# Notes for Lecture 4

## 4 More on Composition of Codes

### 4.1 Review of constructions so far

We begin by reviewing the parameters achieved by constructions of codes we have covered so far.

Recall that the *Reed-Solomon* code gives, for every $q \geq n$, an $[n, k, n-k+1]_q$ code. Viewing this in terms of the *rate* $k/n$ and *relative distance* $\delta = d/n$, this gives a roughly linear trade-off:

$$k/n \approx 1 - \delta.$$

By the *singleton bound* $k \leq n - d + 1$, this is optimal for this parametrization.

Its generalization to multivariate polynomials, the *Reed-Muller* code, gives for every $q \geq s$, an $\left[s^m, \binom{t+m}{m}, s^m\left(1 - \frac{t}{s}\right)\right]_q$ code, where $s = |S|$ is the size of the subset $S \subseteq \mathbf{F}$ of field elements on which the polynomial is evaluated.

For the simplest (bivariate) polynomial case $m = 2$, we get an $\left[s^2, \binom{t+2}{2}, s^2\left(1 - \frac{t}{s}\right)\right]_q$ code or, changing notation, roughly an $\left[n, k, \left(1 - \sqrt{2k/n}\right) n\right]_q$ code. Now the rate $k/n$ is at most $\frac{1}{2}$ (and only gets worse as $m \to \infty$) but, in exchange, we now only need that the alphabet size $q$ be at least $s = \sqrt{n}$ instead of $n$, and this becomes even better as $m \to \infty$.

In the extreme case of a minimal (boolean) alphabet, $\mathbf{F} = \mathbf{F}_2$, $s = 2$, and as $m \to \infty$ we get a family of $\left[n, 1 + \log n, \frac{n}{2}\right]_2$ codes, called the *Hadamard* codes.

### 4.2 Concatenation

We saw earlier that we could use *concatenation* of codes to combine code constructions with various properties to create new codes with combinations of these properties. Recall that concatenation of an $[n_1, k_1, d_1]_q$ code with an $[n_2, k_2, d_2]_2$ code gives us an $[n_1 n_2, k_1 k_2, d_1 d_2]_2$ code (here we assume that the input size of the second code is compatible with the encoding of the first as a sequence of bits, ie $q = 2^{k_2}$). For example, with appropriate notation, the Reed-Solomon and Hadamard constructions give us a family of $[n, \epsilon n, (1-\epsilon)n]_n$ and $[n/2, \log n, n/4]_2$ codes respectively which, concatenated, yield an $[n, \epsilon\sqrt{n}\log n, \frac{n}{2}(1-\epsilon)]_2$ code.

How well can we decode concatenated code constructions? Information-theoretically, we can correct up to $\frac{d_1 d_2}{2} - 1$ errors, but it is not obvious that this can be achieved algorithmically. Consider the naive decoding algorithm: Apply the inner decoding algorithm to each block to obtain an input for the outer decoder; then, apply the outer decoding algorithm to recover the message. Assuming the outer and inner codes can correct $e_1$ and $e_2$ errors, respectively, we see that the procedure fails only if at the intermediate step we had a message with more than $e_1$ errors, where each of these errors required more than $e_2$ errors in its corresponding block. So, this decoding procedure allows us to recover from up to $e_1 e_2$ errors.

But $e_i \leq \frac{1}{2}d_i$ so $e_1 e_2 \leq \frac{d_1 d_2}{4}$ (and one can construct examples where this bound is tight), while we should theoretically be able to recover up to about $\frac{d_1 d_2}{2}$ errors. This naive decoding procedure causes us to lose a factor 2 against the theoretical optimum, and this only becomes worse as additional layers of concatenation are applied. Next time, we will see a more clever procedure which allows us to eliminate this gap.

### 4.3 The Plotkin Bound

Recall that the *Gilbert-Varshamov* bound gave us a construction for a family of boolean codes with rate

$$k/n \approx (1 - H_2(\delta))n.$$

Thinking of the dependence of $k/n$ on $\delta$, codes in the Gilbert-Varshamov family fall along a curve which runs just below the line between $(0,1)$ and $\left(\frac{1}{2}, 0\right)$.

Can we do better? As a first attempt, let us see what happens if we try to construct an $[n, k, \frac{2}{3}n + 1]_2$ code. We claim that, in this case, we can have at most 2 code words. To see this, let $x_1, x_2, x_3$ be three distinct code words in such a code. Without loss of generality, we may assume that the first $\frac{2}{3}$ entries of $x_1, x_2$ are all zeros and all ones, respectively. What could $x_3$ be? Regardless of the choices for the last $\frac{1}{3}$ of its entries, $x_3$ must have at least $\frac{1}{3}n + 1$ 1s in its first $\frac{2}{3}$ entries. But then it will be within distance $\frac{2}{3}n$ to $x_2$, a contradiction. On the other hand, $d = \frac{n}{2}$ is achievable by the Hadamard code.

Can relative distances between $\frac{1}{2}$ and $\frac{2}{3}$ be achieved with boolean codes? The answer is essentially no, though to see this we must use a somewhat more sophisticated argument, known as the *Plotkin bound:*

**Theorem 1** *Suppose there exists an $[n, k, d]_2$ code. Then:*

1. *If $d \geq \frac{n}{2}$ then $k \leq 1 + \log n$.*
2. *If $d \geq \frac{n}{2}(1 + \epsilon)$ then $k \leq \log\left(1 + \frac{1}{\epsilon}\right)$.*
3. *$k \leq n - 2d + \log(4d)$*

The first part of the bound is achieved by the Hadamard code. The second part states that if we wish to reach relative distances bounded away from $\frac{1}{2}$, we can send only a *constant* (depending on $\epsilon$) number of messages. The final part states that the rate $k/n \leq 1 - \frac{2d}{n} + o(1)$, so we cannot move much to the right of the rate/distance trade-off line we saw above.
PROOF: We begin by proving that (1) $\to$ (3): Given an $[n, k, d]_2$ code, list the code words in lexicographic order. By the pigeonhole principle, at least one of the $2^{n-2d}$ possible $(n-2d)$-bit prefixes must occur at least $\frac{2^k}{2^{n-2d}} = 2^{k-n+2d}$ times. Focus attention on codewords with

such a prefix. Suffixes of the first $2^{k-n+2d}$ of these codewords must form a $[2d, k-(n-2d), d]_2$ code. Then, by (1), $k-(n-2d) \leq \log(4d)$, giving (3).

To prove parts (1) and (2), we view codewords $x_i \in \{0,1\}^n$ as points $y_i \in \mathbf{R}^n$ using the standard geometric embedding which sends entries labeled 0, 1 into 1, -1, respectively. Note that, up to normalization by $\frac{1}{\sqrt{n}}$, all points $y_i$ lie on the surface of the unit sphere in $l_2$. Further, the embedding respects the following relation between Hamming distance $d_H$ and dot product:

$$\langle y_1, y_2 \rangle = 2d_H(x_1, x_2) - n.$$

Under this correspondence, parts (1) and (2) of Theorem 1 follow easily from the following lemma. The basic intuition is that if one tries to place points on a sphere so that all points lie in different hemispheres, then one can only place about as many points as there are dimensions (an optimal placement is to use the $2n$ vectors $\pm e_i$, where $e_i$ is the $i$th basis vector). Further, if we strengthen the condition so that all pairs of vectors must form at least a given obtuse angle, then we can only place a *constant* number of such points on the sphere.

**Lemma 2** *Let $a_1, \ldots, a_m$ be vectors in $\mathbf{R}^n$.*

1. *If, for all $i, j$, $a_i \neq 0$ and $\langle a_i, a_j \rangle \leq 0$, then $m \leq 2n$*
2. *If, in addition, $\|a_i\| = 1$ and $\langle a_i, a_j \rangle \leq -\alpha$ for some $\alpha > 0$, then $m \leq 1 + \frac{1}{\alpha}$*

PROOF: Suppose first that there exists $b \in \mathbf{R}^n$ such that $\langle b, a_i \rangle > 0$ for all $i$. We claim that $m \leq n$. If not, then $m > n$ and so the set $\{a_i\}$ is not linearly independent. Then there exists a linear relation

$$\sum_{i \in S^+} \lambda_i a_i + \sum_{i \in S^-} \lambda_i a_i = 0$$

where $S^+, S^-$ are the set of indices $i$ where $\lambda_i$ is strictly positive and negative, respectively, and $|S^+| + |S^-| > 0$. Then

$$\sum_{i \in S^+} \lambda_i a_i = \sum_{i \in S^-} -\lambda_i a_i = z$$

for some vector $z \in \mathbf{R}^n$. Further, $z \neq 0$ since, assuming w.l.o.g. $S^+ \neq \emptyset$,

$$\langle b, z \rangle = \left\langle b, \sum_{i \in S^+} \lambda_i a_i \right\rangle = \sum_{i \in S^+} \lambda_i \langle b, a_i \rangle > 0.$$

(similarly if $S^- \neq \emptyset$). Then

$$0 < \langle z, z \rangle = \left\langle \sum_{i \in S^+} \lambda_i a_i, \sum_{i \in S^-} -\lambda_i a_i \right\rangle = \sum_{i \in S^+, j \in S^-} -\lambda_i \lambda_j \langle a_i, a_j \rangle \leq 0.$$

Contradiction.

To prove part (1), set $b$ above to any vector not orthogonal to $\{a_i\}$, so that $\langle b, a_i \rangle \neq 0$. Then, for at least half of the indices $i$, $\langle \pm b, a_i \rangle > 0$. Choosing the sign on $b$ appropriately, we get $m \leq 2n$.

For part (2), let $z = \sum a_i$. Then

$$0 \leq \langle z, z \rangle = \left\langle \sum a_i, \sum a_i \right\rangle = \sum \langle a_i, a_i \rangle + \sum_{i \neq j} \langle a_i, a_j \rangle \leq m - \alpha m(m-1)$$

where the last step uses the fact that the $\{a_i\}$ are $m$ unit vectors and $\langle a_i, a_j \rangle \leq -\alpha$. This implies

$$m \leq 1 + \frac{1}{\alpha}.$$

□

To prove part (1) of Theorem 1, let $\{a_i\}$ be the images $\{y_i\}$ as before. Then $d \geq \frac{n}{2}$ implies $\langle a_i, a_j \rangle \leq 0$, so part (1) of the lemma gives $2^k = m \leq 2n$ or $k \leq 1 + \log n$.

To prove part (2), note that the stronger precondition $d \geq \frac{n}{2}(1+\epsilon)$ gives $\langle a_i, a_j \rangle \leq -\epsilon n$ so that $\left\langle \frac{1}{\sqrt{n}} a_i, \frac{1}{\sqrt{n}} a_j \right\rangle \leq -\epsilon$ and, as noted earlier, $\frac{1}{\sqrt{n}} a_i$ is a unit vector. Then part (2) of the lemma gives $2^k = m \leq 1 + \frac{1}{\epsilon}$ or $k \leq \log\left(1 + \frac{1}{\epsilon}\right)$. □