Error Correcting Codes: Combinatorics, Algorithms and Applications **(Fall 2007)**

Lecture 19: Elias-Bassalygo Bound

October 10, 2007

Lecturer: Atri Rudra *Scribe: Michael Pfetsch & Atri Rudra*

In the last lecture, we saw the q-ary version of the Johnson bound on the rate and distance of a code, which we repeat below.

1 Johnson bound

Theorem 1.1 (Johnson Bound)**.** *Let $C \subseteq [q]^n$ be a code of distance d. If $\rho < J_q\left(\frac{d}{n}\right)$, then C is a (ρ, qdn)-list decodable code, where the function $J_q(\delta)$ is defined as*

$$J_q(\delta) = \left(1 - \frac{1}{q}\right)\left(1 - \sqrt{1 - \frac{q\delta}{q-1}}\right).$$

Recall that the best upper bound on R (in terms of δ) that we have seen so far is a combination of the Plotkin and Hamming bounds (see Figure 1).

2 Elias-Bassalygo bound

We begin with the statement of a new upper bound on the rate called the Elias-Bassalygo bound.

Theorem 2.1 (Elias-Bassalygo bound)**.** *Every q-ary code of rate R, distance δ, and large enough block length, satisfies the following:*

$$R \leq 1 - H_q(J_q(\delta)) + o(1)$$

The proof of theorem above uses the following lemma:

Lemma 2.2. *Given a q-ary code, $C \subseteq [q]^n$and $0 \leq e \leq n$, there exists a Hamming ball of radius e with at least $\frac{|C| Vol_q(\mathbf{0},e)}{q^n}$ codewords in it.*

Proof. We will prove the existence of the required Hamming ball by the probabilistic method. Pick a received word $\mathbf{y} \in [q]^n$ at random. It is easy to check that the expected value of $|B_q(\mathbf{y}, e) \cap C|$ is $\frac{|C| Vol_q(\mathbf{0},e)}{q^n}$. (We have seen this argument earlier when we proved the negative part of the list decoding capacity.)

This implies the existence of a $\mathbf{y} \in [q]^n$ such that

$$|B_q(\mathbf{y}, e) \cap C| \geq \frac{|C|\, Vol_q(\mathbf{0}, e)}{q^n},$$

as desired. □

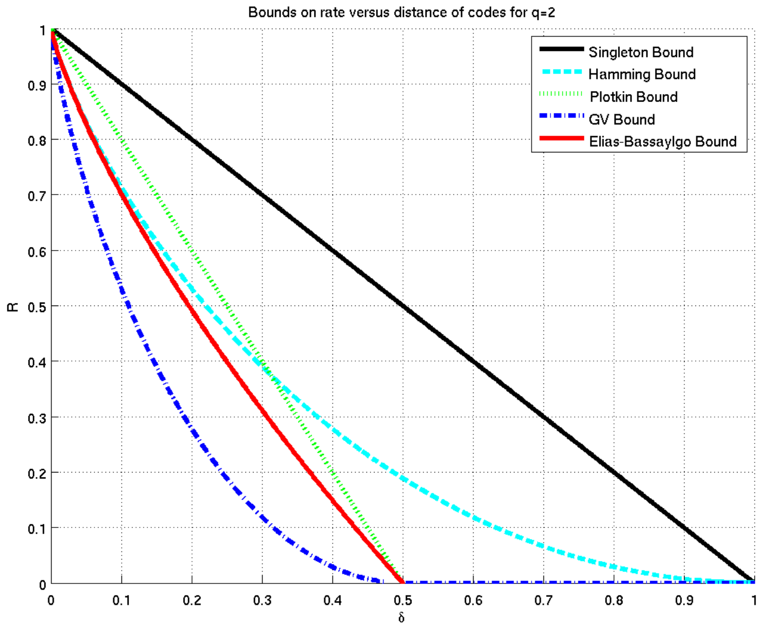

Figure 1: Singleton, Hamming, Plotkin, GV and Elias-Bassalygo bounds on rate versus distance for binary codes. The Elias-Bassalygo bound is shown in red.

Proof of Elias-Bassalygo bound. Let $C \subseteq [q]^n$ be any code with relative distance δ. Define $e = nJ_q(\delta) - 1$ (this choice will allow us to use the Johnson bound). By Lemma 2.2, there exists a Hamming ball with B codewords such that the following inequality is true:

$$B \geq \frac{|C|\, Vol_q(\mathbf{0}, \mathbf{e})}{q^n}.$$

This along with the Johnson bound, which states $B \leq qdn$, the following inequality is true:

$$|C| \leq qnd \cdot \frac{q^n}{Vol_q(\mathbf{0}, e)} \leq q^{n(1-H_q(J_q(\delta))+o(1))},$$

where the second inequality follows from our good old lower bound on the volume of a Hamming ball. This implies that the rate R of C satisfies:

$$R \leq 1 - H_q(J_q(\delta)) + o(1).$$

The proof is complete.

3 The MRRW bound: A better upper bound

The McEliece-Rodemich-Rumsey-Welch (MRRW) bound was introduced in 1977 in the paper [1]. The MRRW bound is based on a linear programming approach introduced by Delsarte to

Shannon	Hamming	
BSC_p	Unique	List
1-H(p) is capacity	$R \geq 1 - H\left(\delta\right)$	$1 - H\left(p\right)$ is list decoding capacity
	$R \leq MRRW$	
Explicit codes at capacity?	Explicit Asymptotically good codes?	Explicit codes at capacity?
Efficient decoding algorithm?	Efficient decoding algorithms?	Efficient decoding algorithms?

Table 1: High level summary of results seen so far.

bound the rate of a code. Unfortunately, unlike the bounds we have seen so far, the MRRW bound does not have a simple closed form expression thought it is a better upper bound than the Elias-Bassalygo bound. However, there is a gap between the Gilbert-Varshamov (GV) bound and the MRRW bound. The gap still exists to this day. To give one data point on the gap, for $\delta = \frac{1}{2} - \varepsilon$ (think of $\varepsilon \to 0$), the GV bound gives a lower bound on R of $\Omega\left(\varepsilon^2\right)$, while the MRRW bound gives an upper bound on R of $O\left(\varepsilon^2 \log\left(\frac{1}{\varepsilon}\right)\right)$.

4 A breather

Given that there is a gap between the best know lower and upper bound on the rate of a code with given relative distance, the following are extremely important open questions:

1. What is the optimal trade-off between R and δ?

2. Does there exist an explicit construction of codes on the GV bound?

If we scale down our ambitions, the following is a natural weaker version of the second question above:

- Do there exist explicit codes with positive rate R and positive relative distance δ?

Table 1 summarizes what we have seen so far for binary codes in Shannon's world and Hamming's world (under both unique and list decoding settings). We also list some natural open questions in each category. The remainder of the course will focus on the questions mentioned in the last two rows of Table 1.

We now change gears with an application of codes (and Reed-Solomon codes in particular).

5 Application: Secret sharing

Secret sharing is the natural cryptographic problem of “dividing" up a “secret" among many players so that the players can reconstruct the secret if and only if a sufficient number of them get together.

More formally, an (ℓ, m)-secret sharing scheme is defined as follows. The inputs to the problem are $s \in \mathbb{D}$ (for some domain $\mathbb{D}$) and n players, $P_1, \cdots, P_n$. The outputs are the secret shares,

$s_1, \cdots, s_n$, such that each output s_i corresponds to the player P_i. Note that the outputs do not have to be elements of $\mathbb{D}$. Finally, the sharing scheme needs to satisfy the following properties:

1. For all $T \subseteq [n]$ such that $|T| \geq m$, $\{P_i\}_{i \in T}$ can recover s $\big($from $\{s_i\}_{i \in T}\big)$. This means that each player can recover the complete secret from the set of secret parts if the number of elements in the set T is greater than or equal to m.

2. For all $T' \subseteq [n]$ such that $|T'| \leq \ell$. $\{P_i\}_{i \in T'}$ *cannot* recover s $\big($from $\{s_i\}_{i \in T'}\big)$. The phrase "cannot recover" means that given $\{P_i\}_{i \in T'}$, all possibilities $s \in \mathbb{D}$ are equally likely. Alternatively, we can say that having $\{s_i\}_{i \in T'}$ is as good as having no secret shares.

Note that the above is an information-theoretic requirement. In particular, these requirement need to hold even if the players have unlimited computation power.

5.1 Example

We now provide an example of secret sharing for $n = 4$ players, to illustrate the requirements of the secret sharing problem.

Let the number of players be $n = 4$ and let the alphabet be $\mathbb{D}$ be the set of all length-8 strings over $\{a, b, \cdots, z\}$. Further let secret, s, be $s =$ password. Now consider the secret sharing scheme that produces the following shares:

$$\begin{aligned} s_1 &= \text{pa} \\ s_2 &= \text{ss} \\ s_3 &= \text{wo} \\ s_4 &= \text{rd} \end{aligned}$$

Note that without the knowledge of any share, all of the $26^8 \approx 208.8$ billion choices for s are possible. However, even knowing one share reduces the number of possibilities to $26^6 \approx 308.9$ million. Thus, this scheme is not a (ℓ, m) secret sharing scheme for any $m \geq 1$.

5.2 An example that works

Let $\mathbb{D} = \mathbb{F}_2^t$. Consider the following $(n-1, n)$-secret sharing scheme.

Pick s_i to be $r_i \in \mathbb{F}_2^t$ uniformly at random (for $1 \leq i \leq n-1$). Finally, define $s_n = s + r_1 + \cdots + r_{n-1}$.

We now briefly argue that the two required properties are satisfied:

1. We claim that given any $\leq n-1$ shares, the secret, s, can be any vector in $\mathbb{F}_2^t$. To see this note that $s_1, \ldots, s_n$ are all random elements of $\mathbb{F}_2^t$. Further, any $n-1$ of them are independent random elements. Thus, knowing at most $n-1$ shares gives off no extra information about s.

2. This condition holds by noting that $s = s_1 + \cdots + s_n$.

References

[1] R. J. McEliece, E. R. Rodemich, H. Rumsey, and L. R. Welch. ”New Upper Bounds on the Rate of a Code via the Delsarte-MacWilliams Inequalities.” *IEEE Trans. Inform. Theory*, vol. IT-23, pp. 157–166, Mar. 1997.