LP Decoding Achieves Capacity

(CORC Technical Report TR-2004-04)

Jon Feldman* Cliff Stein†

Department of Industrial Engineering and Operations Research
Columbia University
New York, NY.
{jonfeld,cliff}@ieor.columbia.edu

Abstract

We give a linear programming (LP) decoder that achieves the capacity (optimal rate) of a wide range of probabilistic binary communication channels. This is the first such result for LP decoding. More generally, as far as the authors are aware this is the first known polynomial-time capacity-achieving decoder with the *maximum-likelihood (ML) certificate* property—where output codewords come with a proof of optimality. Additionally, this result extends the capacity-achieving property of expander codes beyond the binary symmetric channel to a larger family of communication channels.

Perhaps most importantly, since LP decoding performs well in practice on *turbo* codes and *low-density parity-check (LDPC)* codes (comparable to the revered "belief propagation" algorithm), this result exhibits the power of a new, widely applicable "dual witness" technique (Feldman, Malkin, Servedio, Stein and Wainwright, ISIT '04) for bounding decoder performance.

For expander codes over an adversarial channel, we prove that LP decoding corrects a constant fraction of errors. To show this, we provide a new combinatorial characterization of error events that is of independent interest, and which we expect will lead to further improvements.

*Supported by an NSF Mathematical Sciences Postdoctoral Research Fellowship.

†Research partially supported by NSF Grant DMI-9970063 and an Alfred P. Sloan Foundation Fellowship.

1 Introduction

A great deal of current research in coding theory focuses on the code families of turbo codes [BGT93], low-density parity-check (LDPC) codes [Gal62] and expander codes [SS96], using "soft information" message-passing decoders such as belief-propagation (BP) and min-sum. These codes and decoders perform very well in practice on probabilistic channels, yet this performance has not been fully explained.

In this paper we give a new linear-programming (LP) decoder [FK02] for expander codes, and prove that it **achieves the capacity of any memoryless symmetric LLR-bounded (MSB) channel.** (MSB channels include most probabilistic channels commonly considered in practice; see Section 2.1 for a definition). By "achieving capacity," we mean that for any rate less than the capacity (optimal rate) of the channel, the probability of decoding error decreases exponentially in the length of the code. This is the first such result for LP decoding, and also implies the following new results:

- *Polynomial-time maximum-likelihood (ML) certificate decoders can achieve capacity.* An *ML certificate* decoder has the property that any codeword output by the decoder comes with a proof of optimality (maximum-likelihood). (The decoder is also allowed to give an "error.") Since LP decoders are always ML certificate decoders, our result shows for the first time that the requiring a polynomial-time decoder to give an ML certificate does not prevent it from achieving the optimal rate of the channel.

- *Expander codes can achieve capacity for arbitrary MSB channels.* It was known [BZ02, BZ04, BZ03] that expander codes can achieve capacity in the binary symmetric channel (BSC), a particular MSB channel. However these results were not proved for other commonly considered channels, such as the additive white Gaussian noise (AWGN) channel.

- *Experimentally good decoders can achieve capacity.* On LDPC codes, LP decoding has an empirical error-correcting performance that is competitive with min-sum and BP [Fel03]. However, neither of these algorithms have been shown to achieve capacity. The capacity-achieving message-passing decoders for expander codes (over the BSC) use "hard information" steps[1], and such decoders do not perform as well in practice as "soft information" decoders such as LP decoding [Fel03] or belief-propagation.

We also give results for the *adversarial* channel. We show that using an expander code of rate $1 - 2H(\delta)$, the LP decoder corrects an $\delta^2/4 - \epsilon$ fraction of errors, for any $\epsilon > 0$. This matches the result of Barg and Zémor [BZ02] using a bit-flipping decoder on the same code. In proving this result, we give a new combinatorial characterization of a necessary condition for LP decoding failure. This characterization is of independent interest, and has parallels to the "pseudocodeword" results for other code families [Wib96, FKKR01, DPR$^+$02, FKV01, KV03].

This result is a milestone for the general technique of LP decoding. Because LP decoders are less efficient than the linear-time message-passing decoders, their advantages lie in their wide applicability and analytical tractability. Message-passing decoders do not necessarily converge, they need to be redesigned for each code and/or channel, and consequently they are often quite difficult to analyze. LP decoding is a single algorithm that always converges, applies to a wide range of codes (including turbo codes, LDPC codes and expander codes) and channels, and is amenable to very powerful analysis. Achieving capacity illustrates this power.

1.1 Techniques and related work. Linear programming (LP) decoders for binary codes use a polytope $Q \subseteq [0,1]^n$ where the integral vertices of Q are exactly the possible codewords sent over the channel. Decoding success depends on the transmitted codeword being the optimal solution to the LP. To prove decoding success, it suffices to exhibit a dual solution that is a witness to the transmitted codeword being optimal (by complementary slackness).

[1] The decoder in [BZ03] uses one iteration of soft information messages, but then continues with hard information messages.

In recent work [FMS$^+$04], we (along with Malkin, Servedio and Wainwright) used a dual witness to prove that LP decoding corrects a constant fraction of errors using LDPC codes. In the current paper, this technique is extended to expander codes, and to a probabilistic setting. We show that when the code rate is below capacity, then with high probability over the noise in the channel, there exists a dual witness proving that the transmitted codeword is optimal.

Expander codes were introduced in [SS96] (see also [Spi95]). One of the original results gave codes of rate $1 - 2H(\delta)$, built on Ramanujan graphs; it was shown that a bit-flipping algorithm (a variant on the algorithm in [Gal62]) corrects a $\delta^2/64$ fraction of errors. Zémor [Z0́1] uses another variant of the algorithm to improve this to $\delta^2/4$ (which has since been further improved [SR03] to $\delta^2/2$).

In later work, Barg and Zémor [BZ02] give a bit-flipping algorithm that achieves the capacity of the binary symmetric channel. Our paper is very much inspired by their work, and our capacity-achieving code construction is roughly equivalent. In very recent results, using more sophisticated constructions [BZ04, BZ03], they correct a fraction of errors up to the Zyablov [Zya71] bound (and more generally the Blokh-Zyablov bound [BZ82]), and improve the error probability in the BSC. Guruswami and Indyk [GI02] give a different expander-based binary code construction, and also attain the Zyablov/Blokh-Zyablov bound. In later work, they achieve the Gilbert-Varsharmov bound for low rates [GI04]. It would be interesting to see if LP decoding could improve the results for these constructions.

Density evolution [RU01, LMSS98] has given near-capacity rate thresholds for distributions of random graphs under BP and min-sum decoding for more general channels, and represents a major breakthrough in the analysis of the message-passing decoders. However the thresholds computed using density evolution are only estimates of the true behavior of the decoders, because they assume a cycle-free message history, which is not the case in practice. That being said, density evolution has yielded thresholds that are quite close to capacity [CFRU01].

Channel capacity for memoryless symmetric channels, under polynomial-time decoding, was first achieved by Forney [For66] using concatenated codes and generalized minimum-distance (GMD) decoding.

1.2 Outline. In Section 2, we give the necessary background on codes, channel models, expander codes, and LP decoding. In Section 3 we present the LP decoder for a general expander code. We give some necessary graph-theoretic definitions and lemmas in Section 4. Our main result, that LP decoders achieve capacity, is given in Section 5. In Section 6 we give results for the adversarial channel, including our new characterization of a necessary decoding failure condition. We give some open questions in Section 7.

2 Background

2.1 Coding, channel models. A binary code C of length n is a subset of $\{0,1\}^n$, where $|C| = 2^k$. The code C is used to transmit information in the presence of noise. An information word $x \in \{0,1\}^k$ is encoded to a unique codeword $y \in C$, and sent over a noisy channel. A corrupt codeword $\hat{y}$ is received, and the decoding task is to recover the transmitted codeword y. The *rate* of the code is k/n. The *distance* of the code is the minimum Hamming distance $\Delta(y, y')$ between any two distinct codewords $y, y' \in C$. The *relative distance* is the distance divided by the length of the code. A binary *linear* code $C \subseteq \{0,1\}^n$ is a linear subspace of $\mathbb{F}_2^n$; i.e., $0^n \in C$, and for all codeword pairs $y, y' \in C$, we have $(y + y') \in C$.

We will consider both adversarial and probabilistic noise in this paper. In the adversarial model, the channel flips some of the bits arbitrarily. A decoder is said to "correct an α fraction of error" if, for any set of at most αn bits flipped by the channel, the decoder can still recover the original codeword; i.e., the decoder succeeds if $\Delta(y, \hat{y}) \leq \alpha n$.

For the probabilistic model, we will consider an arbitrary *memoryless symmetric LLR-bounded* (MSB) channel, which we now define. Associated with the channel is an alphabet Σ representing the set of possible symbols output by the channel. (Note that this could be a continuous set, such as the reals.) The channel being *memoryless* means that the noise affects each bit transmitted over the channel independently; therefore, the channel is completely specified by transition probabilities $p(\mathtt{a}|\mathtt{b})$ for each $\mathtt{a} \in \Sigma$ and $\mathtt{b} \in \{0,1\}$, where $p(\mathtt{a}|\mathtt{b})$ denotes the probability that symbol $\mathtt{a}$ is output by the channel, given that the bit $\mathtt{b}$ is transmitted. (In continuous alphabets, $p(\mathtt{a}|\mathtt{b})$ is a p.d.f.) The channel being *symmetric* means that the noise affects 0's and 1's symmetrically; formally, the symbol space Σ can be partitioned into pairs $(\mathtt{a}, \mathtt{a}')$ such that $p(\mathtt{a}|0) = p(\mathtt{a}'|1)$ and $p(\mathtt{a}|1) = p(\mathtt{a}'|0)$. (The definition also allows for a single "erasure" symbol to be its own pair.) Finally, we define the *log-likelihood ratio* (LLR) γ_i of a received bit $\hat{y}_i$ to be

$$\gamma_i = \log \frac{p(\hat{y}_i|0)}{p(\hat{y}_i|1)}.$$

The channel being *LLR-bounded* means that there is some number W where $-W < \gamma_i < W$ for all possible received symbols $\hat{y}_i \in \Sigma$.

One common example of an MSB channel is the *binary symmetric channel* (BSC) where each bit is flipped indepentently with probability p. In the BSC, we have $\Sigma = \{0, 1\}$, $\gamma(1|0) = \gamma(0|1) = p$, and $\gamma(1|1) = \gamma(0|0) = 1 - p$. We will use the BSC as a running example throughout the presentation. An important example of an *unbounded* memoryless symmetric channel is the additive white Gaussian noise (AWGN) channel, where $\Sigma = \mathbb{R}$, and for each transmitted bit y_i, we have $\hat{y}_i = (1 - 2y_i) + \mathcal{N}(0, \sigma^2)$, where $\mathcal{N}(0, \sigma^2)$ is a zero-centered Gaussian with variance σ^2. In practice, we could truncate the tails of the Gaussian, and so in this case we get an MSB channel.

A decoder is a *Maximum-likelihood* (ML) decoder if it always outputs the codeword y that maximizes the likelihood of receiving $\hat{y}$, given that y was transmitted. This is equivalent to the codeword y that minimizes the quantity $\sum_i \gamma_i y_i$. In the BSC, the ML decoder finds the codeword that is closest in Hamming distance to the received word. For most interesting codes, including the expander codes considered in this paper, ML decoding is NP-hard.

The *word error rate* P_{err} of a decoder is the probability, taken over the noise in the channel, that the decoder succeeds (outputs the codeword y that was originally transmitted).

2.2 Channel Capacity. A *code family* is a set of codes of a particular fixed rate r, but increasing length n. A major goal in coding theory is to define a code family (and accompanying decoder) with as high a rate as possible such that $P_{\text{err}} \to 0$ as $n \to \infty$. For any memoryless channel, this is achieved by a random code using ML decoding, as long as the rate r is strictly less than the *capacity* $\mathcal{C}$ of the channel [Sha48, Gal68]. Furthermore, this property is not possible if $r \geq \mathcal{C}$ (see [Gal68]).

The capacity $\mathcal{C}$ is a function only of the channel model (and its associated parameters). For example, the capacity of the binary symmetric channel with crossover probability p is equal to $1 - H(p)$, where H is the binary entropy function.

The *random coding exponent* $\mathcal{E}(r)$ is a standard lower bound on the expectation of $-(\log P_{\text{err}})/n$ under ML decoding, taken over a random choice of codes of rate r. (We give more details in Appendix A.) This random coding exponent [Gal68] has the property that $\mathcal{E}(r) > 0$ for all rates $r < \mathcal{C}$, and has been studied extensively for different channel models.

2.3 Expander codes. Let $G = (V, E)$ be a d-regular graph with $M = |V|$ nodes and $N = |E|$ edges. (We choose N for the edges, since this will be the length of the code.) For a node $j \in V$, we let $\Gamma(j)$ be the set of d edges incident to j. We will choose a graph G that is an expander, but we do not need this property to define the code.

An *expander code* [SS96] (see also [BZ02], [Bar98]) is a code based on G, defined as follows.[2] For each node $j \in V$, let C_j be a binary linear code with length d, rate r_j, and relative distance δ_j. For each node $j \in V$, define an arbitrary (but fixed) ordering of the edges incident to j. The *expander code* $\mathbf{EC}(G, \{C_j\}_j)$ is defined as the settings of bits y_e to the edges $e \in E$ such that for every node $j \in V$, the bits $\{y_e\}_{e \in \Gamma(j)}$ (when considered in their fixed ordering) form a codeword of C_j. For a codeword $c \in C_j$, and some edge $e \in \Gamma(j)$, let $c[e] \in \{0, 1\}$ be the bit assigned to edge e in the codeword c. By counting the number of linear constraints on the code, it is easily seen that the rate R of the overall expander code C is at least $1 - 2(\sum_j (1 - r_j))/M$.

[2]We use the definition in [BZ02]; more general expander codes, to which LP decoding can also be applied, can be found in [Spi95].

2.4 Linear programming (LP) decoding. LP decoding was introduced in [FK02, FKW02] for turbo codes, and has since been extended to LDPC codes [FWK03, Fel03, FMS+04] and considered in general for binary codes [FKW03, Fel03]. This is the first consideration of LP decoding for expander codes, and the LP given here is a natural generalization of the one in [FWK03, Fel03].

The idea behind LP decoding is to use LP relaxation to try and find the ML codeword. For a specific code $C \subseteq \{0,1\}^n$, a polytope $Q \in [0,1]^n$ over variables $\{y_i\}$ is specified such that the integral points in the polytope are exactly the codewords of the code; i.e., $Q \cap \{0,1\}^n = C$. Using the objective function $\min \sum_i \gamma_i y_i$, and enforcing $y \in Q$, $y_i \in \{0,1\}$, one obtains an integer linear program that is a maximum-likelihood decoder.

We relax the integer constraints to $0 \leq y_i \leq 1$ to obtain an LP that is solvable in polynomial time. Upon solving the LP, if the solution y is integral, then y must represent the ML codeword; if it is fractional, then an error is declared. This gives LP decoders the *ML-certificate* property: if a codeword is output, it is guaranteed to be the ML codeword. The word error rate of an LP decoder is the probability, taken over the noise in the channel, that the transmitted codeword is the optimal solution to the LP. For a general discussion of LP decoding, see [FKW03, Fel03]. For more details on specific LP decoders, see [FK02, FWK03, Fel03, FMS+04].

3 LP decoding with expander codes

In this section we define an LP decoder that can be used for any expander code (as defined in Section 2.3), and can be seen as generalization of the LP for LDPC codes in [FWK03].

The decoding LP contains a variable $f_e \geq 0$ for every edge in the graph, indicating the value of the code bit y_e. The LP objective is to minimize $\sum_e \gamma_e f_e$, where γ_e is a function of the channel model, and the received word $\hat{y}$. For probabilistic channels, γ_e is defined to be the LLR of the code bit associated with edge e, as discussed in Section 2.1. More precisely, for some received word $\hat{y} \in \Sigma^N$, we have $\gamma_e = \log \frac{p(\hat{y}_e|0)}{p(\hat{y}_e|1)}$. For adversarial channels, where $\hat{y} \in \{0,1\}^N$, we set $\gamma_e = +1$ if $\hat{y}_e = 0$, and $\gamma_e = -1$ if $\hat{y}_e = 1$. This makes the LP objective, for all integral solutions $f_e \in \{0,1\}^N$, equal to $\Delta(f_e, \hat{y}_e) - \sum_e \hat{y}_e$, an adjusted Hamming distance from the received word.

We also have auxiliary variables $w_{j,c} \geq 0$ defined for each node j and local codeword $c \in C_j$, indicating that node j is satisfied by the local codeword c. When $w_{j,c} = 1$ it should be the case that the edges $e \in \Gamma(j)$ take on values $f_e = c[e]$. The LP constraints enforce consistency between the f and w variables in the natural way, specified in the LP below:

$$\textbf{Decoding LP:} \quad \text{minimize} \quad \sum_e \gamma_e f_e \quad \text{s.t.}$$
$$\forall j \in V, \quad \sum_{c \in C_j} w_{j,c} = 1$$
$$\forall e = (j, j') \in E, \quad f_e = \sum_{c \in C_j : c[e]=1} w_{j,c} = \sum_{c \in C_{j'} : c[e]=1} w_{j',c}$$

We claim that solutions (f, w) to this LP where $f \in \{0,1\}^N$ must have $f \in C$. To see this, consider a single node $j \in V$. By the LP constraints, the variables $\{f_e\}_{e \in \Gamma(j)}$ must represent a convex combination of local codewords $c \in C_j$. However, since $f_e \in \{0,1\}$ for all e, the convex combination must put all its weight on a single local codeword. Therefore, since this holds for all j, we have $f \in C$. This also shows that this LP decoder has the ML-certificate property.

Another property we will need is that the decoding polytope is C*-symmetric* (see [FKW03, Fel03]). We include the definition of C-symmetry, and a proof of this fact in Appendix C. The C-symmetry of the decoding polytope allows us to assume that 0^N is the codeword that is transmitted over the channel.

3.1 Bounding the word error rate using a dual witness. In this section we describe the method of proving a word error rate bound using a zero-valued dual feasible point. This method was first described in [FMS+04] in order to show that LP decoders correct a constant fraction of error using LDPC codes.

When we assume 0^N is transmitted, our LP decoder succeeds if it outputs a solution where $f = 0^N$. In fact, there is only one feasible setting of the $\{w_{j,c}\}$ variables when $f = 0^N$; namely, $w_{j,0^d} = 1$ for all j, and all other $w_{j,c} = 0$. We refer to this setting of the $w_{j,c}$ variables as w^*, and so our LP decoder succeeds if $(0^N, w^*)$ is the unique LP optimum. (If there are multiple LP optima, we assume failure.) The solution $(0^N, w^*)$ is always feasible, and always has value zero. Therefore, a necessary and sufficient condition for $(0^N, w^*)$ to be optimal is the existence of a dual feasible solution with value zero. Furthermore, a sufficient condition for $(0^N, w^*)$ to be the *unique* optimum is the existence of zero-valued dual feasible solution with slack in every dual constraint associated with variables f_e. (This follows from complementary slackness.) Our strategy for proving decoding success will be to find such a dual solution.

If we take the LP dual, set the objective value equal to zero, enforce slack in the edge constraints, and simplify, we get the following (open) polytope, defined over variables $\{\tau_{e,j}\}_{j \in V, e \in \Gamma(j)}$:

$$\textbf{Polytope } \hat{P}: \quad \forall j \in V, c \in C_j, \quad \sum_{e \in \Gamma(j)} c[e] \cdot \tau_{e,j} \geq 0 \tag{1}$$

$$\forall e = (j, j') \in E, \quad \tau_{e,j} + \tau_{e,j'} < \gamma_e \tag{2}$$

Any feasible point $\tau \in \hat{P}$ is a proof of the fact that 0^N is the unique primal LP optimum.

As an sanity check, suppose we transmitted over an adversarial channel, and there were no errors; i.e., $\gamma_e = +1$ for all $e \in E$. Then, if we set all $\tau_{e,j} = 0$, we get a point in $\hat{P}$. Thus we have shown that if there are no errors, the decoder succeeds. In later sections, we will demonstrate feasible points in $\hat{P}$ for more interesting situations.

4 Graph expansion and ρ-orientations

In this section we state some purely graph-theoretic lemmas concerning expansion that we will need to construct points in the polytope $\hat{P}$, for both probabilistic and adversarial error. The proofs are all in Appendix B.

A *subgraph* $G' = (V', E')$ of G is a subset of nodes and edges in G where every edge in E' has both of its endpoints in V' (no "hanging" edges). An *induced* subgraph is a subgraph $G = (V', E')$ where E' is exactly the set of edges with both endpoints in V'. The following definition of expansion is not the normal one, but will be more useful to us:

Definition 1 *A d-regular graph G is a* (α, ρ)-expander *if, for all induced subgraphs* $G' = (V', E')$ *with* $|V'| \leq \alpha|V|$*, we have* $|E'| \leq \rho d|V'|$.

The following is clear from the definition of expansion above.

Lemma 1 *In a d-regular* (α, ρ)*-expander* $G = (V, E)$*, if some subgraph* $G' = (V', E')$ *has* $|E'| \leq \alpha\rho d|V|$*, then* $|V'| \geq |E'|/(\rho d)$.

We use expansion in the construction of a point in $\hat{P}$ by spreading out the cost of channel errors. The following definition and lemma will help us do this:

Definition 2 *Let a* ρ-orientation *of a subgraph* $G' = (V', E')$ *of a d-regular graph G be an assignment of directions to every edge in* E' *such that each node in* V' *contains at most* ρd *incoming edges from* E'.

Lemma 2 *If a d-regular graph G is a* (α, ρ)*-expander, where* ρd *is an integer, then all subgraphs* $G' = (V', E')$ *where* $|E'| \leq \alpha\rho d|V|$ *contain a* ρ*-orientation.*

We will also use the following result of Alon and Chung [AC88] to establish the expansion properties of our graphs:

Theorem 3 (Alon-Chung) *Let $G = (V, E)$ be a d-regular graph such that all eigenvalues other than d have absolute value at most λ. Let T be a subset of the vertices of G of size $\gamma|V|$. Then, the number of edges contained in the subgraph induced by T in G is at most $\gamma|V|\left(\frac{d\gamma}{2} + \frac{\lambda}{2}(1-\gamma)\right)$.*

For a particular value ρ, we can use this theorem to construct a d-regular graph that is an (α, ρ)-expander, where $\alpha = 2\rho - (\lambda/d)$.

5 Probabilistic error: achieving capacity

In this section we assume an arbitrary MSB channel with capacity $\mathcal{C}$ and LLR bound W. Our task is to come up with an expander code family of some given rate $R < \mathcal{C}$ such that the word error rate under LP decoding decreases exponentially in the code length $N = |E|$.

We first define, in Section 5.1, the general parameters of the expander code family we will use. In Section 5.2, we show how to find a point in $\hat{P}$, given that a certain condition holds, and then show that this condition is very likely. Finally, in Section 5.3, we show how to instantiate the parameters of the code so that we achieve capacity.

5.1 The parameters of the code. The expander code family we present here is essentially the same as that of Barg and Zémor [BZ02], with some of the parameters set differently. We let G be a balanced bipartite d-regular Ramanujan graph with second-largest eigenvalue $\lambda = \Theta(\sqrt{d})$, as used in [BZ02]. Since G is bipartite, we have $V = \{A, B\}$, with $|A| = |B| = M/2$. We use two codes C_A and C_B, and set $C_j = C_A$ for all $j \in A$, and $C_j = C_B$ for all $j \in B$. Let R be some target rate of our overall code. We let r_{A}, the rate of code C_A, be any rate greater than R, and set r_{B}, the rate of the code C_B, to be equal to $r_{\text{B}} = R - r_{\text{A}} + 1$. Note that since $r_{\text{A}} > R$, we have $r_{\text{B}} < 1$. The overall code will have rate at least $1 - 2(\sum_j r_j)/M = r_{\text{A}} + r_{\text{B}} - 1 = R$, as required.

Let δ_{A} and δ_{B} be the relative distance of the code C_A and C_B, respectively. We use the following definition to further characterize the code C_A:

Definition 3 *For a particular memoryless symmetric channel, a binary linear code C of length n is (β, κ)-*robust *if, with probability at least $1 - 2^{-\kappa n + 1}$ over the noise in channel, all non-zero codewords $y \in C$ have cost $\sum_i \gamma_i y_i \geq \beta n$.*

For now we assume that C_A is (β, κ)-robust for some $\beta, \kappa > 0$. We will show later that this can be achieved for rates r_{A} less than capacity. We define $\rho' = \delta_{\text{B}}/(1 + \delta_{\text{B}}/\delta_{\text{A}} + W/\beta)$. We let ρ be any number where ρd is an integer, and $\rho'/2 \leq \rho \leq \rho'$. By Theorem 3, we can make G a (α, ρ)-expander, where $\alpha = 2\rho - (\lambda/d)$. (Note that we may need to increase d in order to define ρ, and to make $\alpha > 0$.)

We use the notation C_{prob} to represent a particular expander code $\mathbf{EC}(G, \{C_A\}_{j \in A}; \{C_B\}_{j \in B})$ as described above. (The specific parameters will be clear from context.)

5.2 Finding a point in $\hat{P}$. We use the LP decoder from Section 3 on the code C_{prob}, and this defines a polytope $\hat{P}$. Our goal in this section is to construct a point in $\hat{P}$, as long as some high-probability event occurs. Such a point is a dual witness to the optimality of 0^n in the primal decoding LP, and therefore a proof that the decoder succeeds.

We assume a particular received word $\hat{y}$, and the resulting edge costs $\gamma_e = \log \frac{p(\hat{y}_e|0)}{p(\hat{y}_e|1)}$, where $-W < \gamma_e < W$. The *cost* $\gamma(c)$ of a local codeword $c \in C_j$ for some node j is equal to $\gamma(c) = \sum_{e \in \Gamma(j)} c[e]\gamma_e$.

Suppose we have that for every $j \in A$, all non-zero codewords $c \in C_j$ have positive cost. In this case, finding a point in $\hat{P}$ is simple: just set $\tau_{e,j} = \gamma_e - \epsilon$ for all $j \in A$, and set $\tau_{e,j} = 0$ for all $j \in B$, for some small $\epsilon > 0$. Of course this will not always happen, so we need to be more careful about how we set the variables $\tau_{e,j}$, which we will refer to as "edge weights." If a node has a negative-cost local codeword, then we need to bias the incident edge weights to be positive in order to satisfy the node constraints (1) of $\hat{P}$; on the other hand, if a node has all its non-zero codewords with positive cost, then it can afford to "absorb" some incident excess negative weight.

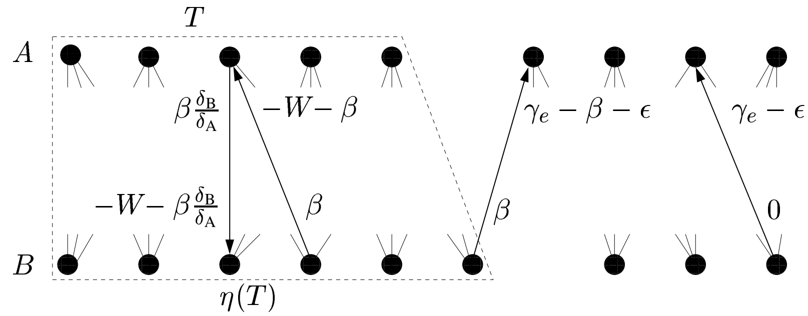

Figure 1: Setting the edge weights $\tau_{e,j}$ for each node j and incident edge $e \in \Gamma(j)$ to satisfy the constraints of $\hat{P}$. The weights are set according to the "bad" nodes T, their neighbors $\eta(T)$, and the orientation of each edge.

This motivates the following definition: let $T \subseteq A$ be the nodes in A that have an incident non-zero local codeword with cost less than or equal to βd. Formally, $T = \{j \in A : \exists c \in C_j \text{ s.t. } \sum_{e\in\Gamma(j)} c[e] \cdot \gamma_e \leq \beta d\}$. These are the "bad" nodes in A, the ones that cannot afford to absorb positive weight, and therefore must be treated carefully. Note that since we made the code C_A robust, it will be unlikely for a node in A to be bad.

Let $\eta(T)$ be nodes in the neighborhood of T; note that $\eta(T) \subseteq B$, since the graph is bipartite. Our weighting scheme is given in the proof of the following theorem (also see Figure 1):

Theorem 4 *If $|T \cup \eta(T)| \leq \alpha M$, then the LP decoder succeeds (outputs the transmitted codeword).*

Proof: We show the LP decoder succeeds by providing a point in $\hat{P}$. To set the edge weights $\tau_{e,j}$, we first define a direction for each edge in the graph. All edges that are not incident to T are directed toward the nodes A. Edges incident to T are directed according to a ρ-orientation of the subgraph induced by $(T \cup \eta(T))$. This is possible using Theorem 2, since $|T \cup \eta(T)| \leq \alpha M$ by assumption, and so $|\{\Gamma(j)\}_{j\in T}| \leq \alpha\rho d M$ by expansion.

We will give each edge $e = (j \to j')$ a "tail-weight" $\tau_{e,j}$ and a "head-weight" $\tau_{e,j'}$. To satisfy the edge constraints (2) of $\hat{P}$, the sum of these two weights should be strictly less than γ_e. We give the assignment in detail below (also in Figure 1), where $\epsilon > 0$ is a small constant to be specified later. There are four cases:

- **(i)** For all edges leaving T, set the tail-weight to $\beta(\delta_B/\delta_A)$ and the head-weight to $-W - \beta(\delta_B/\delta_A)$. Note that the sum is $-W$, which is strictly less than γ_e by definition of W.
- **(ii)** For all edges going into T, set the head-weight to $-W - \beta$ and the tail-weight to β, and again the sum is $-W < \gamma_e$.
- **(iii)** For all edges e incident to $\eta(T)$ but not T, set the tail weight to β, and the head weight to $\gamma_e - \beta - \epsilon$. Note that these edges are all directed away from $\eta(T)$. The sum of the edge weights is $\gamma_e - \epsilon < \gamma_e$.
- **(iv)** For all other edges (those not incident to either T or $\eta(T)$), set the head-weight to $\gamma_e - \epsilon$ and the tail-weight to 0. Recall that these edges are all directed toward A. The sum of these two edge weights is also $\gamma_e - \epsilon < \gamma_e$.

It remains to show that this weight assignment satisfies the node constraints (1) of $\hat{P}$. We show this below for each type of node: those in T, $\eta(T)$, $A - T$ and $B - \eta(T)$.

- **(i)** For a node $j \in T$, we have at most $\rho d \leq \rho' d$ incoming edges e with weight $\tau_{e,j} = -W - \beta$; the remaining (outgoing) edges have weight $\beta(\delta_B/\delta_A)$. Each non-zero codeword $c \in C_j$ has a support set of size at least $\delta_A d$, and so for all $c \in C_j$ we have $\sum_{e\in\Gamma(j)} c[e] \cdot \tau_{e,j} \geq \rho' d(-W - \beta) + (\delta_A - \rho')d\beta(\delta_B/\delta_A) = 0$.
- **(ii)** For a node $j \in \eta(T)$, we have at most $\rho d \leq \rho' d$ incoming edges e with weight $\tau_{e,j} = -W - \beta(\delta_B/\delta_A)$, and the remaining (outgoing) edges have weight β. Therefore, similar to the previous case, every non-zero codeword $c \in C_j$ has $\sum_{e\in\Gamma(j)} c[e] \cdot \tau_{e,j} \geq \rho' d(-W - \beta(\delta_B/\delta_A)) + (\delta_B - \rho')d\beta = 0$.
- **(iii)** For a node $j \in (A - T)$, we have that every incident edge e is incoming, and has weight $\tau_{e,j}$ equal to either $\gamma_e - \epsilon$ or $\gamma_e - \beta - \epsilon$. In the worst case and wlog, they all have weight $\tau_{e,j} = \gamma_e - \beta - \epsilon$, and so every

non-zero codeword $c \in C_j$ has $\sum_{e \in \Gamma(j)} c[e] \cdot \tau_{e,j} \geq (\sum_{e \in \Gamma(j)} c[e] \cdot \gamma_e) - d(\beta + \epsilon)$. By the definition of T, we know that $\sum_{e:c[e]=1} \gamma_e > \beta d$. Therefore, there exists some $\epsilon > 0$ such that $\sum_{e:c[e]=1} \tau_{e,j} \geq 0$.

(iv) For a node $j \in (B - \eta(T))$, we have $\tau_{e,j} = 0$ for all $e \in \Gamma(j)$. ■

We now give the main theorem of the section. It says that if our code C_A is robust, then the word error rate of LP decoding decreases exponentially in the code length $N = |E|$. In the next section we use this theorem to attain capacity.

Theorem 5 *If the code C_A is (β, κ)-robust, for some $\beta, \kappa > 0$, then there exists a sufficiently large degree d such that the word error rate of LP decoding using the code C_{prob} is at most $2^{-\Omega(N)}$.*

Proof: By Theorem 4 the LP decoder succeeds as long as $|T \cup \eta(T)| \leq \alpha M$. Using $|\eta(T)| \leq d|T|$, we have success if $|T| \leq \alpha M/(d+1)$, which is equivalent to $|T| \leq (M/2)(2\alpha/(d+1))$, i.e., if the fraction of bad nodes in A is at most

$$\alpha_2 = \frac{2\alpha}{d+1} \geq \frac{2\rho'}{d+1} - \frac{2\lambda}{d(d+1)}.$$

Since the code C_A is (β, κ)-robust, we have that the probability of a node being bad is at most $2^{-\kappa d+1}$, and so the expected fraction of bad nodes is at most $2^{-\kappa d+1}$. Note that α_2 decreases linearly in d, whereas the expected fraction of bad nodes decreases exponentially in d. Thus, for sufficiently large d, we have $2^{-\kappa d+1} < \alpha_2$. Each node in A is bad independently, since the edges adjacent to them are disjoint. Therefore a Chernoff bound implies a word error rate of at most $2^{-\Omega(M)} = 2^{-\Omega(N)}$, since d is constant. ■

We note that for the binary symmetric channel, the error exponent (the constant in the Ω) is not as good as the one proved by Barg and Zémor [BZ02] using a bit-flipping decoder; in particular, it has an unfortunate inverse dependence on d. It would be interesting to see if a different method of setting the edge weights could yield stronger results; since LP decoding performs better than bit-flipping decoders (at least on LDPC codes), one would expect this to be possible.

5.3 Achieving capacity. In this section we will need the following theorem, proved in Appendix A, which is in essence a slight generalization of Shannon's noisy coding theorem:

Lemma 6 *For any memoryless symmetric channel with capacity $\mathcal{C}$, for sufficiently large n, any rate $r < \mathcal{C}$, and any β where $0 < \beta < \mathcal{C} - r$, there exists a $(\beta, \mathcal{E}(r+\beta))$-robust binary linear code C with length n, rate r, and minimum distance at least $H^{-1}(1-r)$, where $\mathcal{E}$ is the random coding exponent.*

We now define the code C_{cap} that will achieve capacity. We use a particular case of the code C_{prob}. We set r_{A} to some number where $R < r_{\text{A}} < \mathcal{C}$, and β to some number where $0 < \beta < \mathcal{C} - r_{\text{A}}$. We then invoke Lemma 6 above (with $r = r_A$) to obtain the code C_A. Thus, the code C_A is $(\beta, \mathcal{E}(r_A+\beta))$-robust. Note that the random coding exponent $\mathcal{E}(r_A + \beta) > 0$ since $r_A + \beta < \mathcal{C}$. We also have that $\delta_{\text{A}} = H^{-1}(1 - r_{\text{A}})$. Furthermore we make $\delta_{\text{B}} = H^{-1}(1 - r_{\text{B}})$ by using a code C_B on the Gilbert-Varsharmov (G-V) bound (see [Gal68]).

We note that any constants δ_{A} and δ_{B} would suffice to achieve capacity; the fact that the codes are on the G-V bound only affects the error exponent (the constant in front of N in the exponent). In theory, we use exhaustive search to construct the codes C_A and C_B, which takes constant time, since d is constant. (In practice, note that any codes C_A and C_B with decent parameters give exponentially small word error rate for rates close to capacity, just by using Theorem 5.)

Theorem 7 *The word error rate P_{err} of LP decoding using the code C_{cap} is at most $2^{-\Omega(N)}$ for all rates $R < \mathcal{C}$.*

Proof: Follows from Theorem 5 and Lemma 6. ■

6 Adversarial error: correcting a constant fraction

A dual witness can also be used to give bounds for the adversarial channel. In the probabilistic channel we gave a condition on the error pattern that implied a dual witness, and then proved that this condition was likely to hold. In the adversarial channel, we will give a dual witness assuming a bound on the number of bits flipped by the channel.

Specifically, we will show that LP decoding succeeds as long as $\Delta(y, \hat{y}) \leq \alpha N$, where y is the transmitted codeword, $\hat{y} \in \{0,1\}^n$ is the received word, and α is as high a fraction as possible. In this section, our edge costs γ_e are defined so that the LP objective tries to minimize the Hamming distance from the received word $\hat{y}$, as explained in Section 3: we set $\gamma_e = +1$ if $\hat{y}_e = 0$, and $\gamma_e = -1$ if $\hat{y}_e = 1$, and so we have $\sum_e \gamma_e y'_e = \Delta(y', \hat{y}) - \sum_e \hat{y}_e$ for all codewords $y' \in C$.

To prove that LP decoding succeeds, we find a point in the polytope $\hat{P}$, as in the previous section. (We may assume that 0^N is transmitted, since the LP is C-symmetric; see Appendix C.) We first show, in Section 6.1, a general result for an arbitrary expander code, giving a purely combinatorial necessary condition for the LP decoder to fail. We follow this up in Sections 6.2 and 6.3 with a specific construction of an expander code that takes advantage of this condition.

6.1 Necessary combinatorial failure condition: error cores. We define $C_{\text{core}} = \mathbf{EC}(G, \{C_j\})$ to be an arbitrary expander code built on a d-regular graph G where each code C_j has relative distance at least δ. Note that C_{core} assumes nothing about the expansion of the graph. The following combinatorial object will be key to our results in this section:

Definition 4 *A ρ-*error core *is a subgraph $G' = (V', E')$ where* (i) $\hat{y}_e = 1$ *for all $e \in E'$, and* (ii) $\Gamma(j) \cap E' \geq \rho d$ *for all $j \in V'$.*

Note that for an edge e to be in an error core, it must be the case that the code bit y_e was flipped by the channel, and that both endpoints of e are incident to at least ρd edges e' that are also in the error core. This can become quite restrictive. We now state the main theorem in this section, which will later lead to a bound on the adversarial channel.

Theorem 8 *If the LP decoder fails in the adversarial channel using code C_{core}, then there exists an $(\delta/4)$-error core in the graph G.*

This theorem should be of independent interest, since it does not rely on graph expansion; it is merely a graph-theoretic necessary condition for decoding failure. This type of characterization is often referred to as a "pseudocodeword," since it is an object that "fools" a sub-optimal decoder. (For example, the "stopping sets" of an LDPC code represent pseudocodewords for belief-propagation in the binary erasure channel [DPR+02].)

The rest of this section is devoted to proving Theorem 8. For some received vector $\hat{y}$, let S^0 be the set of edges with an error; i.e., $S^0 = \{e \in E : \gamma_e = -1\}$. Define sets $S^1 \supseteq S^2 \supseteq \ldots$ and $T^1 \supseteq T^2 \supseteq \ldots$ inductively as follows: Let $T^i \subseteq V$ be the set of nodes with at least $(\delta/4)d$ incident edges in S^{i-1}. Now define $S^i \subseteq S^{i-1}$ to be the set of edges in S^{i-1} induced by T^i. Note that this definition could produce an infinite sequence of sets (e.g., if $S^0 = E$).

Lemma 9 *If $S^i = \emptyset$ for some finite i, the LP decoder succeeds.*

Proof: We show decoding success by constructing a point in the polytope $\hat{P}$. We set the edge weights $\tau_{e,j}$ as follows, where $\epsilon > 0$ is a small constant that we specify later:

- **(i)** For all $e = (j, j') \notin S^0$: set $\tau_{e,j} = \tau_{e,j'} = 1/2 - \epsilon$. Since $e \notin S^0$, we have $\gamma_e = +1$, and so $\tau_{e,j} + \tau_{e,j'} = 1 - 2\epsilon < \gamma_e$.
- **(ii)** For all i, and edges $e = (j, j') \in S^i$ but not in S^{i+1}: By definition of T^{i+1}, at most one endpoint of e is in T^{i+1}. If neither endpoint is in T^{i+1}, set the two weights $\tau_{e,j}$ and $\tau_{e,j'}$ to $1/2 - \epsilon$ and $-3/2$ arbitrarily. If one endpoint (say j) is in T^{i+1}, set $\tau_{e,j} = 1/2 - \epsilon$ for that endpoint, and $\tau_{e,j'} = -3/2$ for the other endpoint. In both cases, we have $\tau_{e,j} + \tau_{e,j'} = -1 - \epsilon < -1 = \gamma_e$.

Since $S^i = \emptyset$ for some finite i, all edges fall into one of the two cases above. We claim that τ is a feasible point in $\hat{P}$. We have already argued that the edge constraints (2) of $\hat{P}$ are satisfied, and so it remains to show that the node constraints (1) are satisfied.

We claim that every node j has fewer than $(\delta/4)d$ incident edges e with $\tau_{e,j} = -3/2$. There are two cases: (i) Consider a node $j \notin T^1$. This node is incident to fewer than $(\delta/4)d$ edges in S^0, and these are the only edges e that could possibly have $\tau_{e,j} = -3/2$. (ii) Consider a node $j \in T^1$. Since $S^i = \emptyset$ for some i, we have $j \in (T^i - T^{i+1})$ for some i. Since $j \notin T^{i+1}$, there are fewer than $(\delta/4)d$ edges in $\Gamma(j) \cap S^i$. If some edge $e \notin S^i$, then $\tau_{e,j} = 1/2 - \epsilon$. Therefore, fewer than $(\delta/4)d$ incident edges have $\tau_{e,j} = -3/2$, and so there is some $\epsilon' > 0$ such that every node j has at most $(\delta/4 - \epsilon')d$ edges $e \in \Gamma(j)$ with $\tau_{e,j} = -3/2$. Since code C_j has relative distance δ, we have, for all j and $c \in C_j$, $\sum_{e \in \Gamma(j)} c[e] \cdot \tau_{e,j} \geq (-3/2)(\delta/4 - \epsilon')d + (1/2 - \epsilon)(3\delta/4)d = (3\epsilon'/2 - 3\delta\epsilon/4)d$. Setting $\epsilon \leq 2\epsilon'/\delta$, we get $\sum_{e \in \Gamma(j)} c[e] \cdot \tau_{e,j} \geq 0$. ■

Lemma 10 *If there is no $(\delta/4)$-error core in the graph G, then $S^i = \emptyset$ for some finite i.*

Proof: Suppose there is no finite i where $S^i = \emptyset$. Then, for some i, $S^i = S^{i+1} \neq \emptyset$, and so $T^i = T^{i+1}$. This implies, by definition of T^{i+1}, that every node in T^{i+1} has at least $(\delta/4)d$ incident edges in $S^i = S^{i+1}$. Since the edges S^{i+1} are all induced by T^{i+1}, and $S^{i+1} \subseteq S^0$, we have that (T^{i+1}, S^{i+1}) is a $(\delta/4)$-error core. ■

Theorem 8 follows from Lemmas 9 and 10. ■

6.2 Using expansion in the error core. Even if the graph contains an error core, there is still a hope for assigning legal edge weights. In fact, if the graph expands, then we can use a ρ-orientation to assign the edge weights.

Theorem 11 *Suppose $G = (V, E)$ is a $(\alpha, \delta/4 - \epsilon')$-expander, for some $\epsilon' > 0$ where $d(\delta/4 - \epsilon')$ is an integer. Then, if the LP decoder fails, there exists a $(\delta/4)$-error core $G' = (V', E')$ where $|E'| > \alpha(\delta/4 - \epsilon')dM$.*

Proof: (Sketch) If the LP decoder fails, then by Theorem 8 we have an $(\delta/4)$-error core $G' = (V', E')$ in the graph G. Suppose that $|E'| \leq \alpha(\delta/4 - \epsilon')dM$. We show decoding success by constructing a point in $\hat{P}$.

For the edges not in E', set the edge weights exactly as in the proof of Theorem 9, using some value $\epsilon > 0$ that we specify later. Since G is a $(\alpha, \delta/4 - \epsilon')$-expander, $d(\delta/4 - \epsilon')$ is an integer, and $|E'| \leq \alpha(\delta/4 - \epsilon')dM$, there exists a $(\delta/4 - \epsilon')$-orientation of G' (by Theorem 2). Set the weights of edges $(j \to j') \in E'$ according to this orientation by setting $\tau_{e,j} = 1/2 - \epsilon$ and $\tau_{e,j'} = -3/2$.

This setting of the edge weights clearly satisfies the edge constraints of $\hat{P}$. Furthermore, using the argument from Theorem 9 together with the definition of a ρ-orientation, we have that each node j has fewer than $(\delta/4 - \epsilon'')d$ incident edges e with weight $\tau_{e,j} = -3/2$, for some $\epsilon'' > 0$. Using the same argument as in Theorem 9, setting $\epsilon \leq 2\epsilon''/\delta$, we have $\sum_{e \in \Gamma(j)} c[e] \cdot \tau_{e,j} \geq 0$ for all nodes j and non-zero codewords $c \in C_j$. ■

6.3 Correcting a $\delta^2/4$ fraction of errors. In order to use Theorem 11, we define an expander code $C_{\text{adv}} = \mathbf{EC}(G, \{C\}_j)$ using any d-regular Ramanujan graph G with second-largest eigenvalue $\lambda = \Theta(\sqrt{d})$. The codes C_j for each node j will be identical codes C on the G-V bound, of length d and relative minimum distance δ. (We make d sufficiently large to reach the G-V bound.) The overall code C_{adv} thus has rate $R = 1 - 2H(\delta)$.

Theorem 12 *For any $\epsilon > 0$, there exists a sufficiently large degree d such that using the code C_{adv}, LP decoding corrects a $\delta^2/4 - \epsilon$ fraction of errors in an adversarial channel.*

Proof: (Sketch) As before, using Theorem 3, we have that G is a (α, ρ)-expander, where $\alpha = 2\rho - (\lambda/d)$. Setting $\rho = (\delta/4 - \epsilon')$ (we later specify $\epsilon' > 0$ s.t. $d(\delta/4 - \epsilon')$ is an integer), we get $\alpha = \delta/2 - 2\epsilon' - (\lambda/d)$. By Theorem 11, for the LP decoder to fail, there must be an error core with more than $\alpha(\delta/4 - \epsilon')dM$ edges. All edges in an error core represent errors, and so there must have been at least $\alpha(\delta/4 - \epsilon')dM = (\delta - 4\epsilon' - 2\lambda/d)(\delta/4 - \epsilon')N$ errors in the channel. This can be made greater than $(\delta^2/4 - \epsilon)N$ by increasing d and decreasing ϵ', maintaining that $d(\delta/4 - \epsilon')$ is an integer. ■

7 Future work

We have showed that LP decoding is a strong enough technique to achieve the capacity of an arbitrary MSB channel by using expander codes. However, we still have a lot to learn about the superb empirical performance of more practical codes like turbo codes and LDPC codes. Since LP decoders apply to these codes (see [Fel03]), we should be able use the techniques developed here to answer the following open questions:

(i) Can we prove capacity (or near capacity) results for LP decoding on LDPC codes? Can we get such results where the degree does not depend on the gap to capacity?

(ii) Can we give a turbo code where LP decoding has a word error rate of $2^{-\Omega(n^\epsilon)}$ for some constant $0 < \epsilon < 1$, for rates close to capacity? (We know that ϵ must be less than one, since turbo codes are known to have sub-linear distance in general [BMMS01].)

Acknowledgments

We would like to thank Rocco Servedio, Gilles Zémor and G. David Forney Jr. for helpful discussions.

References

[AC88] N. Alon and F. R. K. Chung. Explicit construction of linear sized tolerant networks. *Discrete Math*, 72:15–19, 1988.

[Bar98] A. Barg. Complexity issues in coding theory. *V.S. Pless and W.C. Huffman, editors, Handbook of Coding Theory, Elsevier Science*, pages 649–754, 1998.

[BGT93] C. Berrou, A. Glavieux, and P. Thitimajshima. Near Shannon limit error-correcting coding and decoding: turbo-codes. *Proc. IEEE International Conference on Communication (ICC), Geneva, Switzerland*, pages 1064–1070, May 1993.

[BMMS01] L. Bazzi, M. Mahdian, S. Mitter, and D. Spielman. The minimum distance of turbo-like codes. *manuscript*, 2001.

[BZ82] E. L. Blokh and V. V. Zyablov. Linear concatenated codes. *Nauka, Moscow*, 1982.

[BZ02] A. Barg and G. Zémor. Error exponents of expander codes. *IEEE Transactions on Information Theory*, 48(6):1725–1729, 2002.

[BZ03] A. Barg and G. Zémor. Concatenated codes: Serial and parallel. Manuscript, submitted to IEEE Transactions on Information Theory, 2003.

[BZ04] A. Barg and G. Zémor. Error exponents of expander codes under linear-complexity decoding. *SIAM Journal on Discrete Math*, 17(3):426–445, 2004.

[CFRU01] S.-Y. Chung, G. D. Forney, T. Richardson, and R. Urbanke. On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit. *IEEE Communications Letters*, 5(2):58–60, February 2001.

[DPR+02] C. Di, D. Proietti, T. Richardson, E. Telatar, and R. Urbanke. Finite length analysis of low-density parity check codes. *IEEE Transactions on Information Theory*, 48(6), 2002.

[Fel03] J. Feldman. *Decoding Error-Correcting Codes via Linear Programming*. PhD thesis, Massachusetts Institute of Technology, 2003.

[FK02] J. Feldman and David R. Karger. Decoding turbo-like codes via linear programming. *Proc. 43rd annual IEEE Symposium on Foundations of Computer Science (FOCS)*, November 2002. To appear in *Journal of Computer and System Sciences*.

[FKKR01] G. D. Forney, R. Koetter, F. R. Kschischang, and A. Reznik. On the effective weights of pseudocodewords for codes defined on graphs with cycles. In *Codes, systems and graphical models*, pages 101–112. Springer, 2001.

[FKV01] B. Frey, R. Koetter, and A. Vardy. Signal-space characterization of iterative decoding. *IEEE Transactions on Information Theory*, 47(2):766–781, 2001.

[FKW02] 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 Conference on Communication, Control, and Computing*, October 2002.

[FKW03] J. Feldman, D. R. Karger, and M. J. Wainwright. LP decoding. In *Proc. 41st Annual Allerton Conference on Communication, Control, and Computing*, October 2003.

[FMS+04] 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.

[For66] G. D. Forney. *Concatenated Codes*. M.I.T. Press, 1966.

[FWK03] J. Feldman, M. J. Wainwright, and D. R. Karger. Using linear programming to decode linear codes. *37th annual Conference on Information Sciences and Systems (CISS '03)*, March 2003. Submitted to *IEEE Transactions on Information Theory*, May, 2003.

[Gal62] R. Gallager. Low-density parity-check codes. *IRE Trans. Inform. Theory*, IT-8:21–28, Jan. 1962.

[Gal68] R. Gallager. *Information Theory and Reliable Communication*. John Wiley & Sons, New York, NY, 1968.

[GI02] V. Guruswami and P. Indyk. Near-optimal linear-time codes for unique decoding and new list-decodable codes over smaller alphabets. In *Proc. of the 34th annual Symposium on Theory of Computing (STOC)*, 2002.

[GI04] V. Guruswami and P. Indyk. Efficiently decodable low-rate codes meeting gilbert varshamov bound. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2004.

[KV03] R. Koetter and P. O. Vontobel. Graph-covers and iterative decoding of finite length codes. In *Proc. 3rd International Symposium on Turbo Codes*, September 2003.

[LMSS98] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. Spielman. Improved low-density parity-check codes using irregular graphs and belief propagation. *Proc. 1998 IEEE International Symposium on Information Theory*, page 117, 1998.

[RU01] T. Richardson and R. Urbanke. The capacity of low-density parity-check codes under message-passing decoding. *IEEE Transactions on Information Theory*, 47(2), February 2001.

[Sha48] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656, 1948.

[Spi95] D. Spielman. *Computationally Efficient Error-Correcting Codes and Holographic Proofs*. PhD thesis, Massachusetts Institute of Technology, 1995.

[SR03] V. Skachek and R. Roth. Generalized minimum distance iterative decoding of expander codes. In *Proc. IEEE Information Theory Workshop*, 2003.

[SS96] M. Sipser and D. Spielman. Expander codes. *IEEE Transactions on Information Theory*, 42(6):1710–1722, 1996.

[Wib96] N. Wiberg. *Codes and Decoding on General Graphs*. PhD thesis, Linkoping University, Sweden, 1996.

[ZÓ1] G. Zémor. On expander codes. *IEEE Transaction of Information Theory*, 47(2):835–837, 2001.

[Zya71] V. V. Zyablov. An estimate of the complexity of constructing binary linear cascaded codes. *Problemy Peridachi Informatsii*, 15(2):58–70, 1971.

A Noisy coding theorem generalization

Let Σ be the set of possible symbols received from an MSB channel with capacity $\mathcal{C}$. For some codeword $y \in \{0,1\}^n$, and received word $\hat{y} \in \Sigma^n$, let $p(\hat{y}|y)$ be the probability of receiving $\hat{y}$, given that y was sent over the channel. For some $0 \leq \beta \leq 1$, code C, codeword $y \in C$ and received word $\hat{y} \in \Sigma^n$, let $\mathcal{B}(\beta, C, y, \hat{y})$ be the set of codewords $y' \in C$, $y' \neq y$, such that

$$\frac{p(\hat{y}|y')}{p(\hat{y}|y)} \geq 2^{-\beta n}. \tag{3}$$

For some $0 \leq \beta \leq 1$, code C of rate r, and information word $x \in \{0,1\}^k$, let $P_r(C, x, \beta)$ be the probability, over the noise in the channel (which determines $\hat{y}$), that there exists some codeword $y' \in \mathcal{B}(\beta, C, y, \hat{y})$.

Let $C(x)$ denote the encoding of information word $x \in \{0,1\}^k$ in the code C. Suppose a random code C of rate r has the property that for every information word $x \in \{0,1\}^k$ and word $y \in \{0,1\}^n$, $\Pr[C(x) = y] = 2^{-n}$. Furthermore, we assume pairwise independence between the codewords; i.e., for all distinct information words $x, x' \in \{0,1\}^k$, $x' \neq x$, and words $y, y' \in \{0,1\}^n$, we have $\Pr[C(x) = y] = \Pr[C(x) = y | C(x') = y'] = 2^{-n}$. Let $\overline{P}_r(x, \beta)$ be the expectation of $P_r(C, x, \beta)$ over this random choice of C with rate r.

The classical results in coding theory [Sha48] address the case $\beta = 0$, when the event described in (3) represents an error by an optimal (maximum-likelihood) decoder, and so $P_r(C, x, \beta)$ represents the word error rate. It is known [Gal68] that

$$\overline{P}_r(x, 0) \leq 2^{-n(\mathcal{E}(r))}, \tag{4}$$

where $\mathcal{E}(r) > 0$ for all rates $r < \mathcal{C}$. The function $\mathcal{E}(r)$ is the well studied *random coding exponent*. The following more general statement is most likely known to the coding community, but we show it here for completeness:

Theorem 13 *For any $\beta > 0$, we have $\overline{P}_r(x, \beta) \leq 2^{-n(\mathcal{E}(r+\beta))}$.*

Proof: We generalize the proof in [Gal68]. Let σ be an arbitrary constant, $0 \leq \sigma \leq 1$, and let $\tau = 1/(1 + \sigma)$. Summing over choices of (i) the encoding $C(x)$ of x and (ii) the received word $\hat{y}$, we have

$$\overline{P}_r(x, \beta) = \sum_{y \in \{0,1\}^n} \Pr_C[y = C(x)] \sum_{\hat{y} \in \Sigma^n} p(\hat{y}|y) \overline{\beta}(y, \hat{y}), \tag{5}$$

where

$$\begin{aligned}
\overline{\beta}(y,\hat{y}) &= \Pr_C\left[\exists y' \in \mathcal{B}(\beta, C, y, \hat{y}) | y = C(x)\right] \\
&\le \left(\sum_{x'\in\{0,1\}^k, x'\neq x} \Pr_C\left[C(x') \in \mathcal{B}(\beta, C, y, \hat{y}) | y = C(x)\right]\right)^{\sigma} \\
&= \left(\sum_{\substack{x'\in\{0,1\}^k \\ x'\neq x}} \sum_{y'\in\mathcal{B}(\beta,C,y,\hat{y})} \Pr_C[y' = C(x') | y = C(x)]\right)^{\sigma} \\
&\le \left(\sum_{\substack{x'\in\{0,1\}^k \\ x'\neq x}} \sum_{y'\in\{0,1\}^n} 2^{-n}\left(\frac{p(\hat{y}|y')}{p(\hat{y}|y)} 2^{\beta n}\right)^{\tau}\right)^{\sigma} \\
&= p(\hat{y}|y)^{-\sigma\tau} 2^{\beta\sigma\tau n} 2^{rn\sigma}\left(\sum_{y'\in\{0,1\}^n} 2^{-n} p(\hat{y}|y')^{\tau}\right)^{\sigma}.
\end{aligned}$$

Plugging the bound on $\overline{\beta}(y,\hat{y})$ back into (5) and rearranging the sums, we have

$$\begin{aligned}
\overline{P}_r(x,\beta) &= 2^{(r+\beta\tau)\sigma n} \sum_{\hat{y}\in\Sigma^n} \sum_{y\in\{0,1\}^n} 2^{-n} p(\hat{y}|y)^{1-\sigma\tau}\left(\sum_{y'\in\{0,1\}^n} 2^{-n} p(\hat{y}|y')^{\tau}\right)^{\sigma} \\
&= 2^{(r+\beta\tau)\sigma n} \sum_{\hat{y}\in\Sigma^n} \left(\sum_{y'\in\{0,1\}^n} 2^{-n} p(\hat{y}|y')^{\tau}\right)^{\sigma} \sum_{y\in\{0,1\}^n} 2^{-n} p(\hat{y}|y)^{1-\sigma\tau}.
\end{aligned}$$

Using $\tau = 1/(1+\sigma)$, we get

$$\begin{aligned}
\overline{P}_r(x,\beta) &= 2^{(r+\frac{\beta}{1+\sigma})\sigma n} \sum_{\hat{y}\in\Sigma^n}\left(\sum_{y'\in\{0,1\}^n} 2^{-n} p(\hat{y}|y')^{1/(1+\sigma)}\right)^{1+\sigma} \\
&\le 2^{(r+\beta)\sigma n} \sum_{\hat{y}\in\Sigma^n}\left(\sum_{y'\in\{0,1\}^n} 2^{-n} p(\hat{y}|y')^{1/(1+\sigma)}\right)^{1+\sigma}
\end{aligned}$$

Note that this bound matches the one obtained (via the same analysis) on the quantity $\overline{P}_{r+\beta}(x,0)$. In other words, we have shown that increasing β from 0 to β' has the same effect on this bound as increasing the rate from r to $r+\beta'$. Since the bound above matches the one used in [Gal68] to show (4), we have

$$\overline{P}_r(x,\beta) \le 2^{-n(\mathcal{E}(r+\beta))}.$$

■

This theorem implies that $\overline{P}_r(x,\beta)$ decreases exponentially in n as long as $r+\beta < \mathcal{C}$. In other words, if $r < \mathcal{C}$, then there exists $\beta > 0$ such that $\overline{P}_r(x,\beta)$ decreases exponentially in n. We also note that for many natural channels with continuous alphabets Σ, a similar theorem can be shown using the same argument, replacing the sums over $\hat{y} \in \Sigma^n$ with integrals.

A.1 Binary linear code ensemble. In this paper, we are concerned with the ensemble of random binary linear codes. This ensemble has a distribution (for a particular codeword) that is uniform over $\{0,1\}^n$, and also has the required pairwise independence property [Gal68]. Recall that $\gamma_i = \log \frac{p(\hat{y}_i|0)}{p(\hat{y}_i|1)}$.

Theorem 6 *For any memoryless symmetric channel with capacity $\mathcal{C}$, for sufficiently large n, any rate $r < \mathcal{C}$, and any β where $0 < \beta < \mathcal{C} - r$, there exists a $(\beta, \mathcal{E}(r+\beta))$-robust binary linear code C with length n, rate*

r, and minimum distance at least $H^{-1}(1-r)$, where $\mathcal{E}$ is the random coding exponent.

Proof: By Theorem 13, we have $\overline{P}_r(x,\beta) \leq 2^{-\mathcal{E}(r+\beta)n}$. This means that the expected value of $P_r(C,x,\beta)$ (over random linear codes C) is at most $2^{-\mathcal{E}(r+\beta)n}$. Using Chebychev's inequality, a random linear code C will have $P_r(C,x,\beta) \leq 2 \cdot 2^{-\mathcal{E}(r+\beta)n}$ with probability at least $1/2$. Since a random linear code is known to achieve the G-V bound with high probability, there must be some binary linear code C that achieves the G-V bound, and has $P_r(C,x,\beta) \leq 2^{-\mathcal{E}(r+\beta)n+1}$.

Recall that $P_r(C,x,\beta)$ is the probability, over the noise in the channel, that

$$\frac{p(\hat{y}|y')}{p(\hat{y}|y)} \geq 2^{-\beta n}. \tag{6}$$

Taking the log of both sides, we have

$$\begin{aligned}
\beta n &\geq -\log \frac{p(\hat{y}|y')}{p(\hat{y}|y)} \\
&= -\log \prod_i \frac{p(\hat{y}_i|y'_i)}{p(\hat{y}_i|y_i)} \qquad (7)\\
&= -\sum_{i:y'_i=1,y_i=0} \log \frac{p(\hat{y}_i|1)}{p(\hat{y}_i|0)} - \sum_{i:y'_i=0,y_i=1} \log \frac{p(\hat{y}_i|0)}{p(\hat{y}_i|1)} \\
&= \sum_i \gamma_i y'_i - \sum_i \gamma_i y_i.
\end{aligned}$$

(To get (7), we use the fact that the channel is memoryless.) We conclude that C is $(\beta, \mathcal{E}(r+\beta))$-robust. ■

Note that if the transmitted codeword is 0^n, we have that with probability at least $1 - 2^{-\mathcal{E}(r+\beta)n+1}$, all non-zero codewords have cost at least βn.

B Expansion and orientations

In the lemmas below, when we assume that G is an (α,ρ)-expander, we assume that $0 < \alpha < 1$, $0 < \rho < 1$ and ρd is an integer.

Lemma 1 *In a d-regular (α,ρ)-expander $G=(V,E)$, if some subgraph $G'=(V',E')$ has $|E'| \leq \alpha\rho d|V|$, then $|V'| \geq |E'|/(\rho d)$.*

Proof: Suppose $|V'| < |E'|/(\rho d)$. Then since $|E'| \leq \alpha\rho d|V|$ we have $|V'| \leq \alpha|V|$. So, by expansion, we have $|E'| \leq \rho d|V'|$, a contradiction. ■

Lemma 14 *Suppose G is a (α,ρ)-expander. Let $E' \subseteq E$ be a subset of at most $\alpha\rho d|V|$ edges. For a node $j \in V$, let $Y_j = \max\{|\Gamma(j) \cap E'| - \rho d, 0\}$. Then,*

$$\sum_{j \in V} Y_j \leq |E'|.$$

Proof: Let W be the nodes $j \in V$ where $|\Gamma(j) \cap E'| \geq \rho d$. Suppose $\sum_{j\in V} Y_j > |E'|$; then, we have $\sum_{j\in W} Y_j > |E'|$, since $Y_j = 0$ for all $j \notin W$. Consider the total degree of W w.r.t. E':

$$\begin{aligned}
\sum_{j\in W} |\Gamma(j) \cap E'| &= \sum_{j\in W} (Y_j + \rho d) \\
&> |E'| + \rho d|W|.
\end{aligned}$$

Therefore, the total degree w.r.t. E' on nodes in $V - W$ is less than $|E'| - \rho d|W|$. It follows that for the edges $E'' \subseteq E'$ induced by W, we have $|E''| > \rho d|W|$. However, by Lemma 1, since $|E''| \leq \alpha\rho d|V|$, it must be the case that $|W| \geq |E''|/(\rho d)$, a contradiction. ■

Lemma 2 *If a d-regular graph G is a (α, ρ)-expander, where ρd is an integer, then all subgraphs $G' = (V', E')$ where $|E'| \leq \alpha\rho d|V|$ contain a ρ-orientation.*

Proof: For node sets A and B, let $\deg(A, B)$ be the set of edges with one endpoint in A and the other endpoint in B. For a node set A and node b, we have $\deg(A, b) = \deg(A, \{b\})$.

Consider an arbitrary induced subgraph $G' = (V', E')$ where $|E'| \leq \alpha\rho dm$. Set up a max-flow instance as follows. We will use the nodes from V' in the instance (call them $\hat{V}'$ in the context of the flow instance). Additionally, we "place" a new node $\langle e \rangle$ on every edge $e = (j, j')$ in E'; the edge e becomes two directed edges: $\langle e \rangle \to j$ and $\langle e \rangle \to j'$, each with capacity 1. Let $\hat{E}'$ denote this set of new nodes. Define two additional new nodes s and t as the source and sink of the flow. Add edges from the source s to every node $\langle e \rangle \in \hat{E}'$. with capacity 1. Add edges from every node in $\hat{V}'$ to the sink t with capacity ρd.

Note that each node $\langle e \rangle \in \hat{E}'$ has one incoming edge and two outgoing edges, and that each node in $\hat{V}'$ has d incoming edges and one outgoing edge. Since all capacities are integers, an integral flow of value $|\hat{E}'|$ gives us a set of edges from $\hat{E}'$ to $\hat{V}'$ such that at most one edge is incident to each node in $\hat{E}'$, and at most ρd edges are incident to each node in $\hat{V}'$. This can easily be transformed into an ρ-orientation in the graph G'.

It remains to show that the max-flow has value at least $|\hat{E}'|$. A general min-cut in the flow graph can be described by sets $\hat{E}'_L \subseteq \hat{E}'$ and $\hat{V}'_L \subseteq \hat{V}'$, and the size $\mathcal{M}$ of the cut is

$$\mathcal{M} = |\hat{E}' - \hat{E}'_L| + \deg(\hat{E}'_L, \hat{V}' - \hat{V}'_L) + \rho d|\hat{V}'_L|.$$

We can say that wlog, $\hat{V}'_L$ is the set of nodes $j \in \hat{V}'$ such that $\deg(\hat{E}'_L, j) > \rho d$. (This is because for a node $j \in (\hat{V}' - \hat{V}'_L)$, we must have $\deg(\hat{E}'_L, t) \leq \rho d$, since otherwise, j could be moved into $\hat{V}'_L$, reducing the the cut size; for a node $j \in \hat{V}'_L$, we may assume $\deg(\hat{E}'_L, j) > \rho d$, since otherwise, j could be moved into $\hat{V}' - \hat{V}'_L$, and the cut size is either reduced, or it remains the same.)

Therefore, we have

$$\mathcal{M} = |\hat{E}'| - |\hat{E}'_L| + \sum_{j \in \hat{V}'} \min\{\deg(\hat{E}'_L, j), \rho d\}.$$

By simple counting, we have

$$\sum_{j \in \hat{V}'} \min\{\deg(\hat{E}'_L, j), \rho d\} = 2|\hat{E}'_L| - \sum_{j \in \hat{V}'} \max\{\deg(\hat{E}'_L, j) - \rho d, 0\},$$

and so

$$\mathcal{M} = |\hat{E}'| + |\hat{E}'_L| - \sum_{j \in \hat{V}'} \max\{\deg(\hat{E}'_L, j) - \rho d, 0\}. \tag{8}$$

Consider the set $E'_L \subseteq E'$ in the original graph (these are the edges corresponding to the nodes $\hat{E}'_L$ in the flow graph.) The edges E'_L induce at most $\alpha|V|$ nodes in the original graph by assumption, and so by Lemma 14, we have

$$\sum_{j \in V} \max\{|\Gamma(j) \cap E'_L| - \rho d, 0\} \leq |E'_L|.$$

Therefore, since $|E'_L| = |\hat{E}'_L|$ and $|\Gamma(j) \cap E'_L| = \deg(\hat{E}'_L, j)$ (for all $j \in V$), we can use Equation (8) to get $\mathcal{M} \geq |\hat{E}'|$, as desired. ■

C C-symmetry

In this section we show that it is okay to assume that 0^N is the codeword transmitted over the channel. This will follows from results in [FKW03, Fel03].

Definition 5 [FKW03, Fel03] *Let $Q \subseteq [0,1]^n$ be a polytope, and let C be a binary code, such that $Q \cap \{0,1\}^n = C$. For a point $f \in Q$, and codeword $y \in C$ let the point f^y be defined as follows: For each coordinate i, set $f_i^y = |y_i - f_i|$. The polytope Q is said to be C-symmetric if, for all $f \in Q$ and $y \in C$, we have $f^y \in Q$.*

Let Q be the polytope defined in section 3 (to decode an expander code C). Let $\dot{Q}$ be the polytope Q, projected onto the edge variables f_e.

Lemma 15 *The polytope $\dot{Q}$ is C-symmetric.*

Proof: Let (f, w) be an arbitrary point in Q, and let y denote an arbitrary codeword of C, with a bit y_e for every edge $e \in E$. The point f^y has $f_e^y = |y_e - f_e|$. We will define settings of the variables $w'_{j,c}$ such that $(f^y, w') \in Q$, thus proving the lemma.

For every node j, and codeword $c \in C_j$, set $w'_{j,c+y(j)} = w_{j,c}$, where $y(j)$ denotes the portion of the codeword y on edges incident to j. For each node j, we have $\sum_{c \in C_j} w'_{j,c} = 1$, since $\sum_{c \in C_j} w_{j,c} = 1$; thus we satisfy the dual constraints on the nodes.

Now consider a particular node j and edge $e \in \Gamma(j)$. We have

$$\begin{aligned}
\sum_{c:c[e]=1} w'_{j,c} &= \sum_{c:(c+y(j))[e]=1} w'_{j,c+y(j)} \\
&= \sum_{c:(c+y(j))[e]=1} w_{j,c} \\
&= \left| y_e - \sum_{c:c[e]=1} w_{j,c} \right| \\
&= |y_e - f_e| \\
&= f_e^y
\end{aligned}$$

thus satisfying the dual constraints on the edges. ■

The following corollaries are implied by results in [FKW03, Fel03]:

Corollary 16 *The word error rate of the LP decoder using Q on C is independent of the transmitted codeword, under any memoryless symmetric channel. Thus, one may assume that 0^n is transmitted.*

Corollary 17 *The performance of the LP decoder using Q on C is independent of the transmitted codeword, under the adversarial bit-flipping channel. Thus, one may assume that 0^n is transmitted.*