In this chapter, we will discuss some popularly used graphical models for codes. These graphical models are usually used to design efficient decoding algorithms for such codes.

I  Tanner Graphs for Linear Block Codes

A linear block code is typically specified by a generator matrix $G$ or, equivalently, by a parity check matrix $H$. The parity check matrix $H$ is a matrix such that for any valid codeword $c \in C$, $cH^T = 0$. This is equivalent to saying that the dot product of the vector $c = [c_1, c_2, \ldots, c_n]$ with every row of the parity check matrix must be zero. For example, a parity check matrix for a $(7, 4)$ Hamming code and the corresponding dot products that should sum to zero are shown in Figure 1.

For this example, the dot products that must sum to zero are given by

\[
\begin{align*}
\text{constraint1: } & c_1 \oplus c_5 \oplus c_6 \oplus c_7 = 0 \\
\text{constraint2: } & c_2 \oplus c_4 \oplus c_6 \oplus c_7 = 0 \\
\text{constraint3: } & c_3 \oplus c_4 \oplus c_5 \oplus c_7 = 0
\end{align*}
\]
A vector \( \mathbf{c} \) is a valid codeword iff it satisfies all the three constraints.

These set of constraints can also be specified by a Tanner graph which is shown below. We associate with every coded bit \( c_i \), a bit node or variable node denoted by a filled circle. With every check or constraint, we associate a check node denoted by a square. An edge is drawn between a bit and a check node if that bit participates in the constraint denoted by the check. The Tanner graph corresponding to the parity check matrix is shown in Figure ??.

With every check node we can associate an indicator function given by

\[
1\{c_a \oplus c_b \oplus c_d \oplus c_e = 0\} = \begin{cases} 
1 & \text{if } c_a \oplus c_b \oplus c_d \oplus c_e = 0 \\
0 & \text{otherwise}
\end{cases}
\]

We can also associate an indicator function or a code membership function \( 1\{\mathbf{c} \in \mathcal{C}\} \) which takes the value 1 if \( \mathbf{c} \in \mathcal{C} \) and zero otherwise.

Then, the code membership function is the product of the indicator functions denoted by the check nodes. i.e., for the \((7, 4)\) Hamming code in the example,

\[
1\{\mathbf{c} \in \mathcal{C}\} = 1\{c_1 \oplus c_5 \oplus c_6 \oplus c_7 = 0\} \cdot 1\{c_2 \oplus c_6 \oplus c_7 = 0\} \cdot 1\{c_3 \oplus c_4 \oplus c_5 \oplus c_7 = 0\}.
\]

We will talk more about the Tanner graph of a linear block code a bit later. Right now, I want you to understand how to draw a Tanner graph for a given parity check matrix. Notice, that there are several parity check matrices for the same code. The Tanner graph depends on the parity check matrix and, hence, there are several Tanner graph representations for the same code. Interestingly, it does seem to matter which exact Tanner graph we use with some kinds of decoders.

II Trellises for Convolutional Codes

The other class of codes which can be represented using a graph is convolutional codes. In this case, the graph is a weighted directed graph called a Trellis diagram. An example of a trellis diagram for a rate-1/3 convolutional code with generator polynomials \([1 + D, 1 + D^2, 1 + D + D^2]\) is shown in Figure ??.

The Viterbi algorithm is a maximum likelihood sequence detector that exploits the trellis structure of the convolutional code. This graphical structure is also something that enforces constraints. Not all \( \mathbf{c} \) are valid codewords, rather only those that correspond to a valid path through the trellis are codewords! Again, the point is that a graph can be used to represent a constraint.

Another graphical representation for the convolution code is that of a tree. This graph is used to design sequential decoding algorithms such as the Fano algorithm or the stack algorithm. The tree representation for the convolutional code is given in Fig. ??.

To know more about convolutional codes and the Viterbi algorithm, please see a book such as Lin and Costello’s coding book.
Figure 2: Trellis diagram for a rate-1/3 convolutional code and encoder structure
Figure 3: Tree representation of a convolutional code.