A Study of the Performance of LDPC Codes under Various Decoding Algorithms and Schedules

Sangwon Chae, Hyojeong Choi, Gangsan Kim, Hong-Yeop Song
Yonsei University

Jongsun Ahn
Korea Aerospace Research Institute

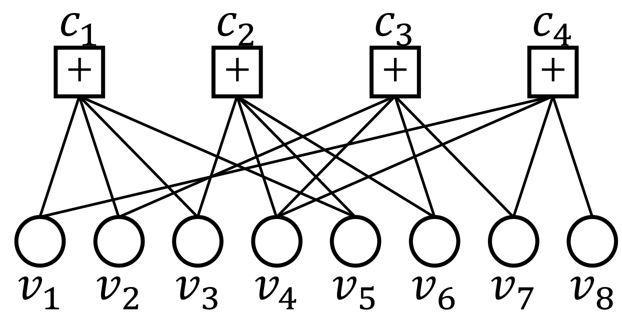
2025 ICUFN

Contents

1. LDPC codes

- **Low-density parity-check (LDPC) codes are linear block error-correcting codes proposed by Gallager in 1962[1].**

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}$$



Check Node Operations:

- Sum-Product (SP)
- Normalized Min-Sum (NMS)
- Offset Min-Sum (OMS)

Scheduling Methods:

- Flooding
- Layered Belief Propagation (LBP)
- Residual Belief Propagation (RBP)

- Balancing decoding complexity and performance has been widely studied [5], [6].

2. Decoding Algorithm

- **Sum-Product [7]**

$$L_{mn}^{(i)} = 2\tanh^{-1}\left(\prod_{n'\in N(m)\backslash n} \tanh\left(\frac{Z_{mn'}^{i-1}}{2}\right)\right)$$

- **Min-Sum [8]**

$$L_{mn}^{(i)} = \left(\prod_{n'\in N(m)\backslash n} \mathrm{sign}\left(Z_{mn'}^{i-1}\right)\right) \cdot \min_{n'\in N(m)\backslash n}\left(Z_{mn'}^{i-1}\right)$$

- **Normalized Min-Sum, Offset Min-Sum [9]**

$$L_{mn}^{(i)} = \boldsymbol{\alpha}\left(\prod\nolimits_{n'\in N(m)\backslash n} \mathrm{sign}\left(Z_{mn'}^{i-1}\right)\right) \cdot \min_{n'\in N(m)\backslash n}\left(Z_{mn'}^{i-1}\right)$$ **(Normalized Min-Sum)**

$$L_{mn}^{(i)} = \left(\prod\nolimits_{n'\in N(m)\backslash n} \mathrm{sign}\left(Z_{mn'}^{i-1}\right)\right) \cdot \max\left\{\min_{n'\in N(m)\backslash n}\left(Z_{mn'}^{i-1}\right) - \boldsymbol{\beta}, 0\right\}$$ **(Offset Min-Sum)**

3. Schedules

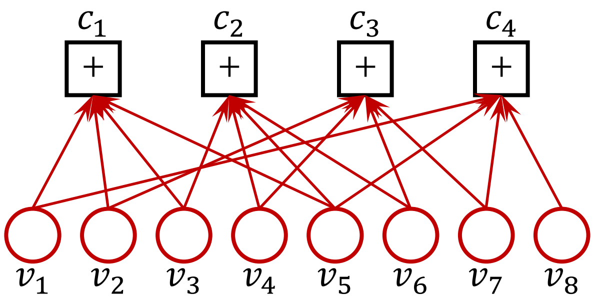
- Schedules used:
  - Flooding [10]
  - Layered Belief Propagation (LBP) [11]
  - Residual Belief Propagation (RBP) [12]

[10] Kschischang and B. J. Frey, "Iterative decoding of compound codes by probability propagation in graphical models," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 2, pp. 219–230, Jan. 1998.
[11] D. E. Hocevar, "A Reduced Complexity Decoder Architecture via Layered Decoding of LDPC Codes," *IEEE Workshop on Signal Processing Systems*, pp. 107–112, Dec. 2004.
[12] G. Elidan, I. McGraw, and D. Koller, "Residual belief propagation: Informed scheduling for asynchronous message passing," in *Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence, UAI 2006*, 2006, pp. 165–173.

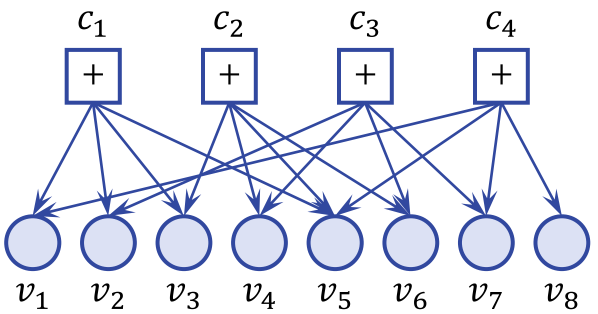
3. 1) Flooding

For each Iteration:

update all ↑

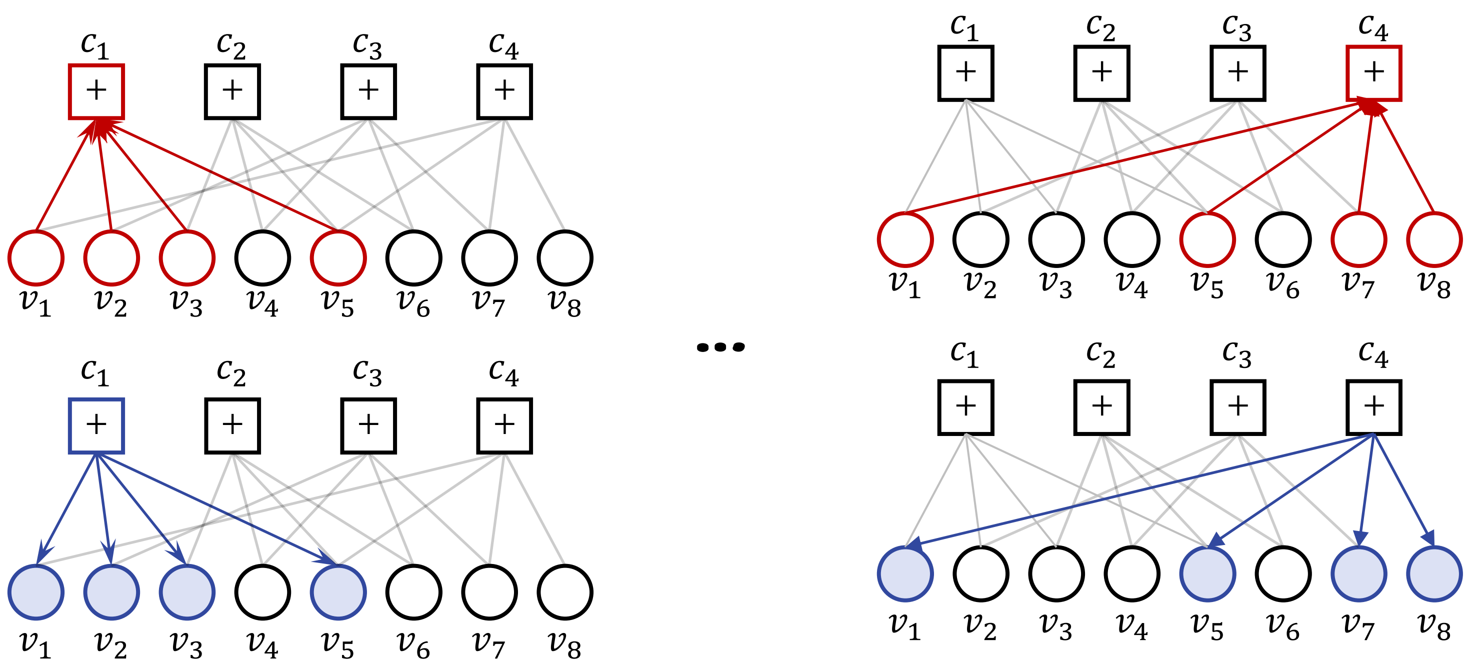


update all ↓ and
update decoded output LLRs



3. 2) Layered Belief Propagation

For each Iteration:



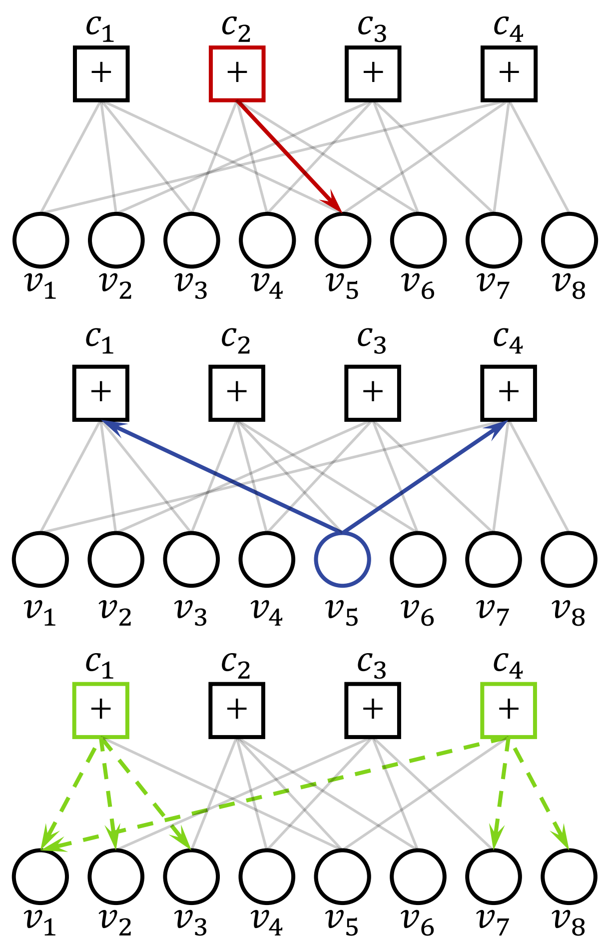
3. 3) Residual Belief Propagation

For each Iteration:

choose maximum residual ↓ of all ↓
then update ↓

update ↑

update the residual of ↓



4. Experimental Result and Analysis

- $(\mathbf{1200}, \mathbf{600})$ **irregular LDPC code** implemented in GPS L1C Subframe 2 [13]
- **Channel Model:** AWGN
- **Modulation:** BPSK

Decoding Algorithm

- Sum-Product
- Normalized Min-Sum ($\alpha = 0.85$)
- Offset Min-Sum ($\beta = 0.5$)

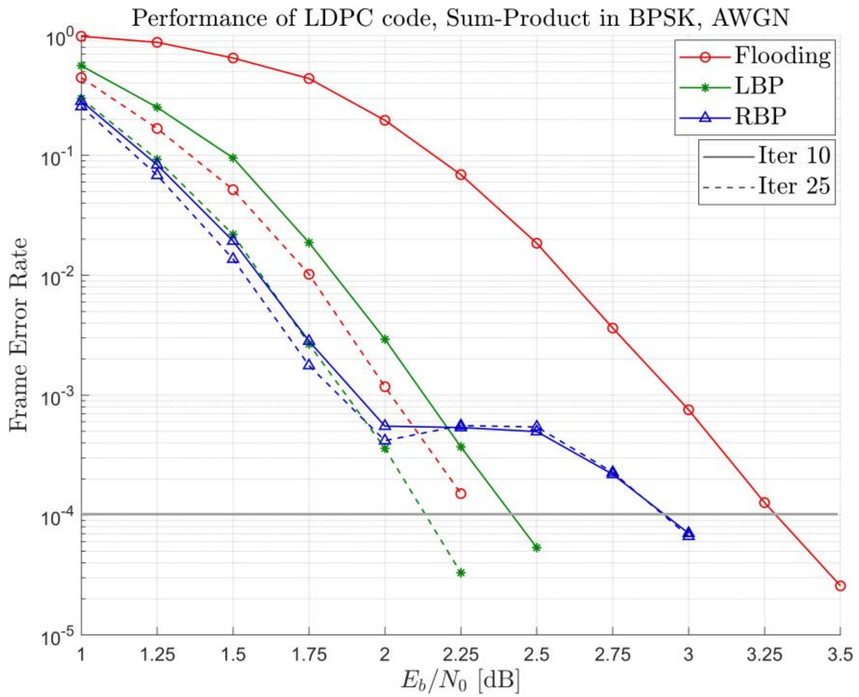
Schedules

- Flooding
- Layered Belief Propagation
- Residual Belief Propagation

Best-performing parameters
selected with a 0.05 interval

4. Experimental Result and Analysis

1) Sum-Product + Flooding, LBP, RBP



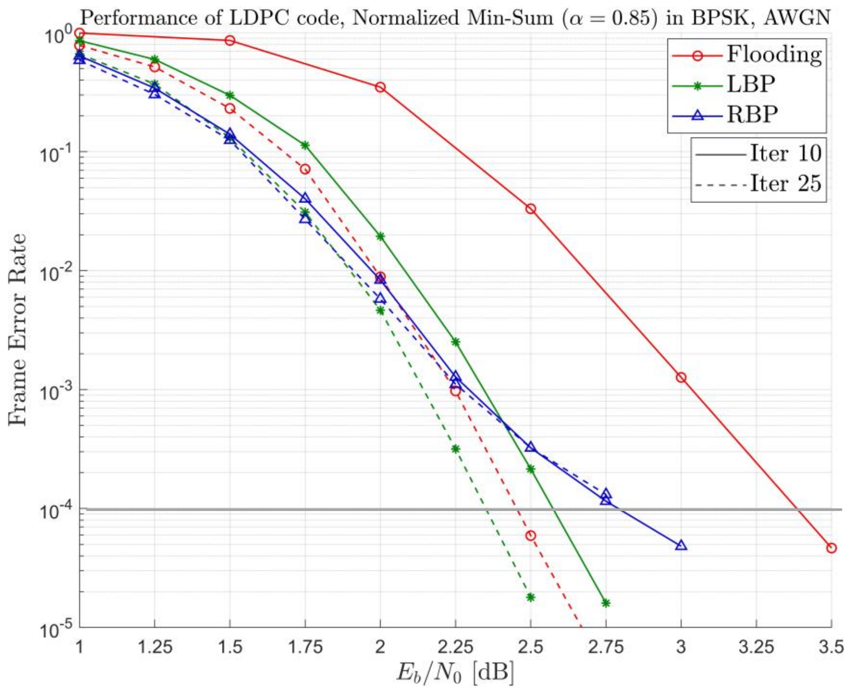
iteration 10 :
LBP > RBP > Flooding

iteration 25 :
LBP > Flooding > RBP

- **LBP** shows the best performance at both iteration 10 and 25.
- **Flooding** shows the slowest convergence speed.
- **RBP** converges fast but exhibits a noticeable error floor.

4. Experimental Result and Analysis

2) Normalized Min-Sum($\alpha = 0.85$) + Flooding, LBP, RBP



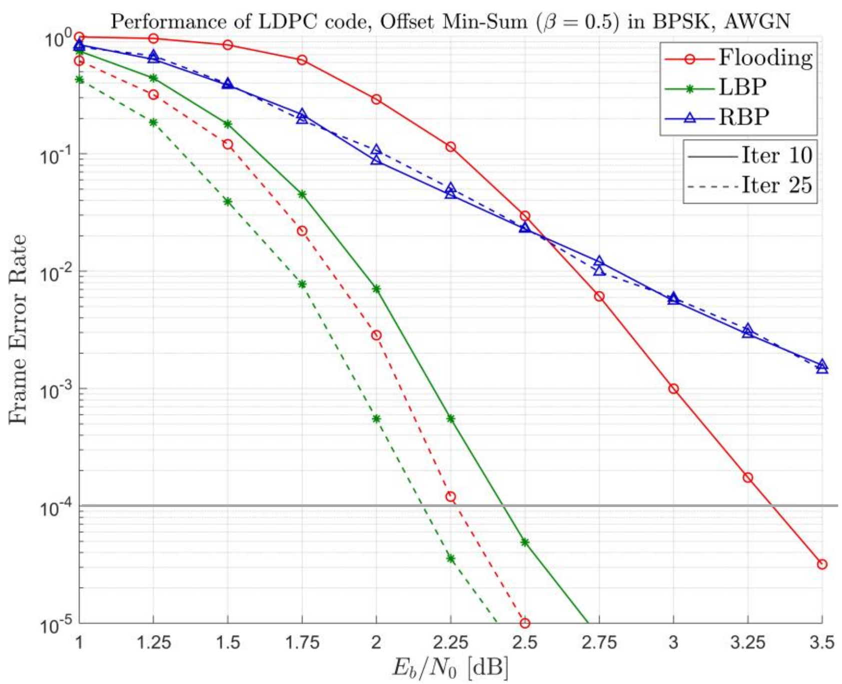
iteration 10 :
LBP > RBP > Flooding

iteration 25 :
LBP > Flooding > RBP

- **LBP** shows the best performance at both iteration 10 and 25.
- **Flooding** shows the slowest convergence speed.
- **RBP** converges fast, but performs the worst at the FER = 10^{-4}.

4. Experimental Result and Analysis

3) Offset Min-Sum($\beta = 0.5$) + Flooding, LBP, RBP



iteration 10, 25 :
LBP > Flooding > RBP

- **LBP** shows the best performance at both iteration 10 and 25.
- **Flooding** shows the slowest convergence speed.
- **RBP** converges fast but shows the worst performance.

5. Conclusion

- We evaluated the decoding performance of a (1200,600) irregular LDPC code using different algorithms and scheduling methods.

- **LBP** shows the best performance in all cases (SP, NMS, OMS) at both 10 and 25 iterations.
- **Flooding** shows the slowest convergence across all decoding algorithms (SP, NMS, OMS).
- **RBP** converges very fast, but shows the worst performance due to an error floor.

- **OMS ($\boldsymbol{\beta = 0.5}$) with LBP** offers a good balance between decoding complexity and performance.

Thank you for listening