Soham Chatterjee
Email: sohamc@cmi.ac.in
Course: Expander Graphs and Applications

Assignment - 1
Roll: BMC202175
Date: March 5, 2024

Problem 1 Problem 4.9 (The Replacement Product): Pseudorandomness By Salil Vadhan

Given a D_1-regular graph G_1 on N_1 vertices and a D_2-regular graph G_2 on D_1 vertices consider the following graph $G_1 \textcircled{r} G_2$ on vertex set $[N_1] \times [D_1]$: vertex (u, i) is connected to (v, j) iff

(a) $u = v$ and (i, j) is an edge in G_2 or,

(b) v is the i'th neighbour of u in G_1 and u is the jth neighbor of v.

That is, we “replace” each vertex v in G_1 with a copy of G_2, associating edge incident to v with one vertex of G_2.

1. Prove that there is a function g such that if G_1 has spectral expansion $\gamma_1 > 0$ and G_2 has spectral expansion $\gamma_2 > 0$ (and both graphs are undirected) then $G_1 \textcircled{r} G_2$ has spectral expansion $g(\gamma_1, \gamma_2, D_2) > 0$.

   [Hint: Note that $(G_1 \textcircled{r} G_2)^3$ has $G_1 \textcircled{z} G_2$ as a subgraph]

2. Show how to convert an explicit construction of constant degree (spectral) expanders into an explicit construction of degree 3 (spectral) expanders.

3. Without using Theorem 4.14, prove an analogue of Part 1 for edge expansion. That is, there is a function h such that if G_1 is an $\left(\frac{N_1}{2}, \epsilon_1\right)$ edge expander and G_2 is a $\left(\frac{D_1}{2}, \epsilon_2\right)$ edge expander then $G_1 \textcircled{r} G_2$ is a $\left(\frac{N_1 D_1}{2}, h(\epsilon_1, \epsilon_2, D_2)\right)$ edge expander where $h(\epsilon_1, \epsilon_2, D_2) > 0$ if $\epsilon_1, \epsilon_2 > 0$.

   [Hint: Given any set S of vertices of $G_1 \textcircled{r} G_2$, partition S into the clouds that are more than “half-full” and those that are not]

4. Prove that the functions $g(\gamma_1, \gamma_2, D_2)$ and $h(\epsilon_1, \epsilon_2, D_2)$ must depend on D_2 by showing that $G_1 \textcircled{r} G_2$ cannot be a $\left(\frac{N_1 D_1}{2}, \epsilon\right)$ edge expander if $\epsilon > \frac{1}{D_2+1}$ and $N_1 \geq 2$

Solution:

1. Let A_1 and A_2 denote the normalized adjacency matrices of G_1 and G_2 respectively. The degree of the new graph $G_1 \textcircled{r} G_2$ is $D_2 + 1$. Now denote $B \triangleq I_{N_1} \otimes A_2$ and A be a $N_1 \cdot D_1 \times N_1 \cdot D_1$ matrix where

$$A[(u, i), (v, j)] = \begin{cases} 1 & \text{when } i\text{th neighbor of } u \text{ is } v \text{ and } j\text{th neighbor of } v \text{ is } u \text{ in } G_1 \\ 0 & \text{otherwise} \end{cases}$$

Therefore the adjacency matrix of the graph $G_1 \textcircled{r} G_1$ is $A + D_2 B$. Therefore the normalized adjacency matrix, M

$$M \triangleq \frac{A + D_2 B}{D_2 + 1}$$

Now notice the graph $(G_1 \textcircled{r} G_2)^3$ contains the graph $G_1 \textcircled{z} G_2$ as a subgraph. Hence

$$M^3 = \left[\frac{A + D_2 B}{D_2 + 1}\right]^3 = \frac{D_2^2}{(D_2+1)^3} BAB + \left[1 - \frac{D_2^2}{(D_2+1)^3}\right] C$$

for some matrix C. Lets denote $p := \frac{D_2^2}{(D_2+1)^3}$. Then $M^3 = pBAB + (1-p)C$. Hence for any $v \perp u$ where u is the uniform vector we have

$$\|M^3 v\| \leq p\|BABv\| + (1-p)\|Cv\|$$

Now we can think as C is a normalized adjacency matrix of an undirected graph. Hence for all $v \perp u$ we have $\|Cv\| \leq \|v\|$. Now we know for all $v \perp u$

$$\|BABv\| \leq (\lambda_1 + \lambda_2 + \lambda_2^2)\|v\|$$

where $\lambda_1 = 1 - \gamma_1$ and $\lambda_2 = 1 - \gamma_2$. Hence

$$\|M^3v\| \leq p(\lambda_1 + \lambda_2 + \lambda_2^2)\|v\| + (1-p)\|v\| = [p(\lambda_1 + \lambda_2 + \lambda_2^2) + (1-p)]\|v\|$$

Suppose $\max\limits_{v \perp u} \frac{\|M^3v\|}{\|v\|} = \lambda$. Then we have $\lambda = (1 - g(\gamma_1, \gamma_2, D_2))^3$. Therefore we have

$$\begin{aligned}\lambda = \max_{v\perp u} \frac{\|M^3v\|}{\|v\|} \leq \max_{v \perp u} \frac{\|(pBAB + (1-p)C)v\|}{\|v\|} \\ \leq \max_{v\perp u} \frac{[p(\lambda_1 + \lambda_2 + \lambda_2^2) + (1-p)]\|v\|}{\|v\|} = [p(\lambda_1 + \lambda_2 + \lambda_2^2) + (1-p)]\end{aligned}$$

Hence

$$(1 - g(\gamma_1, \gamma_2, D_2))^3 \leq= [p(\lambda_1 + \lambda_2 + \lambda_2^2) + (1-p)]$$

Now

$$\begin{aligned}1 - [p(\lambda_1 + \lambda_2 + \lambda_2^2) + (1-p)] &= 1 - (1-p) - p(\lambda_1 + \lambda_2 + \lambda_2^2) \\ &= p - p(\lambda_1 + \lambda_2 + \lambda_2^2) \\ &= p[1 - (\lambda_1 + \lambda_2 + \lambda_2^2)]\end{aligned}$$

Now we know

$$\lambda_1 + \lambda_2 + \lambda_2^2 < 1 \iff 0 < 1 - (\lambda_1 + \lambda_2 + \lambda_2^2) < 1 \qquad \text{and} \qquad 0 < p < 1$$

Then $0 < p[1 - (\lambda_1 + \lambda_2 + \lambda_2^2)] < 1$. Hence

$$0 < p(\lambda_1 + \lambda_2 + \lambda_2^2) + (1-p) < 1$$

Now

$$\begin{aligned}1 - g(\gamma_1, \gamma_2, D_2) &= \left[p(\lambda_1 + \lambda_2 + \lambda_2^2) + (1-p)\right]^{\frac{1}{3}} \\ &= \left[1 - p[1 - (\lambda_1 + \lambda_2 + \lambda_2^2)]\right]^{\frac{1}{3}} \\ &\leq 1 - \frac{1}{3}p[1 - (\lambda_1 + \lambda_2 + \lambda_2^2)] < 1\end{aligned}$$

So

$$g(\gamma_1, \gamma_2, D_2) = 1 - \left[p(\lambda_1 + \lambda_2 + \lambda_2^2) + (1-p)\right]^{\frac{1}{3}} > 0$$

2. First we will prove some lemmas

   Lemma 1: Eigenvalues of the permutation $\sigma \in S_n$ where $\sigma = (12\cdots n)$ are all the n-th roots of unity.

   Proof: The permutation matrix of σ is

$$P = \begin{bmatrix} 0 & 1 \\ I_{n-1} & 0 \end{bmatrix}$$

   Now by Wikipedia: Circulant Matrix Any circulant matrix looks like

$$C = \begin{bmatrix} c_0 & c_{n-1} & \cdots & c_2 & c_1 \\ c_1 & c_0 & c_{n-1} & & c_2 \\ \vdots & c_1 & c_0 & \ddots & \vdots \\ c_{n-2} & & \ddots & \ddots & c_{n-1} \\ c_{n-1} & c_{n-2} & \cdots & c_1 & c_0 \end{bmatrix}$$

Hence P is a circulant matrix with $c_0 = 0$, $c_1 = 1$ and for all $i \in [n] - \{1\}$, $c_i = 0$. Hence from the same reference we get that for all $j \in [n-1] \cup \{0\}$, the jth eigenvalue λ_j is

$$\lambda_j = c_0 + c_1\omega^j + c_2\omega^{2j} + \cdots + c_{n-1}\omega^{(n-1)j} = \omega^j$$

where $\omega = e^{\frac{2\pi i}{n}}$. Hence the eigenvalues of P are the n-th roots of unity. □

Lemma 2: A k-cycle graph is a $\left(k, 2, 1 - \Theta\left(\frac{1}{k^2}\right)\right)$-expander.

Proof: Let P_k denote the matrix

$$P_k = \begin{bmatrix} 0 & 1 \\ I_{k-1} & 0 \end{bmatrix}$$

The the adjacency matrix of k-cycle is just $M = P_k + P_k^T$. Since P_k is unitary matrix Let S be the matrix such that $SP_kS^\dagger$ is diagonalized. Let's denote that D. Then

$$SMS^\dagger = S(P_k + P_k^\dagger)S^\dagger = SP_kS^\dagger + SP_k^\dagger S^\dagger = D + S(SP_k)^\dagger = D + (SP_kS^\dagger)^\dagger = D + D^\dagger$$

Hence the eigenvalues of M are $2\operatorname{Re}(\omega^j)$ for all $j \in [n]$ where $\omega = e^{\frac{2\pi i}{k}}$

Now the normalized adjacency matrix for the k-cycle is $\frac{1}{2}M$. Hence the eigenvalues for the normalized adjacency matrix are $\operatorname{Re}(\omega^j) = \cos\frac{2j\pi}{k}$ for all $j \in [k]$. Hence the second largest eigenvalue is when $j = 1$ i.e.

$$\cos\frac{2\pi}{k} \geq 1 - \frac{1}{2}\left(\frac{2\pi}{k}\right)^2 = 1 - \frac{2\pi^2}{k^2} = 1 - \frac{1}{\Theta(k^2)}$$

Therefore k-cycle is $1 - \frac{1}{\Theta(k^2)}$ expander. □

Now we will show an explicit construction of degree 3 expanders from an constant degree expanders. Let G be an (N, D, λ)-expander. Take H to be a D-cycle. Hence by the Lemma 2 we have H is a $\left(D, 2, 1 - \frac{1}{\Theta(D^2)}\right)$-expander. Take the graph $G' = G \textcircled{r} H$. G' is a 3 regular graph. Hence G' is a $(ND, 3, \lambda')$-expander where $1 - \lambda' > 0$ by part (1). Hence G' is a degree 3 expander.

3. Suppose V_1, V_2 denote the vertex sets of G_1 and G_2 respectively. Hence we can denote the vertex set of $G_1 \textcircled{r} G_2$ to be $V_1 \times V_2$. Now Let S is a vertex subset of $G_1 \textcircled{r} G_2$. Now we will partition S into two groups of vertices A and B. Here A contains all the at least half filled clouds of vertices of V_1 i.e.

$$(u,v) \in S,\ (u,v) \in A \iff |\{v \mid (u,v) \in S\}| \geq \frac{D_1}{2}$$

then take $B = S - A$. Clearly B is the set of all at least half empty clouds of vertices of V_2. Also take $C = \{u \mid \exists v \in V_2, (u,v) \in A\}$. Therefore finally we get this relation

$$|S| \leq |B| + |C|D_1 \implies |C| \geq \frac{|S| - |B|}{D_1}$$

Now we will consider a relation between $|B|$ and $|S|$ of the factor $\frac{\epsilon_1}{\epsilon_1 + x}$ where we will choose x. Now we will consider 5 cases:

Case I: $|B| > \frac{\epsilon_1}{\epsilon_1 + x}|S|$

Now we have all the half filled clouds of S in B. So by only the G_2 edge expansion there are at least $D_2\epsilon_2|B|$ many edges from S to $\bar{S}$. Since $|B| > \frac{\epsilon_1}{\epsilon_1 + x}|S|$ we have at least $\frac{D_2\epsilon_1\epsilon_2}{\epsilon_1 + x}|S|$ many edges from S to $\bar{S}$. So the edge expansion here is

$$\frac{\frac{D_2\epsilon_1\epsilon_2}{\epsilon_1 + x}|S|}{(D_2 + 1)|S|} = \frac{D_2\epsilon_1\epsilon_2}{(D_2 + 1)(\epsilon_1 + x)}$$

Case I: $|B| \le \frac{\epsilon_1}{\epsilon_1+x}|S|$ **and** $|C| \le \frac{N_1}{2}$

Now suppose the set of edges going from C to $\bar{C}$ is E_C. Then some of these edges are going to B also and in $G_1 \textcircled{r} G_2$ each of the edges in E_C have exactly one neighbor in G_1. Hence at least $|E_C| - |B|$ many edges are between S and $\bar{S}$ in $G_1 \textcircled{r} G_2$. Now there are at least $|C|\epsilon_1 D_1$ many edges from C to $\bar{C}$ in G_1. Hence

$$|C|\epsilon_1 D_1 \ge \frac{|S|-|B|}{D_1}\epsilon_1 D_1 \ge |S|\left(1-\frac{\epsilon_1}{\epsilon_1+x}\right)\epsilon_1 = x|S|\frac{\epsilon_1}{\epsilon_1+x} > 2|B|$$

For the last step we want $\boxed{x>2}$.Therefore the the number of edges between S and $\bar{S}$ is at least

$$|C|\epsilon_1 D_1 - |B| > \frac{|S|-|B|}{D_1}\epsilon_1 D_1 - |B| \ge |S| - (1+\epsilon_1)|B| \ge |S|\left(1-\frac{(1+\epsilon_1)\epsilon_1}{\epsilon_1+x}\right) \ge |S|\frac{x-\epsilon_1}{x+\epsilon_1}$$

Therefore the edge expansion is

$$\frac{|S|\frac{x-\epsilon_1}{x+\epsilon_1}}{|S|(D_2+1)} = \frac{x-\epsilon_1}{(x+\epsilon_1)(D_2+1)}$$

Case I: $|B| \le \frac{\epsilon_1}{\epsilon_1+x}|S|$ **and** $\frac{3N_1}{4} \ge |C| \ge \frac{N_1}{2}$

In this case we have $|C| \ge \frac{N_1}{2}$. Therefore $|\bar{C}| \le \frac{N_1}{2}$. Therefore the number of edges between C and $\bar{C}$ is at least

$$|\bar{C}|\epsilon_1 D_1 \ge \frac{N_1\epsilon_1 D_1}{4} \ge \frac{3N_1\epsilon_1 D_1}{3\times 4} \ge \frac{|C|\epsilon_1 D_1}{3} \ge \frac{(|S|-|B|)\epsilon_1}{3} \ge x|S|\frac{\epsilon_1}{3(\epsilon_1+x)} > 2|B|$$

Hence we want $\frac{x}{3} > 2 \iff \boxed{x \ge 6}$. Hence like the earlier case we have the edges between S and $\bar{S}$ is at least

$$|\bar{C}|\epsilon_1 D_1 - |B| \ge \frac{(|S|-|B|)\epsilon_1}{3} - |B| \ge \frac{1}{3}|S| - \frac{3+\epsilon_1}{3}|B| \ge |S|\left(\frac{1}{3}-\frac{(3+\epsilon_1)\epsilon_1}{3(\epsilon_1+x)}\right) \ge |S|\frac{x-3\epsilon_1}{3(x+\epsilon_1)}$$

Therefore the edge expansion is

$$\frac{|S|\frac{x-3\epsilon_1}{3(x+\epsilon_1)}}{(D_2+1)|S|} = \frac{x-3\epsilon_1}{3(x+\epsilon_1)(D_2+1)}$$

Case I: $|B| \le \frac{\epsilon_1}{\epsilon_1+x}|S|$ **and** $\frac{3N_1}{4} \le |C|$

For this case suppose x be the number of clouds in C which have at most $\frac{3D_1}{4}$ many pairs from V_2 inside S. Then

$$x\frac{D_1}{2} + (|C|-x)\frac{3D_1}{4} \le |S| \le \frac{N_1 D_1}{2} \implies \frac{N_1 D_1 1}{2} + x\frac{D_1}{4} \ge |C|\frac{3D_1}{4} \ge \frac{9N_1 D_1}{16} \implies x \ge \frac{N_1}{4}$$

Hence there are at least $\frac{N_1}{4}$ clouds in C which have at most $\frac{3D_1}{4}$ many pairs from V_2. Now for any such v denote the set of parirs as T_v. Then $|T_v| \le \frac{3D_1}{4}$. So the number of edges between T_v and $\bar{T_v}$ is at least $|\bar{T_v}|\epsilon_2 D_2 \ge \frac{D_1}{4}D_2\epsilon_2$ and all these edges will be present in $G_1 \textcircled{r} G_2$. So the number of edges between S and $\bar{S}$ is at least $\frac{\epsilon_2 D_2 D_1 N_1}{4} \ge \frac{|S|\epsilon_2 D_2}{2}$. Hence edge expansion is $\frac{\epsilon_2 D_2}{2(D_2+1)}$.

After all this analysis we can take x to be something big for example 10. Then the corresponding edge expansion function will be

$$h(\epsilon_1,\epsilon_2,D_2) \equiv \min\left\{\frac{D_2\epsilon_1\epsilon_2}{(D_2+1)(\epsilon_1+10)}, \frac{10-\epsilon_1}{(10+\epsilon_1)(D_2+1)}, \frac{10-3\epsilon_1}{3(10+\epsilon_1)(D_2+1)}, \frac{\epsilon_2 D_2}{2(D_2+1)}\right\}$$

4. We will create S where it contains full cloud vertices. Then all the edges of G_2 are not going outwards from S only the edge corresponding to G_1 is going outside. Hence only the G_1 edges are going outside of S. So the number of edges going outside of $S \leq$ the number of G_1 edges of $S = |S|$ therefore $\epsilon_2 \leq \frac{1}{D_2+1}$. So it is not an $\epsilon > \frac{1}{D_2+1}$ edge expander

$\square$

Problem 2 Problem 4.10 (Unbalanced Vertex Expanders and Data Structures): Pseudorandomness By Salil Vadhan

Consider a $(K, (1-\epsilon)D)$ bipartite vertex expander G with N left vertices, M right vertices and left degree D.

1. For a set S of left vertices, a $y \in N(S)$ is called a *unique* neighbor of S if y is incident to exactly one edge from S. Prove that every left-set S of size at most K has at least $(1-2\epsilon)D|S|$ unique neighbors.

2. For a set S of size at most $\frac{K}{2}$, prove that at most $\frac{|S|}{2}$ vertices outside S have at least δD neighbors in $N(S)$ for $\delta = O(\epsilon)$.

Property Π: For all left vertices x, all but a $\delta = O(\epsilon)$ fraction of neighbors of x are assigned the value $\chi_S(x)$ (where $\chi_S(x) = 1$ iff $x \in S$).

3. Show that if we store an assignment satisfying Property Π then we can probabilistically test membership in S with error probability δ by reading just one bit of the data structure.

4. Show that an assignment satisfying Property Π exists provided $|S| \leq \frac{K}{2}$

   [Hint: First assign 1 to all of S's neighbors and 0 to all its non-neighbors, then try to correct the errors.]

Solution:

1. Let U be the set of unique neighbors in $N(S)$. Denote $T = \Gamma(S) - U$. Then we have $|U \cup T| \geq (1-\epsilon)D|S|$. Now we will count the number of edges between S and $\Gamma(S)$. From each vertex in S there are D edges going out. Hence total $D|S|$ many edges are going out from S. Now in $\Gamma(S)$ for each vertex in U there is exactly one edge coming from S and for each edge in T there are at least 2 edges coming from S. Hence there are at least $|U| + 2|T|$ many edges are coming towards $\Gamma(S)$. Hence we have:

   $$\begin{aligned} |U| + 2|T| \leq D|S| \iff & |U| + 2(|\Gamma(S)| - |U|) \leq D|S| \\ & \iff |U| \geq 2|\Gamma(S)| - D|S| \geq (1-\epsilon)D|S| - D|S| = (1-2\epsilon)D|S| \end{aligned}$$

   Hence there are at least $(1-2\epsilon)D|S|$ unique neighbors.

2. Let S be a set of left vertices with $|S| \leq \frac{K}{2}$. Assume the contrary. There exists a set of vertices, T more than $\frac{|S|}{2}$ outside S such that they have at lest δD neighbors in $N(S)$. Suppose $V \subseteq T$ such that $|V| = \frac{|S|}{2} + 1$. Then We can take $|S \cup V| \leq K$. All vertices in V has at least δD many common neighbors with $N(S)$. Hence all vertices have at most $(1-\delta)D$ many neighbors which are not in S. Therefore

   $$|N(S \cup V)| \leq |N(S)| + (1-\delta)D|V| \leq D|S| + (1-\delta)D|V|$$

   Since $|S \cup V| \leq K$ we have

   $$|N(S \cup V)| \geq D(1-\epsilon)|S \cup V| = (|S| + |V|)(1-\epsilon)D$$

   Therefore we have

   $$\begin{aligned} & D(|S| + |V|)(1-\epsilon) \leq D|S| + (1-\delta)D|V| \\ \iff & (1-\epsilon)(|S| + |V|) \leq |S| + (1-\delta)|V| \\ \iff & (\delta - \epsilon)|V| \leq \epsilon|S| \\ \iff & (\delta - \epsilon)\frac{|S|}{2} \leq (\delta - \epsilon)|V| \leq \epsilon|S| \qquad \left[|S| \geq \frac{|V|}{2}\right] \\ \iff & \delta \leq 3\epsilon \end{aligned}$$

   Therefore if we take $\delta > 3\epsilon$ then we will not satisfy the inequality above. This choice of delta will lead to contradiction. Hence for $\delta > 3\epsilon$ we have at most $\frac{|S|}{2}$ many vertices outside S which have at least δD many common neighbors with $N(S)$.

3. For any x we have to decide if $x \in S$ with error probability δ by reading just one bit of the data structure. We will do the following, given x, we will pick any random neighbor $v \in_R N(x)$ then output the value assigned to v.

   In this algorithm all but δ fraction of the neighbors of x are assigned the value of $\chi_S(x)$. Hence error happens when we sample the random neighbor of x from the δ fraction of the neighbors of x which are not assigned the value of $\chi_S(x)$. Therefore with this algorithm the error probability is δ and we only read one bit of the data structure.

4. We will start by first assigning 1 to all of the vertices in $N(S)$ and 0 to all the other vertices.

   Since the error probability is δ then for all $u \in S$ less than δD locations in $N(u)$ should be assigned 0 and $u \notin S$ less than δD locations in $N(u)$ should be assigned 1 . Therefore we have the condition:

   $$\forall\, S \subseteq L, |S| = n,\ \forall u \in L - S,\ |N(u) - N(S)| > (1-\delta)D$$

   for the set of left vertices of G are denoted by L.

   Now if we have $\delta > 3\epsilon$ then there are at most $\frac{|S|}{2}$ many vertices outside S which has at least δD many neighbors in $N(S)$ using part (2). So all other vertices has less than δD neighbors which are in $N(S)$. Hence for all these vertices the probability of accepting them is δ since less than δD many neighbors are colored 1. So the bad set of neighbors.

   We will call let $S \sqcup T = L$. Then by *ϵ-intersection property* of S, T we mean to say

   - $\forall\, S' \subseteq S, |\{v \in T \mid N(v) \cap N(S')| > \epsilon D\}| < |S'|$
   - $\forall\, T' \subseteq T, |\{u \in S \mid N(u) \cap N(T')| > \epsilon D\}| < |T'|$

□

Problem 3 Problem 5.5 (LDPC Codes): Pseudorandomness By Salil Vadhan

Given a bipartite multigraph G with N left-vertices and M right-vertices, we can obtain a linear code $\mathcal{C} \subseteq \{0,1\}^N$ (where we view $\{0,1\}$ as the fiels of two elements):

$$\mathcal{C} = \left\{ c \in \{0,1\}^N : \forall\, j \in [M] \bigoplus_{i \in \Gamma(j)} c_i = 0 \right\}$$

where $\Gamma(j)$ denotes the set of neighbors of vertex j. When G has small left-degree D (e.g. $D = O(1)$), then $\mathcal{C}$ is called a *low-density parity check (LDPC) code.*

1. Show that $\mathcal{C}$ has rate at least $1 - \frac{M}{N}$.
2. Show that if G is a (K, A) expander $A > \frac{D}{2}$, then $\mathcal{C}$ has minimum distance at least $\delta = \frac{K}{N}$.
3. Show that if G is a $(K, (1-\epsilon)D)$ expander for a sufficiently small constant ϵ, then $\mathcal{C}$ has a polynomial-time $(1-3\epsilon)\frac{K}{N}$-decoder. Assume that G is given as input to the decoder.

   [Hint: Given a received word $r \in \{0,1\}^n$, flip all coordinates of r for which at least $\frac{2}{3}$ of the neighboring parity checks are not satisfied, and argue that the number of errors decreases by a constant factor. It may be useful to use the results of Problem 2]

Solution:

1. Suppose A be the $M \times N$ adjacency matrix for the bipartite graph. Then we can say

   $$\mathcal{C} = \{x \in \{0,1\}^N \mid Ax \equiv 0 \bmod 2\}$$

   Because

   $$\forall\, j \in [M], \bigoplus_{i \in \Gamma(j)} x_i = 0 \iff \sum_{j \in \Gamma(j)} x_i \equiv 0 \bmod 2$$

   and A contains 1's in jth row at ith column if ith left-vertex is an neighbor of jth right-vertex. Hence $x \in \mathcal{C} \iff Ax \equiv 0 \bmod 2$.

   Now by Rank-Nullity Theorem we have

   $$rank(A) + \dim(\ker(A)) = N$$

   Now $rank(A) \leq M$. So

   $$\dim(\ker(A)) = N - rank(A) \geq N - M$$

   Hence

   $$|\mathcal{C}| \geq 2^{N-M} \implies \log|\mathcal{C}| \geq N - M$$

   Hence rate of the code $\geq \frac{N-M}{N} = 1 - \frac{M}{N}$.

2. By the above part we get $\mathcal{C} = \ker(A)$. Hence $\mathcal{C}$ is a linear code. Hence it is enough to show that $\forall\, c \in \mathcal{C}$, $wt(c) \geq K$. So assume the contrary. Let $\exists\, c \in \mathcal{C}$ where $c = (c_1, \dots, c_N)$ such that $wt(c) \leq K$. Now we take the set $S_c = \{i \mid c_i = 1\}$. Hence by the assumption $|S_c| \leq K$. Now take $\epsilon = 1 - \frac{A}{D}$. Then $A = (1-\epsilon)D$. Since $A > \frac{D}{2}$ have $\epsilon < \frac{1}{2}$. By Problem 2 part (1) we have the number of unique neighbors of S_c is at least $(1-2\epsilon)D|S_c|$. Since $\epsilon < \frac{1}{2}$ we have $(1-2\epsilon)D|S_c| > 0$. Hence there exists one unique neighbor $i \in [M]$. which is neighbor of only one $v \in S_c$. So the constraint at i is not satisfied since i has only one neighbor in S_c. So ith coordinate of Ac is not 0. But we took $c \in \mathcal{C} \iff Ac = 0 \bmod 2$. Hence contradiction.

   Therefore $\forall\, c \in \mathcal{C}$, $wt(c) \geq K$. Hence the distance of the code $\mathcal{C}$ is at least $\frac{K}{N}$.

3. First we introduce two notions which we will use. Let $G = (L, R, E)$ be an left D-regular $(K, (1-\epsilon)D)$ bipartite expander, then we define

   $$\Gamma^{odd}(S) = \{j \in R \mid |\Gamma(j) \cap S| = \text{odd}\} \quad \Gamma^{+}(S) = \{j \in R \mid |\Gamma(j) \cap S| = 1\}$$

So $\Gamma^{odd}(S)$ is the set of vertices of R which have odd neighbors in S and $\Gamma^{+}(S)$ is the set of unique neighbors of S. Now we will prove a lemma for distance of the code.

Lemma 1: $G = (L, R, E)$ is left D-regular $(K, (1-\epsilon)D)$-expander for some $\epsilon \in \left(0, \frac{1}{2}\right)$ then

$$\delta(\mathcal{C}(G)) > 2\delta(1-\epsilon)$$

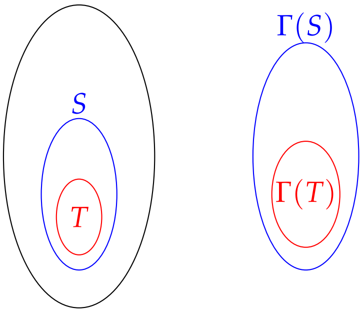
where $\delta = \frac{K}{N}$

Proof: Let c is the min weight nonzero codeword. Take $S = \{i \in L \mid c_i = 1\}$. From the previous part we have $|S| \geq \delta n$. Suppose $|S| < 2\delta(1-\epsilon)n$ for contradiction. So we have

$$\delta n \leq |S| < 2\delta(1-\epsilon)n$$

Fix any subset $T \subseteq S$ such that $|T| = \delta n$. Now

$$\begin{aligned}|\Gamma^{odd}(S)| &\geq |\Gamma^{+}(S)| \\ &\geq |\Gamma^{+}(T)| - |\Gamma(S \setminus T)| \\ &\geq (D(1-2\epsilon)\delta n) - D|S \setminus T| \\ &> (D(1-2\epsilon)\delta n) - D(\delta(1-2\epsilon))n = 0\end{aligned}$$



So $|\Gamma^{odd}(S)| > 0$. Hence there is a vertex $v \in R$ such that there is odd number of neighbors in S. Hence the constraint v is not satisfied. Hence contradiction.

□

Now let r be the received word and $c \in \mathcal{C}(G)$ be the unique codeword such that $\delta(r, c) < \delta(1-2\epsilon)N$. Denote

$$S^{(k)} = \{i \in L \mid x_i^{(k)} \neq c_i\}$$

Hence we have $|S^{(0)}| < \delta(1-2\epsilon)N$. Also we will use the set $\mathsf{UNSAT}^{(k)}$ to denote the set of unsatisfied right constraints at kth step. Similarly for $\mathsf{SAT}^{(k)}$. Now we will state the decoding algorithm then we will analyze the algorithm to show the given statement.

1 Decoding Algorithm

Algorithm 1: Linear Time Decoding Algorithm for Expander Code

```
Input: $r = (r_1, \dots, r_n)$ with promise $\exists!\ c \in \mathcal{C}(G)$ such that $\delta(r, c) < \delta(1-2\epsilon)n$
begin
    $k \longleftarrow 0$
    $x^{(k)} \longleftarrow r$
    foreach $j \in R$ do
        if $\sum_{i \in \Gamma(j)} x_i = 0$ then
            label $j$ as "SAT"
        else
            label $j$ as "UNSAT"
    foreach $i \in L$ do
        $\mathsf{SAT}_i^{(k)} = \{j \in \Gamma(i) \mid j$ labeled "SAT"$\}$
        $\mathsf{UNSAT}_i^{(k)} = \{j \in \Gamma(i) \mid j$ labeled "UNSAT"$\}$
    while $\exists\ i \in L$ s.t. $|\mathsf{UNSAT}_i^{(k)}| > \frac{2}{3}|\Gamma(i)|$ do
        $x_i^{(k+1)} \longleftarrow 1 - x_i^{(k)}$
        $x_{i'}^{(k+1)} \longleftarrow x_i^{(k)}$ for all $i' \neq i$
        Update $\mathsf{SAT}_i^{(k)}$ and $\mathsf{UNSAT}_i^{(k)}$
        $k \longleftarrow k + 1$
    return $x^k$
```

2 Analysis

Lemma 2: If $\epsilon \in \left(0, \frac{1}{6}\right)$ and $0 < |S^{(k)}| \leq \delta N$ then $\exists\ i \in L$ such that $|\mathsf{UNSAT}_i^{(k)}| > \frac{2}{3}|\Gamma(i)|$.

Proof: First notice that all unique neighbors of $S^{(k)}$ are unsatisfied at kth iteration. $\epsilon \in \left(0, \frac{1}{6}\right)$ hence the graph has $(1-2\epsilon)D|S|$ unique neighbors for any $S \subseteq L$ with $|S| \leq \delta N$ by Problem 2 part (1). Hence

$$|\mathsf{UNSAT}^{(k)}| \geq |\Gamma^+(S^{(k)})| \geq (1-2\epsilon)D|S^{(k)}| > \frac{2D}{3}|S^{(k)}|$$

Hence, $\exists\ i \in S^{(k)}$ such that $|\mathsf{UNSAT}_i^{(k)}| > \frac{2D}{3}$. Now the degree of i is $D \implies |\Gamma(i)| = D$. Hence $|\mathsf{UNSAT}_i^{(k)}| > \frac{2}{3}|\Gamma(i)|$.

□

Observation:

- The number of unsatisfied right constraints is always decreasing.
- $|S^{(k)} - S^{(k+1)}| = 1$

Lemma 3: $|S^{(0)}| < \delta(1-2\epsilon)N \implies |S^{(k)}| < \delta N$.

Proof: Initially $\mathsf{UNSAT}^{(0)} \subseteq \Gamma(S^{(0)})$ since the unsatisfied constraints are the subset of the neighbors of errors. Hence

$$|\mathsf{UNSAT}^{(0)}| \leq |\Gamma(S^{(0)})| \leq D|S^{(0)}| < D|S^{(0)}| < \delta(1-2\epsilon)DN$$

Suppose there exists a k' such that $|S^{(k')}| \geq \delta N$. By the observation there exists $k \leq k'$ such that $|S^{(k)}| = \delta N$. Hence

$$|\mathsf{UNSAT}^{(k)}| > |\Gamma^{+}(S^{(k)})| \geq \delta N \cdot (1 - 2\epsilon)D$$

But the $|\mathsf{UNSAT}^{(k)}|$ keeps decreasing so it can not start with less than $\delta(1-2\epsilon)DN$ and after that at some point is $\geq \delta N \cdot (1-2\epsilon)D$. Hence contradiction.

□

Since for each iteration the distance between $x^{(k)}$ and c is at most δN, c is the only codeword which is nearest to $x^{(k)}$. Hence the nearest codeword for each iteration stays the same.

At kth iteration suppose the number of unsatisfied constraints is nonzero and $|S^{(k)}| < \delta n$. Since number of unsatisfied constraints is nonzero $|S^{(k)}| > 0$. By Lemma 2 there exists an $i \in L$ such that $|\mathsf{UNSAT}_i^{(k)}| > \frac{2}{3}|\Gamma(i)|$. Hence the algorithm will find some vertex which has more unsatisfied constraints than satisfied constraints and flip its bit and proceed to the next iteration. With this process the number of unsatisfied constraints reduced by at least 1. Thus the algorithm will keep reducing the number of unsatisfied constraints till it becomes zero because if its not zero at any jth iteration and then $|S^{(j)}| > 0$ and hence by the above argument it will proceed. Once the number of unsatisfied constraints becomes zero cause then the final output, suppose x satisfies all the right constraints. Hence it is indeed a codeword and since the nearest codeword at each iteration stays the same $x = c$.

Therefore the above algorithm can decode with $(1-2\epsilon)\frac{K}{N}$ fraction errors. Since $(1-3\epsilon)\frac{K}{N} < (1-2\epsilon)\frac{K}{N}$ as $\epsilon \in \left(0, \frac{1}{6}\right)$ as in Lemma 2 we have a $(1-3\epsilon)\frac{K}{N}$-decoder algorithm. Now in the next section we will prove that it is a polynomial time (in fact linear time) algorithm.

3 Time Complexity

(a) Preprocessing Stage: For each $j \in R$ to check $\sum_{i \in \Gamma(j)} x_i = 0$ it takes $O(d)$ time. Hence the first for loop takes $O(md)$ time. Now for each vertex in L we keep the number of unsatisfied constraints which are neighbor of that vertex. We also keep a list of vertices in L which have more unsatisfied constraints than satisfied constraints. This can be done in $O(cn)$ time.

(b) In each iteration of the while loop instead of searching for a vertex with more unsatisfied constraints than satisfied constraints we remove an element of Q.

After flipping the vertex we update the list of unsatisfied constraints in R in $O(c)$ time. Then we will update the number of unsatisfied constraints associated with each element of in L which are neighbors of the neighbors of i i.e. the vertices in $\Gamma(\Gamma(i))$ in $O(cd)$ time. Since after the bit flip the previously unsatisfied constraints are satisfied in $\Gamma(i)$ and the previously satisfied constraints are now unsatisfied. For each vertex $j \in \Gamma(i)$ if j was previously unsatisfied then we will subtract 1 from the number of unsatisfied constrains of the neighbors of j and if j was previously satisfied then we will add 1 for any previously satisfied constraint to the number of unsatisfied constrains of the neighbors of j. Now from Q we will remove the elements which have lesser unsatisfied constraints than satisfied constraints and add the.

After updating the number of unsatisfied constraints of each vertex in $\Gamma(\Gamma(i))$ we will add the vertices which have more unsatisfied constraints than satisfied constraints into Q and remove the vertices which have lesser unsatisfied constrains than satisfied constraints. This all can be done in $O(cd)$ time since $|\Gamma(\Gamma(i))| \leq cd$. Since c, d are constants every thing inside each iteration can be done in constant time.

(c) In each iteration the number of unsatisfied constraints reduces by at least 1. The original number of unsatisfied constraints is at most $c\delta(1-2\epsilon)n$. (Lemma 2). Then the total number of iterations is at most $c\delta(1-2\epsilon)n = O(n)$.

Hence the algorithm decodes the received word in $O(n)$ time.

□

Problem 4

Write a short analysis of the Zig-Zag product construction the way we used int he algorithm of Reingold