Concatenated Codes for Deletion Channels

Johnny Chen, Michael Mitzenmacher, Chaki Ng and Nedeljko Varnica
Division of Engineering and Applied Sciences, Harvard University
johnnyc@eecs.harvard.edu, michaelm@eecs.harvard.edu, chaki@eecs.harvard.edu, varnica@hrl.harvard.edu

Abstract — **We design *concatenated* codes suitable for the deletion channel. The inner code is a combination of a single deletion correcting Varshamov-Tenengolts block code and a marker code. The outer code is a low-density parity-check (LDPC) code. The inner decoder detects the synchronizing points in the received symbol sequence and feeds the outer LPDC decoder with soft information. Our simulation results with regular LDPC outer codes demonstrate that the bit error rates of 10^{-6} can be obtained at rate 0.21 when the probability of deletion is 8%.**

I. Channel Model and Code Structure

In the memoryless deletion channel model [1], each transmitted symbol is independently deleted with probability P_d; otherwise it is transmitted correctly. As the codewords are passed through the deletion channel the location and the size of each codeword become unclear.

Our coding scheme is shown in Figure 1. Information bits are first encoded by an outer low-density parity-check (LDPC) encoder [2]. We denote the LDPC block length with N. Then the blocks of N encoded bits are broken into blocks of length k. The inner code consists of Varshamov-Tenengolts (VT) code [3] and Marker code [4]. The VT encoder encodes k-bit blocks into blocks of length n. The marker code is used to solve the synchronization problem. A marker (header) is a set of bits with specific length (marker length), inserted between a predetermined number of bits in the code sequences encoded by the outer LDPC and inner VT encoder.

The VT code $VT_a(n)$ is a single-deletion correcting set of length n binary strings $\underline{x} = x_1 \ldots x_n$ satisfying

$$\sum_{i=1}^{n} i x_i \equiv a (mod\ (n+1)).$$

Let $f : \{0,1\}^k \rightarrow \mathcal{C} \subset VT_a(n)$ be an encoding function and $\underline{b} = b_1 \ldots b_k$ be a message. Our inner decoder g assigns the probability vector $g(\underline{w}) = (p_1, \ldots, p_k)$ to each string $\underline{w}$ observed from the channel, where

$$p_i = \Pr[b_i = 1 | \underline{w}] = \sum_{b : b_i = 1} \Pr[\underline{b} | \underline{w}].$$

For small k it is possible to preprocess $g(\underline{w})$ for all $\underline{w}$. Let $Z_i^f(\underline{w})$ indicate if hard decoding on p_i gives the correct b_i. Call f *optimal* if it maximizes

$$\delta(f) = \mathrm{E}_{\underline{w}} \left[\sum_{i=1}^{k} Z_i^f(\underline{w}) \right].$$

We use locally optimal f; details appear in [5].

To facilitate header detection we chose headers consisting entirely of 0's in conjunction with VT codewords with high Hamming weight.

[1] This work was supported by the NSF Grant CCR-0118701.

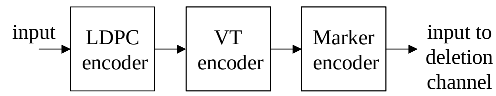

Figure 1: Encoder structure.

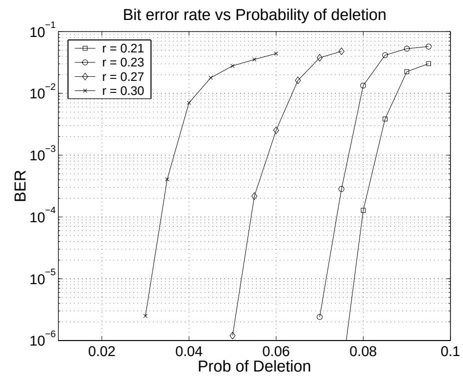

Figure 2: BER vs P_d, with $M = 5$, $N = 5000$, $k = 5$ and $n = 10$.

Our decoder is simple and is organized as follows. The inner decoder regains synchronization and obtains the probabilities passed to the outer message-passing LDPC decoder. Synchronization involves finding (possibly shortened) marker sequences, probabilities for the outer decoder are returned with table lookups, and the LDPC decoder takes $2 - 20$ iterations to recover all the bits. For details, see [5].

II. Simulation Results

We simulated our codes with $M = 5$ marker bits inserted after every $n = 10$-bit VT codeword in the encoded bitstream. The rate of the VT code is $r_{vt} = k/n = 1/2$. The long (size $L = 2M$) markers are inserted after every $B = 50$ codewords [5]. Thus, the rate of the inner code is $r_{in} = 0.329$. The rate r_{out} of the outer LDPC code is varied to obtain appropriate overall rates $r = r_{out} \cdot r_{in}$. We used simple $(3, R)$-regular LDPC codes as outer codes and obtained results shown in Figure 2.

Our scheme works well for P_d above 3%. At $P_d = 8\%$ our codes, with $N = 20000$ bits, achieve bit error rate (BER) 10^{-6} at $r = 0.21$. We note that our codes are still far from the lower bound on the capacity [1], which stands at $r = 0.598$ for $P_d = 8\%$.

References

[1] S.N. Diggavi and M. Grossglauser, "On Transmission Over Deletion Channels," *in the Proc. of ISIT 2001*, July 2001.

[2] R.G. Gallager, "Low-Density Parity-Check Codes," *MIT Press*, Cambridge, MA, 1962.

[3] N.J.A. Sloane, "On Single-Deletion-Correcting Codes," *D. Ray-Chaudhuri Festschrift*, 2001, available at http://www.research.att.com/njas/doc/dijen.ps

[4] F.F. Sellers Jr., "Bit Loss and Gain Correction Code," *IRE Trans. Inform. Theory*, vol. 8(1), pp. 35-38, Jan. 1962.

[5] J. Chen, M. Mitzenmacher, C. Ng and N. Varnica, "Concatenated Codes for Deletion Channels," *Technical Report No-???*, Harvard University, April 2003.