

# Node-wise Variable-to-check Residual Belief Propagation Decoding of LDPC Codes



Mi-Young Nam , Jung-Hyun Kim, and Hong-Yeop Song  
{my.nam, jh.kim06, hysong}@yonsei.ac.kr

Coding and Crypto Lab  
YONSEI University

# Contents

---

- Introduction
- Belief Propagation
- Residual BP algorithm
- Variable-to-check RBP algorithm
- Proposed algorithm
- Simulation Result



# Introduction

---

## ■ Belief Propagation

- Provides ML decoding over a cycle-free code
- In some cases, loopy factor graphs of codes have near ML performance
- Iterative algorithm → requires a message-passing schedule
  - ✓ Flooding (simultaneous scheduling)
  - ✓ Sequential
    - Layered BP
    - Shuffled BP
  - ✓ Sequential scheduling allows the convergence speed increase at no cost
- Sequential updating poses the problem of finding the ordering of message updates
  - ✓ Residual BP
  - ✓ Variable-to-check RBP



# Belief Propagation for LDPC

---

- The channel information of the variable node  $v_j$

- $C_{v_j} = \log\left(\frac{p(y_j|v_j=0)}{p(y_j|v_j=1)}\right)$

- Message from variable node  $v_j$  to check node  $c_i$

- $m_{v_j \rightarrow c_i} = \sum_{c_a \in N(v_j) \setminus c_i} m_{c_a \rightarrow v_j} + C_{v_j}$

- Message from check node  $c_i$  to variable node  $v_j$

- $m_{c_i \rightarrow v_j} = 2 \times \operatorname{atanh}\left(\prod_{v_b \in N(c_i) \setminus v_j} \tanh\left(\frac{m_{v_b \rightarrow c_i}}{2}\right)\right)$



# Residual BP [5][6]

---

- A residual is the absolute value of the difference of the LLRs
  - $r(m_k) = \|f_k(\mathbf{m}) - m_k\|$
- RBP calculates the residual using  $m_{c_i \rightarrow v_j}$
- RBP is an informed scheduling strategy that **updates first the message that maximizes the residual**
- RBP is the **edge-based** algorithm(selects an edge)
- Maximum residual means
  - The probability that the value of the node will be changed is high
  - Updating that node first may make BP converge at a higher speed
- It can solve the trapping set problem

[5] A. I. Vila Casado, M. Griot, and R. D. Wesel, "Informed Dynamic Scheduling for Belief-Propagation Decoding of LDPC Codes," *In Proc. IEEE ICC 2007*, Glasgow, Scotland, June, 2007.

[6] A. I. Vila Casado, M. Griot, and R. D. Wesel, "Improving LDPC Decoders via Informed Dynamic Scheduling," *IEEE Information Theory Workshop 2007*, Lake Tahoe, CA, September, 2007.

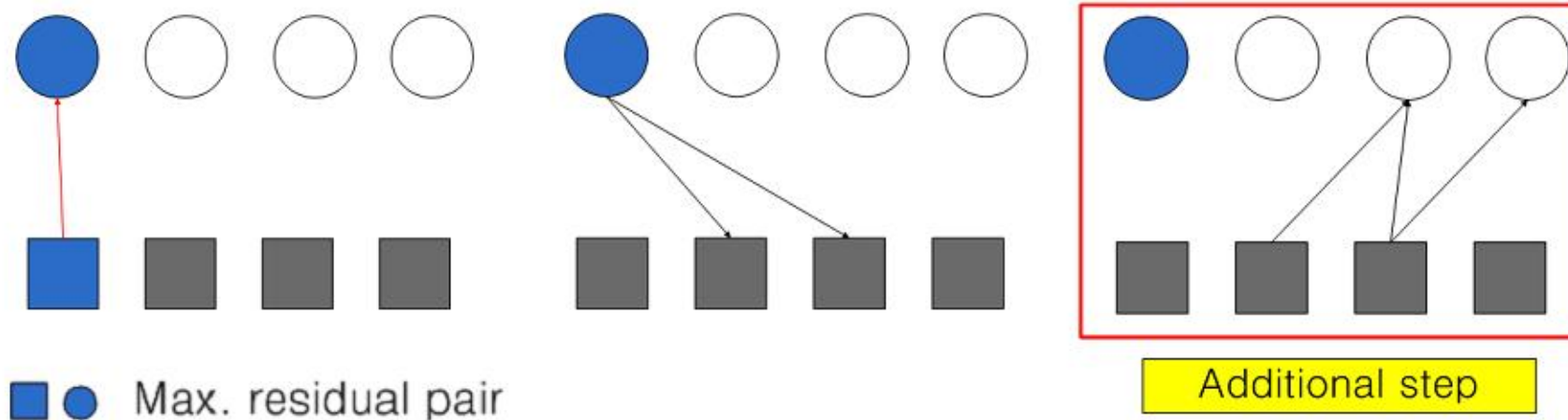




# Residual BP

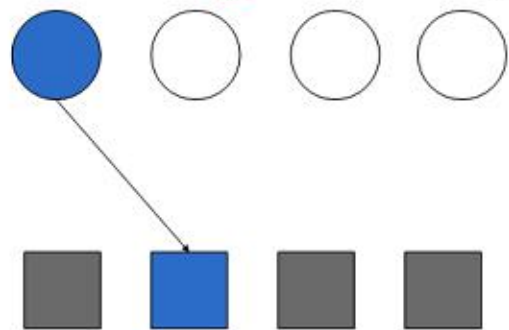
## ■ Some problems

- Greedy algorithm  $\rightarrow$  its error-rate performance for a large enough number of iterations is worse
- Needs additional step for calculating the residual
- There's unnecessary reorder step

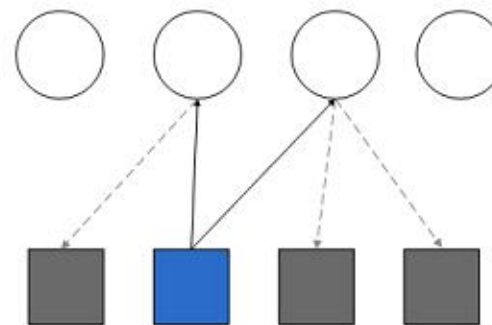


# Variable-to-check RBP[7]

- **Less greedy algorithm**
  - Overcoming negative effect of greediness of RBP
- **One step is reduced than RBP**
  - Lower complexity
- **Calculate the residual using  $m_{v_j \rightarrow c_i}$**
- **Also an edge-based algorithm**



■ ● Max. residual pair



[7] Jung-Hyun Kim, and Ki-Moon Lee, Hong-Yeop Song, "Variable-to-Check Residual Belief Propagation for LDPC Codes," Submitted to *IEEE Electronics Letters*.

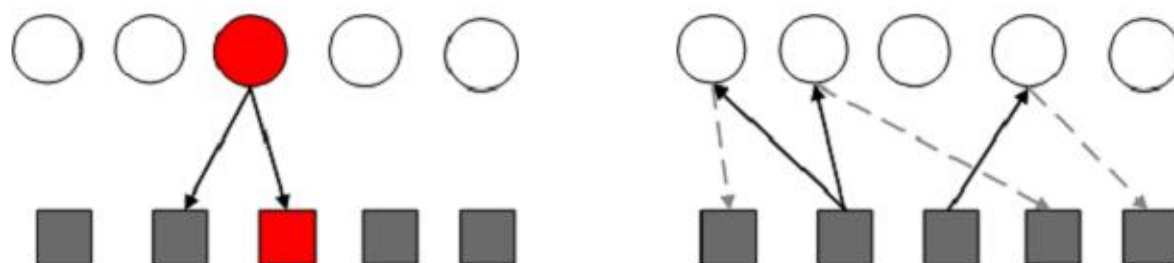


# Node-wise VC-RBP (Proposed)

## ■ Node-based scheduling algorithm

- Just all nodes should be chosen to complete one iteration
  - ✓ cf. Edge-based : all edges should be chosen to complete one iteration
- (# of nodes =  $N$ ) < (# of edges =  $N \cdot d_v$ )  
where  $d_v$  is degree of variable node

→ Lower the complexity

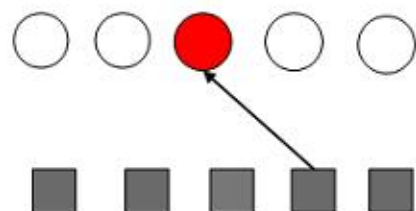




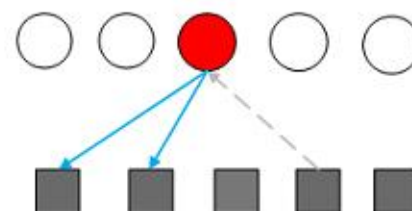
# Node-wise VC-RBP (Proposed)

## ■ Degradation is negligible

- Residual is calculated based on the message  $m_{v_j \rightarrow c_i}$
- So every edge has similar residual value except one incoming edge



If incoming message value to one node is very different from previous value



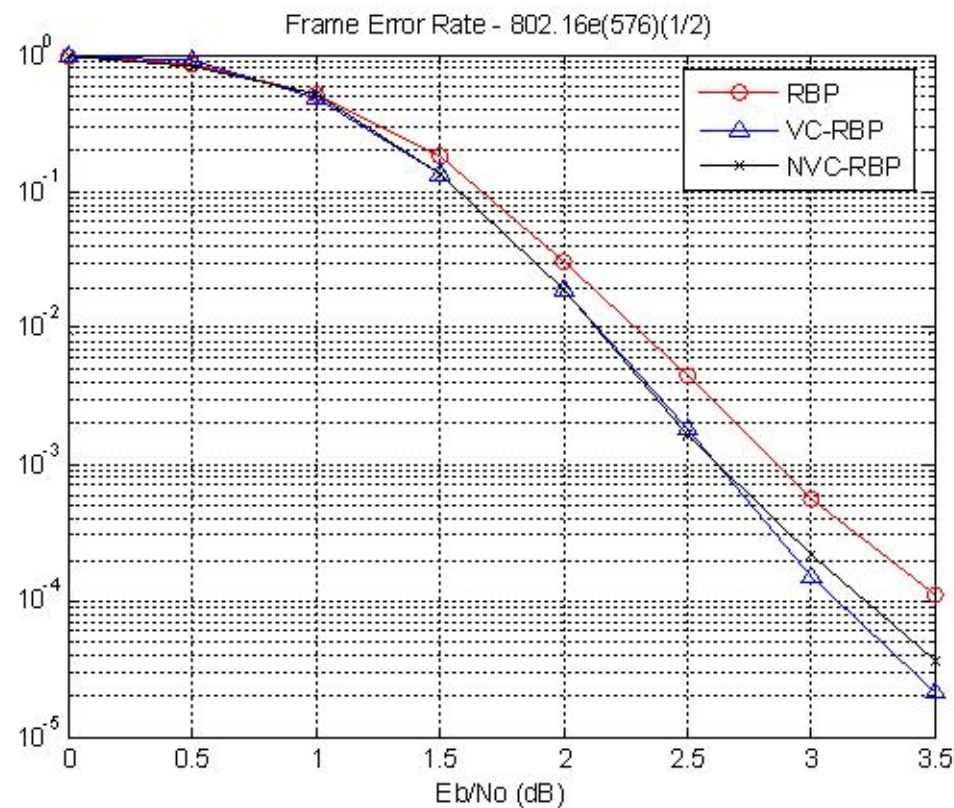
Then all the residuals of the edges outgoing from that node will be very large and the residuals are similar



Therefore, the performance degradation is negligible!!

# Simulation result

- Performance with IEEE 802.16e block length-576, code rate-1/2, maximum 8 iterations



# Conclusion

---

- Node-wise VC-RBP makes **the convergence** of LDPC decoding **fast**.
- Node-wise VC-RBP **reduce the complexity** of the VC-RBP efficiently.
- Its performance **degradation is negligible** compared with the VC-RBP.

