GEORGIA TECH FALL'04
CS 8002 : PCPS AND HARDNESS OF APPROXIMATION

Lecture 6: Minimum Distance of a Linear Code

Lecturer: *Subhash Khot* Scribe: *Parikshit Gopalan*

1 Introduction

In this lecture we will prove hardness results for some natural algorithmic problems in coding theory. We will work with linear codes over $\mathbb{F}_2$. The results can be extended to linear codes over any finite field.

DEFINITION 1 *An $[n, k]_2$ linear code $\mathcal{C}$ is a $k-$dimensional subspace of $\mathbb{F}_2^n$.*

The code $\mathcal{C}$ can be succinctly presented in two ways:

1. By an $n \times k$ generator matrix $\mathbf{G}$. Let $\{\mathbf{v_1}, \cdots, \mathbf{v_k}\}$ be a basis for $\mathcal{C}$. Let $\mathbf{G} = [\mathbf{v_1}, \cdots, \mathbf{v_k}]$. Then $\mathcal{C} = \{\mathbf{Gm} | \mathbf{m} \in \{0,1\}^k\}$.
2. By an $n-k \times n$ parity check matrix $\mathbf{H}$. Since $\mathcal{C}$ is a k-dimensional space, the orthogonal space $\mathcal{C}^\perp$ has dimension $n-k$. Let $\{\mathbf{w_1}, \cdots, \mathbf{w_{n-k}}\}$ be a basis for $\mathcal{C}^\perp$. Let $\mathbf{H}^t = [\mathbf{w_1}, \cdots, \mathbf{w_{n-k}}]$. Then $\mathcal{C} = \{\mathbf{c} \in \{0,1\}^n | \mathbf{Hc} = \mathbf{0}\}$.

Given the matrix $\mathbf{G}$, one can compute $\mathbf{H}$ via Gaussian elimination and vice versa, hence we will assume that both matrices are given as input.

Let $\mathbf{d}(\mathbf{x}, \mathbf{y})$ denote the Hamming distance between $\mathbf{x}$ and $\mathbf{y}$. Let $\mathbf{wt}(\mathbf{x})$ denote the Hamming weight of $\mathbf{x}$.

DEFINITION 2 *The minimum distance d of a code is defined as*

$$d = \min_{\mathbf{c}, \mathbf{c'} \in \mathcal{C}, \mathbf{c} \neq \mathbf{c'}} \mathbf{d}(\mathbf{c}, \mathbf{c'}).$$

For a linear code, since $\mathbf{d}(\mathbf{c}, \mathbf{c'}) = \mathbf{d}(\mathbf{c} + \mathbf{c'}, \mathbf{0})$ and $\mathbf{c} + \mathbf{c'}, \mathbf{0} \in \mathcal{C}$

$$d = \min_{\mathbf{c} \neq \mathbf{0} \in \mathcal{C}} \mathbf{wt}(\mathbf{c}).$$

In this lecture we will study two natural algorithmic problems about linear codes.

§1 **Minimum Distance of a Code (MDC):** Given a linear code $\mathcal{C}$ find its minimum distance d.

§2 **Nearest Codeword Problem (NCP):** Given a linear code $\mathcal{C}$ and a target vector $\mathbf{t} \in \{0,1\}^n$, find a codeword $\mathbf{c}$ such that $\mathbf{d}(\mathbf{c}, \mathbf{t})$ is minimized.

One of the motivations to study **MDC** is that in many scenarios randomly generated linear codes are known to perform very well and in some cases are even better than the best known explicit constructions. Hence a natural approach to design a code might be to generate a code at random and check that it indeed has good distance. The next theorem says that it is hard to even approximate the minimum distance for any arbitrary linear code. This theorem is due to Dumer, Miccancio and Suidan [1]. The proof we present here is a slight variation of their proof.

THEOREM 1

- **MDC** *is hard to approximate within any constant factor unless* $\mathbf{N}P \subseteq BPP$.
- **MDC** *is hard to approximate within factor* $2^{\log^{1-\varepsilon} n}$ *for any* $\varepsilon > 0$ *unless* $\mathbf{N}P \subseteq Quasi - BPP$.

To prove this theorem, we will first show a hardness result for **NCP**.

THEOREM 2

NCP *is hard to approximate within any constant factor unless* $\mathbf{P} = \mathbf{NP}$.

PROOF:We use a reduction from **MAX-3-LIN**. By Hastad's theorem, given a system of equations

$$\mathbf{A}\mathbf{x} = \mathbf{t} \tag{1}$$

for any $\varepsilon_0 > 0$, it is NP-complete to distinguish the following two cases.

1. YES case: More than $(1 - \varepsilon_0)$ fraction of the equations are satisfiable.
2. NO case: At most $(\frac{1}{2} + \varepsilon_0)$ fraction of the equations are satisfiable.

Consider the code $\mathcal{C}_1$ generated by the columns of $\mathbf{A}$. In other words, consider a submatrix $\mathbf{G}$ of $\mathbf{A}$ whose columns are a basis for the columns of $\mathbf{A}$. Note that the sets $\mathbf{Ax}$ and $\mathbf{Gm}$ are the same. Assume that $\mathbf{G}$ has $2D$ rows and $k < 2D$ columns. Consider the $[2D, k]_2$ code $\mathcal{C}_1$ generated by $\mathbf{G}$ and the target vector t.

1. YES case: There exists $\mathbf{c} \in \mathcal{C}_1$ such that

$$\mathbf{d}(\mathbf{c}, \mathbf{t}) \leq 2\varepsilon_0 D = \varepsilon D$$

2. NO case: For every $\mathbf{c} \in \mathcal{C}_1$,

$$\mathbf{d}(\mathbf{c}, \mathbf{t}) \geq (1 - 2\varepsilon_0)D = (1 - \varepsilon)D$$

Hence **NCP** is hard to approximate within any constant. □

We will make a first attempt at proving hardness for **MDC**. Consider the code $\mathcal{C}_2$ generated by the columns of $\mathbf{G}$ and the vector $\mathbf{t}$. Codewords in $\mathcal{C}_2$ are of the form $\mathbf{c}'$ or $\mathbf{t} + \mathbf{c}'$ for $\mathbf{c}' \in \mathcal{C}_1$. In the Yes case, there is a vector in $\mathbf{t} + \mathbf{c}' \in \mathcal{C}_2$ of weight at most εD. In the No case, clearly $\mathbf{wt}(\mathbf{t} + \mathbf{c}') \geq (1 - \varepsilon)D$. However if the codeword is of the form $\mathbf{c}' \in \mathcal{C}_1$, then it could have small weight. (Indeed in Hastad's construction, each column has weight constant.) So there are small weight vectors even in the No case, and the construction fails.

The actual proof of hardness will proceed in three steps:

1. We give a reduction from the **NCP** instance above to an intermediate code $\mathcal{C}_1$ which has the following property.

(a) In the Yes case, there are *several* codewords of weight at most $(\frac{3}{4}+\varepsilon)D$. Call these Good codewords.

(b) In the No Case, there are a *few* codewords of weight less than $(1-\varepsilon)D$. Call these Bad codewords.

The crucial ingredient in this step is the use of BCH codes (it is important that BCH codes beat the Gilbert-Varshamov bound in the parameter range we are interested in).

2. We take a random subspace $\mathcal{C}_2$ of $\mathcal{C}_1$ by introducing some random constraints. The number of constraints is such that with high probability $\mathcal{C}_2$ contains some Good codewords but no Bad codewords. Thus we get a gap of some constant factor $C > 1$.

3. We amplify the gap to an arbitrary constant by taking the k-wise tensor product. Taking tensor product $\log^k n$ times gives hardness up to almost polynomial factors.

2 BCH Codes

Given a binary linear code of length n and distance D, how large can k be? It is easy to see that a code has minimum distance D iff every $D-1$ columns in the parity check matrix $\mathbf{H}$ are linearly independent. Let us set $n-k=h$. The columns of $\mathbf{H}$ are vectors in $\mathbb{F}_2^h$. Hence the problem above asks how large does h need to be to find n vectors of which any $D-1$ are linearly independent.

Theorem 3
There exist codes with $h = D\log n$ (these parameters correspond to the Gilbert Varshamov Bound).

Proof: Let $t=\lceil \log n \rceil$. Take n distinct element $x_1,\cdots,x_n$ in the extension field $\mathbb{F}_{2^t}$. Consider the matrix

$$\mathbf{H}' = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \\ x_1^2 & x_2^2 & \cdots & x_n^2 \\ \cdots & \cdots & \cdots & \cdots \\ x_1^{D-2} & x_2^{D-2} & \cdots & x_n^{D-2} \end{bmatrix} \tag{2}$$

Any $D-1$ columns of $\mathbf{H}'$ give a Vandermonde matrix, hence the columns are linearly independent over $\mathbb{F}_{2^t}$ and hence also over $\mathbb{F}_2$. To get a matrix over $\mathbb{F}_2$ we write each element of $\mathbb{F}_{2^t}$ as a t dimensional vector over $\mathbb{F}_2$. This construction gives $h=t(D-2)+1$ which is roughly $D\log n$. $\square$

Theorem 4
There exist codes with $h = \frac{D}{2}\log n$.

PROOF: We take the previous construction and omit all the rows which have non-zero even degree. Let

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \\ x_1^3 & x_2^3 & \cdots & x_n^3 \\ \cdots & \cdots & \cdots & \cdots \\ x_1^{D-2} & x_2^{D-2} & \cdots & x_n^{D-2} \end{bmatrix} \tag{3}$$

We show that any $D-1$ columns of $\mathbf{H}$ are linearly independent over $\mathbb{F}_2$. A linear dependence over $\mathbb{F}_2$ means that a set of $d \leq D-1$ columns sum to $\mathbf{0}$. Assume these columns correspond to $y_1, \cdots, y_d \in \mathbb{F}_{2^t}$. Then we have

$$\begin{aligned} \sum_{i=1}^{d} y_i^j &= 0 \qquad \text{for } j \leq D-2, \text{ odd} \\ \Rightarrow \quad \left(\sum_{i=1}^{d} y_i^j\right)^2 &= 0 \\ \Rightarrow \quad \sum_{i=1}^{d} y_i^{2j} &= 0 \qquad \text{Since the field has characteristic 2} \end{aligned}$$

By repeated squaring, we can show for all $j \leq D-2$

$$\sum_{i=1}^{d} y_i^j = 0 \tag{4}$$

But this contradicts the fact that any d columns of $\mathbf{H}'$ are linearly independent over $\mathbb{F}_2$. Writing the columns of $\mathbf{H}$ as vectors over $\mathbb{F}_2$ gives $h = \frac{D}{2} \log n$. □

In fact this construction is optimal [2]. The $[n, n - \frac{D}{2} \log n]_2$ code constructed above are BCH codes. We will denote them by $\mathcal{B}$ and their generator matrix by $\mathbf{B}$. Since $\mathcal{B}$ had distance D, balls of radius $\frac{D}{2}$ around any point can contain at most one codeword. We show that there exists a ball of radius $\frac{3D}{4}$ (or even $(\frac{1}{2} + \varepsilon)D$ for any $\varepsilon > 0$) in a BCH code containing several codewords and that such a dense ball can be found efficiently. This will be used in our intermediate construction to show that there are *several* Good codewords in the Yes case.

LEMMA 5
There is a ball of radius $\frac{3D}{4}$ in $\mathcal{B}$ containing $\binom{n}{\frac{3D}{4}} \cdot n^{-\frac{D}{2}}$ codewords.

PROOF: A ball of radius $\frac{3D}{4}$ in $\{0,1\}^n$ contains at least $\binom{n}{\frac{3D}{4}}$ points. Hence a random ball of this radius will contain at least

$$\binom{n}{\frac{3D}{4}} \cdot \frac{2^{n - \frac{D}{2} \log n}}{2^n} \quad = \quad \binom{n}{\frac{3D}{4}} \cdot n^{-\frac{D}{2}} \quad \text{codewords.}$$

□

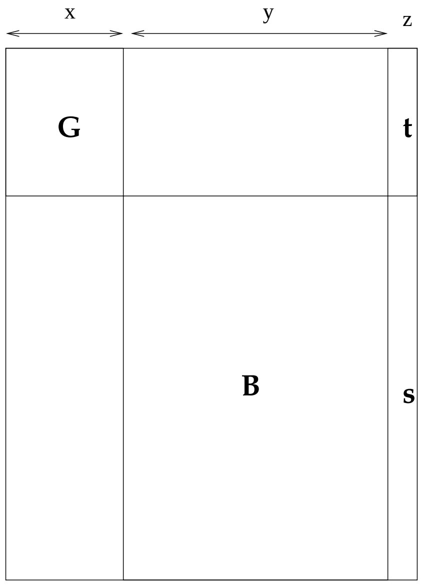

Figure 1: Generator matrix for $\mathcal{C}_1$

LEMMA 6
There is a **BPP** *algorithm that outputs* $\mathbf{s} \in \{0,1\}^n$ *such that with probability* $\frac{99}{100}$*, the ball of radius* $\frac{3D}{4}$ *centered at* $\mathbf{s}$ *contains at least*

$$\frac{1}{100}\binom{n}{\frac{3D}{4}} \cdot n^{-\frac{D}{2}} \quad \textit{codewords of } \mathcal{B}.$$

PROOF: The algorithm first picks a codeword $\mathbf{c} \in \mathcal{B}$ at random, then picks a random point $\mathbf{s}$ such that $\mathbf{d}(\mathbf{c},\mathbf{s}) = \frac{3D}{4}$ and outputs it. For $\mathbf{s} \in \{0,1\}^n$ let $b_{\mathbf{s}}$ denote the number of codewords at distance $\frac{3D}{4}$ from $\mathbf{s}$. Then

$$\Pr[\mathbf{s} \text{ is output}] = \frac{b_{\mathbf{s}}}{\binom{n}{\frac{3D}{4}} \cdot 2^{n-\frac{D}{2}\log n}}$$

$$\text{Hence if } b_{\mathbf{s}} \leq \frac{1}{100}\binom{n}{\frac{3D}{4}} \cdot n^{-\frac{D}{2}}, \quad \Pr[\mathbf{s} \text{ is output}] \leq \frac{1}{100}\frac{1}{2^n}$$

$$\Rightarrow \quad \Pr[\mathbf{s} \text{ is output and } b_{\mathbf{s}} \leq \frac{1}{100}\binom{n}{\frac{3D}{4}} \cdot n^{-\frac{D}{2}}] \leq \frac{1}{100} \quad \text{by the union bound.}$$

□

3 The Intermediate Construction

Take $\mathbf{G}$ and $\mathbf{t}$ be an instance of **NCP** as in Theorem 2. Let $\mathbf{B}$ and $\mathbf{s}$ as in Lemma 6. Recall that $\mathbf{G}$ has dimension $2D \times k$ for some $k < 2D$ and $\mathbf{B}$ has dimension $n \times n - \frac{D}{2}\log n$. Consider the code $\mathcal{C}_1$ whose generator matrix is as in Figure 1. Each codeword consists of two parts $\mathbf{Gx} + z\mathbf{t}$ and $\mathbf{By} + z\mathbf{s}$.

Lemma 7
In the Yes case of **NCP**, *there are at least* $\frac{1}{100}\binom{n}{\frac{3D}{4}} \cdot n^{-\frac{D}{2}}$ *codewords in* $\mathcal{C}_1$ *of weight less than* $(\frac{3}{4}+\varepsilon)D$.

Proof: Set $z = 1$. In the Yes case of **NCP** by Theorem 2, $\exists\, \mathbf{x}$ such that $\mathbf{wt}(\mathbf{Gx}+\mathbf{t}) \leq \varepsilon D$. Further by Lemma 6, there are at least $\frac{1}{100}\binom{n}{\frac{3D}{4}} \cdot n^{-\frac{D}{2}}$ vectors $\mathbf{y}$ so that $\mathbf{By}+\mathbf{s} \leq \frac{3D}{4}$. This gives at least $\frac{1}{100}\binom{n}{\frac{3D}{4}} \cdot n^{-\frac{D}{2}}$ codewords in $\mathcal{C}_1$ of weight at most $(\frac{3}{4}+\varepsilon)D$. □

We call these vectors Good vectors. By setting $n = D^{21}$, we can ensure that

$$\frac{1}{100}\binom{n}{\frac{3D}{4}} \cdot n^{-\frac{D}{2}} \quad > \quad 2^{4D}$$

using standard estimates.

Lemma 8
In the No case of **NCP**, *there are at most* 2^{2D} *codewords in* $\mathcal{C}_1$ *of weight less than* $(1-\varepsilon)D$.

Proof:If $z = 1$, then by Theorem 2, for any $\mathbf{x}$, $\mathbf{wt}(\mathbf{Gx}+\mathbf{t}) \geq (1-\varepsilon)D$. Hence, we may assume $z = 0$. If $\mathbf{y} \neq \mathbf{0}$, then since $\mathcal{B}$ has distance D, $\mathbf{wt}(\mathbf{By}) \geq D$. Hence, we may also assume $\mathbf{y} = \mathbf{0}$.

This leaves us with vectors of the form $\mathbf{Gx}$. These could have small weight, but clearly there are at most 2^{2D} such vectors. □

We call these vectors Bad vectors.

4 Adding Random Constraints

Lemma 9
For $\mathbf{w} \xleftarrow{R} \{0,1\}^n$ *and distinct non-zero vectors* $\mathbf{c_1}, \mathbf{c_2} \in \{0,1\}^n$, *the variables* $\mathbf{w}\cdot\mathbf{c_1}$ *and* $\mathbf{w}\cdot\mathbf{c_2}$ *are independent.*

Proof: We use the fact that the orthogonal space to a k-dimensional subspace of $\mathbb{F}_2^n$ has 2^{n-k} vectors. It follows that $\Pr[\mathbf{w}\cdot\mathbf{c_1} = \mathbf{0}] = \Pr[\mathbf{w}\cdot\mathbf{c_2} = \mathbf{0}] = \frac{1}{2}$. Also, since $\mathbf{c_1}, \mathbf{c_2}$ are linearly independent, $\Pr[\mathbf{w}\cdot\mathbf{c_1} = \mathbf{w}\cdot\mathbf{c_2} = \mathbf{0}] = \frac{1}{4}$. From this it follows that the variables are independent. □

We will take a random subspace $\mathcal{C}_2$ of $\mathcal{C}_1$ by choosing a set of $3D$ vectors $\mathbf{w_i}$ at random and adding the constraints $\mathbf{w}\cdot\mathbf{c} = \mathbf{0}$.

Lemma 10
$\mathcal{C}_2$ contains at least one Good vector in the Yes case and no Bad vectors in the No case with high probability.

Proof: Note that for any $\mathbf{c} \neq \mathbf{0}$, $\Pr[\forall i, \mathbf{w_i}\cdot\mathbf{c} = \mathbf{0}] = \frac{1}{2^{3D}}$. In the No case, using the Union bound, the probability that some bad vector satisfies all the constraints is at most $\frac{1}{2^D}$.

In the Yes case, let G denote the number of Good vectors that satisfy all constraints. Note that $E[G] = 2^D$, and $Var[G] < 2^D$ using pairwise independence of the events that any pair of non-zero vectors satisfy all constraints. Hence $P[G = 0] < \frac{1}{2^D}$. □

THEOREM 11
MDC *is hard to approximate within factor* β *for some* $\beta > 1$ *unless* $\mathbf{NP} \subseteq \mathbf{BPP}$.

PROOF: We have given a randomized reduction from **NCP** to **MDC** so that with probability at least $\frac{9}{10}$ in the Yes case there is a vector of weight at most $(\frac{3}{4} + \varepsilon)D$ and in the No case, every vector has length $(1 - \varepsilon)D$. This gives a hardness factor of $\beta = \frac{1-\varepsilon}{\frac{3}{4}+\varepsilon}$. □

5 Amplifying Hardness via the Tensor Product

DEFINITION 3 *Given an* $[n, k]_2$ *code* $\mathcal{B}$ *and an* $[m, \ell]_2$ *code* $\mathcal{C}$ *generated by* $\{\mathbf{u_1}, \cdots, \mathbf{u_n}\}$ *and* $\{\mathbf{v_1}, \cdots, \mathbf{v_m}\}$ *respectively, their tensor product* $\mathcal{B} \otimes \mathcal{C}$ *is the* $[n \times m, k \times \ell]_2$ *code generated by* $\{\mathbf{u_i} \otimes \mathbf{v_j}\}$ *for* $1 \leq i \leq n$ *and* $1 \leq j \leq m$.

EXERCISE 1 Verify that the following definitions of tensor product of codes are equivalent to the one above.

1. $\mathcal{B} \otimes \mathcal{C}$ is the set of all vectors of the form $\mathbf{B} \cdot \mathbf{X} \cdot \mathbf{C^t}$ where $\mathbf{B}$ and $\mathbf{C}$ are generators for $\mathcal{B}$ and $\mathcal{C}$ respectively and $\mathbf{X} = [x_{ij}]$ is a $k \times \ell$ matrix of indeterminates.
2. $\mathcal{B} \otimes \mathcal{C}$ is the set of all $n \times m$ matrices whose columns belong to $\mathcal{B}$ and whose rows belong to $\mathcal{C}$.

LEMMA 12
Let the minimum distances of $\mathcal{B}$ *and* $\mathcal{C}$ *be* d *and* d' *respectively. The minimum distance of* $\mathcal{B} \otimes \mathcal{C}$ *is* $d \cdot d'$.

PROOF: It is easy to show that the minimum distance is at most $d \cdot d'$. Let b and c be codewords of minimum weight in the two codes. Then $b \otimes c$ is a codeword in $\mathcal{B} \otimes \mathcal{C}$ of weight $d \cdot d'$.

To show the other direction we can use either of the alternative definitions of $\mathcal{B} \otimes \mathcal{C}$. We use the second definition. Given a non-zero codeword in $\mathcal{B} \otimes \mathcal{C}$, pick a non-zero column $\mathbf{b_j}$. Since $\mathbf{b_j} \in \mathcal{B}$, it has at least d non-zero entries. Hence the codeword has at least d non-zero rows. Each row being a codeword of $\mathcal{C}$ has weight at least d'. Hence the codeword has weight at least $d \cdot d'$. □

We use the tensor product to complete the proof of the main Theorem.

THEOREM 13
- **MDC** *is hard to approximate within any constant factor unless* $\mathbf{NP} \subseteq \mathbf{BPP}$.
- **MDC** *is hard to approximate within factor* $2^{\log^{1-\varepsilon} n}$ *for any* $\varepsilon > 0$ *unless* $\mathbf{NP} \subseteq \mathbf{Quasi-BPP}$.

PROOF: We have shown that **MDC** is hard to approximate for some constant $\beta > 1$. Given a gap instance $\mathcal{C}$ of **MDC** of length n take the k-wise tensor product. This boosts the gap to β^k, while the length of the code $\mathcal{C}^{\otimes k}$ is n^k which is polynomial in the input size n. This proves a hardness factor of any constant.

If we repeat the tensor product $\log^k n$ times, this gives a gap of $\beta^{\log^k n}$ on inputs of length $n^{\log^k n}$ which gives a hardness factor of $2^{\Omega(\log^{\frac{k}{k+1}} n)}$. The reduction runs in quasipolynomial time. □

References

[1] Ilya Dumer and Daniele Micciancio and Madhu Sudan, Hardness of Approximating the Minimum Distance of a Linear Code, FOCS'99.

[2] Noga Alon and Joel Spencer, The Probabilistic Method. Chapter 15, Derandomization.