**Error Correcting Codes: Combinatorics, Algorithms and Applications** **(Fall 2007)**

# Lecture 34: Iterative Message Passing Decoder

November 11, 2007

*Lecturer: Atri Rudra* *Scribe: Michael Pfetsch & Atri Rudra*

The last lecture introduced the Low Density Parity Check (LDPC) codes and their decoding on the binary erasure channel with erasure probability $\alpha$, $BEC_\alpha$. We now complete a description of the iterative message passing algorithm for decoding regular LDPC codes on the $BEC_\alpha$.

## 1 Iterative message passing decoder for $BEC_\alpha$ (Regular LDPC codes)

The iterative message passing decoder for the $BEC_\alpha$, with regular LDPC codes, is described as follows:

Variable to check nodes

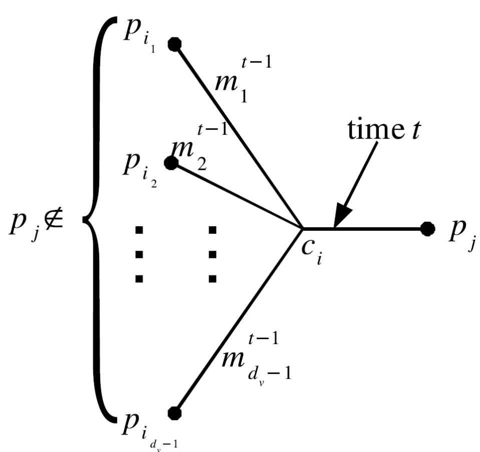


Figure 1: A factor graph for the mapping of message bits to a parity check bit.

$$\Psi_{c_i}^{t,p_j}\left(y, m_1^{t-1}, \cdots, m_{d_v-1}^{t-1}\right) = \begin{cases} b & \text{if at least one of } y_i, m_1^{t-1}, \cdots, m_{d_v-1}^{t-1} \text{ is } b \in \{0,1\} \\ ? & \text{if } y_i = m_1^{t-1} = \cdots = m_{d_v-1}^{t-1} = ? \end{cases} \quad (1)$$

$$\Psi_{p_j}^{t,c_i}\left(y, m_1^{t-1}, \cdots, m_{d_c-1}^{t-1}\right) = \begin{cases} ? & \text{if any one of } m_1^{t-1} = ? \\ m_1^{t-1} \oplus m_2^{t-1} \oplus \cdots \oplus m_{d_c-1}^{t-1} & \text{otherwise} \end{cases} \quad (2)$$

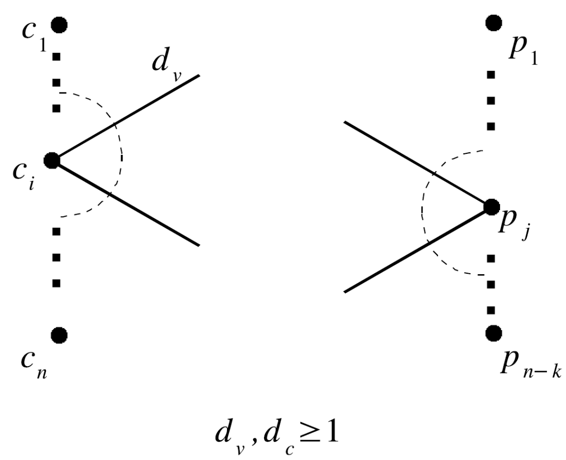


Figure 2: A factor graph for a regular LDPC code $((d_v, d_c) \geq 1)$.

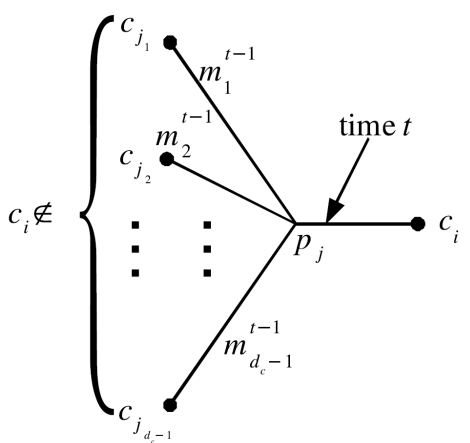


Figure 3: A factor graph for the mapping of parity check bits to a message bit.

**Claim 1.1.** *When a value in $\{0, 1\}$ is sent as a message, it is correct. $p_j \to c_i \Rightarrow i^{th}$ value $= b$*

We now provide a sketch of the proof of this claim.

*Proof.* For the scope of this proof, let us temporarily define $A$ to be the event that the correct value was sent.

By induction, we conclude that at $t = 0$ (var→check), $y \in \{0, 1\} \iff A$.

For $t > 0$ (check $\to$ var). By induction, $m_1^{t-1}, \cdots, m_{d_c-1}^{t-1}$ are correct values.

By the parity check condition, $c_i \oplus m_i^{t-1} \oplus ... \oplus m_{d_c}^{t-1} = 0$ (var→check) if $y \in \{0, 1\} \Rightarrow$done.

By induction, any $m_i^{t-1} \in \{0, 1\}$ is a correct value (no "conflict"). $\square$

**Remark 1.2.** *Messages are taken from the set $\{-1, 0, 1\}$, where a "1" is encoded as a "-1", a "0" is encoded as a "1", and and erasure is encoded as a "0". We can then make the following* *conclusion:*

$$\Psi_{p_i}^{t,c_j} = \prod_{i=1}^{d_c-1} m_i^{t-1} \tag{3}$$

We can now begin a discussion of the decoding algorithm.

# 2 Decoding algorithm

Let us define the variable $l$ as a parameter. Run the following steps for $l$ rounds:

Round $l$

Phase 1: $c_i$ sends message to $p_j$ using $\Psi_{c_i}^{p,2l}(\cdots)$ (for $l = 0$, $c_i$ sends $y_i$ to $p_j$)

Phase 2: $p_j$ sends message $c_i$ using $\Psi_{p_j}^{c_i,2l_t}(\cdot)$

We now describe the paradigm proposed by R. G. Gallager, which can be divided into the following three steps:

1. Step 1: Code construction: The code is constructed by picking an explicit graph of girth $g = \Omega(\log(n))$ as a factor graph. $(4l < g)$
2. Step 2: Analysis of decoder: Given an edge, let $s_e^r$ be the probability that $e_{in}$ erasure is passed over $e$ at round $r$. It is now necessary to derive a recurrence relation between $s_e^{r+1}$ and $s_e^r$.
3. Step 3: Threshold computation: The threshold $\alpha^*$ is computed (either analytically or experimentally) such that, for every $\alpha < \alpha^*$, the probability of error, $p_e$, approaches zero, as $p_e^l \to 0$, for any $e$ (If $\alpha > \alpha^*$, then $p_e^l > 0$). We conclude that one can have reliable transmission over $BEC_\alpha$ for every $\alpha < \alpha^*$.

We will not cover Step 1 at this time. We refer the reader to the book *Low-Density Parity-Check Codes*, by R. G. Gallager [1], for further details of Step 1. The following three claims are made for Step 2:

**Claim 2.1.** $s_e^r = s_{e'}^r, \forall e, e'$

*To prove this claim, we simply use $s^r$, and the rest if the proof follows from induction.*

**Claim 2.2.** *Messages received by any node in round $r \leq l$ are all independent.*

Claim 2.2 can be proved by the following proof by picture:

*Proof.* (by picture) Consider a check node $p_i$. The "dependence" tree is unraveled up to round 0. In Phase 1 of the decoding algorithm, the leaves of the tree are $y_i's$.

□

Before making the final claim, note that every message depends on distinct $y_i's$, each of which are independent random variables by a property of the $BEC_\alpha$.

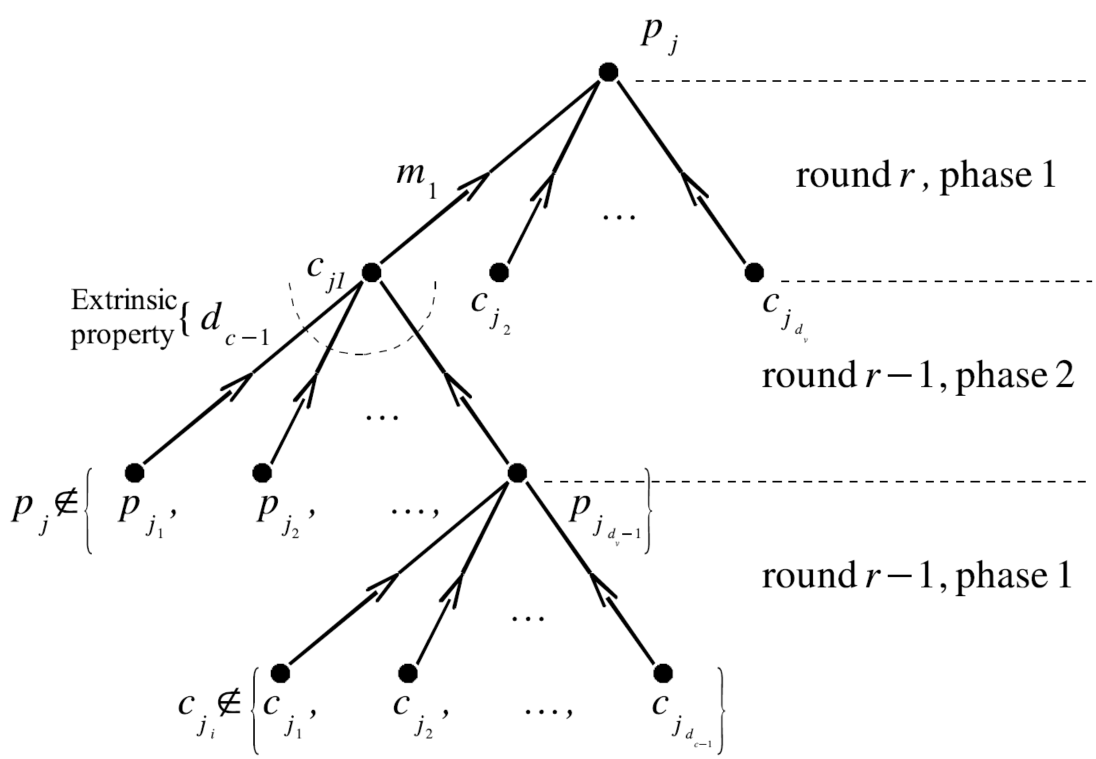


Figure 4: Parity-check “dependence” tree

**Claim 2.3.** *All of the leaves of the tree correspond to $y_i's$ for distinct $i$ values.*

This claim can be proved with the following proof by contradiction:

*Proof.* For the scope of this proof, let us temporarily define $A$ as be the event that all of the leaves of the tree do not correspond to $y_i's$ for distinct $i$ values, and let us also temporarily define $B$ as the event that a cycle in the factor graph exists. The event $B$ could occur if the leaves $w$ and $w'$ corresponded to the same $y_i$ value (or variable node). We assume that $A \implies B$, and consider the following cycle in the factor graph:

$$\underbrace{w \leftrightsquigarrow p_j \leftrightsquigarrow w'}_{\leq 2l}$$

Thus, there is a cycle in the factor graph of length $\leq 4l < g$, and the event $B$ is true, which is a contradiction. We conclude that the final claim is true.

□

Thus, we can observe that the random variables, $y_i$, which correspond to distinct messages, $i$, in the tree, are independent random variables.

# References

[1] R. G. Gallager. *Low-Density Parity-Check Codes*. M.I.T. Press, Cambridge, MA, 1963.