A GMD Decoding of Concatenated Codes

We present a proof of Proposition 11.9, restated below, which was used in the construction of linear-time binary codes from Chapter 11. The result in particular implies an efficient algorithm to decode concatenated codes up to the product bound provided there exists an efficient errors-and-erasures decoding algorithm for the outer code, and the decoding of the inner codes can also be performed efficiently (which is usually easy since the dimension of the inner code is typically small).

Proposition A.1. *Let C_{out} be an $(N, K)_Q$ code where $Q = q^k$ and let C_{in} be an $(n, k)_q$ code with minimum distance at least d. Let $\mathbf{C}$ be the $(Nn, Kk)_q$ code obtained by concatenating C_{out} with C_{in}. Assume that there exists an algorithm running in time T_{in} to uniquely decode C_{in} up to less than $d/2$ errors. Assume also the existence of an algorithm running in time T_{out} that uniquely decodes C_{out} from S erasures and E errors as long as $2E + S < \tilde{D}$ for some $\tilde{D} \le \text{dist}(C_{\text{out}})$. Then there exists an algorithm $\mathcal{A}$ running in $O(NT_{\text{in}} + dT_{\text{out}})$ time that uniquely decodes $\mathbf{C}$ from any pattern of less than $\frac{d\tilde{D}}{2}$ errors.*

The proof is based on the same approach as the GMD decoding algorithm due to Forney [60, 59] and its use by Justesen [110] to decode his explicit asymptotically good code constructions. The exact style of presentation is inspired by that of [180]. One technical aspect in the proof is that we show that GMD decoding works with as many rounds of decoding of the outer code as there are distinct weights passed by the inner stage. In particular this implies that one has to invoke the outer errors-and-erasures decoding algorithm at most $\lfloor d/2 \rfloor + 1$ times.

A.1 Proof

Let $\mathbf{r} \in [q]^{Nn}$ be a received word which is at a Hamming distance less than $d\tilde{D}/2$ from a codeword $\mathbf{z}$ of the concatenated code $\mathbf{C}$. We divide $\mathbf{r}$ and $\mathbf{z}$ into N blocks of n symbols each corresponding to the n encodings by the inner codes. Denote by r_i (resp. z_i) the i'th block of $\mathbf{r}$ (resp. $\mathbf{z}$), for $1 \le i \le N$. Let y_i be the unique codeword of C_{in} with $\Delta(y_i, r_i) < d/2$, if one exists.

V. Guruswami: List Decoding of Error-Correcting Codes, LNCS 3282, pp. 333-335, 2004.

The inner decoder can find such an y_i if it exists in time T_{in}. If the inner decoder fails to find any codeword within distance $d/2$ of r_i, we set y_i to be an arbitrary codeword of C_{in}. For each y_i, we compute a weight $w_i = \min\{\Delta(r_i, y_i), \lfloor d/2 \rfloor\}$. The inner codewords $y_1, y_2, \ldots, y_N$ together with the weights $w_1, \ldots, w_N$ can all be found in NT_{in} time.

Assume without loss of generality that $w_1 \le w_2 \le \cdots \le w_N$. Let s be the number of distinct weights among $w_1, w_2, \ldots, w_N$. By the definition of the weights, we clearly have $s \le \left\lfloor \frac{d}{2} \right\rfloor + 1$. Let S_j be the block of (contiguous) indices with the same value of w_i for $1 \le j \le s$, and denote by $\tilde{w}_j$ this common weight. Let $n_j = |S_j|$.

The decoding of $\mathbf{r}$ is now finished as follows. For each p, $1 \le p \le s$, we run the assumed errors-and-erasures decoding algorithm for C_{out} on the received word $\langle y_1, y_2, \ldots, y_N \rangle$, by declaring the y_i's in the last p blocks S_j, $s - p + 1 \le j \le s$, as erasures. If any of these decodings finds a message x such that $\Delta(\mathbf{r}, \mathbf{C}(x)) < d\tilde{D}/2$, we output the codeword $\mathbf{C}(x)$ and terminate the algorithm, otherwise we report that there exists no codeword of $\mathbf{C}$ at a Hamming distance less than $d\tilde{D}/2$ from $\mathbf{r}$.

Since the algorithm runs the outer decoding algorithm s times, the total time of the decoding algorithm is $O(NT_{\text{in}} + dT_{\text{out}})$, as claimed. We next proceed to prove the correctness of the algorithm. That is, if there exists $\mathbf{z} \in \mathbf{C}$ with $\Delta(\mathbf{r}, \mathbf{z}) < d\tilde{D}/2$, then the above algorithm will find and output $\mathbf{z}$.

Let $\ell_i = \Delta(r_i, z_i)$ — then by our definition of w_i, we have $\ell_i \ge w_i$. Also, if $y_i \ne z_i$ (i.e., the inner decoder makes a mistake in position i), then clearly $\ell_i \ge d - w_i$ (by triangle inequality). So if we denote by a_i the indicator variable for $y_i \ne z_i$, we have $\ell_i \ge a_i(d - w_i)$. Together with $\ell_i \ge w_i$, this gives

$$\ell_i \ge (1 - a_i)w_i + a_i(d - w_i) = w_i + a_i(d - 2w_i) \ . \tag{A.1}$$

We would like to prove that if the decoding failed to find the codeword $\mathbf{z}$, then we must have $\Delta(\mathbf{r}, \mathbf{z}) \ge d\tilde{D}/2$ errors. In our notation this means we want to prove

$$\sum_{i=1}^{N} \ell_i \ge \frac{\tilde{D}d}{2} \ . \tag{A.2}$$

Define the quantities $A_j = \sum_{i \in S_j} a_i$ and $L_j = \sum_{i \in S_j} \ell_i$. We have by (A.1), for $1 \le j \le s$,

$$L_j \ge n_j \tilde{w}_j + A_j(d - 2\tilde{w}_j) \ . \tag{A.3}$$

Rewriting (A.2), recall that our goal is to prove that

$$\frac{1}{d} \sum_{j=1}^{s} L_j \ge \frac{\tilde{D}}{2} \ . \tag{A.4}$$

Define $x_j = (1 - 2\tilde{w}_j/d)$. Clearly $1 \ge x_1 > x_2 > \cdots > x_s \ge 0$. We have from (A.3) that

$$\frac{L_j}{d} \ge n_j \frac{(1 - x_j)}{2} + A_j x_j \ . \tag{A.5}$$

Define $\varDelta_j = \frac{n_j}{2} - A_j$. Using (A.5) above and the fact that $\sum_{j=1}^{s} n_j = N$, we get that in order to prove (A.4), it suffices to prove that

$$\sum_{j=1}^{s} \varDelta_j x_j \leq \frac{N - \tilde{D}}{2} \ . \tag{A.6}$$

Now if each of the s errors-and-erasures outer decodings fail to find the codeword $\mathbf{z}$, then in each run the $E + S/2 < \tilde{D}/2$ condition must fail. In such a case we must have, for each p, $1 \leq p \leq s$,

$$\sum_{j=1}^{p} A_j + \frac{1}{2} \cdot \sum_{j=p+1}^{s} n_j \geq \frac{\tilde{D}}{2} \ , \tag{A.7}$$

which is the same as

$$\sum_{j=1}^{p} \varDelta_j \leq \frac{N - \tilde{D}}{2} \ . \tag{A.8}$$

Define $x_{s+1} = 0$. Multiplying the p'th equation above with the non-negative quantity $(x_p - x_{p+1})$ for $1 \leq p \leq s$, and adding up the resulting inequalities, we get

$$\sum_{j=1}^{s} \varDelta_j x_j \leq \frac{N - \tilde{D}}{2} \cdot x_1 \leq \frac{N - \tilde{D}}{2} \ , \tag{A.9}$$

which is exactly Equation (A.6) that we had to prove. □