Lecture : Peeling decoder (contd.) and an introduction to message-passing decoding

Recall the definition of the peeling decoder, from last time. We now present a bound on its performance, via the idea of using stopping sets.

Def 1 (Stopping set): A stopping set (ss) $\mathcal{S} \subseteq \mathcal{V}$ is a collection of variable nodes whose induced subgraph has no check nodes with degree 1.

By definition, we let the empty set be a ss.

Remark: Let $\text{supp}(\underline{c})$ denote the support of a codeword $\underline{c}$ in the code of interest. In order to satisfy the parity-check equations, all check nodes in $N(\text{supp}(\underline{c}))$ must be connected at least twice to variable nodes in $\text{supp}(\underline{c})$. Hence, $\text{supp}(\underline{c})$, for any codeword $\underline{c}$, is a stopping set.

Stopping sets have the following properties:

Proposition 1: (1) If $\mathcal{S}_1$ and $\mathcal{S}_2$ are ss, then so is $\mathcal{S}_1 \cup \mathcal{S}_2$.

(2) Each subset $\mathcal{W} \subseteq \mathcal{V}$ has a unique maximal ss.

(3) If $\mathcal{W} \subseteq \mathcal{V}$ is the collection of erased variable nodes, then the peeling decoder recovers the value of all variable nodes except those in the unique maximal ss. in $\mathcal{W}$.

Proof:

(1) Note that if $i \in N(\mathcal{S}_1 \cup \mathcal{S}_2)$, then $i \in N(\mathcal{S}_1)$ or $i \in N(\mathcal{S}_2)$, implying that i is connected at least twice to nodes in $\mathcal{S}_1 \cup \mathcal{S}_2$.

(2) Define Z to be the union of all ss contained in W; by (1), it follows that Z is the unique maximal ss in W as any ss in W is contained in Z.

(3) Note that since $Z = Z(W)$ is a ss, the variable nodes in W will not be recovered by the peeling decoder. Further, any $T \subseteq W \setminus Z$ is such that $T \cup Z$ is not a s.s. and will be recovered in part by one step of the peeling decoder. ▨

Def 2: A ss in minimal if the only ss it contains besides itself is the empty set.

For a given code $\mathcal{C}$ with a fixed parity-check matrix H (that defines a Tanner graph), let $A_{ss}(h)$ denote the number of minimal stopping sets of cardinality h. Now, over a BEC, note that the peeling decoder is unsuccessful at decoding from the erasures introduced (at some set $\eta \subseteq \mathcal{V}$), only if η contains some minimal ss [Why?].

Hence, it follows that

$$\Pr[\hat{\underline{c}} \neq \underline{c}] = P_{err,peeling}(\epsilon) \leqslant \sum_{h=1}^{n} A_{ss}(h)\, \epsilon^h.$$

FACT [Luby et al. (2001), "Efficient erasure correcting codes"] : Cascaded LDPC codes (followed by an MDS code) achieve the capacity of an erasure channel, when a peeling decoder is used for decoding the LDPC codes.

An aside: MAP decoding over the BEC(ϵ)

Recall that the MAP decoder (or equivalently, the ML decoder for equiprobable codewords) computes

$$\underline{\hat{c}}^{MAP} = \arg\max_{\underline{x} \in \mathcal{C}} P(\underline{x} | \underline{y})$$

$$= \arg\max_{\underline{x} \in \mathcal{C}} P(\underline{y} | \underline{x}) .$$

For an erasure channel, given $\underline{y}$, let $\mathcal{X}(\underline{y})$ denote the set of codewords "compatible" with $\underline{y}$, i.e.,

$$\mathcal{X}(\underline{y}) = \{ \underline{c} \in \mathcal{C} : \; c_j = y_j, \; \text{if } y_j \neq ? \}.$$

If this set $\mathcal{X}(\underline{y})$ has cardinality 1, it follows that MAP decoding will be successful. [Why?]

Now, in order to find the set $\mathcal{X}(\underline{y})$, note that we only need to solve a system of linear equations (over $\mathbb{F}_2$):

Let η be the set of erased locations & for any set S, let H_S denote the submatrix of H obtained by retaining the columns in S.

Then, note that

$$X(\underline{y}) = \{ \underline{x} \in \mathcal{C} : x_{[n]\setminus\eta} = y_{[n]\setminus\eta} \text{ and } H_\eta x_\eta^T = H_{[n]\setminus\eta} y^T_{[n]\setminus\eta} \}.$$

Solving this system of linear equations takes time $O(n^3)$, but decoding via a peeling decoder is expected to be faster.

Remark: Note that the MAP decoder fails if and only if η contains the support of a codeword. However, the peeling decoder fails if and only if η contains a stopping set.

The peeling decoder is hence suboptimal compared to the MAP decoder.

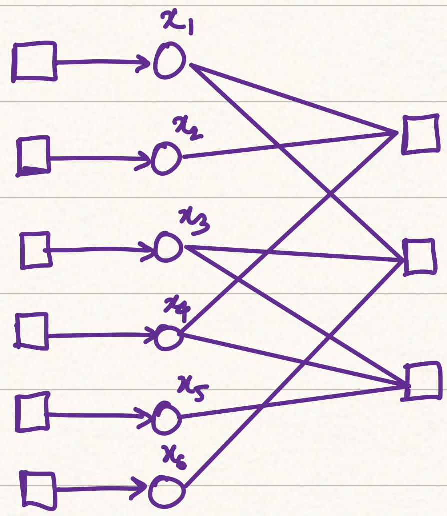
An introduction to message-passing decoding for general channels

The peeling decoder structure informs a more general algorithm for decoding codes on graphs (such as LDPC codes), over channels other than the BEC.

This general algorithm, called the message-passing decoder, is based on the well-studied sum-product algorithm (or "generalized distributive law") that is applied to the problem of computing marginals in probabilistic graphical models.

↳ also called a belief-propagation (BP) decoder

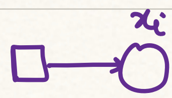
<u>An illustration of the BP decoder</u> : Fix a received sequence $\underline{y}$ over some channel $P_{Y|X}$ and let the Tanner graph of the code be as shown below.



Initial messages from the channel

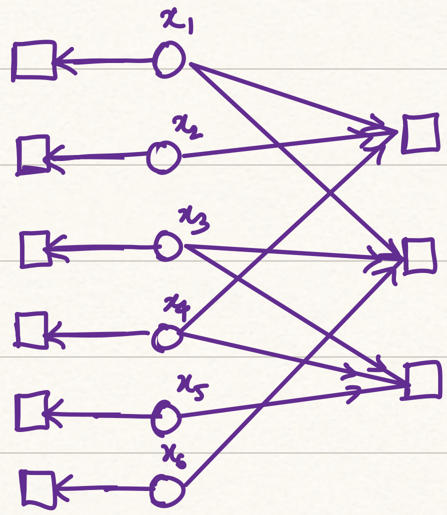
Each message is a vector
$\underline{u} = [u(0), u(1)]$.

(0)



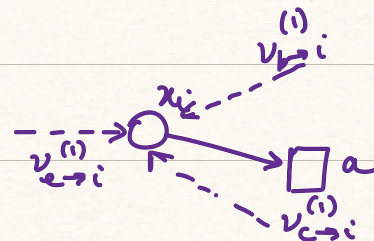
The channel forwards the message

$\underline{m_i} = [P(y_i|0), P(y_i|1)]$



Variable to factor node messages

1. (a)

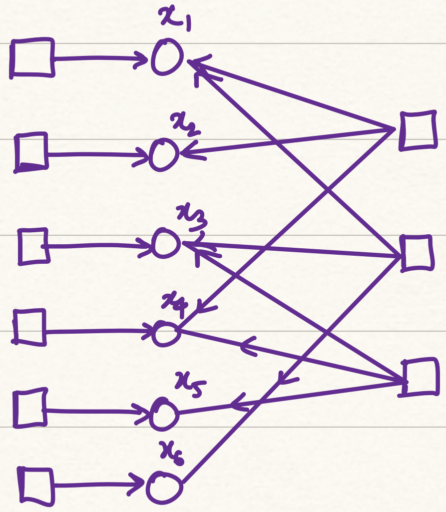


The variable node forwards the message

$$\mu^{(1)}_{i\to a}(x_i) = \prod_{b\in N(i)\setminus\{a\}} \nu^{(1)}_{b\to i}(x_i)$$

to the factor node $a \in N(i)$, for each $x_i \in \{0,1\}$.

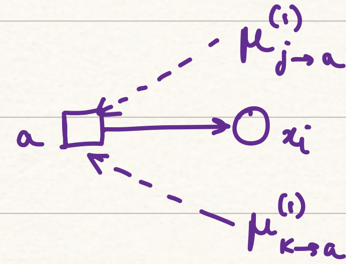
[The vector $\underline{\mu}^{(1)}_{i\to a}$ can be thought of as the msg. vector]



Factor to variable node messages

Here, $\nu^{(1)}_{b \to i}(x)$ are initialized to 1, for all b.

1. (b)



The factor nodes forward the message

$$\nu^{(2)}_{a \to i}(x_i)$$

$$= \sum_{x_{N(a) \setminus \{i\}}} 1\{\bigoplus_{j \in N(a)} x_j = 0\} \cdot \prod_{j \in N(a) \setminus \{i\}} \mu^{(1)}_{j \to a}(x_j)$$

to variable node x_i, for each $x_i \in [0,1]$.

[Here again, $\underline{\nu^{(2)}_{a \to i}}$ can be thought of as the message vector]

... and the steps 1.(a) and 1.(b) are recursed over until convergence of the messages or until a fixed no. of iterations.