princeton univ. F'07 cos 597D: thinking like a theorist

Lecture 13: LP-based Decoding of Expander Codes

Lecturer: *Sanjeev Arora* Scribe:*Aravindan Vijayaraghavan*

1 Introduction

This lecture continues in the series of lectures illustrating the use of thinking continuously. In this lecture we will demonstrate the use of Linear Programming to decode expander codes, which were introduced in *Lecture 6*. We will formulate a Linear programming relaxation for decoding expander codes and prove that the LP optimum is unique and gives the required code word. We do this by constructing a witness, from which we prove optimality and uniqueness by LP duality (or Farkas Lemma). In the next section we give a quick review of LP duality, followed by a recap of expander codes. We then proceed to describe the LP decoding algorithm and prove the correctness of the algorithm assuming the existence of a witness. Finally, we describe how to construct the witness efficiently. In fact, we use a property(symmetry) of the LP(decoder) polytope (shown in [**?**]) to give a simpler description of the decoding algorithm.

LP duality

We now give an alternate view of LP duality. We consider the system of linear inequalities

$$\bar{a_i}.\bar{x} \geq b_i \ \forall i \in \{1, 2, \ldots, m\} \tag{1}$$

We want to know when this system of linear inequalities is feasible over $\mathbf{R}$. We can try to show infeasibility by finding a positive combination of these inequalities which gives us a contradiction (like $0 \geq 1$). However, it is not clear whether this method of deriving contradictions is complete. LP duality (Farkas lemma) answers precisely this question by stating that it is always possible to come up with a contradiction for an infeasible system of linear inequalities, using a positive combination of these system of linear inequalities. Farkas lemma states that

Lemma 1
For $A \in \mathbf{R}^{m \times n}, b \in \mathbf{R}^m$, exactly one of the following is true:

- $\exists x \in \mathbf{R}^n$ *such that* $Ax \leq b$
  $x \geq 0$.

- $\exists y \in \mathbf{R}^m$ *such that* $A^T y \geq 0$
  $y^T b < 0$
  $y \geq 0$.

Recap of Expander codes (LDPC)

We now give a brief description of Expander codes (Low Density Parity Check Codes) which were covered in *Lecture 6*. Consider a bipartite (α, β)-expander graph $G(V, W, E)$ where V and W represent the two partitions of the graph. Further, we assume that all vertices of V have degree c. Let $n = |V|$ and $m = |W|$. We also need that the expansion $\beta = \delta c$, where $\delta > \frac{2}{3} + \frac{1}{3c}$ and that β is an integer. The subset of vertices V (of size n) represents the n bits in the code and subset W of vertices represents the constraints satisfied by the bits of any codeword. The constraint corresponding to a vertex $j \in W$ is $g(j)$, given by

$$\sum_{i \in \Gamma(j)}^{\oplus} x_i = 0 \tag{2}$$

where $\Gamma(j)$ represents the neighbours of j, and the addition here refers to parity or exclusive-or ($\oplus$). The expander code C is thus defined implicity as the set of vectors $x \in GF(2)^n$ which satisfies the constraint $g(j)$ for every vertex j in partition W. We further assume that the expander graph is randomly constructed (w.r.t edges) and that $m < \frac{n}{2}$. Clearly the number of vectors in code (size of code space) $|C| \geq \frac{2^n}{2^m} \geq 2^{\frac{n}{2}}$. Further, using the expansion of the graph, it is shown that $MinDistance(C) \geq \alpha$ (from the unique neighbours property).

2 Decoding Expander codes

We now briefly describe the decoding procedure. The received word is $\bar{b} = (b_1, b_2, \ldots, b_n) \in GF(2)^n$ and we are given that at most ρ bits are erroneous where $\rho \leq \frac{3\delta-2}{2\delta-1}(\alpha n - 1)$. For the purpose of this discussion we assume that vertices in W are also degree-bounded. We need to find the codeword $\bar{f} = (f_1, f_2, \ldots, f_n)$ which is closest to $\bar{b}$ (w.r.t l_1 distance) such that all the following constraints are satisfied.

$$\forall j \in W, \quad \sum_{i \in \Gamma(j)}^{\oplus} f_i = 0 \tag{3}$$

However, the constraints specified here are not linear constraints. The key idea here is to capture each non-linear constraint using linear constraints so that we can use a linear program for decoding. With this intention, we define for every constraint $j \in W$, variables $w_{j,S}$ where $S \subseteq \Gamma(j)$. The intention is that $w_{j,S}$ is 1 if all the binary variables b_i corresponding to the elements $i \in S$ are 1 and the rest of the elements ($\notin S$) are 0. We would want $w_{j,S}$ as $\{0, 1\}$ variables as well. The constraints are given below,

$$\forall j \in W, \quad \sum_{S \subseteq \Gamma(j)} w_{j,S} = 1 \tag{4}$$

$$\forall i \in V, j \in W, \qquad f_i = \sum_{S \subseteq \Gamma(j), S \ni i} w_{j,S} \tag{5}$$

$$\forall S \subseteq \Gamma(j) \text{ s.t } |S| \text{ is odd, } w_{j,S} = 0 \tag{6}$$

Figure 1: Figure shows the expander graph with $|W| = |V|/2$. For the constraint node y_1 shown in the figure, the associated constraint is $x_1 \oplus x_2 \oplus x_3 = 0$.

The objective function that needs to be minimized is dependent on the received word b and on the property of the transmitting channel. We now make a simplifying assumption about the received code-word. We describe the decoding algorithm assuming that the transmitted code-word is 0^n and that the received word $\bar{b}$ has weight $\leq \rho$. If the decoding algorithm works under this simplified assumption, it can shown [**?**] using some symmetry properties of the LP decoder polytope, that the decoding works for other received words too. A polytope P for a code C is said to be C-symmetric if for any point f in the polytope P, and codeword $y \in C$, the point f relative to y, $f^{[y]}(= |f_i - y_i|)$ is also in P. We now state the relevant result from [**?**]

Theorem 2
For any LP decoder using a C-symmetric polytope to decode C under a binary input memoryless symmetric channel, the probability that the LP decoder fails is independent of the codeword that is transmitted.

Thus, it suffices to show that the decoding algorithm works for the transmitted codeword being 0^n.

2.1 The LP Decoder

Let us assume that the channel is a binary symmetric channel. The decoder for the expander code consists of solving a linear program, which depends on the received word $\bar{b}$. The LP under the assumption that the received word $\bar{b}$ has weight ρ is given by

$$minimize \sum_{i \in V} \gamma_i f_i \tag{7}$$

$$\forall j \in W, \qquad \sum_{S \subseteq \Gamma(j)} w_{j,S} = 1 \tag{8}$$

$$\forall i \in V, j \in W, \qquad f_i = \sum_{S \subseteq \Gamma(j), S \ni i} w_{j,S} \tag{9}$$

$$\forall S \subseteq \Gamma(j) \text{ s.t } |S| = \text{odd}, \qquad w_{j,S} = 0 \tag{10}$$

$$w_{j,S}, f_i \geq 0 \tag{11}$$

where

$$\gamma_i = \begin{cases} -1 & \text{if } b_i = 1 \\ +1 & \text{if } b_i = 0 \end{cases}$$

If the channel is not a binary symmetric channel, then γ_i represents a log-likelihood ratio of the i^{th} code bit.

It is clear that this LP is solvable in polynomial time (number of constraints and variables is polynomial in n, because of the degrees being bounded). We now need to show that the vector 0^n is the unique optimal solution of the LP. The dual of the LP presented above is

$$maximize \sum_{j \in W} v_j \tag{12}$$

$$\forall j \in W, S \subseteq \Gamma(j), \sum_{i \in S} \tau_{ij} \geq v_j \tag{13}$$

$$\forall i \in V, \sum_{j \in \Gamma(i)} \tau_{ij} \leq \gamma_i \tag{14}$$

The variables $v_i, \tau_{ij} \in \Re$. The objective function value for $\bar{f} = 0, w_{j,\phi} = 1, w_{j,S \neq \phi} = 0$ is 0. We will henceforth denote this assignment of variables as $(0^n, w^0)$. We now try to obtain a certificate or witness, consisting of an assignment to the dual variables, which will prove the unique optimality of$(0^n, w^0)$.

Definition 1 *A setting of the dual variables* $\{\tau_{ij}\}$ *is a witness if*

(a) For all checks $j \in W$*, and distinct* $i, i' \in \Gamma(j)$*, we have* $\tau_{ij} + \tau_{i'j} \geq 0$.

(b) For all $i \in V$*, we have* $\sum_{j \in \Gamma(i)} \tau_{ij} < \gamma_i$.

We now use *Farkas Lemma* and *LP duality* to show that if a witness exists, $(0^n, w^0)$ is the unique optimum.

Proposition 3
If there exists a witness i.e a setting of the dual variables $\{\tau_{ij}\}$ *satisfying conditions* $(a), (b)$ *in Definition 1, then* $(0^n, w^0)$ *is the unique optimum to the LP defined above.*

PROOF: Let $\{\tau_{ij}\}$ be a witness. We first show that $(0^n, w^0)$ is an optimum solution by finding a feasible assignment to the dual variables v_j with dual objective function value 0. The required assignment is $v_j = 0\ \forall j$. Clearly, conditions $(a), (b)$ in Definition 1 imply that constraints (13) and (14) are satisfied respectively. The dual objective function value is 0, and thus by LP duality $(0^n, w^0)$ is an optimal solution.

To show that $(0^n, w^0)$ is a unique optimum, we show that if $f_i = \epsilon > 0$ for some $i \in V$, then we show by *Farkas lemma* that the objective value is > 0. If $f_{i'} = \epsilon > 0$ for some $i' \in V$, we add the constraint

$$-f_{i'} \leq -\epsilon \text{ where } \epsilon > 0$$

to the primal. Let the corresponding dual variable be z'. We need $\sum_{iinV} \gamma_i f_i \leq 0$. To show that this system is infeasible, by Farkas lemma, it suffices to find a solution to the dual variables $\{t_{ij}\}, \{v_j\}, z'$ satisfying the constraints

$$maximize \sum_{j \in W} v_j \tag{15}$$

$$\forall j \in W, S \subseteq \Gamma(j),\ \sum_{i \in S} \tau_{ij} \geq v_j \tag{16}$$

$$\forall i \in V \setminus \{i'\},\ \sum_{j \in \Gamma(i)} \tau_{ij} \leq \gamma_i \tag{17}$$

$$\sum_{j \in \Gamma(i')} \tau_{i'j} + z' \leq \gamma_{i'} \tag{18}$$

$$\sum_j v_j - \epsilon z' < 0 \tag{19}$$

Such a solution clearly exists with the values of τ_{ij}, v_j as before, with $0 < z' < \gamma_i' - \sum_{j \in \Gamma(i')} \tau_{ij}$ (this value of z' exists from condition (b) of Def 1). Thus $(0^n, w^0)$ is a unique optimum for the LP as required. □

The LP described above is clearly solvable in polynomial time (all vertices are of bounded degree) and this completes the description of the decoding algorithm. This procedure can also be extended to the case where the degree of vertices in W is not bounded, but it is left as an exercise.

3 Using the Expansion to find a Witness

We now use the expansion of the graph to find such a witness i.e. assign appropriate edge weights τ_{ij} that satisfy Definition (1). To do this, we now define a δ-matching and show that if such a δ-matching exists, then a witness can be constructed. We then show such a δ-matching exists using the expansion of the graph. For the purpose of the rest of the lecture, we use $\lambda = 2(1-\delta) + \frac{1}{c}$, where $\beta = \delta c$ is the expansion of the graph. Let $U = \{i \in V | \gamma_i = -1\}$ and let $\dot{U}$ be the set of variables $\notin U$ that have more that $(1-\lambda)c$ neighbours in $\Gamma(U)$. Also define $U' = U \cup \dot{U}$. Here U corresponds to the bits of the received word which are 1.

Definition 2 *A δ-matching of U is a subset of edges M which are incident to U satisfying the following conditions:*

1. *Every constraint $j \in \Gamma(U)$ is incident to at most one edge in M.*
2. *Every node $i \in U$ is incident to at least δc edges of M.*
3. *Every node $i' \in \dot{U}$ is incident to at least λc edges of M.*

Intuitively, we see that such a δ-matching of U can be constructed by applying the matching algorithm used in *Lecture 5* repeatedly. However, we prove the existence of this δ-matching in the next subsection 3.1. We now show that if such a δ-matching for U exists, then we can obtain a witness i.e. edge weights $\{\tau_{ij}\}$ satisfying Definition 1.

Lemma 4
If a δ-matching for U exists, then a witness i.e. a setting of values $\{\tau_{ij}\}$ satisfying Definition 1 exists.

Proof: Let x be a positive constant such that $\frac{1}{(2\delta-1)c} < x < \frac{1}{(1-\lambda)c}$. We now set the values $\{\tau_{ij}\}$ as follows:

- For every edge $e = (i, j) \in M$, we set $\tau_{ij} = -x$ and we set $\tau_{i'j} = x$ for all $i' \in \Gamma(j)$. We observe that this is consistent since each constraint $j \in W$ is incident on at most one edge of M.
- We set all other $\tau_{ij} = 0$.

Clearly this assignment satisfies property (a) of Definition 1, since no two edges incident on $j \in W$ get negative weights and it is impossible for exactly one vertex of U to be a neighbour of j. We now check that property (b) is also satisfied. We observe that all edges $e \in M$ incident on $i \in U$ get weight $-x$, other edges of M receive weight 0 and other edges get weight 0 or x.
If $i \in U$, $\gamma_i = -1$. There are atleast δc edges from M incident on i, each having weight $-x$. Therefore

$$\sum_{j \in \Gamma(i)} \tau_{ij} \leq \delta c(-x) + (1-\delta)cx = (1-2\delta)cx < -1 < \gamma_i$$

If $i \in \dot{U}$, $\gamma_i = 1$. There are at most $(1-\lambda)c$ edges which are $\notin M$ and they contribute at most $(1-\lambda)cx$ weight, while the remaining edges (atleast λc of them) are in M and have weight 0. Hence

$$\sum_{j \in \Gamma(i)} \tau_{ij} \leq (1-\lambda)cx < 1 < \gamma_i$$

Finally, if $i \notin U'$, still $\gamma_i = 1$. At most $(1-\lambda)c$ edges are incident on $\Gamma(U)$ and only these can contribute non-zero weight. Hence

$$\sum_{j \in \Gamma(i)} \tau_{ij} \leq (1-\lambda)cx = (1-\lambda)cx < 1 < \gamma_i$$

□

3.1 Existence of δ-matching of U

We now prove (as in [?]) that there exists a δ-matching of U when the graph G has expansion $(\alpha, \delta c)$ expansion, where $\delta > \frac{2}{3} + \frac{1}{3c}$. We first show that the size of $\dot{U}$ is small by arguing that if $\dot{U}$ is large, then the expansion property is violated as many neighbours of $\dot{U}$ are in $\Gamma(U)$.

Lemma 5
If the number of erroneous bits $|U| \leq \rho$, then $|\dot{U}| \leq \frac{1-\delta}{3\delta-2}|U|$, where $\rho \leq \frac{3\delta-2}{2\delta-1}(\alpha n - 1)$.

Proof: For convenience, let $\nu = \frac{1-\delta}{3\delta-2}$. Assume to the contrary that $|\dot{U}| > \nu|U|$. Then there exists some $\tilde{U} \subseteq \dot{U}$ such that $|\tilde{U}| = \lfloor \nu|U| \rfloor + 1$. Now, $|U \cup \tilde{U}| \leq \rho(1+\nu) + 1 \leq \alpha n$. By the expansion property, $|\Gamma(U \cup \tilde{U})| \geq c\delta(|U| + |\tilde{U}|)$.

Further $|\Gamma(U \cup \tilde{U})| = \leq c|U| + |\Gamma(\tilde{U}) \setminus \Gamma(U)|$. But each node in $\tilde{U}$ has at most $\lambda c - 1$ edges which are not incident on $\Gamma(U)$. Thus $|\Gamma(U \cup \tilde{U})| \leq c|U| + (\lambda c - 1)|U|$ which contradicts the lower bound due to the expansion. □

Now, note that $|U'| = |U| + |\dot{U}| \leq \alpha n$, and thus the expansion property holds for U'. We construct the $\delta - matching$ of U by finding the maximum flow of a directed graph H which is constructed from G as follows. The vertices of H correspond to the vertices U', $\Gamma(U')$ and two other vertices designated as the source s and sink t. For every edge $(i,j) \in G$, where $i \in U', j \in \Gamma(U')$, we include a directed edge (i,j) in H with capacity 1. We also add directed edges of capacity δc from s to every node $i \in U'$. Similarly, we include a directed edge of capacity 1 from each vertex of $\Gamma(U')$ to t. We note that since every edge capacity is integral, the maximum flow is integral.

Proposition 6
If there is a flow of value $\delta c|U'|$ in H, there is a δ-matching M.

Proof: To prove the claim, we consider an integral flow(f) and we set

$$M = \{e = (i,j) | i \in U', j \in \Gamma(U') \text{ and } f(e) = 1$$

Since f has value $\delta c|U'|$, every edge $(s,i), i \in U'$ is saturated. Hence, exactly δc edges out of each vertex $i \in U'$ is saturated with a unit flow. This satisfies the second and third conditions in Def2 ($\lambda < \delta$). Since edges to t from $\Gamma(U')$ is of capacity 1 each, there is atmost one edge of non-zero flow incident on each vertex $\Gamma(U')$. Thus the first condition of Def2 is also satisfied. □

Now to complete the proof, we just need to show that the value of the maximum flow in H is $\delta c|U'|$. Equivalently, we show that the value of the min-cut in H is $\delta c|U'|$. Let V_s, W_s represent the vertices of V and W respectively, on the source side of the min-cut. V_t, W_t be the corresponding vertices on the sink-side in the min-cut. It is easily seen that without loss of generality, there are no edges in the min-cut from V_s to W_t (for those edges (i,j), we move j to the source-side).
The edges contributing to the min-cut are those from s to V_t and from W_s to t. Thus

$$MinCut\ value \geq \delta c|V_t| + |W_s| \geq \delta c|V_t| + |\Gamma(V_s)| \geq \delta c|V_t| + \delta c|V_s| = \delta c|U'|$$

This gives the construction (and existence) of δ-matching for U.

Thus, the LP decoder described by the linear program (7) decodes correctly when the transmitted codeword is 0^n and the number of bit errors is $\leq \rho$. As described earlier, this suffices to show that an LP decoder works for all other transmitted codewords as well.

References

[1] J. Feldman, T. Malkin, R. Servedio, C. Stein, and M. Wainwright. Lp decoding corrects a constant fraction of errors. In *Technical Report TR-2003-08, Operations Research, Columbia University*, 2003.

[2] J. Feldman, D. R.Karger, and M. Wainwright. Lp decoding. In *Proceedings of 41st Annual Allerton Conference on Communiction, Control and Computing*, 2003.