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

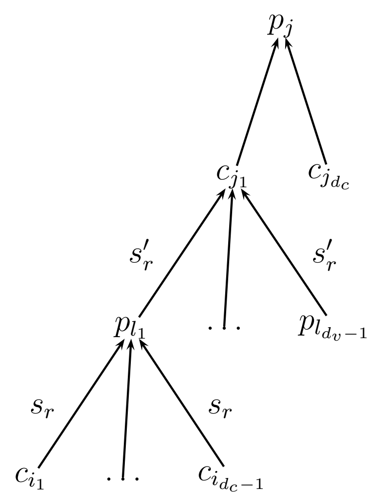
Lecture 35: Threshold Computation

November 16, 2007

Lecturer: Atri Rudra *Scribe: Nathan Russell*

We recall that in BEC_α, we can receive 0, 1 or $?$ at each node.

We recall that when considering communications sent from a variable node to a check node, if the variable node in question got some $b \in \{0, 1\}$ as its received word, it will always send b since it now knows the value.



In the last lecture, we saw that the message sent from check node to variable node c_i depends on the messages it received from nodes other than c_i itself. Using this property and the fact that the number of iterations is at most a fourth of the girth of the factor graph, we showed that all messages received by a check (or, for that matter, variable) node in any round $i < \ell$ are independent random variables. In another lecture, we will see that we can implement this message passing algorithm in $O(n)$ time.

Recall, we defined s_r as the probability of an erasure being passed from variable to check nodes in round r, and s'_r as being the probability of an erasure being passed in the other drection. Next, we define s_{r+1} in terms of s_r.

Recall that in round $r + 1$, c_i sends an erasure to p_j if and only if all its incoming messages in round r were erasures and it received $y_i =?$. Thus, $s_{r+1} = \alpha \cdot (s'_r)^{dv-1}$, where we used the fact that all the messages received by c_1 are independent random variables and the fact that s'_r is independent of the choice of edge.

p_i will send an erasure to c_j if any one of the incoming messages was an erasure. Thus, we have that $s'_r = (1 - s_r)^{d_c-1}$ and

$$s_{r+1} = \alpha(1 - (r - s_r)^{d_c-1})^{d_v-1}.$$

1 Threshold Computation

Next, we will analyze the performance of the message passing algorithm. To do that we will need the following definition.

Definition 1.1. $\alpha^* = \min_{x \in [0,1]} \frac{x}{(1-(1-x)^{d_c-1})^{d_v-1}}$

Theorem 1.2. *If $\alpha < \alpha^*$, the message decoder recovers the transmitted code word with probability $1 - 2^{-n^{\Omega(1)}}$.*

Proof. Pick $\ell = \lfloor \frac{g-1}{4} \rfloor$ (as we need $4\ell < g$, where g is the girth in round ℓ and ℓ is the total number of rounds for our proof that all messages are independent variables). Then, $\ell = \Omega(\log n)$.

We will show that $s_\ell \leq 2^{-n^{\Omega(1)}}$.

By the union bound the probability that there's no erasure sent in round ℓ is at least $1 - (\#\text{ edges})s_\ell$, and since the number of edges is $O(n)$, this is at least $1 - 2^{-n^{\Omega(1)}}$, as required.

We show this in two steps:

1. After $t = O(1)$ rounds, s_t is less than $\min(\frac{1}{d_c+1}) \triangleq b - 1$.

2. For any round $r \geq t$, $s_{r+1} < s_r^{1+\varepsilon}$ for some $\varepsilon > 0$.

If we can show these two steps to be true, we will have that $s_\ell \leq 2^{-n^{\Omega(1)}}$. This holds since $s_\ell < (s_t)^{(1+\varepsilon)^{\ell-t}}$ and so, by Step 1, $s_\ell < (\frac{1}{a})^{(1+\varepsilon)^{\ell-t}}$ for some $a > 1$. Finally, as $t = O(1)$ and $\ell = \Omega(\log n)$, $s_\ell \leq 2^{-n^{\Omega(1)}}$.

We will now show that the statements above are true.

We begin with step 1.

Define $g(x) \triangleq \frac{x}{(1-(1-x)^{d_c-1})^{d_v-1}}$, and note that we have $\alpha^* = \min_{x\in[0,1]} g(x)$.Define $f(\alpha_1, x) = \alpha(1-(1-x)^{d_c-1})^{d_v-1}$. Note that $s_{r+1} = f(\alpha, s_r)$.

Further, by definition,

$$f(\alpha, x) = \frac{\alpha x}{g(x)} = (\frac{\alpha}{\alpha^*})(\frac{\alpha^* x}{g(x)}) \leq (\frac{\alpha}{\alpha^*})x,$$

where the inequalitry follows from the fact that $\alpha^* \leq g(x)$.

Thus, for all r, $s_{r+1} < (\frac{\alpha}{\alpha^*})s_r$, and note that $\frac{\alpha}{\alpha^*} < 1$.

To make sure that $s_r < b$ where b is NEED DEF HERE, we can use the above equation, $t = O(\log_{\frac{\alpha}{\alpha^*}}(\frac{\alpha}{b}))$ times. Note that $t = O(1)$ as claimed.

This proved Step 1. We now move to Step 2. For this, we will need the following fact: Fix $r/geqt$. If $a \geq 1$ is an integer and $ax < 1$, $(1-x)^a \geq 1 - ax$. We leave the proof as an exercise.

Using the fact above and the fact that $s_r \leq \frac{1}{d_c-1}$ (as $s_t \leq \frac{1}{d_c-1}$ and $s_r \leq s_t$ we get:

$$s_{r+1} = \alpha(1-(1-s_r)^{d_c-1})^{d_v-1} \leq \alpha((d_c-1)s_r)^{d_v-1}$$

By Step 1, we also have $s_r \leq s_t < \frac{1}{(\alpha(d_c-1)^{d_v-1})^{\frac{1}{d_v-2-\varepsilon}}}$ and it is also the case that $s_{r+1} < s_r^{(1+\varepsilon)}$. This completes the proof. □

Using standard calculus, it can be shown that α^* is is the root of the polynomial $P(x) = (\frac{d_v-1}{d_c-1} - 1)x^{d_c} - 2 - \sum_{i=0}^{d_c-3} x^i$.

Remark 1.3. *As a few concrete notes, from this formula, note that when $d_v = 2$, $\alpha^* = 0$ so for any meaningful performance we need $d_v \geq 3$ which then requires $d_c \geq 4$ for positive rate. If we choose these exact values, $d_v = 3$ and $d_c = 4$, we have $\alpha^* = 0.6474$. At capacity, we would have $\alpha = 1 -$ rate and since rate is $1 - \frac{d_v}{d_c}$, this is $\frac{3}{4} = 0.75 < \alpha^*$. In fact, it can be shown that capacity is never achieved for any fixed values of d_v and d_c.*