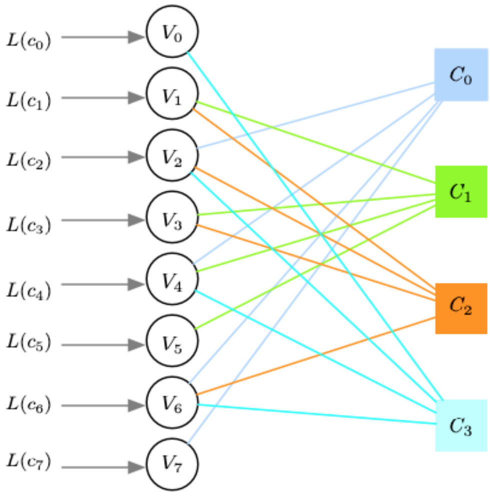
LDPC Decoding

Parity check matrix & Tanner graph example

$$\mathbf{H} = \begin{array}{c} \begin{array}{cccccccc} V_0 & V_1 & V_2 & V_3 & V_4 & V_5 & V_6 & V_7 \end{array} & \\ \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix} & \begin{array}{c} C_0 \\ C_1 \\ C_2 \\ C_3 \end{array} \end{array}$$

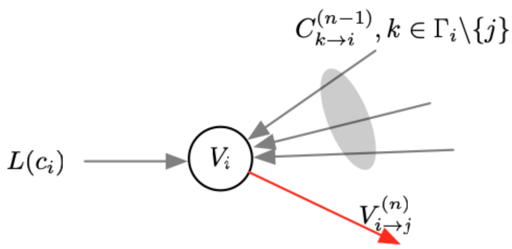


Notations:

- Left side :variable nodes $V_i, i \in \{0, 1, 2 \cdots, 7\}$
- Right side: check nodes $C_j, j \in \{0, 1, 2, 3\}$
- Γ_i denote the set of all checked nodes connected to variable node V_i
- Υ_j denote the set of all variable nodes that connected to check node C_j
- $L(c_i)$ denote the LLR of encoded bit
- $V_{i\to j}$ message from variable node i to check node j
- $C_{j\to i}$ message from check node j to variable node i

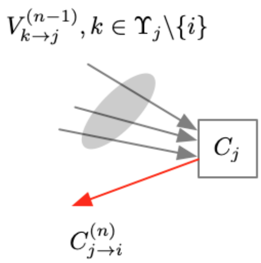
Message(Belief) passing from variable node to check node in the nth round

$$V_{i\to j}^{(n)} = L(c_i) + \sum_{k\in\Gamma_i\backslash\{j\}} C_{k\to i}^{(n-1)}$$



Message passing from check node to variable node in the nth round

$$C_{j\to i}^{(n)} = 2\tanh^{(-1)}\left(\prod_{k\in\Upsilon_j\backslash\{i\}}\tanh\left(\frac{V_{k\to j}^{(n-1)}}{2}\right)\right)$$
$$\approx \prod_{k\in\Upsilon_j\backslash\{i\}}\operatorname{sgn}\left(V_{k\to j}^{(n-1)}\right)\cdot\min_{k\in\Upsilon_j\backslash\{i\}}\left(\left|V_{k\to j}^{(n-1)}\right|\right)$$



Message Passing Decoding algorithm(belief propagation decoding)[2]

For transmitted encoded codeword $\mathbf{c} = [c_0, c_1, ...]$ with channel output $y = [y_0, y_1, \cdots]$, the input to the LDPC decoder is the log-likelihood ratio (LLR) value:

$$L(c_i) = \log\left\{\frac{Pr(c_i = 0|y_i)}{Pr(c_i = 1|y_i}\right\}$$

Initialization : $V_{i\to j}^{(0)} = 0, \quad C_{j\to i}^{(0)} = 0, \quad n = 0$

for each iteration n

- for each variable node i uses received messages from previous round and its accumulated LLR from all previous round, calculate and send messages to its connected check nodes.

$$V_{i\to j}^{(n)} = L(c_i)^{(0)} + \sum_{k\in\Gamma_i\backslash\{j\}} C_{k\to i}^{(n-1)}$$

- each check node j receives message from its connected variable nodes, calculate and send message back.

$$C_{j\to i}^{(n)} = 2\tanh^{(-1)}\left(\prod_{k\in\Upsilon_j\backslash\{i\}}\tanh\left(\frac{V_{k\to j}^{(n-1)}}{2}\right)\right)$$

- update LLR

$$L(c_i)^{(n)} = L(c_i)^{(n-1)} + \sum_{k \in \Gamma_i} C_{k \to i}^{(n)}$$

Make decision based on final $L(c_i)$

Layered decoding algorithm (layered BP) [3]

same performance but converge faster

Initialization : $V_{i \to j}^{(0)} = 0, \quad C_{j \to i}^{(0)} = 0, \quad n = 0$

for each iteration n

for each layer j (each layer corresponds to a check node or a row of exponent matrix)

- for each variable node i uses received messages from previous round and its accumulated LLR from all previous round, calculate and send messages to its connected check nodes.

$$L(y_i) = L(c_i) - C_{j \to i}^{(n-1)}$$
$$V_{i \to j}^{(n)} = L(c_i)$$

- the check node j receives message from its connected variable nodes, calculate and send message back.

$$C_{j \to i}^{(n)} = 2 \tanh^{(-1)} \left(\prod_{k \in \Upsilon_j \setminus \{i\}} \tanh \left(\frac{V_{k \to j}^{(n-1)}}{2} \right) \right)$$
$$L(c_i) = L(c_i) + C_{j \to i}^{(n)}$$

Make decision based on final $L(y_i)$

min-sum decoding algorithm [4]

replace above $C_{j \to i}^{(n)}$ in layered decoding algorithms with its min-sum approximation

$$C_{j \to i}^{(n)} = \prod_{k \in \Upsilon_j \setminus \{i\}} \operatorname{sgn} \left(V^{(n-1)} k \to j \right) \cdot \min k \in \Upsilon_j \setminus \{i\} \left(\left| V_{k \to j}^{(n-1)} \right| \right)$$

Normalised min-sum decoding algorithm [4]

scale min-sum approximation with a positive factor less than 1

$$C_{j\to i}^{(n)} = \alpha C_{j\to i}^{(n)}, 0 < \alpha < 1$$

offset min-sum decoding algorithm [4]

if absolute of min-sum approximation less than offset, make it 0

$$C_{j\to i}^{(n)} = \max(C_{j\to i}^{(n)} - \beta, 0), 0 < \alpha < 1$$

Performance

min-sum < normalized min-sum < offset min-sum < layered BP = BP

[2] Gallager, Robert G. Low-Density Parity-Check Codes, Cambridge, MA, MIT Press, 1963.

[3] Hocevar, D.E. "A reduced complexity decoder architecture via layered decoding of LDPC codes." In IEEE Workshop on Signal Processing Systems, 2004. SIPS 2004. doi: 10.1109/SIPS.2004.1363033

[4] Chen, Jinghu, R.M. Tanner, C. Jones, and Yan Li. "Improved min-sum decoding algorithms for irregular LDPC codes." In Proceedings. International Symposium on Information Theory, 2005. ISIT 2005. doi: 10.1109/ISIT.2005.1523374