CSE 533: Error-Correcting Codes **(Autumn 2006)**

Lecture 4: Proof of Shannon's theorem and an explicit code

October 11, 2006

Lecturer: Venkatesan Guruswami *Scribe: Atri Rudra*

1 Overview

Last lecture we stated Shannon's theorem (specifically for the Binary Symmetric Channel with crossover probability p, BSC_p) and sketched the proof of its converse. In today's lecture, we will first prove Shannon's theorem. Then we will look at an explicit (and very "hands-down") construction of a code due to Elias [1] that achieves a positive rate for *some* positive crossover probability.

2 Proof of Shannon's theorem

We first recall the Shannon's theorem (for the special case of BSC_p).

Theorem 2.1. *For every p, such that $0 \leq p < \frac{1}{2}$, and every $\varepsilon > 0$, there exists a δ and a code with rate $\frac{k}{n} = 1 - H(p+\varepsilon)$, which can be decoded for the BSC_p channel with error probability at most $2^{-\delta n}$.*

First note that as $p < \frac{1}{2}$, $H(p) < 1$ and hence, the claimed rate is positive. See Figure 1 for a plot of the capacity function.

Proof. (of Theorem 2.1) The encoding function $\mathcal{E} : \{0,1\}^k \rightarrow \{0,1\}^n$, is chosen randomly, that is, for every message $m \in \{01,\}^k$, the corresponding codeword, $\mathcal{E}(m)$ is chosen uniformly at random from $\{0,1\}^n$. [1] The decoding function will be the *maximum likelihood decoding*, that is, choose the closest codeword to the received word. More formally, the decoding function $\mathcal{D} : \{0,1\}^n \rightarrow \{0,1\}^k$ is defined as follows:

$$\mathcal{D}(y) = \arg \min_{m \in \{0,1\}^k} \Delta(y, \mathcal{E}(m)).$$

Note that we are concerned with the *existence* of a suitable code and not with the complexity of decoding. Thus, we can perform maximum likelihood decoding by going over all possible codewords.

We now present the general idea of the proof. We will first show that for any fixed message m, the average error probability (over the choice of $\mathcal{E}$), is small (for appropriate choice of k). This

[1] Note that this might assign the same codeword to two different messages but this (tiny) probability will be absorbed into the decoding error probability.

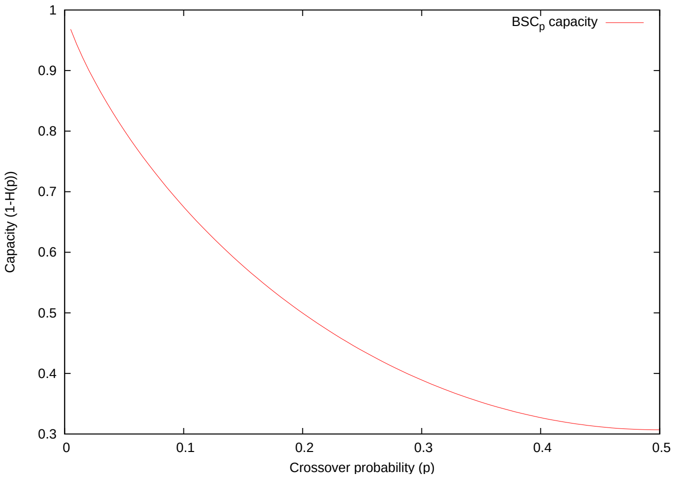

Figure 1: Plot of the BSC_p capacity function.

will imply, that there *exists* a code, which has very low decoding error probability for the message m. Then to prove the result for every message, we will remove certain codewords from the code. This procedure is quite standard and also has a name– *random coding with expurgation*.

Consider the scenario when $\mathcal{E}(m)$ is transmitted and y is received on the other side of the BSC_p channel. Consider the ball of radius roughly p around $\mathcal{E}(m)$. By the Chernoff bound, we know that with overwhelming probability, y is contained inside this ball. Now, our decoding function might decode y to some message $m' \neq m$ if and only if $\mathcal{E}(m')$ lies in a ball of radius (roughly) p around y (see Figure 2).

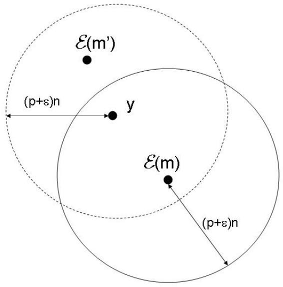

Figure 2: $\mathcal{E}(m)$ is the original codeword, which is received as y. With overwhelming probability, y is within a hamming distance of $(p+\varepsilon)n$ from $\mathcal{E}(m)$. Now a decoding error might occur if there is another codeword $\mathcal{E}(m')$ within the hamming ball around y of radius $(p+\varepsilon)n$.

Since $\mathcal{E}(m')$ was chosen randomly, the probability of such an event happening is the ratio of the volume of a ball of radius (roughly) p to the number of points in the ambient space. The details follow.

Set $\varepsilon' = \frac{\varepsilon}{2}$ and fix a message m. Also fix an encoding function $\mathcal{E}$. We now estimate the probability of the decoding error (over the channel noise), where below $p(y|\mathcal{E}(m))$ denotes the probability (over the channel noise) that on transmitting the codeword $\mathcal{E}(m)$, the received word is y; $\mathbf{1}_{(\cdot)}$ is the *indicator function*[2] and $\mathbb{B}(x, r)$ is the Hamming ball of radius r around the vector x.[3]

$$\begin{aligned}\Pr_{\mathrm{BSC}_p \text{ noise}}[\mathcal{D}(\mathcal{E}(m) + \text{noise}) \neq m] &= \sum_{y \in \{0,1\}^n} p(y|\mathcal{E}(m)) \cdot \mathbf{1}_{\mathcal{D}(y) \neq m} \\ &\leq \sum_{y \notin \mathbb{B}(\mathcal{E}(m),(p+\varepsilon')n)} p(y|\mathcal{E}(m)) + \sum_{y \in \mathbb{B}(\mathcal{E}(m),(p+\varepsilon')n)} p(y|\mathcal{E}(m)) \cdot \mathbf{1}_{\mathcal{D}(y) \neq m} \\ &\leq 2^{-\varepsilon'^2 n/2} + \sum_{y \in \mathbb{B}(\mathcal{E}(m),(p+\varepsilon')n)} p(y|\mathcal{E}(m)) \cdot \mathbf{1}_{\mathcal{D}(y) \neq m},\end{aligned}$$

where the last inequality follows from the Chernoff bound (recall that the BSC_p flips each bit independently with probability p). Now taking expectations on both sides and using z to denote the (random) noise vector, we get

$$\mathbb{E}_{\text{choice of } \mathcal{E}} \Pr_{\text{ noise}}[\mathcal{D}(\mathcal{E}(m) + \text{noise}) \neq m] \leq 2^{-\varepsilon'^2 n/2} + \sum_{z \in \mathbb{B}(\mathbf{0},(p+\varepsilon')n)} \Pr[\text{noise } = z] \cdot \mathbb{E}\left[\mathbf{1}_{\mathcal{D}(z+\mathcal{E}(m)) \neq m}\right], \tag{1}$$

where in the random choice of $\mathcal{E}(\cdot)$, the choice for $\mathcal{E}(m)$ is already fixed. We now estimate the expectation in the sum on the RHS.

$$\begin{aligned}\mathbb{E}\left[\mathbf{1}_{\mathcal{D}(z+\mathcal{E}(m)) \neq m}\right] &\leq \Pr_{\text{choice of } \mathcal{E}}\left[\exists m' \neq m, \text{ such that } \Delta(z + \mathcal{E}(m), \mathcal{E}(m')) \leq (p + \varepsilon')n\right] \\ &\leq \sum_{m' \neq m} \Pr\left[\Delta(\mathcal{E}(m) + z, \mathcal{E}(m')) \leq (p + \varepsilon')n\right] \\ &\leq 2^k \cdot \frac{\mathrm{Vol}(\mathbf{0}, (p + \varepsilon')n)}{2^n} \\ &\leq 2^k \cdot \frac{2^{H(p+\varepsilon')n}}{2^n},\end{aligned} \tag{2}$$

where the third inequality follows from the fact that $\mathcal{E}(m')$ was chosen uniformly at random and the fact that the volume of hamming balls are translation invariant. The last inequality follows from the bound on the volume of Hamming balls that was proved in the last lecture. Now using

[2] For a boolean predicate P, $\mathbf{1}_P$ evaluates to 1 if P is true and 0 otherwise.

[3] That is, $\mathbb{B}(x, r) = \{y | \Delta(x, y) \leq r\}$.

(2) in (1), we get:

$$\begin{aligned}\mathbb{E}_{\text{choice of }\mathcal{E}}\Pr{}_{\text{noise}}[\mathcal{D}(\mathcal{E}(m)+\text{noise})\neq m] &\leq 2^{-\varepsilon'^2 n/2}+\sum_{z\in\mathbb{B}(\mathbf{0},(p+\varepsilon')n)}\Pr\left[\text{noise}=z\right]\cdot 2^{k+(H(p+\varepsilon')-1)n}\\ &\leq 2^{-\varepsilon'^2 n/2}+2^{k-(1-H(p+\varepsilon'))n}\\ &= 2^{-\varepsilon^2 n/8}+2^{-(H(p+\varepsilon)-H(p+\frac{\varepsilon}{2}))n}\\ &\leq 2^{-\delta' n}, \qquad (3)\end{aligned}$$

where the second inequality follows from the fact that $\sum_{z\in\mathbb{B}(\mathbf{0},(p+\varepsilon')n)}\Pr\left[\text{noise}=z\right]\leq 1$, the equality follows from substituting $\varepsilon'=\varepsilon/2$ and $k=(1-H(p+\varepsilon))n$ and the last inequality follows by setting (for example) $\delta'=\min\left(\frac{\varepsilon^2}{8},H(p+\varepsilon)-H(p+\frac{\varepsilon}{2})\right)-\frac{1}{n}$.[4]

As (3) is true for every message m, the following bound on the "average" decoding error is also true:

$$\mathbb{E}_m\mathbb{E}_{\text{choice of }\mathcal{E}}\left[\Pr{}_{\text{noise}}\left[\mathcal{D}(\mathcal{E}(m)+\text{noise})\neq m\right]\right]\leq 2^{-\delta' n}.$$

By changing the order of expectations, we have:

$$\mathbb{E}_{\text{choice of }\mathcal{E}}\mathbb{E}_m\left[\Pr{}_{\text{noise}}\left[\mathcal{D}(\mathcal{E}(m)+\text{noise})\neq m\right]\right]\leq 2^{-\delta' n}.$$

Thus, from the above inequality we conclude that there exists some encoding function $\mathcal{E}^*:\{0,1\}^k\rightarrow\{0,1\}^n$ such that

$$\mathbb{E}_m\left[\Pr{}_{\text{noise}}\left[\mathcal{D}^*(\mathcal{E}^*(m)+\text{noise})\neq m\right]\right]\leq 2^{-\delta' n}. \qquad (4)$$

We are almost done except, we want (4) to hold for *every* message m. One could try using the union bound on (4) but the upper bound on the decoding error probability is too weak. Instead to achieve our goal, sort in ascending order the messages m by their decoding error probabilities. We claim that for the "median" message (and hence for the first 2^{k-1} messages) the decoding error probability is at most $2\cdot 2^{-\delta' n}$. To see why this is true assume for contradiction that this is not true. Then for at least 2^{k-1} messages (in particular for the top 2^{k-1} messages) the error probabilities are strictly greater than $2\cdot 2^{-\delta' n}$, which contradicts (4). Thus, to get our final encoding function, $\mathcal{E}'$, we just "throw" away the top 2^{k-1} messages. This implies that that for $\mathcal{E}':\{0,1\}^k\rightarrow\{0,1\}^n$, the decoding function has an error of at most $2^{-\delta n}$, where $\delta=\delta'-\frac{1}{n}$. Further, the rate of $\mathcal{E}'$ is the same as that of $\mathcal{E}^*$,[5] which is $1-H(p+\varepsilon)$ as required. □

Remark 2.2. *For the Binary erasure channel* BEC_α*, the capacity is* $1-\alpha$*. The intuitive argument for* $1-\alpha$ *being an upper bound on rate is the following. If* n *bits are sent then (in expectation) about* $(1-\alpha)n$ *bits are not erased. Thus, for successful decoding one cannot have more than* $(1-\alpha)n$ *message bits to begin with. The proof of the other direction, that is, a rate of* $(1-\alpha)$ *can be achieved is similar to the proof of Theorem 2.1 and is left as an exercise.*

[4] Tighter expression for δ' (also known as the *error exponent*) are known. The calculations here were done just to show the existence of some δ' that depends on ε and p.

[5] The rate of the code corresponding to $\mathcal{E}'$ is less than that of $\mathcal{E}^*$ as that of $\mathcal{E}^*$ by $\frac{1}{n}$, which is negligible in comparison to $1-H(p+\varepsilon)$.

3 Hamming vs Shannon

As a brief interlude, let us compare the salient features of the works of Hamming and Shannon.

Hamming	Shannon
No mention of encoding or decoding. Focus on the set of codewords itself.	The encoding and decoding functions are explicit.
More combinatorial and constructive (explicit codes).	Non-constructive (and begs construction),
The basic tradeoff is between rate and distance, which gives a better handle on the code.	The tradeoff is between rate and error or mis-communication probability (seems more difficult to directly optimize).
Worst case errors (limit on the number of errors but no restriction on their distribution).	Probabilistic/stochastic noise (Capacity is well defined).

We note that the biggest difference is the last one in the table above. As an interesting historical remark, the paper of Shannon mentions the work of Hamming (even though Hamming's paper appeared two years after that of Shannon). However, Hamming's paper does not mention the work of Shannon.

There are two main "defects" of Shannon's work. First, the code is random and hence not explicit. Second, the decoding "algorithm" is brute force. Thus, his work raises the following natural question(s).

Question 3.1. *How does one construct* explicit[6] *codes with rate close to the capacity of* BSC_p *? How about efficient decoding ?*

The above is the grand challenge. However, even before we talk about the above question, here is a related question.

Question 3.2. *Are there* linear *codes that achieve the capacity for* BSC_p *? Note that here we are not insisting on explicit codes.*

Remark 3.3. *The answer to Question 3.2 is yes. In fact, the proof of Shannon becomes "easier" for linear codes, which is left as an exercise.*

Finally, we turn to the following simpler version of Question 3.1.

Question 3.4. *Is there an explicit code with efficient decoding that achieves a positive rate with negligible decoding error probability for* some *cross over probability ?*

Note that the above is the analogue of the positive rate and positive distance question in the Hamming world.

[6] An explicit code has a polynomial time encoding function.

4 An explicit code construction due to Elias

We now look at an explicit code construction due to Elias ([1]) from 1954 that answers Question 3.4. The construction is very "hands-on" and interestingly such a hands on answer is not known for the corresponding rate vs distance question in the Hamming world.

The code of Elias are based on the Hamming codes, which have been studied in previous lectures. Recall that these are $[2^r - 1, 2^r - r - 1, 3]_2$ codes (for some $r \geq 2$). Consider the scenario, where the received word has exactly two errors. In such a case, the decoding algorithm for the Hamming codes will introduce an extra error (this is because Hamming codes are perfect codes and thus, with two bit flips, the received word will be decoded to another codeword, which implies that the decoded codeword is at a distance 3 from the original codeword). Thus, it will be good if one can detect if there are two bit flips.

The idea to achieve the above is pretty simple– add an overall parity check bit to the Hamming codewords.[7] Note that the distance of this new code is now 4.[8] Thus, we have a $[2^r, 2^r - r - 1, 4]_2$ code– this code is called the *Extended Hamming code*. In particular the extended Hamming code has the following parity check matrix:

$$\begin{pmatrix} & & & 0 \\ & \mathbf{H} & & \vdots \\ & & & 0 \\ 1 & 1 & \dots & 1 \end{pmatrix}$$

In other words, the parity check matrix of the extended Hamming code can be obtained by adding an all ones row to $\mathbf{H}$, the parity check matrix of the Hamming code (and the last column will have all zeroes except for the last row).

The decoding algorithm for this code is also natural. First check the overall parity of the received word. If it is 0 (that is, an even number of errors have taken place), then do nothing. Otherwise run the decoding algorithm for the Hamming code. Thus, in particular the following are true:

1. One bit flip is always corrected.
2. Even number of bit flips (and in particular 2 bit flips) are left as it is.
3. For strictly greater than 2 bit flips, the decoding algorithm could introduce at most one bit flip (recall that in this case the decoder for the Hamming code always flips one bit).

Now consider the scenario when codewords from the extended Hamming code of block length n_1 are transmitted over BSC_p. After the received word has been decoded (by the decoding procedure for the extended Hamming code discussed above), what is the expected fraction of errors in

[7] That is, add a bit such that the parity of the resulting codeword is 0.

[8] If two Hamming codewords differed in at least 4 positions, then we are done. So consider two codewords that differ in exactly 3 positions. However, the overall parity bits for such codewords are different.

the resulting word ? To estimate this, we will need the following notation

$$p(i) \overset{def}{=} \binom{n_1}{i} p^i (1-p)^{n_1 - i}.$$

In other words, $p(i)$ denotes the probability that BSC_p flips exactly $0 \leq i \leq n_1$ bits.

Now, from the observations made above, the expected number of bits that are erroneous in the output of the decoding procedure of the extended Hamming code is

$$\begin{aligned}
&\leq 0 \cdot p(0) + 0 \cdot p(1) + 2 \cdot p(2) + \sum_{i=3}^{n_1} (i+1) \cdot p(i) \\
&= n_1 p - p(1) + \sum_{i=1}^{n_1} p(i) \\
&= n_1 p - p(1) + (1 - p(0) - p(1) - p(2)) \\
&= 1 + n_1 p - p(0) - 2p(1) - p(2) \\
&< n_1^2 p^2,
\end{aligned}$$

where the first equality follows from the fact that $n_1 p = \sum_i ip(i)$ while the last inequality follows by setting $pn_1 \leq 3$ (the detailed calculations are in the appendix). Thus, the expected fraction of errors is $n_1 p^2$, which can be made to be strictly less than p for a small enough value of p (that is, $p < \frac{1}{n_1}$, which is fine for some constant value of n_1 like 16). Also the probability that any position has a bit flip can be shown to be at most $n_1 p^2$ (though this probability need not be independent across different positions). Thus, one can try "recursing" this step.

Thus, the main idea is to take a bunch of (extended) Hamming codewords and to cascade the decodings such that the bit error probability decreases from one step to the next. This is achieved through the following "composition" of codes.

Definition 4.1 ((Tensor) Product of codes)**.** *Given two codes $C_1 = [n_1, k_1, d_1]$ and $C_2 = [n_2, k_2, d_2]$, their* tensor product*, denoted by $C_1 \otimes C_2$, consists of codewords that are $n_1 \times n_2$ matrices such that every row is a codeword in C_2 and every column is a codeword in C_1.*

See Figure 3 for an illustration.

Remark 4.2. *It turns out that $C_1 \otimes C_2$ is an $[n_1 n_2, k_1 k_2, d_1 d_2]$ code. The proof is left as an exercise.*

Thus, if we start with two extended Hamming codes, $C_1 = [n_1, k_1, 4]$ and $C_2 = [n_2, k_2, 4]$ then $C_1 \otimes C_2$ is an $[n_1 n_2, k_1 k_2, 16]$ code. Now $C_1 \otimes C_2$ has a natural decoding scheme– first decode each column independently (using the decoding algorithm for C_1) and then decode each row (using the decoding algorithm for C_2). Note that this procedure returns a $n_1 \times n_2$ matrix (which potentially is a codeword in $C_1 \otimes C_2$).

Consider the matrix after all the columns have been decoded. Now let us look at some arbitrary row. By the calculation, done before, for every bit the probability that it is flipped is at most $n_1 p^2 < p$. However, since the columns were decoded independently, the errors in the bits in the

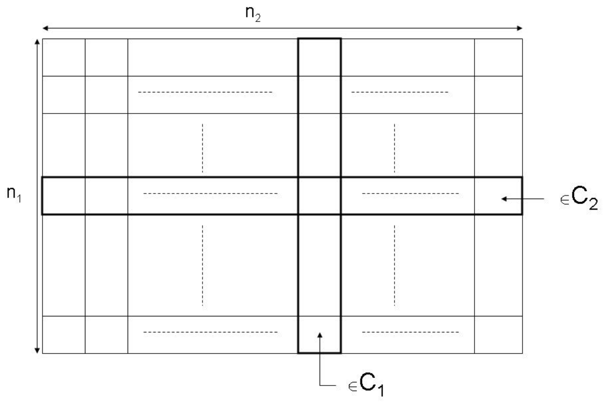

Figure 3: An “illustration” of an $n_1 \times n_2$ codeword in $C_1 \otimes C_2$.

row are still *independent*. Note that the matrix structure was important for this phenomenon. Thus, BSC_p now gets reduced to $\mathrm{BSC}_{n_1 p^2}$. After the rows have also been decoded, every bit is flipped with probability $n_2 n_1^2 p^4$.

Thus, the idea is now to take product of m extended Hamming codes $[n_i, k_i, d_i]$ for $1 \le i \le m$, where the product of m codes is defined recursively using Definition 4.1. Also the decoding procedure also generalizes in an analogous manner. Further, it can be shown that at any stage of decoding, the bit errors (that come from the previous stages) are independent. Let us now look at the parameters. At the end of the last stage, the probability that any bit is in error is at most

$$n_m n_{m-1}^2 n_{m-2}^4 \dots n_1^{2^{m-1}} p^{2^m}. \tag{5}$$

The rate of the code is

$$R = \frac{k_1}{n_1} \cdot \frac{k_2}{n_2} \dots \frac{k_m}{n_m}. \tag{6}$$

Since the probability that any (of the possible $n_1 n_2 \dots n_m$) bit is in error is upper bounded by (5), the total decoding error probability, by the union bound is at most

$$(n_1 n_2 \dots n_m) n_m n_{m-1}^2 n_{m-2}^4 \dots n_1^{2^{m-1}} p^{2^m}. \tag{7}$$

To complete the construction, we now specify the parameters. For some integer $a \ge 5$, and for every $1 \le i \le m$, set

$$n_i = 2^{a+i-1}.$$

Note that this implies (recall we are dealing with extended Hamming codes, which are $[2^r, 2^r - r - 1, 4]$ codes) that

$$\frac{k_i}{n_i} = 1 - \left(\frac{a+i}{2^{a+i-1}}\right).$$

Thus, from (6) we have the following bound on the rate

$$\begin{aligned} R = & \prod_{i=1}^{m} \left(1 - \frac{a+i}{2^{a+i-1}}\right) \\ > & \prod_{i=1}^{\infty} \left(1 - \frac{a+i}{2^{a+i-1}}\right) \\ \geq & \ 1 - \sum_{i=1}^{\infty} \left(\frac{a+i}{2^{a+i-1}}\right) \\ = & \ 1 - \frac{a+2}{2^{a-1}} \\ > & \ 0, \end{aligned}$$

where the last inequality follows from the choice of a and the first inequality is true as for any $i > m$, $1 - (a+i)2^{-a-i+1} < 1$. The calculations for the second inequality (and the equality) can be found in the appendix.

We now return to the calculation of the decoding error probability. Plugging in $n_i = n_1 2^{i-1}$ into (7), it follows that the decoding error probability is at most

$$\begin{aligned} & \left(n_1^m \prod_{i=0}^{m-1} 2^i\right) \cdot n_1^{\sum_{i=0}^{m-1} 2^i} \cdot 2^{\sum_{i=1}^{m} i 2^{m-i}} \cdot p^{2^m} \\ = & \ n_1^m (2^{m(m-1)/2}) \cdot n_1^{2^m - 1} \cdot 2^{2^{m+1} - m - 1} \cdot p^{2^m} \\ < & \ n_1^m 2^{m^2} \left(2 n_1 p\right)^{2^m}, \end{aligned}$$

which approaches 0 as m increases by choosing $p < \frac{1}{2n_1}$.

Remark 4.3. *The distance of the code though is not good. In particular the relative distance is at most $\frac{4^m}{2^{m^2}}$, which vanishes for large m.*

Remark 4.4. *In (5) it is assumed that the order of decoding is as follows: first use the decoding procedure of C_1, then C_2 and so on (and finally C_m). Of course one can use any order but this is the "best" one. To give an intuition reason for this note that the the extended hamming decoder can decode up to 1 bit flip. However, the fraction of bit flip(s) corrected is highest for C_1 and lowest for C_m. Thus, it makes sense that one decodes C_1 first as the "gain" is maximum. Also note that even though C_m can only correct only $1/n_m$ fraction of errors, by the time we use the decoder for C_m the error probability has gone down by quite a bit.*

References

[1] Peter Elias. Error-free coding. *IEEE Transactions on Information Theory*, 4(4):29–37, 1954.

A Calculation of the new bit error rate

We want to show the following inequality:

Lemma A.1. *Let $n \geq 32$ be a power of 2. If $pn \leq 3$ then*

$$1 + np - p(0) - 2p(1) - p(2) < n^2 p^2.$$

Proof. Substituting $p(i) = \binom{n}{i} p^i (1-p)^{n-i}$, we get that the LHS is

$$\begin{aligned} & 1 + pn - (1-p)^n - 2np(1-p)^{n-1} - \binom{n}{2} p^2 (1-p)^{n-2} \\ = \ & 1 + pn - 1 + np + \sum_{i=2}^{n} \binom{n}{i} (-1)^{i+1} p^i - 2np + 2n \sum_{i=1}^{n} \binom{n-1}{i} (-1)^{i+1} p^{i+1} \\ & + \binom{n}{2} \sum_{i=0}^{n} \binom{n-2}{i} (-1)^{i+1} p^{i+2} \\ = \ & \sum_{i=2}^{n} e_i p^i, \end{aligned} \tag{8}$$

where

$$\begin{aligned} e_i = \ & (-1)^{i+1} \binom{n}{i} + 2n(-1)^i \binom{n-1}{i-1} + (-1)^{i-1} \binom{n}{2} \binom{n-2}{i-2} \\ = \ & (-1)^i \binom{n}{i} \left(-1 + 2n \cdot \frac{i}{n} - \frac{n(n-1)}{2} \cdot \frac{i(i-1)}{n(n-1)} \right) \\ = \ & (-1)^i \binom{n}{i} \left(\frac{-2 + 4i - i^2 + i}{2} \right) \\ = \ & (-1)^{i+1} \binom{n}{i} \left(\frac{i^2 - 5i + 2}{2} \right). \end{aligned} \tag{9}$$

We re-arrange (8) as follows:

$$e_2 p^2 + (e_3 p^3 + e_4 p^4 + e_5 p^5) + \sum_{i=3}^{n/2-1} (e_{2i} p^{2i} + e_{2i+1} p^{2i+1}) + e_n p^n.$$

Note that $e_2 = n(n-1) < n^2$. Thus to complete the proof we will show that

$$e_3 p^3 + e_4 p^4 + e_5 p^5 < 0, \tag{10}$$

for every $3 \leq i \leq n/2 - 1$

$$e_{2i} p^{2i} + e_{2i+1} p^{2i+1} < 0, \tag{11}$$

and

$$e_n p^n < 0. \tag{12}$$

From (9), we get $e_3 = -2\binom{n}{3}$, $e_4 = \binom{n}{4} < \frac{n^4}{24}$ and $e_5 = \binom{n}{5} < \frac{n^5}{120}$. Thus, to prove (10) it suffices to show that

$$2p^3\binom{n}{3} - \frac{(np)^4}{24} + \frac{(np)^5}{120} > 0.$$

Indeed this follows from the following sequence of relations:

$$\begin{aligned}
\frac{1}{(np)^3}\left(2p^3\binom{n}{3} - \frac{(np)^4}{24} - \frac{(np)^5}{120}\right) &= \frac{1}{3}\left(1-\frac{1}{n}\right)\left(1-\frac{2}{n}\right) - \frac{np}{24} - \frac{(np)^2}{120} \\
&= \frac{1}{3} - \frac{np}{24} - \frac{(np)^2}{120} - \frac{3}{n} + \frac{2}{n^2} \\
&\geq \frac{1}{3} - \frac{3}{24} - \frac{9}{120} - \frac{3}{n} + \frac{2}{n^2} \\
&= \frac{2}{15} - \frac{3}{n} + \frac{2}{n^2} \\
&> 0,
\end{aligned}$$

where the first inequality follows from the fact that the function achieves its minimum for the largest value of np $(= 3)$ while the last inequality follows from that fact that $\frac{3}{n} - \frac{2}{n^2} < \frac{2}{15}$ or equivalently $2n^2 - 45n + 30 > 0$ for every $n \geq 32$.

We now return to the proof of (11) and (12). First note that $i^2 - 5i + 2 \geq 0$ for all $i \geq 6$. Along with (9) this proves (12). Further, note that this implies that for $3 \leq i \leq n/2 - 1$, $e_{2i} < 0$ and $e_{2i+1} > 0$. Thus, to prove (11) we will show that

$$\frac{|e_{2i}p^{2i}|}{|e_{2i+1}p^{2i+1}|} > 1.$$

Indeed, for any $j \geq 6$,

$$\begin{aligned}
\frac{|e_j p^j|}{|e_{j+1}p^{j+1}|} &= \frac{\binom{n}{j}(j^2 - 5j + 2)}{p\binom{n}{j+1}((j+1)^2 - 5(j+1) + 2)} \\
&= \frac{(j+1)(j^2 - 5j + 2)}{p(n-j)(j^2 - 3j - 2)} \\
&> \frac{7(j^2 - 5j + 2)}{3(j^2 - 3j - 2)} \\
&\geq 1,
\end{aligned}$$

where the first inequality follows from the facts that $(j+1) \geq 7$ and $p(n-j) < np \leq 3$ and the last inequality is true as $3(j^2 - 3j - 2) \leq 7(j^2 - 5j + 2)$ or equivalently, $2j^2 - 13j + 10 \geq 0$ for $j \geq 6$. □

B Missing steps in the calculation of the rate

We need to show the following two results.

Lemma B.1.

$$\prod_{i=1}^{\infty}\left(1-\frac{a+i}{2^{a+i-1}}\right) \geq 1-\sum_{i=1}^{\infty}\left(\frac{a+i}{2^{a+i-1}}\right).$$

Lemma B.2.

$$\sum_{j=a+1}^{\infty}\left(\frac{j}{2^{j-1}}\right) = \frac{a+2}{2^{a-1}}.$$

Proof. (of Lemma B.1) Define

$$x_i = \frac{a+i}{2^{a+i-1}}.$$

Note that $x_1 > x_2 > \dots$. Thus, we have for any $i \geq 1$:

$$(1-x_i) \geq (1-x_1)^{x_i/x_1}.$$

The above is true as $\frac{x_i}{x_1} < 1$ and $x_1 < 1$.[9] Thus, we have

$$\prod_{i=1}^{\infty}(1-x_i) \geq \prod_{i=1}^{\infty}(1-x_1)^{x_i/x_1} = (1-x_1)^{(\sum_{i=1}^{\infty} x_i)/x_1}.$$

From Lemma B.2 and the definition of x_i, it follows that $\sum_{i=1}^{\infty} x_i/x_1 > 2$, which implies that[10]

$$(1-x_1)^{(\sum_{i=1}^{\infty} x_i)/x_1} \geq 1 - x_1\left(\frac{\sum_{i=1}^{\infty} x_i}{x_1}\right) = 1-\sum_{i=1}^{\infty} x_i,$$

as desired. □

Proof. (of Lemma B.2) Define

$$S_a = \sum_{j=a+1}^{\infty}\left(\frac{j}{2^{j-1}}\right).$$

Note that

$$\frac{S_a}{2} = S_a - \frac{a+1}{2^a} - \sum_{j=a+2}^{\infty}\frac{1}{2^{j-1}},$$

[9] Consider $(1-b)^c$, where $b, c < 1$. Expanding, we get $(1-b)^c = 1 - bc + \sum_{i=2}^{\infty}\binom{c}{i}(-i)^i b^i$. It can be verified that the sign of every term in the sum alternates (and the first term is negative). Finally it can be verified that for any $i \geq 2$, $\left|\binom{c}{i}b^i\right| \geq \left|\binom{c}{i+1}b^{i+1}\right|$.

[10] This again follows by expanding $(1-x_1)^{(\sum_{i=1}^{\infty} x_i)/x_1}$ and suitably grouping terms.

which implies that

$$\begin{aligned} S_a = &\ 2\left(\frac{a+1}{2^a} + \sum_{j=a+2}^{\infty} \frac{1}{2^{j-1}}\right) \\ = &\ 2\left(\frac{a+1}{2^a} + \frac{2}{2^{a+1}}\right) \\ = &\ \frac{a+2}{2^{a-1}}. \end{aligned}$$

□