iterative decoding algorithms for LDPC codes—notably the method of density evolution [18, 20]—are asymptotic in nature, based on exploiting the high girth of very large random graphs. Therefore, the thresholds computed using density evolution are only estimates of the true algorithm behavior, since they assume a cycle-free message history. In fact, the predictions of such methods are well-known to be inaccurate for specific codes of intermediate blocklength (e.g., codes with a few hundreds or thousands of bits). For this reason, our current understanding of practical decoders for smaller codes, which are required for applications with delay constraints (e.g., high throughput applications), is relatively limited.

The focus of this paper is the probabilistic analysis of linear programming (LP) decoding, a technique first introduced by Feldman et al. [7, 12] as an alternative to iterative algorithms for decoding LDPC codes. The underlying idea is a standard one in combinatorial optimization—namely, to solve a particular *linear programming* (LP) relaxation of the integer program corresponding to maximum likelihood (optimal) decoding. Although the practical performance of LP decoding is comparable to message passing decoding, a significant advantage is its relative amenability to non-asymptotic analysis. Moreover, there turn out to a number of important theoretical connections between the LP decoding and standard forms of iterative decoding [16, 24]. These connections allow theoretical insight from the LP decoding perspective to be transferred to iterative decoding algorithms.

Previous work: The technique of LP decoding was introduced for turbo-like codes [7], extended to LDPC codes [8, 12], and further studied by various researchers (e.g., [22, 9, 6, 11, 14]). For concatenated expander codes, Feldman and Stein [11] showed that LP decoding can achieve capacity; see also Barg and Zemor [1] for analysis of these generalized constructions. For the standard LDPC codes used in practice, the best positive result from previous work [10, 9] is the existence of a constant $\beta > 0$, depending on the rate of the code, such that LP decoding can correct *any* bit-flipping pattern consisting of at most βn bit flips. (In short, we say that LP decoding can correct a β-fraction of errors.) As a concrete example, for suitable classes of rate $1/2$ LDPC codes, Feldman et al. [9] established that $\beta = 0.000177$ is achievable. However, this analysis [9] was worst-case in nature, essentially assuming an adversarial channel model. Such analysis yields overly conservative predictions for the probabilistic channel models (e.g., each bit flipped with some probability α) that are of more practical interest. Consequently, an important direction—and the goal of this paper—is to develop methods for finite-length and *average-case analysis* of the LP decoding method.

Our contributions: This paper initiates the average-case analysis of LP decoding for LDPC codes. In particular, we analyze the following question: what is the probability, given that a random subset of αn bits is flipped by the channel, that LP decoding succeeds in recovering correctly the transmitted codeword? As one concrete example, we prove that for bit-regular LDPC codes of rate $1/2$ and a random error pattern with αn bit flips, LP decoding will recover the correct codeword, with probability converging exponentially to one, for all α up at least 0.002. This guarantee is roughly ten times higher than the best guarantee from prior work [9]. Our proof is based on analyzing the dual of the decoding linear program, as was done in previous work [9, 10]. The key innovation is a simple graph-theoretic condition for certifying a zero-valued solution the dual LP, which (by strong duality) ensures that the LP decoder correctly recovers the transmitted codeword. The core of the proof involves establishing that such a dual witness exists with high probability under the appropriate conditions. The

argument itself entails a fairly delicate sequence of union bounds and concentration inequalities, exploiting expansion and matchings on random bipartite graphs. An interesting by-product of our analysis is the proof of the existence of "almost-all expanders"—that is, bipartite graphs in which almost all sets of vertices of size up to αn have large expansion. In any such graphs, large randomly selected subsets of vertices have high probability of expanding. In effect, by relaxing the expansion requirement from every set to almost all sets of a given size, we show that one can obtain much larger expansion factors, and hence stronger guarantees on error correction. The remainder of the paper is organized as follows. We begin in Section 2 with background on error-control coding and low-density parity-check codes, as well as the method of linear programming (LP) decoding. Section 3 describes our main result and Section 4 provides the proof in a series of lemmas, with more technical details deferred to the appendices.

2 Background and Problem Formulation

We begin with some background on low-density parity-check codes. We then describe the LP decoding method, and formulate the problem to be studied in this paper.

Low-density parity-check codes: The purpose of an error-correcting code is to introduce redundancy into a data sequence so as to achieve error-free communication over a noisy channel. Given a binary vector of length k (representing information to be conveyed), the encoder maps it onto a codeword, corresponding to a binary vector of length $n > k$. The code rate is given by $\tilde{r} = k/n$, corresponding to the ratio of information bits to transmitted bits. In a binary linear code, the set of all possible codewords corresponds to a subspace of $\{0,1\}^n$, with a total of 2^k elements (one for each possible information sequence). The codeword is then transmitted over a noisy channel. In this paper, we focus on the *binary symmetric channel* (BSC), in which each bit is flipped independently with probability α. Given the received sequence from the channel, the goal of the decoder is to correctly reconstruct the transmitted codeword (and hence the underlying information sequence).

Any binary linear code can be described as the null space of a parity check matrix $H \in \{0,1\}^{(n-k)\times n}$; more concretely, the code $\mathcal{C}$ is given by the set of all binary strings $x \in \{0,1\}^n$ such that $Hx = 0$ in modulo two arithmetic. A convenient graphical representation of such a binary linear code is in terms of its factor graph [17]. The factor graph associated with a code $\mathcal{C}$ is a bipartite graph $G = (V, C)$, with a $n = |V|$ variable nodes corresponding to the codeword bits, and $m = n-k = |C|$ nodes corresponding to the parity checks (rows of the matrix H). Edges in the factor graph connect each variable node to the parity checks which constrain it; that is, the parity check matrix H specifi es the adjacency matrix of the graph. See Figure 1 for an illustration of a particular factor graph. A *low-density parity-check* code is a binary linear code that can be expressed with a sparse factor graph with $O(n)$ edges.

Although this paper focuses on the binary symmetric channel (BSC), our methods are extensible to the more general family of binary-input memoryless symmetric channels. Given a received sequence $y \in \{0,1\}^n$ from the BSC, the optimal Maximum Likelihood (ML) decoding problem is to determine the closest codeword (in Hamming distance). It is well known that the problem of optimal decoding for general binary linear codes NP-hard [2]. This complexity motivates the study of sub-optimal but practical algorithms for decoding.

LP decoding: We now describe how the problem of optimal decoding can be reformulated as a linear program over the *codeword polytope*, i.e. the convex hull of all codewords of the code $\mathcal{C}$. For every bit $\hat{y}_i$ of the received codeword $\hat{y}$, defi ne its log-likelihood as $\gamma_i = \log\left(\frac{\Pr[\hat{y}_i|y_i=0]}{\Pr[\hat{y}_i|y_i=1]}\right)$. Using the memoryless property of the channel, it can be seen that the maximum likelihood (ML) codeword is $\widehat{y}_{\mathrm{ML}} = \mathrm{argmin}_{y\in\mathcal{C}} \sum_{i=1}^n \gamma_i y_i$. Without changing

the outcome of the maximization, we can change the set we are optimizing over to its convex hull conv($\mathcal{C}$) [1] and express ML decoding as the linear program $\widehat{y}_{\mathrm{ML}} = \mathrm{argmin}_{y \in \mathrm{conv}(\mathcal{C})} \sum_{i=1}^{n} \gamma_i y_i$. Although the problem is now just a linear program, it remains intractable because the codeword polytope does not have a simple description.

The natural approach is to relax the linear program by taking only a polynomial set of constraints that provide an outer bound on the codeword polytope $\mathrm{conv}(\mathcal{C})$. The first-order LP decoding method [12] makes use of a relaxation that follows by looking at each parity check (row of H) independently. For each check $a \in C$ in the code, denote by $\mathcal{C}_a$ the set of binary sequences that satisfy it—that is, $\mathcal{C}_a$ corresponds to the local parity check subcode defined check a and its bit neighbors. Observe that the full code $\mathcal{C}$ is simply the intersection of all the local codes, and the codeword polytope has the *exact* representation $\mathrm{conv}(\mathcal{C}) = \mathrm{conv}(\bigcap_{a=1}^{m} \mathcal{C}_a)$. The first-order LP decoder simply ignores interactions between the various local codes, and performs the optimization over the relaxed polytope given by $\mathcal{P} := \bigcap_{a=1}^{m} \mathrm{conv}(\mathcal{C}_a)$. Note that $\mathcal{P}$ is a convex set that contains the codeword polytope $\mathrm{conv}(\mathcal{C})$, but also includes additional vertices with fractional coordinates (called *pseudocodewords* in the coding literature). (It can be shown [24] that if the LDPC graph had no cycles, this relaxation would be exact, hence it can be thought of as a tree-based relaxation.) In contrast to the codeword polytope, the relaxed polytope $\mathcal{P}$ for LDPC codes consists of a linear number of constraints; see Appendix B for an exact description of the inequality constraints defining $\mathcal{P}$. Consequently, LP decoding consists of solving the relaxed linear program:

$$\widehat{y}_{\mathrm{LP}} = \mathrm{argmin}_{y \in \mathcal{P}} \sum_{i=1}^{n} \gamma_i y_i, \tag{1}$$

which can solved exactly in polynomial time, or even faster with iterative and/or approximate methods [3, 23, 24].

3 Description of Main Result

In this section, we describe our main result characterizing the performance of LP decoding for a random ensemble of LDPC codes, before turning to its proof in Section 4.

Random code ensemble: We consider the random ensemble of codes constructed according to the following procedure. Given a code rate $\tilde{r} \in (0, 1)$, form a bipartite *factor graph* $G = (V, C)$ with a set of $n = |V|$ variable nodes, and $m = |C| = \lfloor (1 - \tilde{r})n \rfloor$ check nodes as follows: (i) Fix a variable degree $d_v \in \mathbb{N}$; and (ii) For each variable $j \in V$, choose a random subset $N(j)$ of size d_v from C, and connect variable j to each check in $N(j)$. For obvious reasons, we refer to this as the *bit-regular random ensemble*, and use $\mathcal{C}(d_v)$ to denote a randomly-chosen LDPC code from this ensemble.

The analysis of this paper focuses primarily on the binary symmetric channel (BSC), in which each bit of the transmitted codeword is flipped independently with some probability α. By concentration of measure for the binomial distribution, it is equivalent (at least asymptotically) to assume that a constant fraction αn of bits are flipped by the channel. Let $\mathbb{P}$ denote the joint measure, over both the space of bit-regular random codes, and the space of αn bit flips. Our goal is to obtain upper bounds on the LP error probability $\mathbb{P}[\text{LP fails}]$.

Our analysis will be based on the *expansion* of the factor graph of the code. Specifically, the factor graph of a code will be a (k, Δ)-*expander* if all sets S of *variable nodes*, which are small enough $|S| \leq k$, are connected to at least $\Delta|S|$ checks. Note that throughout this paper, we will be working with codes that have simple parity check constraints (LDPC codes) which are different from the generalized expander codes [21],[11], that can have large linear codes as constraints.

[1] Assume that there is a unique optimum; otherwise declare decoding failure.

Before stating our main result, we note that, as can be easily verified (see e.g. [9]), the bit-regular random construction yields a code with good expansion, with constant probability:

Lemma 1 (Good expansion [9])**.** *For any fixed code rate $\tilde{r} \in (0,1)$ and constant $\delta \in (0,1)$ such that $(1-\delta)d_v$ is an integer greater than or equal to two, a code $\mathcal{C}(d_v)$ from the bit-regular ensemble has probability larger than $1/2$ of being a $(\upsilon n, \delta d_v)$ expander, where*

$$\upsilon \;=\; (2e^{\delta d_v+1}\,(\delta d_v/(1-\tilde{r}))^{(1-\delta)d_v})^{-\frac{1}{(1-\delta)d_v-1}} \;>\; 0. \tag{2}$$

Statement of main result: Our main result is that, for the joint measure over expander bit-regular codes and $\lceil \alpha_c n \rceil$ (or less) bit flips by the channel, LP Decoding will succeed in recovering the correct codeword with high probability. The fraction of correctable errors α_c we establish, is at least ten times higher than the previously known (worst case) result [9]. More formally,

Theorem 1. *There exist constants $\tilde{r}, d_v, c, \upsilon, p > 0$ such that, for all $\alpha \in (0, \alpha_c)$, the LP decoder succeeds with high probability over the space of $(\upsilon n, p)$-expander bit-regular random codes and $\lceil \alpha n \rceil$ bit flips; in other words,*

$$\mathbb{P}[\text{ LP succeeds } | \, \mathcal{C}(d_v) \text{is a } (\upsilon n, p) \text{ expander }] \geq 1 - e^{-cn}. \tag{3}$$

The fraction of correctable errors α_c is a function of the code ensemble, specified by the code rate $\tilde{r}$, variable degree d_v, expansion parameters υ and p, and the error exponent c.

In the sequel, we provide specific parameters for rate $\tilde{r} = 1/2$ that yield the fraction $\alpha_c = 0.002$. (For this parameter setting, a random bit-regular code is a $(\upsilon n, p)$-expander with probability at least $\frac{1}{2}$.) We now state a corollary associated with this particular result.

Corollary 1. *For code rate $\tilde{r} = \frac{1}{2}$, there exist constants $d_v, c > 0$ such that, for all $\alpha \in (0, 0.002)$, the LP decoder succeeds with probability at least $\frac{1}{2} - o(1)$ over the space of bit-regular random codes and $\lceil \alpha n \rceil$ bit flips; in other words,* $\mathbb{P}[\text{ LP succeeds }] \geq \frac{1}{2} - o(1)$.

Improved combinatorial witness – The $(p,q)-$matching The condition that we are going to use to prove that the LP decoder succeeds will be a *dual witness*, i.e a dual feasible point, which will exhibit that the primal linear program has an integral optimal solution. Using the symmetry of the relaxed polytope, it can be shown[9] that the failure, or success, of the LP decoder only depends *on which bits the channel flipped and not on the transmitted codeword*. Using this symmetry, Feldman et al. [10] demonstrated that a dual witness can be graphically interpreted as a set of weights on the edges of the factor graph of the code as the following lemma specifies.

Lemma 2 (Dual witness [9])**.** *Suppose that the channel flipped the bits of set F and left the bits of set $F^c := V \setminus F$ unchanged. Set $\gamma_i = -1$, for all $i \in F$, and $\gamma_i = 1$, for all $i \in F^c$. Linear Programming Decoding will succeed for this error pattern* if and only if *there exist weights $\tau_{i,a}$ for all checks $a \in C$ and adjacent bits $i \in N(a)$ such that the following conditions hold:*

$$\tau_{i,a} + \tau_{j,a} \;\geq\; 0 \qquad \textit{for all checks } a \in C \textit{ and adjacent bits } i, j \in N(a). \tag{4a}$$

$$\sum_{a \in N(i)} \tau_{i,a} \;<\; \gamma_i \qquad \textit{for all } i \in V \textit{ with } \gamma_i < 0. \tag{4b}$$

The key requirement now is a *combinatorial characterization* of when it is possible to assign such weights and hence establish that LP Decoding succeeds. To provide some intuition, the flipped variables need to "push" one unit of negative weight while the unflipped can absorb up to one unit. One way to achieve this is to match each flipped bit with a number of checks, say p checks, to which it has the exclusive privilege to push flow, suppose in a uniform fashion. Let us refer to the checks that are actually used in such a matching as *dirty*, and to all the checks in $N(F)$ as *potentially dirty*. The challenge is that there might be unflipped variables that are adjacent to multiple dirty checks, and hence fail to satisfy the condition (4b); roughly speaking, they receive more weight than they can actually absorb. Thus, the goal is to construct the matching of the flipped bits with p checks each in a careful way so that no unflipped bit has too many dirty neighbors. The δ-matching witness, used by Feldman et al. [9, 10], avoids this difficulty in a brute force manner by matching *all* of the bits adjacent to potentially dirty checks with $\delta = p$ checks each. Our approach circumvents this difficulty using the more refined combinatorial object that we call a (p, q)-matching. For each bit $j \in F^c$, let $Z_j := |N(j) \cap N(F)|$ be the number of its edges adjacent to checks in $N(F)$.

Definition 1. Given non-negative integers p and q, a $(p, q)-$*matching* is defined by the following conditions:

- each bit $i \in F$ must be matched with p (distinct) checks.
- each bit $j \in F^c$ must be matched with $r_j := \max\{q - d_v + Z_j, 0\}$ checks from the set $N(F)$.

We will refer to the number of checks with which each variable node needs to be matched as its *requests*. In this language, all flipped bits have p requests while each unflipped bit j has a variable number of requests r_j which depends on how many of its edges land on checks which have flipped neighbors. The following lemma summarizes an important property of our construction:

Lemma 3. *A (p, q)-matching guarantees that* all the flipped bits are matched with p checks, and all the non-flipped bits have q or more non-dirty check neighbors.

This fact follows by observing that any unflipped bit j with Z_j edges in $N(F)$ has $d_v - Z_j$ clean neigboring checks, and requests $q - (d_v - Z_j)$ extra checks from the potentially dirty ones.

The following lemma, whose proof we omit, establishes that a (p, q)-matching is a certificate of LP decoding success:

Lemma 4. *For any p and q such that $2p + q > 2d_v$, a (p, q)-generalized matching can be used to generate a set of weights $\tau_{i,a}$ which satisfy the dual conditions* (4).

In fact, it is easy to verify that our witness corresponds to a weaker condition for LP Decoding success than the condition used in by Feldman et al. [9]. This strength of our witness along with the randomized analysis are the two ingredients that allow us to establish a much larger fraction α_c of correctable errors.

4 Proof of Theorem 1

The key step in our proof will be to establish that, with high probability over the selection of random expander bit-regular codes and random subsets of $\lceil \alpha n \rceil$ flipped bits, a (p, q)-matching exists, for suitable values of p, q to be specified later. In order to analyze the existence of such a matching, we will make use of Hall's theorem (see also [9]), which, in our context, states that a matching exists if and only if *every subset of the variable nodes* have (jointly) enough neighbors in $N(F)$ to cover the sum of their requests. Observe, however, this inconvenient asymmetry in the definition of our generalized matching: the bits of set F^c need to be matched

with checks from the neighborhood of the flipped bits F and not from the whole set of checks from which they select their neighbors anyway. This correlation between $N(F)$ and the number of requests from set F^c creates severe complications in the analysis. Indeed, any attempt to use Hall's condition through union bounds seems to require independence among different edges in the creation of the code and crude upper-bounds on the number of requests from set F^c seem inadequate to *decorrelate* the requests of F^c from the size of $N(F)$. Rather, establishing our claim requires a somewhat involved sequence of union bounds, concentration inequalities and partitions of our probability space in regions with different properties.

4.1 Partitioning the Probability Space

Under the described probabilistic model, an equivalent description of the neighborhood choices for each variable $j \in F^c$ is as follows. Each node $j \in F^c$ picks a random number $Z_j \in \{0, 1, \ldots, d_v\}$ according to the binomial distribution $\mathrm{Bin}(d_v, \frac{|N(F)|}{m})$, and picks a subset of $N(F)$ of size Z_j. This subset corresponds to the intersection of its check neighborhood $N(j)$ with the check neighborhood $N(F)$ of the flipped bits. The remaining $d_v - Z_j$ edges from bit j connect to checks outside $N(F)$. With this set-up, we now define the following "bad event" which corresponds to the existence of a pair $(S_1, S_2) \in 2^F \times 2^{F^c}$ of sets that *contracts* (i.e., has more requests than neighbors):

$$\mathcal{A} \;:=\; \left\{ \exists S_1 \subseteq F,\ S_2 \subseteq F^c \;\middle|\; |N(S_1) \cup [N(S_2) \cap N(F)]| \leq p|S_1| + \sum_{j \in S_2} \max\{0, q - (d_v - Z_j)\} \right\} \quad (5)$$

Notice that only the neighbors in $N(F)$ are counted, since a (p, q)-matching involves only checks in $N(F)$. By Lemma 4, the event $\mathcal{A}$ must occur whenever LP decoding fails so that we have the inequality

$$\mathbb{P}[\text{LP decoding fails}] \;\leq\; \mathbb{P}[\mathcal{A}]. \quad (6)$$

As was mentioned above, it seems to be useful to partition the space $A := 2^F \times 2^{F^c}$ into three subsets controlled by the parameters $\epsilon_2, \upsilon > 0$. Parameter $\epsilon_2 > 0$ is a small constant to be specified later in the proof, whereas υ is the expansion coefficient specified by equation (2) for $\delta = \frac{p}{d_v}$. The three subsets of interest are given by $A_1 := \{(S_1, S_2) \mid (S_1, S_2) \in A,\ |S_1| + |S_2| < \upsilon n\}$, $A_2 := \{(S_1, S_2) \mid (S_1, S_2) \in A - A_1,\ |S_1| \geq \epsilon_2 n\}$, and $A_3 := A - A_1 - A_2$. This partition, as illustrated in Figure 2 in Appendix C, decomposes $\mathcal{A}$ into sub-events

$$\mathcal{A}(A_i) := \left\{ \exists (S_1, S_2) \in A_i \middle|\ |N(S_1) \cup [N(S_2) \cap N(F)]| \leq p|S_1| + \sum_{j \in S_2} \max\{0, q - (d_v - Z_j)\} \right\} \quad (7)$$

for $i = 1, 2, 3$. Now, a series of union bounds provides the following bound for the probability of failure

$$\begin{aligned} \mathbb{P}[\text{ LP fails} \mid \mathcal{C}(d_v) \text{ is a } (\upsilon n, p) \text{ expander }] \;&\leq\; \mathbb{P}[\,\mathcal{A} \mid \mathcal{C}(d_v) \text{ is a } (\upsilon n, p) \text{ expander }] \\ &\leq\; \sum_{i=1}^{3} \mathbb{P}[\,\mathcal{A}(A_i) \mid \mathcal{C}(d_v) \text{ is a } (\upsilon n, p) \text{ expander }]. \end{aligned}$$

However, all subsets $(S_1, S_2) \in A_1$ of an expander have a p-matching and, because $q < p$, it follows that

$$\mathbb{P}[\,\mathcal{A}(A_1) \mid \text{ is a } (\upsilon n, p) \text{ expander }] = 0$$

and, therefore, we only have to deal with the remaining two terms of the summation. For $i = 2, 3$, we have

$$\begin{aligned}\mathbb{P}[\,\mathcal{A}(A_i) \mid \text{ is a } (\upsilon n, p) \text{ expander }] &= \frac{\mathbb{P}[\,\mathcal{A}(A_i) \ \wedge\ \mathcal{C}(d_v) \text{ is a } (\upsilon n, p) \text{ expander }]}{\mathbb{P}[\mathcal{C}(d_v) \text{ is a } (\upsilon n, p) \text{ expander}]} \\ &\leq \frac{\mathbb{P}[\,\mathcal{A}(A_i)\,]}{\mathbb{P}[\mathcal{C}(d_v) \text{ is a } (\upsilon n, p) \text{ expander}]} \leq 2\mathbb{P}[\,\mathcal{A}(A_i)\,] \quad \text{(By Lemma 1)}\end{aligned}$$

$$\text{So, putting everything together,} \qquad \mathbb{P}[\text{ LP fails } \mid \mathcal{C}(d_v) \text{ is a } (\upsilon n, p) \text{ expander }] \leq 2\sum_{i=2}^{3} \mathbb{P}[\,\mathcal{A}(A_i)\,]. \tag{8}$$

4.2 Simplifying the probability model

In an attempt to *decorrelate* the requests of F^c from the size of $N(F)$, observe that the number of requests from each bit in F^c is a linear function of the number of edges that this bit has in $N(F)$. This observation through an easy coupling argument shows that, if $x, x' \in \{0, \ldots, d_v\}^{|F^c|}$ are two vectors of requests from the bits in F^c, where $x \leq x'$ elementwise, then the probability that a (p, q)-matching exists is larger conditioned on x than on x'.

This suggests the following alternative experiment. Suppose that each node $j \in F^c$ picks a random number $Z_j \in \{0, 1, \ldots, d_v\}$ according to the *modified* binomial distribution $\mathrm{Bin}\left(d_v, \frac{d_v\lceil \alpha n\rceil}{m}\right)$ and then chooses Z_j checks from $N(F)$ *with replacement*. The key distinction is that, since $|N(F)| \leq d_v\lceil \alpha n\rceil$, the bits of set F^c will tend to have more edges in $N(F)$ and, therefore, more requests in this new experiment than in the original one, as suggested by the natural coupling between the two processes. Moreover, since checks are now chosen with replacement, for each bit $j \in F^c$, the size of the intersection $N(j) \cap N(F)$ is less than or equal to Z_j in size, since the same check might be chosen more than once. Intuitively, the existence of matchings is less likely in the new experiment than in the original one and this can be verifi ed by combining these observations with the coupling argument used in the previous paragraph. The benefi t from switching from the original experiment to this new experiment is in allowing us to *decouple* the process of deciding the number of requests made by each bit in F^c from the cardinality of the random variable $N(F)$.

Let us use $\mathbb{Q}$ to denote the probability distribution over random graphs in this new model. Setting $F^c(q) = \{i \in F \mid q > d_v - Z_j\}$, we can defi ne the alternative "bad event"

$$\mathcal{B} \ := \ \left\{ \exists S_1 \subseteq F,\ S_2 \subseteq F^c(q) \ \middle|\ |N(S_1) \cup [N(S_2) \cap N(F)]| \leq p|S_1| + \sum_{j \in S_2} [q - (d_v - Z_j)] \right\} \tag{9}$$

and the corresponding sub-events $\mathcal{B}(A_i)$, $i = 1, 2, 3$. As argued above it must hold that $\mathbb{P}[\mathcal{B}(A_i)] \leq \mathbb{Q}[\mathcal{B}(A_i)]$, for all i, and, therefore, as inequality (8) suggests, in order to upper bound the probability of LP decoding failure, it suffi ces to obtain upper bounds on the probabilities $\mathbb{Q}[\mathcal{B}(A_i)]$ for $i = 2, 3, 4$. For future use, we defi ne for fi xed subsets $S_1 \subseteq F$ and $S_2 \subseteq F^c(q)$, the event

$$\mathcal{B}(S_1, S_2) \ := \ \left\{ |N(S_1) \cup [N(S_2) \cap N(F)]| \leq p|S_1| + \sum_{v \in S_2} [q - (d_v - Z_j)] \right\}. \tag{10}$$

We now proceed, in a series of steps, to obtain suitable upper bounds on the probabilities $\mathbb{Q}[\mathcal{B}(A_i)]$ and, hence, on the probability of LP decoding failure.

4.3 Conditioning on requests from F^c

For each $i \in \{1, \ldots, q\}$, we define the random variable $R_i := \Big|\{j \in F^c \mid Z_j = d_v - q + i\}\Big|$, corresponding to the number of bits in F^c with $d_v - q + i$ edges that lie inside the "contaminated" neighborhood $N(F)$. So if we define, for each i, the probability $q_i := \binom{d_v}{d_v - q + i} \left(\frac{\lceil \alpha n \rceil d_v}{m}\right)^{d_v - q + i} \left(1 - \frac{\lceil \alpha n \rceil d_v}{m}\right)^{q-i}$, then each R_i is binomial with parameters q_i and $\lfloor (1-\alpha)n \rfloor$. Since $\mathbb{E}[R_i] = q_i \lfloor (1-\alpha)\, n \rfloor$, applying Hoeffding's inequality [15] yields the sharp concentration $\mathbb{Q}\left[|R_i - q_i \lfloor (1-\alpha)\, n \rfloor| \geq \epsilon_1 n\right] \;\leq\; 2\exp\left(-2\epsilon_1^2 n\right)$ for any $\epsilon_1 > 0$. Hence, if we define the event

$$\mathcal{T}(\epsilon_1) := \bigcap_{i=1}^{q} \left\{|R_i - q_i \lfloor (1-\alpha)n \rfloor| \leq \epsilon_1 n\right\},$$

then a simple union bound yields that $\mathbb{Q}[\mathcal{T}(\epsilon_1)] \;\leq\; 1 - 2q \;\exp\left(-2\epsilon_1^2 n\right)$, so that *it suffices to bound the conditional probabilities* $\mathbb{Q}[\mathcal{B}(A_i) \mid \mathcal{T}(\epsilon_1)]$, $i = 2, 3$. Note that conditioned on $\mathcal{T}(\epsilon_1)$, we are guaranteed that

$$\frac{R_i}{n} \;\;\leq\;\; q_i(1-\alpha) + \epsilon_1 \;=:\; \bar{R}_i^{\mathrm{up}}. \tag{11}$$

4.4 Bounding $\mathbb{Q}[\mathcal{B}(A_2) \mid \mathcal{T}(\epsilon_1)]]$

We now turn to bounding the probability of the bad event $\mathcal{B}$. Since, by symmetry, the probability of the event $\mathcal{B}(S_1, S_2)$ is the same for different sets S_1 of the same size, a union bound gives $\mathbb{Q}[\mathcal{B}(A_2) \mid \mathcal{T}(\epsilon_1)] \leq \sum_{s_1 = \lceil \epsilon_2 n \rceil}^{\lceil \alpha\, n \rceil} D(s_1)$, where

$$D(s_1) \;\;:=\;\; \binom{\lceil \alpha n \rceil}{s_1} \mathbb{Q}[\exists\, S_2 \subseteq F^c(q) \text{ with } (S_1, S_2) \in A_2 \text{ s.t. } \mathcal{B}(S_1, S_2) \mid \mathcal{T}(\epsilon_1), \text{fixed set } |S_1| = s_1].$$

Before bounding these terms, we first partition the values of s_1 into two sets $\{\lceil \epsilon_2 n \rceil, \ldots, \lceil \bar{s}_{\mathrm{crit}} n \rceil\}$ and $\{\lceil \bar{s}_{\mathrm{crit}} n \rceil + 1, \ldots, \lceil \alpha n \rceil\}$ for some value of $\bar{s}_{\mathrm{crit}}$ to be specified formally in Lemma 5. To give some intuition, in the conditional space $\mathcal{T}(\epsilon_1)$, the total number of matching-requests from the bits of set F^c is at most $V := n \sum_{i=1}^{q} i \bar{R}_i^{\mathrm{up}} =: n\bar{V}$. Therefore, if $\mathbb{Q}[\mathcal{B}(A_2) \mid \mathcal{T}(\epsilon_1)]]$ is relatively small, we would expect that, if the set S_1 is large enough (say $|S_1| \approx |F|$), then with high probability, the size of its image $N(S_1)$ should be large enough not only to cover its own requests but also V additional requests—viz. $|N(S_1)| \geq p|S_1| + V$. If this condition holds, then there cannot exists any set S_2 such that the event $\mathcal{B}(S_1, S_2)$ occurs. We formalize this intuition in the following:

Lemma 5 (Upper Regime)**.** *Define the constant* $\bar{V} := \sum_{i=1}^{q} i\, \bar{R}_i^{\mathrm{up}}$*, and the function* $f(s) := \alpha H\left(\frac{s}{\alpha}\right) + (1 - \tilde{r}) H\left(\frac{ps + \bar{V}}{(1-\tilde{r})}\right) + d_v s \log_2\left(\frac{ps + \bar{V}}{(1-\tilde{r})}\right)$*, where* $H(\cdot)$ *is the binary entropy (see Appendix D), and set*

$$\bar{s}_{\mathrm{crit}} := \begin{cases} \inf\left\{s \in [0, \alpha] | f(s') < 0, \forall s' \in [s, \alpha]\right\}, & \textit{if infimum exists} \\ \alpha, & \textit{otherwise} \end{cases}$$

Then for all $s_1 \in \{\lceil \bar{s}_{\mathrm{crit}} n \rceil + 1, \ldots, \lceil \alpha n \rceil\}$*, the quantity* $D(s_1)$ *decays exponentially fast in* n*.*

It remains to bound $D(s_1)$ for $s_1 \in \{\lceil \epsilon_2 n \rceil, \ldots, \lceil \bar{s}_{\mathrm{crit}} n \rceil\} := L_I$. Consider a fixed set S_1 of size $s_1 \in L_I$, and check neighborhood $N(S_1)$ of size $\gamma_1 := |N(S_1)|$. By conditioning, we have the decomposition $D(s_1) =$

$\sum_{\gamma_1=1}^{d_v s_1} E(\gamma_1, s_1)$, where

$$E(\gamma_1, s_1) := \binom{\lceil \alpha n \rceil}{s_1} \mathbb{Q}' \left[\exists\, S_2 \text{ with } (S_1, S_2) \in A_2 \text{ s.t. } \mathcal{B}(S_1, S_2) \mid |N(S_1)| = \gamma_1, |S_1| = s_1 \right] \times \mathbb{Q}' \left[|N(S_1)| = \gamma_1 \mid\mid S_1| = s_1 \right].$$

Here we have used $\mathbb{Q}'$ to denote the conditional probability distribution of $\mathbb{Q}$ conditioned on the event $\mathcal{T}(\epsilon_1)$. The following lemma allows us to restrict our attention to linearly-sized check neighborhoods $N(S_1)$ in analyzing the individual terms $E(\gamma_1, s_1)$ of the summation:

Lemma 6 (Linear Sized Neighborhood)**.** *Define*

$$\bar{\gamma}_{\text{crit}}(\bar{s}_1) \;\; = \;\; \sup \left\{ \bar{\gamma}_1 \in (0, d_v \bar{s}_1] \;\mid\; 2 + d_v \bar{s}_1 \log_2 \left(\frac{\bar{\gamma}_1}{(1-\tilde{r})} \right) < 0 \right\},$$

where note that the supremum always exists. Then, for set sizes $s_1 \geq \lceil \epsilon_2 n \rceil$ and neighborhood sizes $\gamma_1 \leq \bar{\gamma}_{\text{crit}}(\epsilon_2) n$, the quantity $E(\gamma_1, s_1)$ decays exponentially fast in n.

A summary and some intuition: To summarize our progress thus far, we first argued that in order to bound the probability $\mathbb{Q}[\mathcal{B}(A_2) \mid \mathcal{T}(\epsilon_1)]]$, it suffices to bound the quantities $D(s_1)$, for $s_1 \in \{\lceil \epsilon_2 n \rceil, \ldots, \lceil \alpha n \rceil\}$. Next we partitioned the range of s_1 into two sets: the lower set $L_I = \{\lceil \epsilon_2 n \rceil, \ldots, \lceil \bar{s}_{\text{crit}} n \rceil\}$, and the upper set $U_I := \{\lceil \bar{s}_{\text{crit}} n \rceil + 1, \ldots, \lceil \alpha n \rceil\}$. The upper set has the property that for all sets $S_1 \subseteq F$ of size $|S_1| \in U_I$, then with high probability, the neighborhood $N(S_1)$ is big enough to accommodate not only the matching requests from set S_1, but also all possible matching-requests from any set $S_2 \subseteq F^c$. Having established this property of large S_1 sets, it remains to focus on small S_1. In this regime, the neighborhood $N(S_1)$ on its own is no longer sufficient to cover the joint set of requests from S_1 *and* from any possible set $S_2 \subseteq F^c$. Consequently, one has to consider for every choice $(S_1, S_2) \in A_2$, whether the joint neighborhood $N(S_1) \cup (N(S_2) \cap N(F))$ is large enough to cover the matching requests from S_1 and S_2.

At this point, one might imagine that a rough concentration argument applied to the sizes of $N(S_1)$ and $N(S_2) \cap N(F) - N(S_1)$ would suffice to complete the proof. Unfortunately, any concentration result must be sufficiently strong to dominate the factor $\binom{\lceil \alpha n \rceil}{s_1}$ that leads the expression $D(s_1)$. Consequently, we study the exact distribution of the size of $N(S_1)$, and bound the quantities $E(\gamma_1, s_1)$ for $s_1 \in L_I$ and $\gamma_1 \in \{1, \ldots, d_v s_1\}$. Of course, since s_1 is linear in size, the bulk of the probability mass is concentrated on linear values for γ_1. Therefore, by Lemma 6, we need only bound $E(\gamma_1, s_1)$ for $s_1 \in L_I$ and $\gamma_1 \geq \bar{\gamma}_{\text{crit}}(\epsilon_2) n$. We complete these steps in the following subsection.

Establishing the bound: Let us fix sizes $s_1 \in L_I$ and $\gamma_1 \geq \bar{\gamma}_{\text{crit}}(\epsilon_2) n$. For a set S_1 of size s_1 with neighborhood $N(S_1)$ of size γ_1, define its *residual neighborhood* to be the set $N(F) \backslash N(S_1)$ and use $\gamma_2 := |N(F) \backslash N(S_1)|$ to denote its size. Moreover, for a configuration of requests[2] $r \in \prod_{i=1}^{q} \{0, .., \lceil \bar{R}_i^{\text{up}} n \rceil\}$, let us denote by $\beta(s_1, \gamma_1, r)$ the number of checks missing from the neighborhood of S_1 to cover the total number of requests from S_1 and a set $S_2 \subseteq F^c$ with configuration of requests r. Also, let $\nu(r)$ be the number of edges that the checks of set S_2 have inside $N(F)$. More precisely, the quantities $\beta(s_1, \gamma_1, r)$ and $\nu(r)$ are given by the following formulas: $\beta(s_1, \gamma_1, r) := p s_1 - \gamma_1 + \sum_{i=1}^{q} i r_i$ and $\nu(r) := \sum_{i=1}^{q} (d_v - q + i) r_i$. With these definitions, we have the following exponential upper bound:

[2]Recall that we have conditioned on the event $\mathcal{T}(\epsilon_1)$, so that the number of bits in F^c with i matching requests is concentrated, for every $i \in \{1, \ldots, q\}$.

Lemma 7 (Exponential upper bound)**.** *If* $\bar{s}_{\text{crit}} < \frac{\alpha}{2}$, $\alpha d_v < \frac{(1-\tilde{r})-d_v\bar{s}_{\text{crit}}}{2}$ *and, moreover,* $\alpha H\left(\frac{\bar{s}_{\text{crit}}}{\alpha}\right) + d_v(\alpha - \bar{s}_{\text{crit}})\log_2\left(\frac{d_v\bar{s}_{\text{crit}}}{(1-\tilde{r})}\right) < 0$*, then the probability* $\mathbb{Q}[\mathcal{B}(A_2) \mid \mathcal{T}(\epsilon_1)]]$ *is upper bounded by* $2^{nF(\alpha)} + o(1)$*, where the* $o(1)$ *term is exponentially decreasing in* n*, and the function in the exponent is given by*

$$F(\alpha) := \sup_{\bar{s}_1\in[0,\bar{s}_{\text{crit}}]} \sup_{\bar{\gamma}_1\in[0,d_v\bar{s}_1]} \sup_{\bar{\gamma}_2\in[0,\, d_v\,(\alpha-\bar{s}_1)]} \sup_{\bar{r}_i\in[\bar{R}_i^{\text{up}}/2,\bar{R}_i^{\text{up}}]} G(\bar{s}_1,\bar{\gamma}_1,\bar{\gamma}_2,\bar{r}_1,\ldots,\bar{r}_q),$$

where the intermediate function $G = G(\bar{s}_1,\bar{\gamma}_1,\bar{\gamma}_2,\bar{r}_1,\ldots,\bar{r}_q)$ *is*

$$\begin{aligned}\alpha H\left(\frac{\bar{s}_1}{\alpha}\right) + \sum_{i=1}^{q}\bar{R}_i^{\text{up}}H\left(\frac{\bar{r}_i}{\bar{R}_i^{\text{up}}}\right) + \min\left\{0,(1-\tilde{r})H\left(\frac{\bar{\gamma}_1}{(1-\tilde{r})}\right) + d_v\bar{s}_1\log_2\left(\frac{\bar{\gamma}_1}{(1-\tilde{r})}\right)\right\} + \\ + \min\left\{0,((1-\tilde{r})-\bar{\gamma}_1)H\left(\frac{\bar{\gamma}_2}{((1-\tilde{r})-\bar{\gamma}_1)}\right) + d_v(\alpha-\bar{s}_1)\log_2\left(\frac{\bar{\gamma}_1+\bar{\gamma}_2}{(1-\tilde{r})}\right)\right\} \\ + \min\left\{0,\bar{\gamma}_2 H\left(\frac{\min\{\bar{\gamma}_2,\bar{\beta}(\bar{s}_1,\bar{\gamma}_1,\bar{r})\}}{\bar{\gamma}_2}\right) + \nu(\bar{r})\log_2\left(\frac{\bar{\gamma}_1+\min\{\bar{\gamma}_2,\bar{\beta}(\bar{s}_1,\bar{\gamma}_1,\bar{r})\}}{\bar{\gamma}_1+\bar{\gamma}_2}\right)\right\}\end{aligned}$$

See Appendix G for a proof of this lemma.

4.5 Bounding $\mathbb{Q}[\mathcal{B}(A_3) \mid \mathcal{T}(\epsilon_1)]]$ and combining the pieces

It remains to upper bound the probability of the bad-event $\mathcal{B}(A_3)$ which is equivalent to the existence of a pair of contracting sets (S_1, S_2), where the size of set $S_1 \subseteq F$ is at most $\epsilon_2 n$ and the size of set $S_2 \subseteq F^c$ is at least $(\rho - \epsilon_2)n$. Note that we haven't yet specified the constant ϵ_2. The following lemma establishes that there exists a value of ϵ_2 so that $\mathbb{Q}[\mathcal{B}(A_3) \mid \mathcal{T}(\epsilon_1)]]$ is bounded by an exponentially decreasing function in n provided that the function $F(\alpha)$ of the previous section is negative. The proof of this final lemma is provided in Appendix H.

Lemma 8. *If* $F(\alpha) < 0$*, where* $F(\cdot)$ *is the function defined in Lemma 7, then there exists* ϵ_2 *so that the probability* $\mathbb{Q}[\mathcal{B}(A_3) \mid \mathcal{T}(\epsilon_1)]]$ *is decreasing exponentially in* n*.*

Combining Inequality (8), Lemma 8 and the analysis of Section 4.4, we get

Lemma 9. *Fix constants* $\tilde{r}$*,* d_v*,* p *and* q *such that* $2p + q > 2d_v$ *and define* $\bar{s}_{\text{crit}}$ *as in the statement of Lemma 5. Then, if* $\bar{s}_{\text{crit}} < \frac{\alpha}{2}$, $\alpha d_v < \frac{(1-\tilde{r})-d_v\bar{s}_{\text{crit}}}{2}$, $\alpha H\left(\frac{\bar{s}_{\text{crit}}}{\alpha}\right) + d_v(\alpha - \bar{s}_{\text{crit}})\log_2\left(\frac{d_v\bar{s}_{\text{crit}}}{(1-\tilde{r})}\right) < 0$ *and, moreover, the function* $F(a)$*, defined in the statement of Lemma 7, is strictly negative, then*

$$\mathbb{P}[\text{ LP decoding fails} \mid \mathcal{C}(d_v) \text{ is a } (\upsilon n, p) \text{ expander }]$$

decays exponentially in n*, where* $\mathbb{P}$ *is the uniform measure over the set of bit-regular codes and selections of* $\lceil \alpha n \rceil$ *bit flips, and* υ *is given by Equation (2).*

Using Lemma 9, we can investigate fractions of correctable errors on specific code ensembles. As a concrete example, for code rate $\tilde{r} = 1/2$, if we choose variable degrees $d_v = 8$ and generalized matching parameters $(p, q) = (6, 5)$, one can numerically verify that the conditions of Lemma 9 are satisfied for all $\alpha \leq \alpha_{\text{crit}} = 0.002$. Therefore, for that rate, we establish that the correctable fraction or error that is more than ten times higher than previously known results, as claimed. More generally, it remains to further explore the consequences of our analysis technique for other rates and code ensembles.

References

[1] A. Barg and G. Zémor. Error exponents of expander codes under linear-complexity decoding. *SIAM Journal on Discrete Math*, 17(3):426–445, 2004.

[2] E. Berlekamp, R. McEliece, and H. van Tilborg. On the inherent intractability of certain coding problems. *IEEE Trans. Info. Theory*, pages 384–386, 1978.

[3] D. Bienstock. *Potential Function Methods for Approximately Solving Linear Programming Problems*. Kluwer Academic, Boston, 2002.

[4] S.Y. Chung, T. Richardson, and R. Urbanke. Analysis of sum-product decoding of low-density parity check codes using a Gaussian approximation. *IEEE Trans. Info. Theory*, 47:657–670, February 2001.

[5] T.M. Cover and J.A. Thomas. *Elements of Information Theory*. John Wiley and Sons, New York, 1991.

[6] A. G. Dimakis and M. J. Wainwright. Guessing Facets: Improved LP decoding and Polytope Structure. In *International Symposium on Information Theory*, Seattle, Washington, July 2006.

[7] J. Feldman and D. Karger. Decoding turbo-like codes in polynomial time with provably good error-correcting performing via linear programming. In *FOCS*, July 2002.

[8] J. Feldman, D. R. Karger, and M. J. Wainwright. Linear programming-based decoding of turbo-like codes and its relation to iterative approaches. In *Proc. 40th Annual Allerton Conf. on Communication, Control, and Computing*, October 2002.

[9] J. Feldman, T. Malkin, R. A. Servedio, C. Stein, and M. J. Wainwright. LP decoding corrects a constant fraction of errors. Technical Report CORC Technical Report TR-2003-08, Operations Research, Columbia University, December 2003.

[10] J. Feldman, T. Malkin, R. A. Servedio, C. Stein, and M. J. Wainwright. LP decoding corrects a constant fraction of errors. In *Proc. IEEE International Symposium on Information Theory*, 2004.

[11] J. Feldman and C. Stein. LP decoding achieves capacity. In *Symposium on Discrete Algorithms (SODA '05)*, January 2005.

[12] J. Feldman, M. J. Wainwright, and D. R. Karger. Using linear programming to decode binary linear codes. *IEEE Transactions on Information Theory*, 51:954–972, March 2005.

[13] R. G. Gallager. *Low-density parity check codes*. MIT Press, Cambridge, MA, 1963.

[14] N. Halabi and G. Even. Improved bounds on the word error probability of RA(2) codes with linear-programming-based decoding. *IEEE Trans. Info. Theory*, 51:265–280, January 2005.

[15] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58:13–30, 1963.

[16] R. Koetter and P. O. Vontobel. Graph-covers and iterative decoding of finite length codes. In *Proc. 3rd International Symp. on Turbo Codes*, September 2003.

[17] F.R. Kschischang, B.J. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Trans. Info. Theory*, 47(2):498–519, February 2001.

[18] M. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. Spielman. Improved low-density parity check codes using irregular graphs. *IEEE Trans. Info. Theory*, 47:585–598, February 2001.

[19] D. MacKay. Good error correcting codes based on very sparse matrices. *IEEE Trans. Info. Theory*, 45(2):399–431, 1999.

[20] T. Richardson and R. Urbanke. The capacity of low-density parity check codes under message-passing decoding. *IEEE Trans. Info. Theory*, 47:599–618, February 2001.

[21] Michael Sipser and Daniel A. Spielman. Expander codes. In *IEEE Symposium on Foundations of Computer Science*, pages 566–576, 1994.

[22] P. O. Vontobel and R. Koetter. Bounds on the threshold of linear programming decoding. In *Information Theory Workshop*, March 2006.

[23] P.O. Vontobel and R. Koetter. "Towards low-complexity linear-programming decoding". In *Proc. of the 4th Int. Symp. on Turbo Codes and Related Topics*, Munich,Germany, April 2006.

[24] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. Exact MAP estimates via agreement on (hyper)trees: Linear programming and message-passing. *IEEE Trans. Information Theory*, 51(11):3697–3717, November 2005.

A Illustration of factor graph

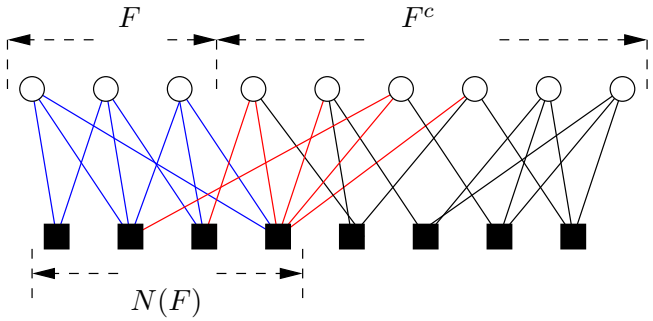

Figure 1. Illustration of the structure of a generalized matching. The subset $F \subseteq V$ corresponds to the set of bits $i \in V$ with negative log-likelihoods ($\gamma_i < 0$), and F^c denotes its complement. The set $N(F)$ corresponds to checks that are connected to flipped bits; a generalized matching requires that this set has sufficient connectivity to the unflipped set F^c.

B Inequality description of relaxed polytope

Here we give a precise description of the inequalities that characterize the relaxed polytope $\mathcal{P}$. For every check a connected to variables $N(a)$ and for all subsets $S \subseteq N(a)$, $|S|$ odd, we introduce the following constraints

$$\sum_{i \in N(a) \setminus S} y_i + \sum_{i \in S} (1 - y_i) \geq 1. \tag{12}$$

It can be shown that by constraining the ℓ_1 distance to be one, we are not excluding any legal codewords from our relaxed polytope. We will call these inequalities *forbidding inequalities*.

We also need to add $2n$ inequalities $0 \leq y_i \leq 1$ to ensure that we remain inside the unit hypercube. The set of forbidding inequalities along with these $[0,1]$-box inequalities define the relaxed polytope, Given a check of degree d_c, there are 2^{d_c-1} local forbidden sequences; for a constant check degree code then, the total number of forbidden sequences would be $2^{d_c-1}m$. Fortunately, in the case of low-density parity-check codes, d_c is either a fixed constant (for regular) or small with high probability (for irregular) so the number of local forbidden sequences remains small. Therefore, in the cases of practical interest, the relaxed polytope can be described by a linear number of inequalities. Finally, it can be shown that if the LDPC graph had no cycles, the local forbidden sequences would identify all the possible non-codewords and the relaxation would be exact [24, 12]. However if the graph has cycles, there exist vertices of the relaxed polytope (called pseudocodewords) with non-integral coordinates that satisfy all the local constraints individually and yet are not codewords nor linear combinations of codewords.

C Partitioning the space

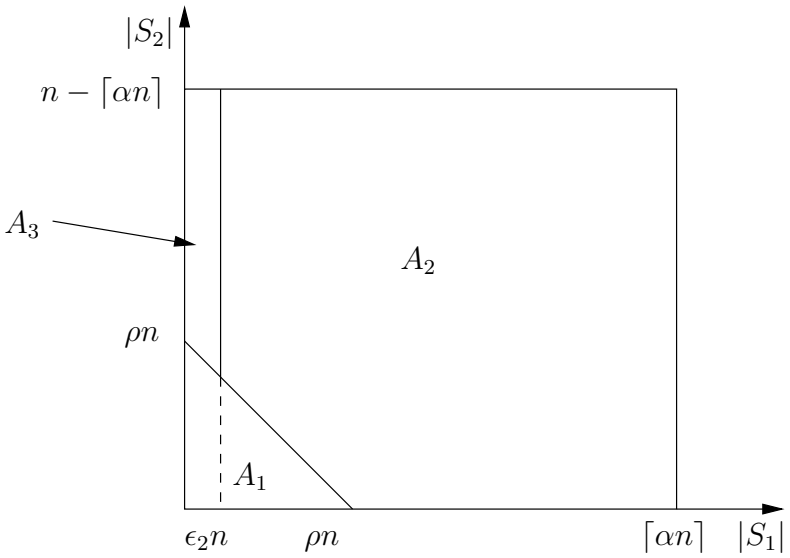

Figure 2: Partitioning the space $2^F \times 2^{F^c}$.

D Elementary bounds on binomial coefficients

For each $\beta \in (0,1)$, define the binomial entropy $H(\beta) := -\beta \log_2 \beta - (1-\beta)\log_2(1-\beta)$ (and $H(0) = H(1) = 0$ by continuity). We make use of the following standard bounds [5] on the binomial coefficients

$$n\left[H\left(\frac{k}{n}\right) - \frac{\log_2(n+1)}{n}\right] \;\leq\; \log_2 \binom{n}{k} \;\leq\; n\left[H\left(\frac{k}{n}\right) + \frac{\log_2(n+1)}{n}\right]. \tag{13}$$

E Proof of Lemma 5

Note that conditioned on the event $\mathcal{T}(\epsilon_1)$, we are guaranteed that $\sum_{i=1}^{q} iR_i \leq \bar{V}n$. Now by union bound, we have

$$\begin{aligned} Y(s_1) &:= \binom{\lceil \alpha n \rceil}{s_1} \mathbb{Q}\left[\text{some fixed } S_1 \text{ of size } s_1 \text{ satisfies } |N(S_1)| < ps_1 + \bar{V}n\right] \\ &\leq \binom{\lceil \alpha n \rceil}{s_1} \binom{\lfloor (1-\tilde{r})n \rfloor}{\lfloor ps_1 + \bar{V}n \rfloor} \left(\frac{\lfloor ps_1 + \bar{V}n \rfloor}{\lfloor (1-\tilde{r})n \rfloor}\right)^{d_v s_1}. \end{aligned}$$

Setting $\bar{s}_1 = \frac{s_1}{n}$ and using standard bounds on binomial coefficients (see Appendix D), the log of $Y(s_1)$ is upper bounded by

$$n\left[\alpha H\left(\frac{\bar{s}_1}{\alpha}\right) + (1-\tilde{r})H\left(\frac{p\bar{s}_1 + \bar{V}}{1-\tilde{r}}\right) + d_v \bar{s}_1 \log_2 \frac{(p\bar{s}_1 + \bar{V})}{(1-\tilde{r})} + o(1)\right].$$

Defining the function f and value $\bar{s}_{\text{crit}}$ as in the lemma statement, we are guaranteed that $Y(s_1)$ decays exponentially in n for all $s_1 \in \{\lceil \bar{s}_{\text{crit}} n \rceil + 1, \ldots, \lceil \alpha n \rceil\}$. To complete the proof of the claim, we write for $s_1 \in \{\lceil \bar{s}_{\text{crit}} n \rceil + 1, \ldots, \lceil \alpha n \rceil\}$

$$\begin{aligned} D(s_1) &:= \binom{\lceil \alpha n \rceil}{s_1} \mathbb{Q}[\exists\, S_2 \subseteq F^c(q) \text{ with } (S_1, S_2) \in A_2 \text{ s.t. } \mathcal{B}(S_1, S_2) \mid \mathcal{T}(\epsilon_1), S_1 \text{ some fixed set of size } s_1] \\ &\leq \binom{\lceil \alpha n \rceil}{s_1} \mathbb{Q}[\exists\, S_2 \subseteq F^c(q) \text{ with } (S_1, S_2) \in A_2 \text{ s.t. } \mathcal{B}(S_1, S_2) \mid \mathcal{T}(\epsilon_1), |N(S_1)| > ps_1 + \bar{V}n] \;+\; Y(s_1) \\ &= Y(s_1), \end{aligned}$$

because, as argued in Section 4.3, in the conditional space $|N(S_1)| > p|S_1| + \bar{V}n$, there can be no S_2 such that the event $\mathcal{B}(S_1, S_2)$ holds.

F Proof of Lemma 6

We have the bound $E(\gamma_1, s_1) \leq \binom{\lceil \alpha n \rceil}{s_1} \mathbb{Q}[\, |N(S_1)| = \gamma_1 \mid |S_1| = s_1]$, where we have used the fact that the event $\{|N(S_1)| = \gamma_1\}$ is independent of $\mathcal{T}(\epsilon_1)$ under the probability measure $\mathbb{Q}$. An exact computation yields

$$\begin{aligned} \log_2 \{\mathbb{Q}[|N(S_1)| = \gamma_1 \mid |S_1| = s_1]\} &\leq \log_2 \left\{ \binom{\lfloor (1-\tilde{r})n \rfloor}{\gamma_1} \left(\frac{\gamma_1}{\lfloor (1-\tilde{r})n \rfloor}\right)^{d_v s_1} \right\} \\ &\leq n \left\{ \frac{\log_2(n+1)}{n} + (1-\tilde{r}) H\left(\frac{\gamma_1}{\lfloor (1-\tilde{r})n \rfloor}\right) + d_v \frac{s_1}{n} \log_2 \left(\frac{\gamma_1}{\lfloor (1-\tilde{r})n \rfloor}\right) \right\}, \end{aligned}$$

where we have used standard bounds on binomial coefficients (see Appendix D). Overall, we have

$$\begin{aligned} \log_2 E(\gamma_1, s_1) &\leq n \left\{ 2\frac{\log_2(n+1)}{n} + H\left(\frac{s_1}{\lceil \alpha n \rceil}\right) + H\left(\frac{\gamma_1}{\lfloor (1-\tilde{r})n \rfloor}\right) + d_v \frac{s_1}{n} \log_2 \left(\frac{\gamma_1}{\lfloor (1-\tilde{r})n \rfloor}\right) \right\} \\ &\leq n \left\{ 2\frac{\log_2(n+1)}{n} + 2 + d_v \frac{s_1}{n} \log_2 \left(\frac{\gamma_1}{\lfloor (1-\tilde{r})n \rfloor}\right) \right\}, \end{aligned}$$

since $\alpha < 1$, $\tilde{r} < 1$ (first line) and each entropy term remains bounded within $[0, 1]$ (second line).

Finally, setting $\bar{s}_1 = s_1/n$, $\bar{\gamma}_1 = \gamma_1/n$, consider the function

$$g(\bar{\gamma}_1) \;:=\; 2 + d_v \bar{s}_1 \log_2 \left(\frac{\bar{\gamma}_1}{(1-\tilde{r})} \right).$$

We have $\lim_{\bar{\gamma}_1 \to 0^+} g(\bar{\gamma}_1) = -\infty$, implying that $E(\gamma_1, s_1)$ decays exponentially fast in n for all $s_1 \geq \lceil \epsilon_2 n \rceil$ and neighborhood sizes $\gamma_1 \leq \bar{\gamma}_{\text{crit}}(\epsilon_2) n$, where $\bar{\gamma}_{\text{crit}}(\cdot)$ is defi ned as in the statement of the lemma.

G Proof of Lemma 7

We begin by proving the following lemma, which provides an upper bound on the quantity $E(\gamma_1, s_1)$.

Lemma 10 (Lower Regime)**.** *If $\bar{s}_{\text{crit}} < \frac{\alpha}{2}$, $\alpha d_v < \frac{(1-\tilde{r}) - d_v \bar{s}_{\text{crit}}}{2}$ and, moreover,*

$$\alpha H\left(\frac{\bar{s}_{\text{crit}}}{\alpha} \right) + d_v(\alpha - \bar{s}_{\text{crit}}) \log_2 \left(\frac{d_v \bar{s}_{\text{crit}}}{(1-\tilde{r})} \right) < 0,$$

then, for all $s_1 \in \{\lceil \epsilon_2 n \rceil, \ldots, \lceil \bar{s}_{\text{crit}} n \rceil\}$ and $\gamma_1 \geq \bar{\gamma}_{\text{crit}}(\epsilon_2) n$, there exists some $\gamma_2^* = \gamma_2^*(\bar{s}_{\text{crit}}, \epsilon_2) > 0$ such that $E(s_1, \gamma_1)$ is upper bounded by

$$\begin{aligned}
&\text{poly}(n) \binom{\alpha n}{s_1} \cdot \min\left\{ 1, \binom{(1-\tilde{r})n}{\gamma_1} \left(\frac{\gamma_1}{(1-\tilde{r})n} \right)^{d_v s_1} \right\} \cdot \\
&\max_{\gamma_2 \in \{\lceil \gamma_2^* n \rceil, \ldots, d_v(\alpha n - s_1)\}} \; \max_{r_i \in \mathcal{R}_i} \Bigg[\binom{\lceil \bar{R}_1^{\text{up}} n \rceil}{r_1} \cdots \binom{\lceil \bar{R}_q^{\text{up}} n \rceil}{r_q} \cdot \\
&\qquad \cdot \min\left\{ 1, \binom{(1-\tilde{r})n - \gamma_1}{\gamma_2} \left(\frac{\gamma_2 + \gamma_1}{(1-\tilde{r})n} \right)^{(\alpha n - s_1) d_v} \right\} \cdot \\
&\qquad \cdot \min\left\{ 1, \binom{\gamma_2}{\min\{\beta(s_1, \gamma_1, r), \gamma_2\}} \left(\frac{\gamma_1 + \min\{\gamma_2, \beta(s_1, \gamma_1, r)\}}{\gamma_1 + \gamma_2} \right)^{\nu(r)} \right\} \Bigg] + o(1), \qquad (14)
\end{aligned}$$

where $\mathcal{R}_i := \left\{ \left\lfloor \frac{\bar{R}_i^{\text{up}} n}{2} \right\rfloor, \ldots, \lceil \bar{R}_i^{\text{up}} n \rceil \right\}$, for all i, and the $o(1)$ is an exponentially in n decreasing function.

Proof. We begin with the decomposition

$$E(s_1, \gamma_1) \;=\; \binom{\lceil \alpha n \rceil}{s_1} \sum_{\gamma_2 = 1}^{d_v \lceil \alpha n \rceil - s_1} U_1(\gamma_1, \gamma_2) U_2(\gamma_1, \gamma_2) \qquad (15)$$

where

$$\begin{aligned}
U_1(\gamma_1, \gamma_2) \;&:=\; \mathbb{Q}' \left[\exists\, S_2 \text{ with } (S_1, S_2) \in A_2 \text{ s.t. } \mathcal{B}(S_1, S_2) \mid |N(S_1)| = \gamma_1,\ |N(F) \backslash N(S_1)| = \gamma_2, |S_1| = s_1 \right] \\
U_2(\gamma_1, \gamma_2) \;&:=\; \mathbb{Q}' \left[|N(S_1)| = \gamma_1,\ |N(F) \backslash N(S_1)| = \gamma_2 \;\mid |S_1| = s_1 \right],
\end{aligned}$$

and recall that $\mathbb{Q}'$ is the measure $\mathbb{Q}$ conditioned on the event $\mathcal{T}(\epsilon_1)$. We now require a lemma that allows us to restrict appropriately the range of summation over linear in n values of γ_2.

Lemma 11. *The conditions of Lemma 10 imply that there exists some value $\gamma_2^* = \gamma_2^*(\bar{s}_{\text{crit}}) > 0$ for which the quantity*

$$G(s_1, \gamma_1) := \binom{\lceil \alpha n \rceil}{s_1} \sum_{\gamma_2=1}^{\lfloor \gamma_2^* n \rfloor} U_1(\gamma_1, \gamma_2) U_2(\gamma_1, \gamma_2)$$

decays exponentially in n for any s_1 , γ_1 that satisfy $\lceil \epsilon_2 n \rceil \leq s_1 \leq \lceil \bar{s}_{\text{crit}} n \rceil$ and $\gamma_1 \geq \bar{\gamma}_{\text{crit}}(\epsilon_2) n$.

Proof. The proof is similar in spirit to the proof of Lemma 6. Take a term of summation (15). We can bound it as follows:

$$\begin{aligned} B(s_1, \gamma_1, \gamma_2) :&= \binom{\lceil \alpha n \rceil}{s_1} U_1(\gamma_1, \gamma_2) U_2(\gamma_1, \gamma_2) \\ &\leq \binom{\lceil \alpha n \rceil}{s_1} U_2(\gamma_1, \gamma_2) \\ &\leq \binom{\lceil \alpha n \rceil}{s_1} \mathbb{Q}' \left[|N(F) \backslash N(S_1)| = \gamma_2 \mid |N(S_1)| = \gamma_1, |S_1| = s_1 \right] \end{aligned}$$

Note that

$$\mathbb{Q}' \left[|N(F) \backslash N(S_1)| = \gamma_2 \mid |N(S_1)| = \gamma_1, |S_1| = s_1 \right] \leq \binom{\lfloor (1-\tilde{r}) n \rfloor - \gamma_1}{\gamma_2} \left(\frac{\gamma_2 + \gamma_1}{\lfloor (1-\tilde{r}) n \rfloor} \right)^{(\lceil \alpha n \rceil - s_1) d_v}$$

Therefore,

$$\begin{aligned} &\log_2 B(s_1, \gamma_1, \gamma_2) \leq \\ &\quad \leq n \left\{ \alpha H\left(\frac{s_1/n}{\alpha} \right) + H\left(\frac{\gamma_2/n}{(1-\tilde{r}) - \gamma_1/n} \right) + d_v \left(\alpha - \frac{s_1}{n} \right) \log_2 \left(\frac{\gamma_2/n + \gamma_1/n}{(1-\tilde{r})} \right) + o(1) \right\} \\ &\quad \leq n \left\{ \alpha H\left(\frac{\bar{s}_{\text{crit}}}{\alpha} \right) + H\left(\frac{\gamma_2/n}{(1-\tilde{r}) - d_v \bar{s}_{\text{crit}}} \right) + d_v (\alpha - \bar{s}_{\text{crit}}) \log_2 \left(\frac{\gamma_2/n + d_v \bar{s}_{\text{crit}}}{(1-\tilde{r})} \right) + o(1) \right\}. \end{aligned}$$

where we have used that $\bar{s}_{\text{crit}} < \frac{\alpha}{2}$ [3] and $\alpha d_v < \frac{(1-\tilde{r}) - d_v \bar{s}_{\text{crit}}}{2}$ [4]. So, if we defi ne, the function

$$b(\gamma) := \alpha H\left(\frac{\bar{s}_{\text{crit}}}{\alpha} \right) + H\left(\frac{\gamma}{(1-\tilde{r}) - d_v \bar{s}_{\text{crit}}} \right) + d_v (\alpha - \bar{s}_{\text{crit}}) \log_2 \left(\frac{\gamma + d_v \bar{s}_{\text{crit}}}{(1-\tilde{r})} \right),$$

it follows that $\lim_{\gamma \to 0} b(\gamma) < 0$ from the assumption that

$$\alpha H\left(\frac{\bar{s}_{\text{crit}}}{\alpha} \right) + d_v (\alpha - \bar{s}_{\text{crit}}) \log_2 \left(\frac{d_v \bar{s}_{\text{crit}}}{(1-\tilde{r})} \right) < 0$$

We fi nish the claim as we did in the proof of Lemma 6. ■

By Lemma 11, it suffi ces to provide upper bounds for the terms $B(s_1, \gamma_1, \gamma_2)$ for $s_1 \in \{\lceil \epsilon_2 n \rceil, \ldots, \lceil \bar{s}_{\text{crit}} n \rceil\}$, $\gamma_1 \geq \bar{\gamma}_{\text{crit}}(\epsilon_2) n$ and $\gamma_2 \geq \gamma_2^* n$. In the proof of Lemma 11 we established that

$$\mathbb{Q}' \left[|N(F) \backslash N(S_1)| = \gamma_2 \mid |N(S_1)| = \gamma_1, |S_1| = s_1 \right] \leq \binom{\lfloor (1-\tilde{r}) n \rfloor - \gamma_1}{\gamma_2} \left(\frac{\gamma_2 + \gamma_1}{\lfloor (1-\tilde{r}) n \rfloor} \right)^{(\lceil \alpha n \rceil - s_1) d_v}$$

[3] so that the fi rst entropy term in the fi rst line is increasing in s_1/n

[4] so that the second entropy term is increasing in γ_1 and the third term is increasing in s_1/n and in γ_1/n

In Appendix F we established that

$$\mathbb{Q}'\left[|N(S_1)| = \gamma_1 |S_1| = s_1\right] \leq \binom{\lfloor (1-\tilde{r})n \rfloor}{\gamma_1} \left(\frac{\gamma_1}{\lfloor (1-\tilde{r})n \rfloor}\right)^{d_v s_1}$$

The only missing piece is an upper bound on

$$\mathbb{Q}'\left[\exists\, S_2 \subseteq F^c(q) \text{ with } (S_1, S_2) \in A_2 \text{ s.t. } \mathcal{B}(S_1, S_2) \mid |N(S_1)| = \gamma_1,\ |N(F) \backslash N(S_1)| = \gamma_2, |S_1| = s_1\right]$$

But, in the conditional space $\mathcal{T}(\epsilon_1)$, every set $S_2 \in F^c(q)$ corresponds to a request vector $r \in \prod_{i=1}^{q} \{0, .., \lceil \bar{R}_i^{\text{up}} n \rceil\}$. Moreover, for a set $S_2 \in F^c(q)$ and its corresponding request vector r, the event $\mathcal{B}(S_1, S_2)$ is equivalent to the following condition being satisfied:

$$\mathcal{B}(S_1, S_2) \Leftrightarrow |(N(S_2) \cap N(F)) - N(S_1)| \leq \beta(s_1, \gamma_1, r)$$

Therefore, a union bound over all the possible choices of sets S_2 gives the following upper bound for the probability of interest:

$$\sum_{r_1=0}^{\lceil \bar{R}_1^{\text{up}} n \rceil} \cdots \sum_{r_q=0}^{\lceil \bar{R}_q^{\text{up}} n \rceil} \underbrace{\binom{\lceil \bar{R}_1^{\text{up}} n \rceil}{r_1} \cdots \binom{\lceil \bar{R}_q^{\text{up}} n \rceil}{r_q} \lambda(r_1, \ldots, r_q, \gamma_1, \gamma_2)}_{\Lambda(r_1, r_2, \ldots, r_q, \gamma_1, \gamma_2)}$$

Where $\lambda(r_1, \ldots, r_q, \gamma_1, \gamma_2)$ is the probability

$$\mathbb{Q}'\left[|(N(S_2) \cap N(F)) \setminus N(S_1)| \leq \beta(s_1, \gamma_1, r) \;\middle|\; \begin{array}{l} |S_1| = s_1, |N(S_1)| = \gamma_1, |N(F) \setminus N(S_1)| = \gamma_2 \\ S_2 \text{ corresponds to request vector } r \end{array}\right]$$

Before completing the proof we need a final observation.

Lemma 12. *$\forall i$, if $\{r_j\}_{j \neq i}, \gamma_1, \gamma_2$ are fixed then $\Lambda(r_1, r_2, \ldots, r_q, \gamma_1, \gamma_2)$ is increasing for $r_i \in \left\{1, \ldots, \lfloor \frac{\bar{R}_1^{\text{up}} n}{2} \rfloor\right\}$.*

Proof. Clearly $\binom{\lceil \bar{R}_i^{\text{up}} n \rceil}{r_i}$ is increasing for $r_i \in \left\{1, \ldots, \lfloor \frac{\bar{R}_1^{\text{up}} n}{2} \rfloor\right\}$. Therefore, it is enough to establish that $\lambda(r_1, \ldots, r_q, \gamma_1, \gamma_2)$ is increasing for $r_i \in \left\{1, \ldots, \lfloor \frac{\bar{R}_1^{\text{up}} n}{2} \rfloor\right\}$. But to show this we can just use the coupling argument that we used in Section 4.2. The coupling which we omit here is built upon the intuition is that, since the number of matching requests and the number of edges that a bit in F^c has in $N(F)$ are linearly related, by increasing the number of edges–requests the probability that event $\mathcal{B}(S_1, S_2)$ happens becomes larger. ■

Having established the above we can now conclude the claim. If we denote by $\mathcal{R}_i := \left\{\left\lfloor \frac{\bar{R}_1^{\text{up}} n}{2} \right\rfloor, \ldots, \lceil \bar{R}_1^{\text{up}} n \rceil\right\}$ we have that

$$\sum_{r_1=0}^{\lceil \bar{R}_1^{\text{up}} n \rceil} \cdots \sum_{r_q=0}^{\lceil \bar{R}_q^{\text{up}} n \rceil} \binom{\lceil \bar{R}_1^{\text{up}} n \rceil}{r_1} \cdots \binom{\lceil \bar{R}_q^{\text{up}} n \rceil}{r_q} \lambda(r_1, \ldots, r_q, \gamma_1, \gamma_2) \leq$$
$$poly(n) \max_{r_i \in \mathcal{R}_i} \binom{\lceil \bar{R}_1^{\text{up}} n \rceil}{r_1} \cdots \binom{\lceil \bar{R}_q^{\text{up}} n \rceil}{r_q} \lambda(r_1, \ldots, r_q, \gamma_1, \gamma_2),$$

where the $\mathrm{poly}(n)$ factor accounts for the fact that there are polynomially many terms in the summation. Now, by a union bound we get

$$\lambda(r_1,\ldots,r_q,\gamma_1,\gamma_2) \leq \binom{\gamma_2}{\min\{\beta(s_1,\gamma_1,r),\gamma_2\}} \left(\frac{\gamma_1+\min\{\beta(s_1,\gamma_1,r),\gamma_2\}}{\gamma_1+\gamma_2}\right)^{\nu(r)}$$

and putting everything together we get the claim.

■

Based on the preceding analysis, we can now complete our proof of Lemma 7. Indeed, using Lemmas 5, 6, 10 we can upper bound $\mathbb{Q}[\mathcal{B}(A_2) \mid \mathcal{T}(\epsilon_1)]]$ by the quantity (14), with the addition of further polynomial pre-factors. Since in the upper bound all relevant quantities, i.e. s_1, γ_1, γ_2, $r_1,\ldots,r_q$, scale linearly with n, standard bounds on binomial coeffi cients (see Appendix D) lead to the claimed form of F.

H Proof of Lemma 8

First we have

$$\mathbb{Q}[\mathcal{B}(A_3) \mid \mathcal{T}(\epsilon_1)] \leq$$
$$\sum_{s_1=1}^{\lfloor \epsilon_2 n \rfloor} \underbrace{\binom{\lceil \alpha n \rceil}{s_1} \mathbb{Q}[\exists\, S_2 \subseteq F^c(q) \text{ with } (S_1,S_2) \in A_3 \text{ s.t. } \mathcal{B}(S_1,S_2) \mid \mathcal{T}(\epsilon_1), S_1 \text{ some fi xed set of size } s_1]}_{D'(s_1)} \tag{17}$$

Note that for ϵ_2 suffi ciently small, we have that, for all $s_1 \in \{1,\ldots,\lfloor \epsilon_2 n \rfloor\}$,

$$\binom{\lceil \alpha n \rceil}{s_1} \leq \binom{\lceil \alpha n \rceil}{\lfloor \epsilon_2 n \rfloor} \leq n\left(\alpha H\left(\frac{\epsilon_2}{\alpha}\right) + o(1)\right)$$

The rest of the analysis is based on the intuition that, for ϵ_2 suffi ciently small and any set S_2 of size at least ρn, if r is the vector of requests from S_2, then, with high probability,

$$|N(S_2) \cap (N(F) - N(S_1))| \geq \sum_{i=1}^{q} i r_i + p\epsilon_2 n := \beta'(\epsilon_2, r).$$

in other words the neighborhood of set S_2 inside $N(F) \setminus N(S_1)$ is suffi ciently large not only to cover the requests from set S_2 but also from S_1. Indeed,

$$\mathbb{Q}'[\exists\, S_2 \subseteq F^c(q) \text{ with } (S_1,S_2) \in A_3 \text{ s.t. } \mathcal{B}(S_1,S_2) \mid S_1 \text{ some fi xed set of size } s_1] \leq$$
$$\mathbb{Q}'[\exists\, S_2 \subseteq F^c(q) \text{ with } |S_2| \geq \rho n \; s.t. |N(S_2) \cap (N(F) - N(S_1))| \geq \sum_{i=1}^{q} i r_i + p\epsilon_2 n \mid S_1 \text{ fi xed}, |N(S_1)| \leq d_v \epsilon_2 n]$$

By similar analysis as in the proof of Lemma 10, we get that

$$D'(S_1) \leq 2^{nF'(\alpha,\epsilon_2)} + o(1)$$

where

$$F'(\alpha,\epsilon_2) := \sup_{\bar{\gamma}_2\in[0,\, d_v\,\alpha]}\ \sup_{\bar{r}_i\in[\bar{R}_i^{\rm up}/2,\bar{R}_i^{\rm up}]} G'(\bar{\gamma}_2,\bar{r}_1,\ldots,\bar{r}_q,\epsilon_2),$$

and the intermediate function $G' = G'(\bar{\gamma}_2,\bar{r}_1,\ldots,\bar{r}_q,\epsilon_2)$ is

$$\begin{aligned}
&\alpha H\left(\frac{\epsilon_2}{\alpha}\right) + \sum_{i=1}^{q} \bar{R}_i^{\rm up} H\left(\frac{\bar{r}_i}{\bar{R}_i^{\rm up}}\right)\\
&+ \min\left\{0, ((1-\tilde{r}) - d_v\epsilon_2) H\left(\frac{\bar{\gamma}_2}{((1-\tilde{r}) - d_v\epsilon_2}\right) + d_v(\alpha-\epsilon_2)\log_2\left(\frac{d_v\epsilon_2+\bar{\gamma}_2}{(1-\tilde{r})}\right)\right\}\\
&+ \min\left\{0, \bar{\gamma}_2 H\left(\frac{\min\{\bar{\gamma}_2, \bar{\beta}'(\epsilon_2,\bar{r})\}}{\bar{\gamma}_2}\right) + \nu(\bar{r})\log_2\left(\frac{d_v\epsilon_2+\min\{\bar{\gamma}_2,\bar{\beta}'(\epsilon_2,\bar{r})\}}{d_v\epsilon_2+\bar{\gamma}_2}\right)\right\}
\end{aligned}$$

Note that $\lim_{\epsilon_2\to 0} G'(\bar{\gamma}_2,\bar{r}_1,\ldots,\bar{r}_q,\epsilon_2) = \lim_{\bar{s}\to 0,\bar{\gamma}_1\to 0} G(\bar{s},\bar{\gamma}_1,\bar{\gamma}_2,\bar{r}_1,\ldots,\bar{r}_q,\epsilon_2)$. Therefore,

$$\lim_{\epsilon_2\to 0} F'(\alpha,\epsilon_2) \le F(\alpha).$$

So, if $F(\alpha) < 0$ it follows $\lim_{\epsilon_2\to 0} F'(\alpha,\epsilon_2) < 0$ and by continuity there exists some value $\epsilon_2 > 0$ such that $F'(\alpha,\epsilon_2) < 0$ and, therefore, for this value of ϵ_2 the probability $\mathbb{Q}[\mathcal{B}(A_3) \mid \mathcal{T}(\epsilon_1)]$ is decreasing exponentially in n.