LDPC

$H \cdot c^T = 0$; H = parity check matrix $(N-k \times N)$, c = codeword $(1 \times N)$

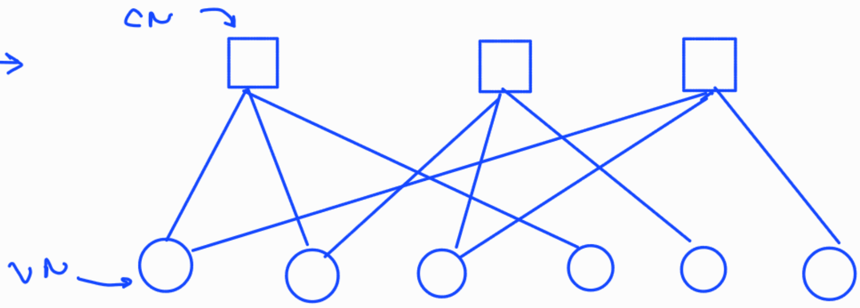
$H \cdot G^T = 0$; G = generator matrix $(k \times N)$

Decode → find c such that $Hc^T = 0$ even after noise.

Let $H = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$; $N = 6$, $N-k = 3$, $k = 3$

Rate $= \frac{k}{N} = \frac{3}{6} = 0.5$

Tanner graph →



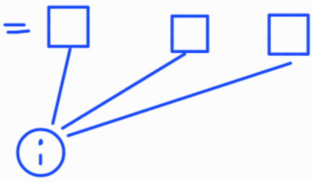
Log likelihood Ratio (LLR) $= L_i = \log \frac{P(y_i \mid c_i = 0)}{P(y_i \mid c_i = 1)}$; $\hat{c} = (c_1\ c_2 \cdots c_N)$

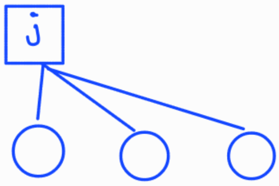
$y = \hat{c} + n = (y_1\ y_2 \cdots y_N)$

variance $= \sigma^2$

$L_i = \frac{2y_i}{\sigma^2}$

$N(i) =$





$= M(j)$

Standard BP algorithm

step 1 :

for each variable node i ;

$$L_i^{(0)} = \text{channel LLR} = \frac{2y_i}{\sigma^2}$$

and initialize messages;

$$L_{i \to j}^{(0)} = L_i^{(0)}$$

Step ~~1~~ 2: Check Node Update (CN $\rightarrow$ VN)

For each check node j and connected variable node i:

$$L_{j\to i} = 2\tanh^{-1}\left(\prod_{i'\in\mathcal{M}(j)\setminus i} \tanh\left(\frac{L_{i'\to j}}{2}\right)\right)$$

This enforces the parity constraint among bits connected to CN j.

—

Step ~~2~~ 3: Variable Node Update (VN $\rightarrow$ CN)

For each variable node i and connected check node j:

$$L_{i\to j} = L_i^{(0)} + \sum_{j'\in\mathcal{N}(i)\setminus j} L_{j'\to i}$$

Each variable node updates its belief using both the channel LLR and messages from other check nodes.

—

Step ~~3~~ 4: A Posteriori LLR and Decision

After each iteration, compute the *a posteriori* LLR:

$$L_i^{(\text{post})} = L_i^{(0)} + \sum_{j\in\mathcal{N}(i)} L_{j\to i}$$

Make hard decisions:

$$\hat{c}_i = \begin{cases} 0, & L_i^{(\text{post})} \geq 0 \\ 1, & L_i^{(\text{post})} < 0 \end{cases}$$

—

Step ~~4~~ 5: Parity Check and Stopping Criterion

If

$$H \cdot \hat{c}^T = 0,$$

decoding stops successfully. Otherwise, repeat the CN and VN updates for a fixed number of iterations (e.g., 30).

—

4. Min-Sum Approximation (Simplified BP)

To reduce computational complexity, the **Min-Sum Approximation** replaces the non-linear tanh and $\tanh^{-1}$ operations with sign and minimum:

$$L_{j\to i} \approx \left(\prod_{i'\in\mathcal{M}(j)\setminus i} \text{sign}(L_{i'\to j})\right) \min_{i'\in\mathcal{M}(j)\setminus i} |L_{i'\to j}|$$

—

5. Summary of Equations

$$\begin{aligned}
\text{Initialization: } & L_{i\to j} = L_i^{(0)} \\
\text{CN update: } & L_{j\to i} = 2\tanh^{-1}\left(\prod_{i'\neq i} \tanh\frac{L_{i'\to j}}{2}\right) \\
\text{VN update: } & L_{i\to j} = L_i^{(0)} + \sum_{j'\neq j} L_{j'\to i} \\
\text{A posteriori: } & L_i^{(\text{post})} = L_i^{(0)} + \sum_{j} L_{j\to i} \\
\text{Decision: } & \hat{c}_i = \begin{cases} 0, & L_i^{(\text{post})} \geq 0 \\ 1, & L_i^{(\text{post})} < 0 \end{cases}
\end{aligned}$$

—

6. Algorithm Summary

1. Initialize $L_{i\to j} = L_i^{(0)}$
2. For each iteration:
   (a) Update CN $\to$ VN messages using parity constraints.
   (b) Update VN $\to$ CN messages using channel and CN inputs.
   (c) Compute posterior LLRs and make hard decisions.
   (d) Stop if all parity checks satisfied.

—

7. Key Insights

- BP exchanges probabilistic information between bits and parity checks.
- Check nodes ensure even-parity consistency.
- Variable nodes combine channel evidence with parity constraints.
- The algorithm converges when all parity checks are satisfied.