Preliminary of Channel Coding

Basic idea of coding

- Shannon's fundamental theorem for channel coding does not tell us how to build good codes.
- Building good codes is the subject of coding theory.
- A code is good and useful if
  - It achieves small error probabilities at rates close to the channel capacity.
  - It can be encoded and decoded efficiently.

  The latter consideration is even more important than the former.
- The basic idea of channel coding is to embed controlled redundancy into a codeword, so that it can withstand channel noise.

- Searching over all possible codebooks to find a good code is prohibitively complex and clearly impractical.
- For efficient encoding and decoding, we seek for structured code design that enables us to implement the code using practical methods (say, using integrated circuits).

Parity check codes

- A simplest parity check code is to append to each message sequence a 0/1 bit so that the total number of 1s is even.
- Example:
  - message $= 0110111$, codeword $= 01101111$.
  - message $= 0001010$, codeword $= 00010100$.
- Considering a BSC, if there are an odd number of bits flipped during transmission, the decoder can detect that the received signal is not a valid codeword and some error must have occurred.
- But this simple scheme does not work if an even number of bits are flipped.
- Furthermore, this scheme is not able to correct the erroneously received codeword.

Hamming codes

- A more useful parity check code is the (7, 4, 3) Hamming code.
- Set $M = 16$, so that the messages can be represented in binary format from 0000 through 1111.
- Consider to append 3 parity check bits to these message sequences, so as to form a codeword.
- We choose to set the parity check bits as the following three modulo sums: $x_1 \oplus x_2 \oplus x_4$, $x_1 \oplus x_3 \oplus x_4$, and $x_2 \oplus x_3 \oplus x_4$.

So a codeword is like:

$$x_1, x_2, x_3, x_4, x_1 \oplus x_2 \oplus x_4, x_1 \oplus x_3 \oplus x_4, x_2 \oplus x_3 \oplus x_4$$

Codewords are:

0000000	0100101	1000011	1100110
0001111	0101010	1001100	1101001
0010110	0110011	1010101	1110000
0011001	0111100	1011010	1111111

There are a number of beautiful facts about this code.

- Except for the all-0 codeword, all the codewords have at least three 1s.
- In fact, any two codewords differ in at least three digits.
- So we say that the minimum distance of the code is 3. This is why we name this code the (7, 4, 3) Hamming code.

- The code is a linear code, that is, the modulo sum between any two codewords is also a codeword.
- In fact each codeword can be obtained from its corresponding message following a matrix multiplication (in modulo 2):

$$\underline{x} = \underline{w} \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

- This matrix is called the generator matrix of the code, denoted by **G** in coding theory.
- Here, the first square sub-matrix of **G** is diagonal, implying that the first 4 digits of a codeword are exactly the message it encodes. Such a code is called systematic; otherwise, it is called non-systematic.

- From $\mathbf{G}$ we can further construct another matrix $\mathbf{H}$ satisfying $\mathbf{H}\mathbf{G}^T = \mathbf{0}$ as

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

  Such a matrix is called the parity check matrix of the code.

- So if we encode a message $\underline{w}$ to codeword $\underline{x} = \underline{w}\mathbf{G}$, transmit $\underline{x}$ over a BSC, and receive $\underline{y} = \underline{x} \oplus \underline{e}$, we have

$$\begin{aligned} \underline{s} &= \mathbf{H}\underline{y}^T \\ &= \mathbf{H}\underline{x}^T \oplus \mathbf{H}\underline{e}^T \\ &= \mathbf{H}\mathbf{G}^T\underline{w}^T \oplus \mathbf{H}\underline{e}^T \\ &= \mathbf{H}\underline{e}^T \end{aligned}$$

  which depends on the error pattern $\underline{e}$ only, and is called the syndrome of the code.

- The (7, 4, 3) Hamming code can detect and correct all error patterns that contain exactly one error (i.e., only one 1 in $\underline{e}$).
- This can be done by listing all possible syndromes and looking up the syndrome table:

error pattern $\underline{e}$	syndrome $\underline{s}$
0000001	001
0000010	010
0000100	100
0001000	111
0010000	011
0100000	101
1000000	110

Beyond Hamming codes

- Many error correcting codes have been developed.
- For nonbinary alphabets, Reed-Solomon codes were invented in 1950s and have been widely used in applications spanning from digital TV to hard disk drives.
- BCH (Bose and Ray-Chaudhuri, Hocquenghem, in 1950s) codes extend the idea of Hamming codes to correct an arbitrary number of errors. Both Hamming and BCH codes belong to cyclic codes, which are based on polynomials in the Galois field. Efficient decoding of BCH codes has been tackled by a series of works by leading coding theorists, including Berlekamp, Massey, and others.
- The theory of algebraic codes has grown into a field connecting algebra and communication engineering.

- Gallager, in his MIT Ph.D. dissertation, invented the low-density parity check (LDPC) codes, in the early 1960s. LDPC codes essentially follow the random coding idea of Shannon, but was largely ignored for many years due to its complexity at that time.
- Convolutional codes, first described by Elias in 1950s, have received extensive research in 1960s and 1970s and have been widely used in applications spanning from deep space communication to cellular networks.
- The popularity of convolutional codes is largely attributed to its relative simplicity and the famous Viterbi decoding algorithm, due to Viterbi and popularized by Forney.

- Unfortunately, almost all the practical codes people constructed until 1970s were viewed as being bad in the sense that for any positive code rate, the ratio between the minimum distance of code and codeword block length vanishes towards zero.
- So it was unclear at that point how one could get close to the channel capacity.
- A joke went like "all codes are good except the ones we know".
- One exception is the idea of concatenated coding due to Forney, who proposed to use "short" random inner code and Reed-Solomon outer code, to approach the channel capacity while retaining the decoding complexity polynomial in the codeword block length. But this analysis is still largely in theory only.

- In 1972, Justesen described a class of codes with strictly positive rate and asymptotically non-vanishing normalized minimum distance.
- In 1993, a practical and empirically "good" code construction was described by Berrou, Glavieux, and Thitimajshima, which they termed as "turbo codes". The basic idea of turbo codes is to iteratively process the information between two connected (via a random-like permutator) convolutional codes.
- This discovery also renewed interest in the LDPC codes invented in Gallager's Ph.D. dissertation in early 1960s.
- LDPC codes were proved to be capacity-achieving, and can be efficiently encoded and decoded using iterative message-passing algorithms on graphs.
- Both turbo codes and LDPC codes have been widely used in numerous applications.

- But the story has not ended.
- In 2008, Erdal Arikan proposed a class of deterministic construction of capacity-achieving codes, called polar codes.
- Polar codes are interesting because
  - it achieves channel capacity;
  - it has a deterministic construction and is highly structured (as opposed to random coding!);
  - it has a rather low encoding/decoding complexity (encoding basically $O(N \log N)$ where N is the codeword length; decoding facilitated by successive decoding).
- Polar codes have also sparked interest in some of the earliest codes. For example, Reed-Muller codes which were invented in 1954 have been found to be intimately related to polar codes and recently proved to be capacity-achieving over binary input symmetric output channels.