

Multi-Gbit/sec Low-Density Parity-Check Decoders with Reduced Interconnect Complexity

Ahmad Darabiha, Anthony Chan Carusone,
Frank R. Kschischang
University of Toronto, Toronto, Canada

ISCAS, Kobe, Japan

May 2005



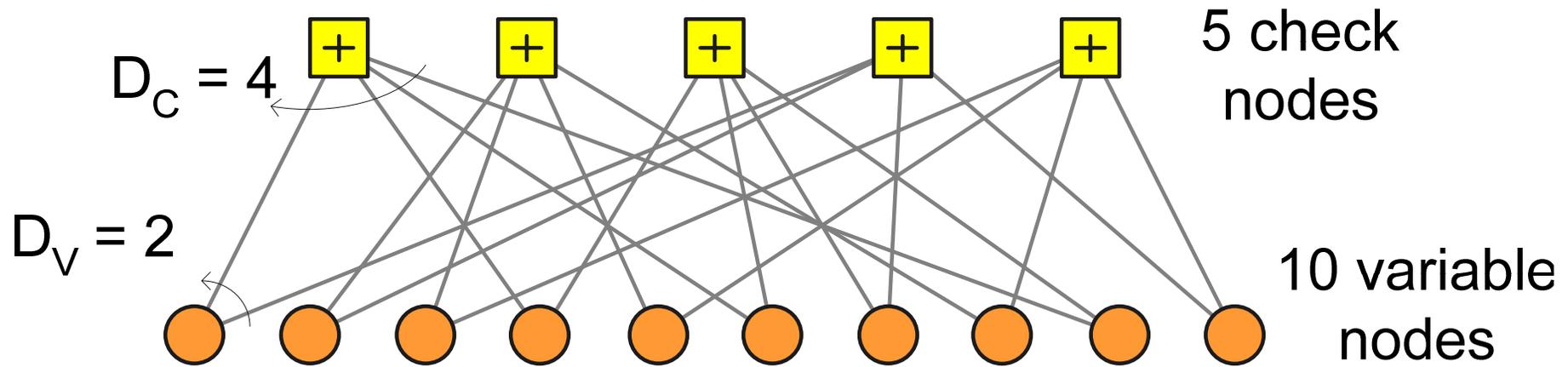
Outline

- Introduction to LDPC codes/decoders
- Proposed techniques
 - Half-Broadcasting
 - Column reordering
- Decoder design
- Conclusion

Introduction

- Low Density Parity Check (LDPC) codes
 - A sub-class of Error Correction Codes (ECC)
 - Defined as the null space of a sparse parity check matrix H
- Motivation
 - Potential for high performance and high throughput decoders
 - LDPC code is adopted for Digital Video Broadcast (DVB-S2) standard
 - The main candidate for 10Gbit Ethernet standard over twisted pair wire

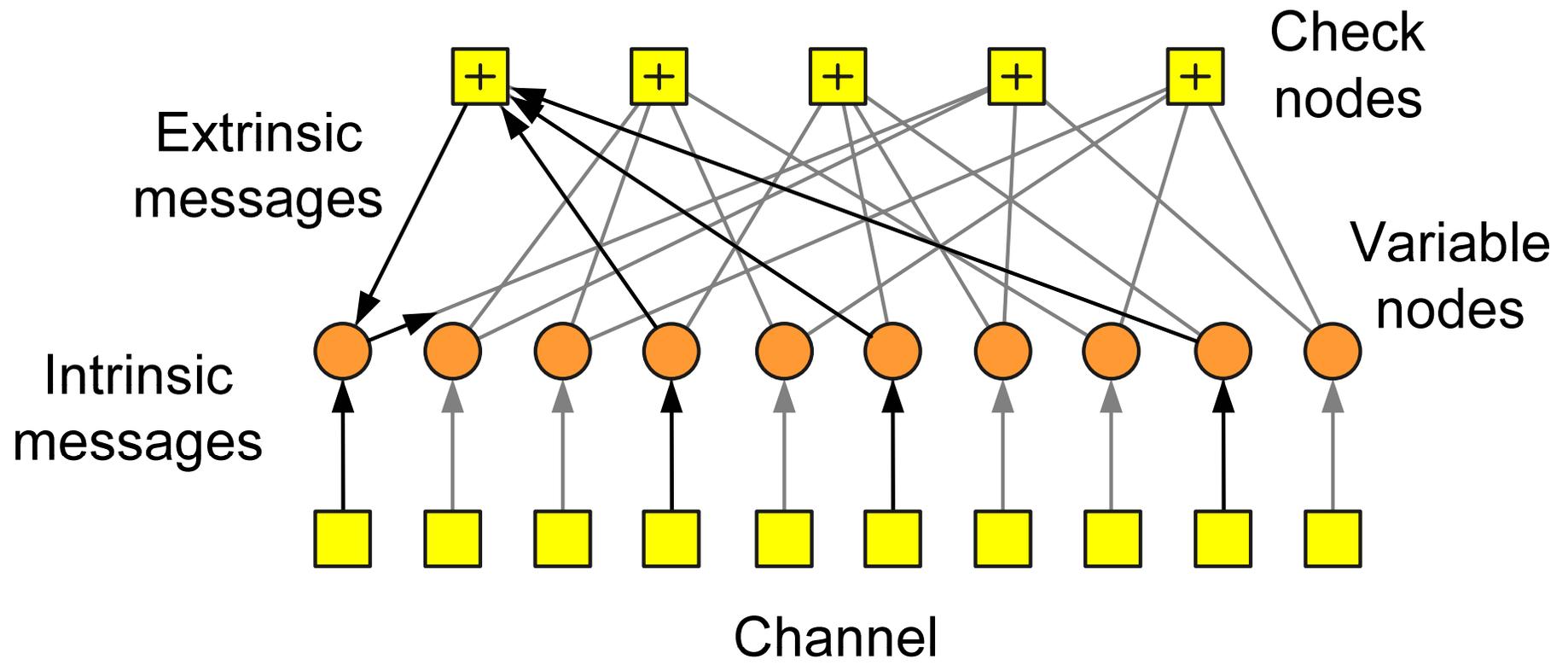
LDPC Codes: Structure



$$H_{5 \times 10} = \begin{vmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \end{vmatrix}$$

- Good LDPC codes are
 - Long (>1,000 bits)
 - With random pattern

LDPC Decoding: Message Passing



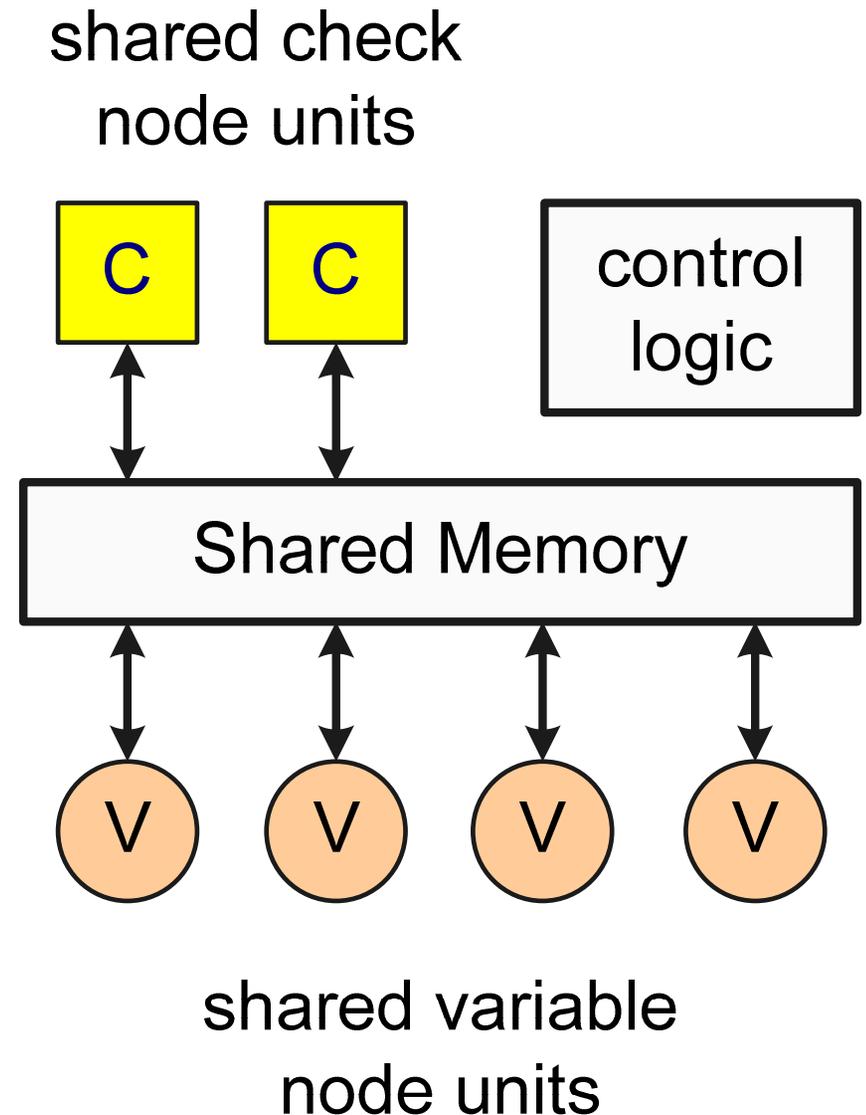
• Example for full-precision MP update functions:

$$c = \text{Check}(v_1, v_2, \dots, v_n) = 2 \tanh^{-1}(\tanh(v_1/2) \tanh(v_2/2) \dots \tanh(v_n/2))$$

$$v = \text{Var}(c_1, c_2, \dots, c_n) = c_1 + c_2 + \dots + c_n$$

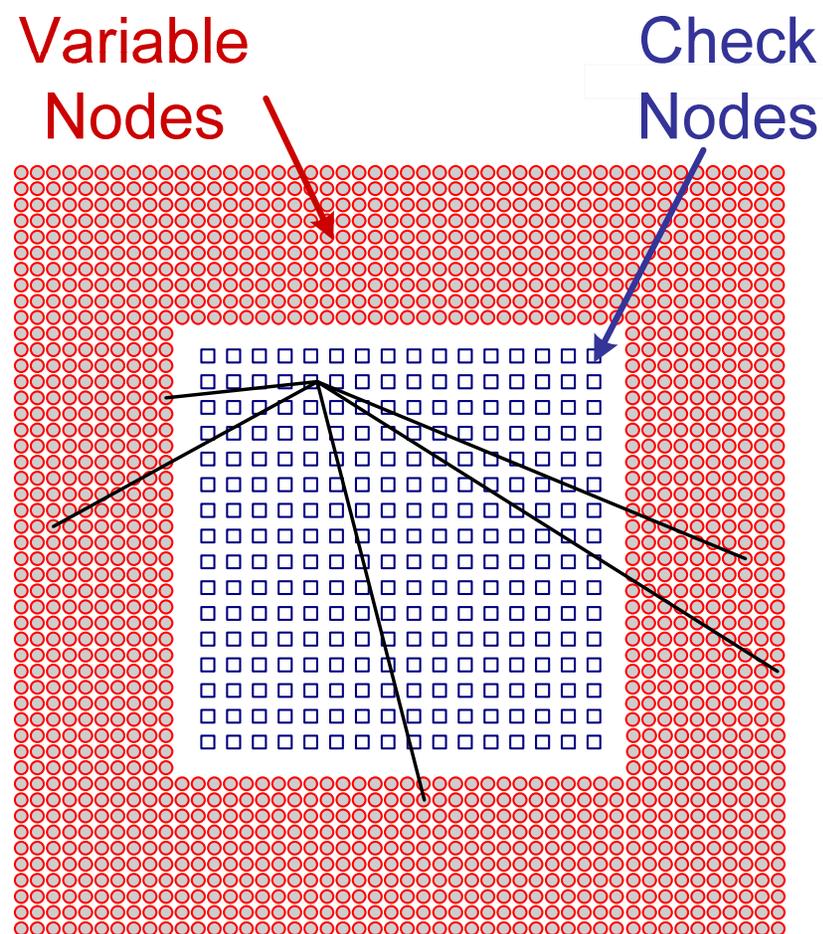
LDPC Decoders: Architecture

- Option 1: Partially parallel architecture [Urard05, Mansour03]
 - A few variable/check processing units
 - **Smaller** area
 - **Lower** throughput
 - Nodes communicate through memory
 - Memory access is the bottleneck



LDPC Decoders: Architecture (cont.)

- Option 2: Fully parallel architecture [Blanksby02]
 - Code graph directly mapped to hardware
 - **Higher** throughput
 - **Larger** area
 - Nodes communicate through hard wires
 - Routing global wires the bottleneck



Outline

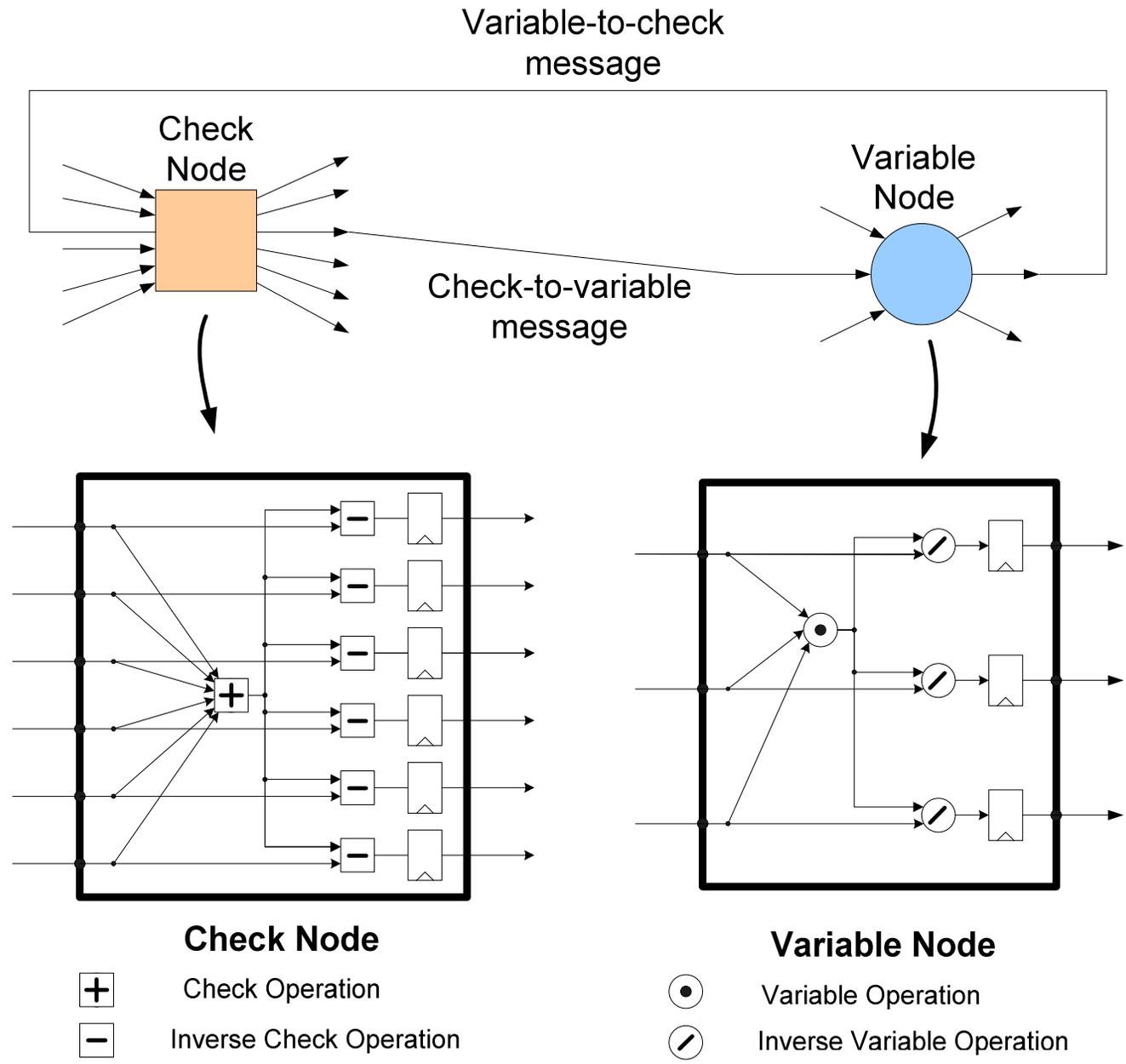
- Introduction to LDPC codes/decoders
- Proposed techniques
 - Half-broadcasting
 - Column reordering
- Decoder design
- Conclusion

Implementation Challenges

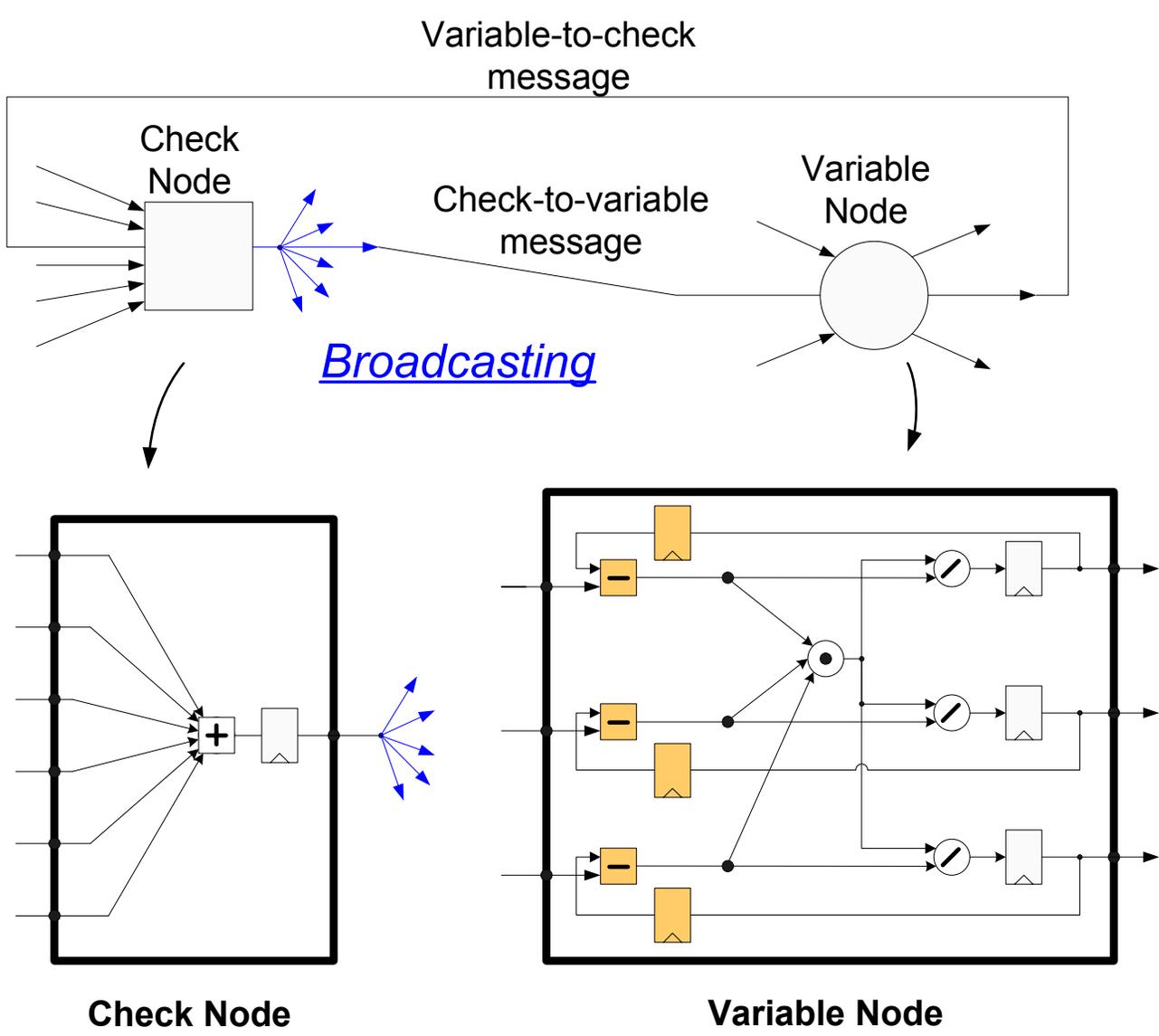
Fully parallel architecture is adopted because of its potential for higher throughput (multi-Gbit/sec).

- Two major challenges:
 - Routing congestion due to random-like interconnections
 - Delay and power consumption introduced by long wires
- Solutions proposed in this work:
 - **Half-broadcasting** reduces total wiring
 - **Column reordering** reduces maximum wirelength

Half-broadcasting

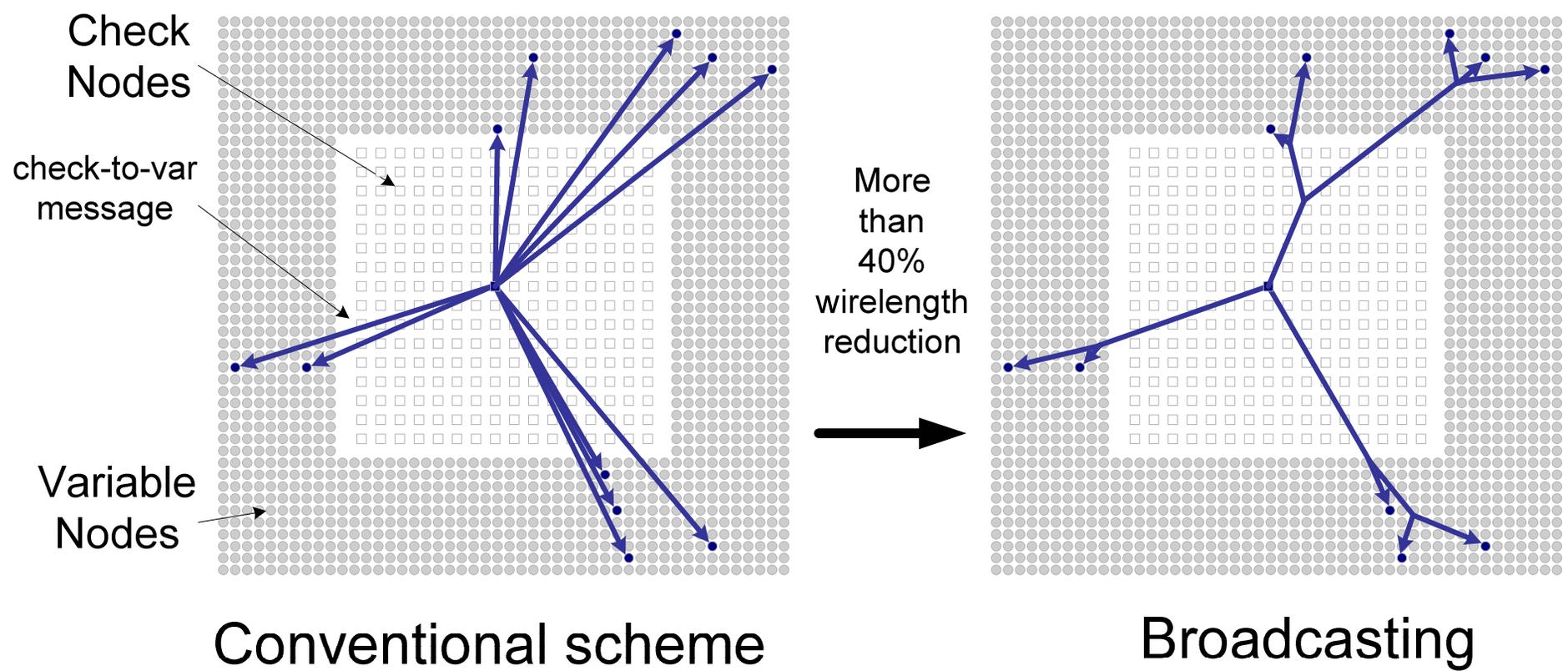


Half-broadcasting (cont.)

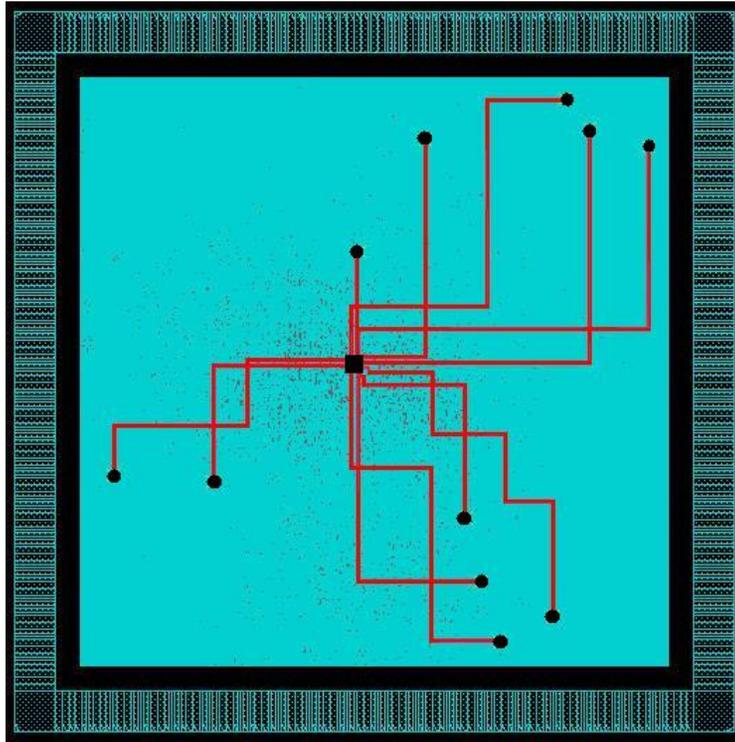


Half-broadcasting (cont.)

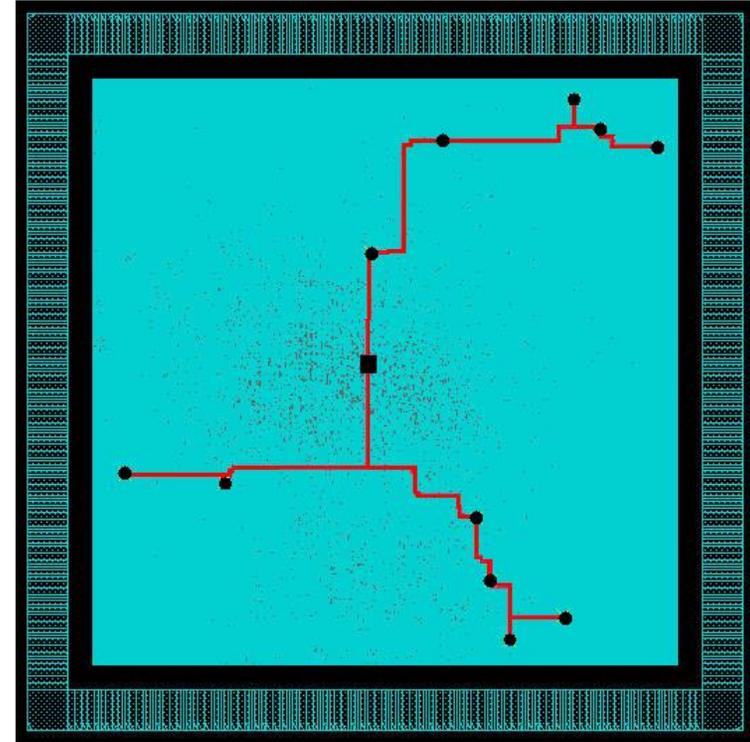
- Half-broadcasting shares portions of the interconnects
 - ⇒ reduces the total wire length
 - ⇒ prevents routing congestion, take less chip area



Half-broadcasting: Results



Without broadcasting

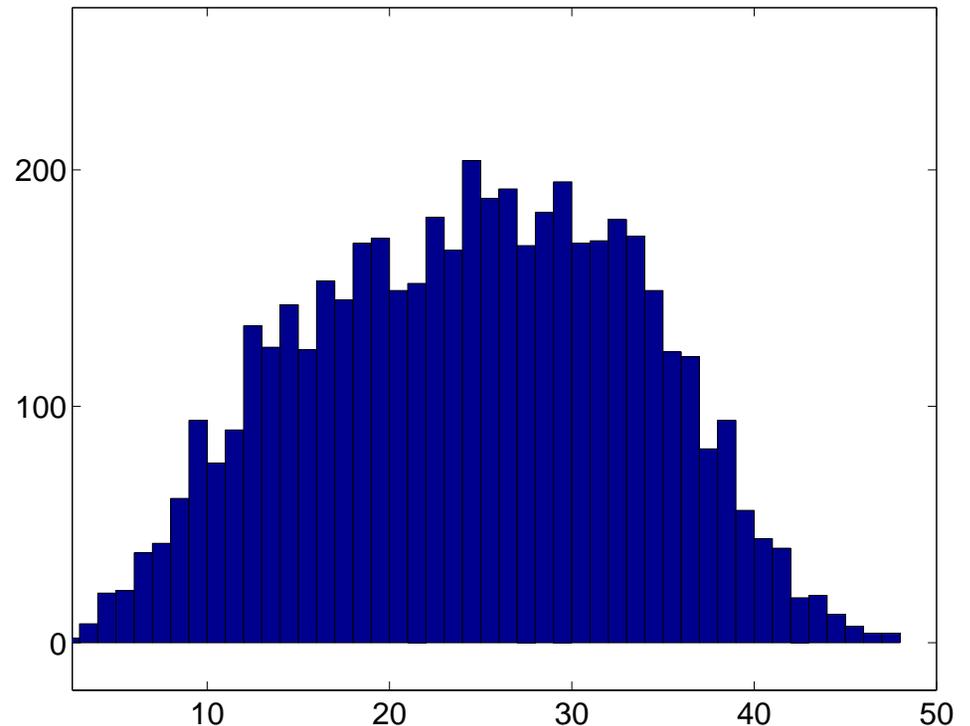


With broadcasting

- More than 40% reduction in total check-to-variable wirelength

Column Reordering

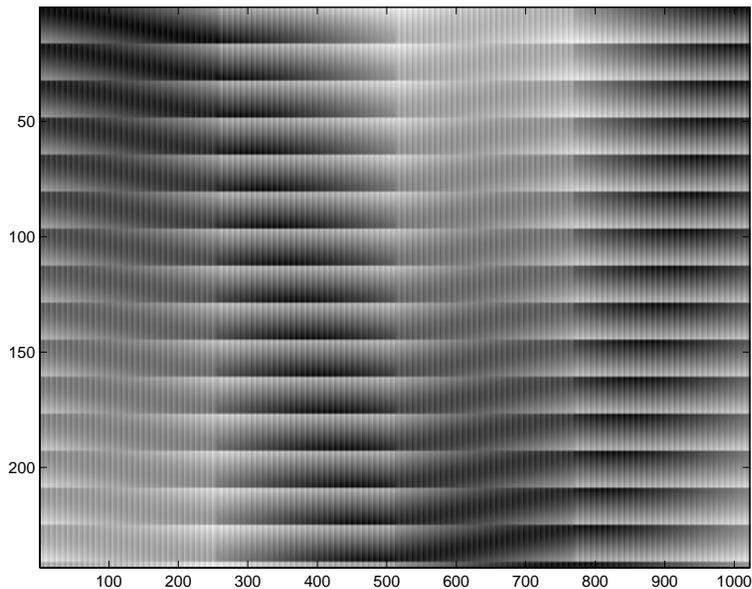
- A placement algorithm that optimizes location of variable nodes in the design layout
- **Removing long wires**
 - **Higher clock frequency**
 - **Lower power consumption**
- **Column reordering**
 - A heuristic algorithm
 - Reorders the columns of parity matrix
 - Globally optimum reordering is NP-hard



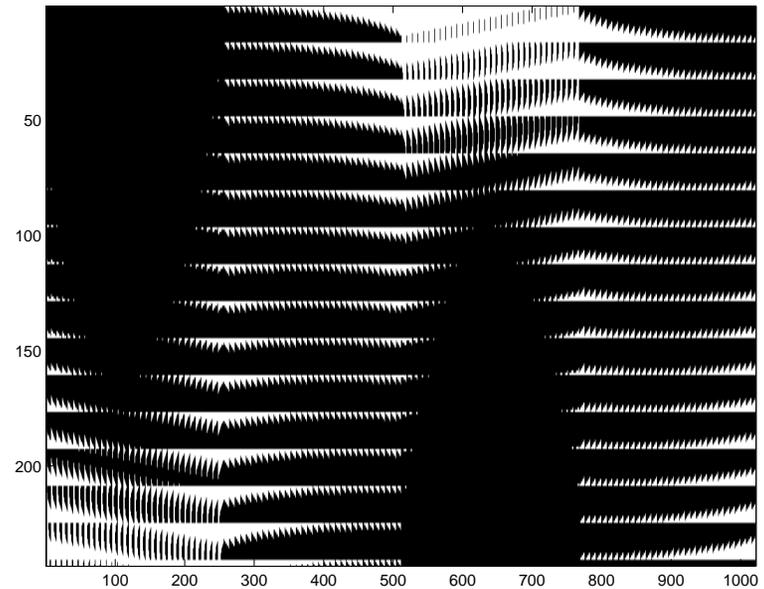
Histogram of Manhattan distances
without column reordering

Column Reordering

1. Generate the cost map matrix



2. Generate threshold cost map

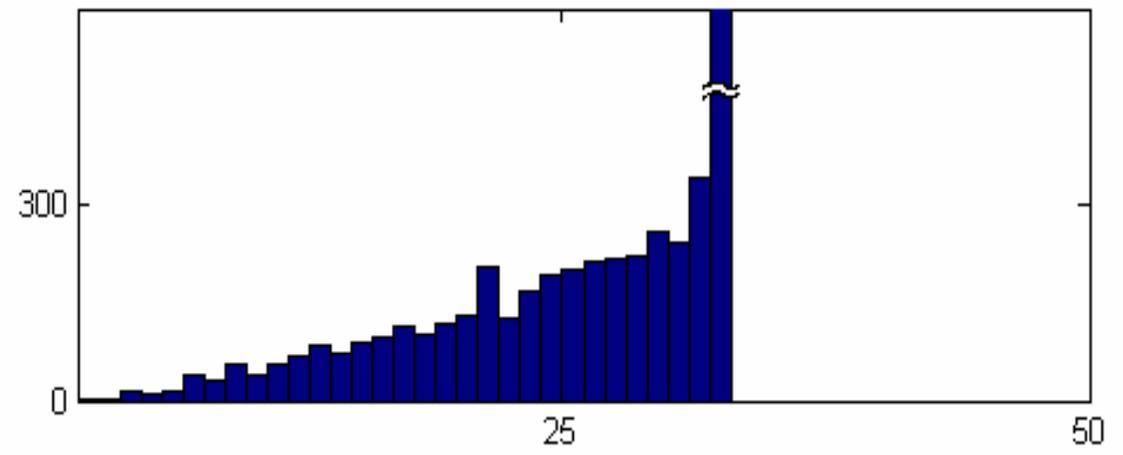
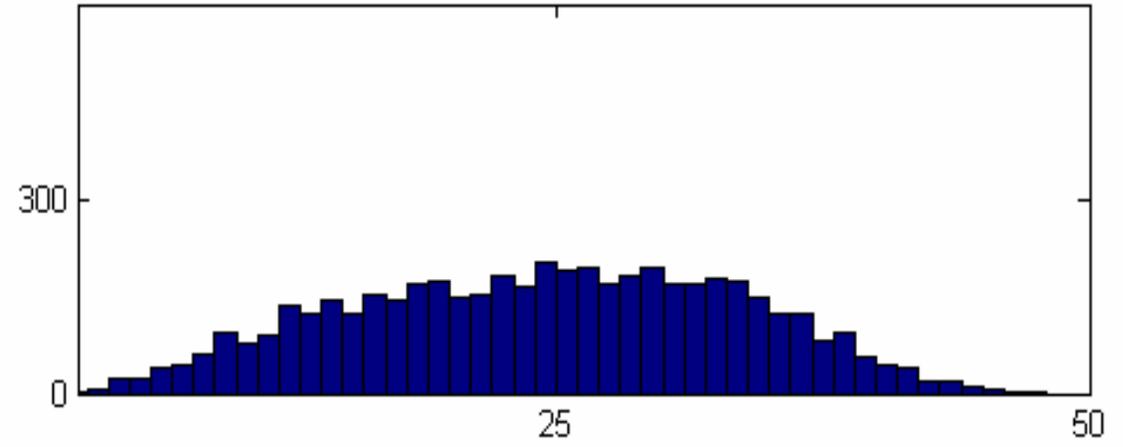


3. Reorder the columns of H such that all '1's are in black region

4. Repeat the algorithm with a lower threshold until the algorithm stalls

Column Reordering: Results

- Variable reordering leads to **37%** reduction in maximum wirelength for a 2048 bit LDPC code.
- No compromise in code performance
- No constraint on type of H matrix



Histogram of Manhattan distances without and with column reordering

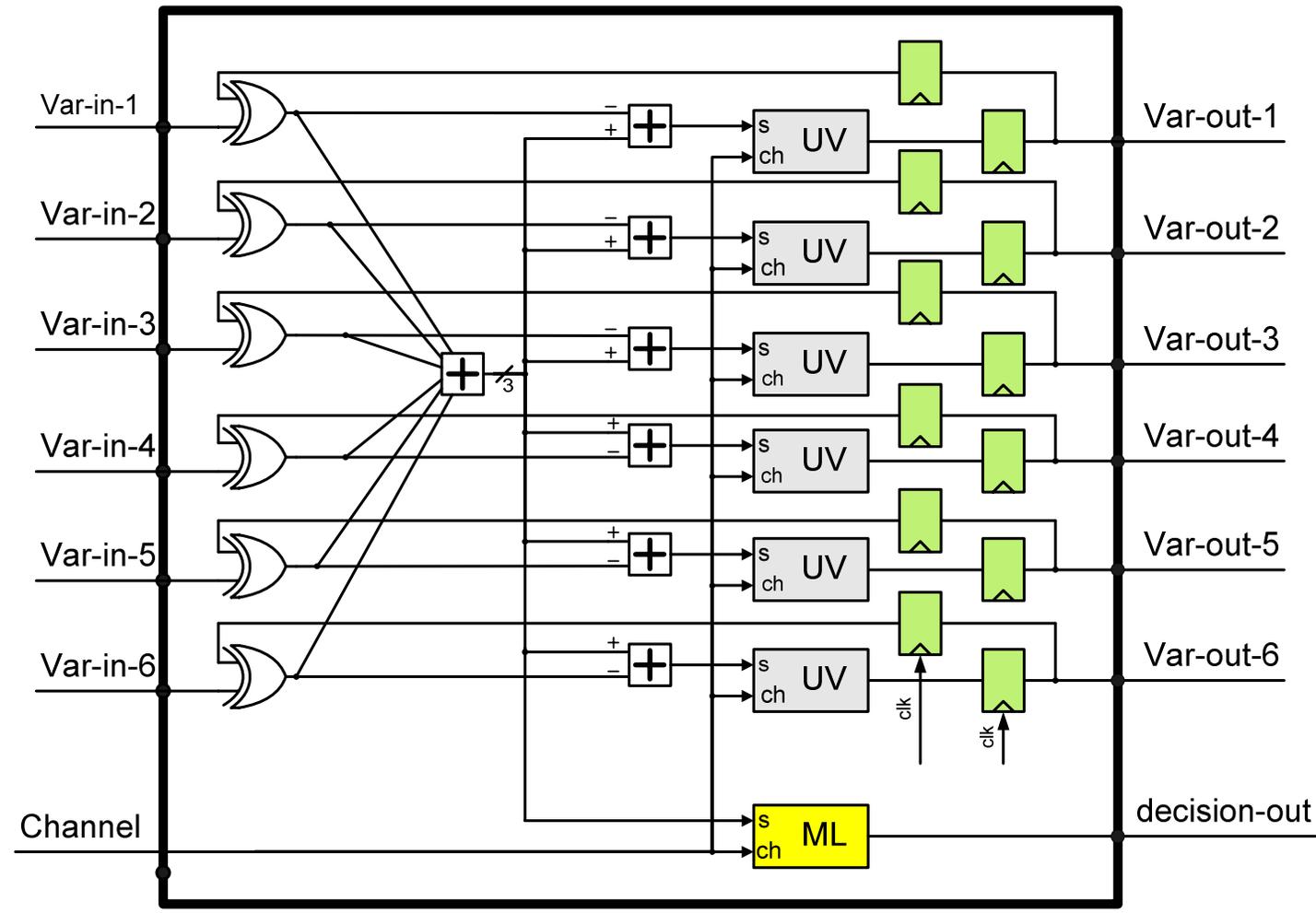
Outline

- Introduction to LDPC codes
- LDPC decoder hardware implementation
 - Implementation challenges
 - Half-broadcasting
 - Variable Reordering
- Decoder design
 - Variable node
 - Check node
 - Top level
- Conclusion

Implemented decoder

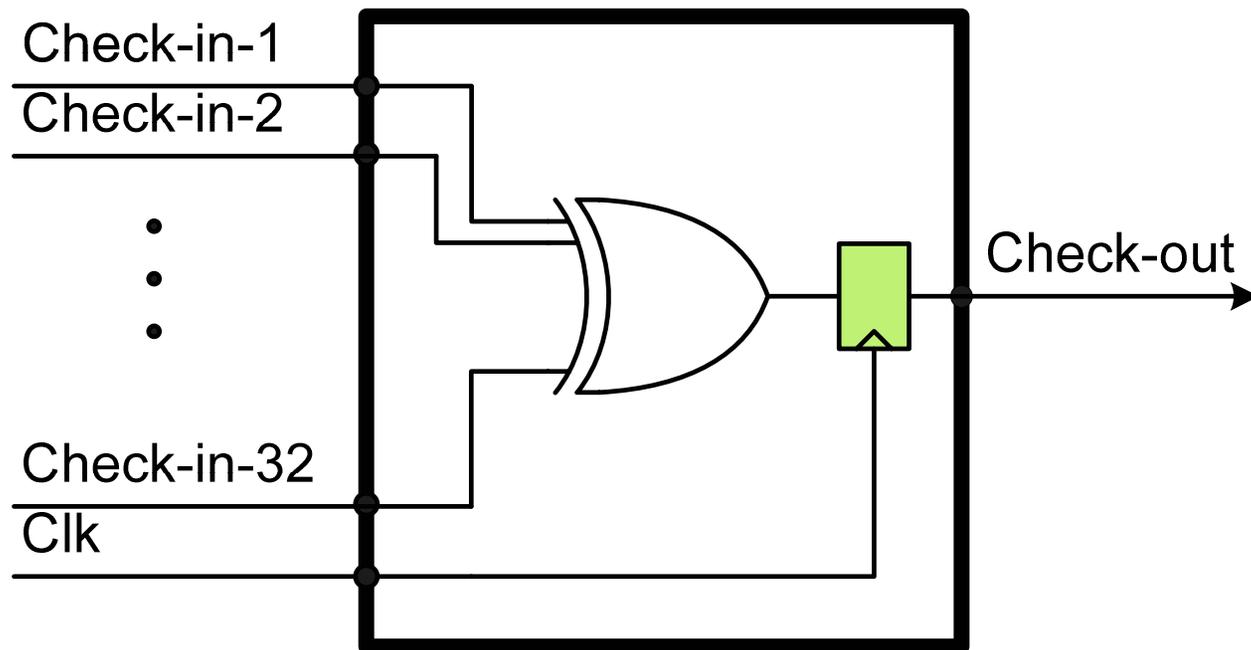
- (2048,1723) Gallager (6,32)-regular LDPC code
- RS-LDPC code is proposed for 10Gbase-T Ethernet standard
- Hard decision message passing algorithm
 - The two techniques are also immediately applicable for higher bit-precisions
- A bottom-up hierarchical design methodology is used
 - One variable and check node synthesized based on Half broadcasting technique
 - Column reordering is used to place the check and variable nodes

Variable Node

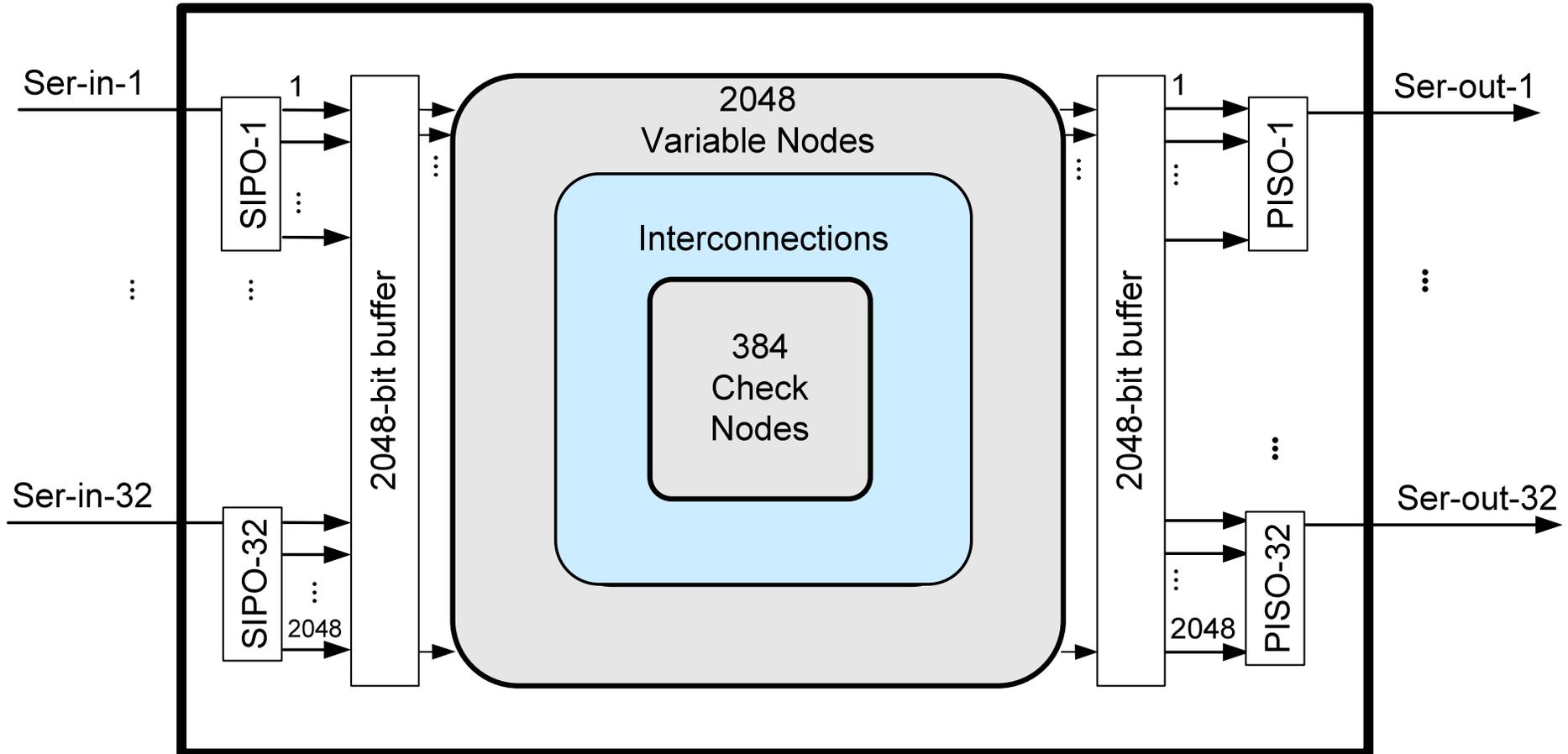


- *UV* output is the same as input *Ch* unless all other inputs oppose
- *ML* output is based on the majority of all inputs and the channel input

Check Node

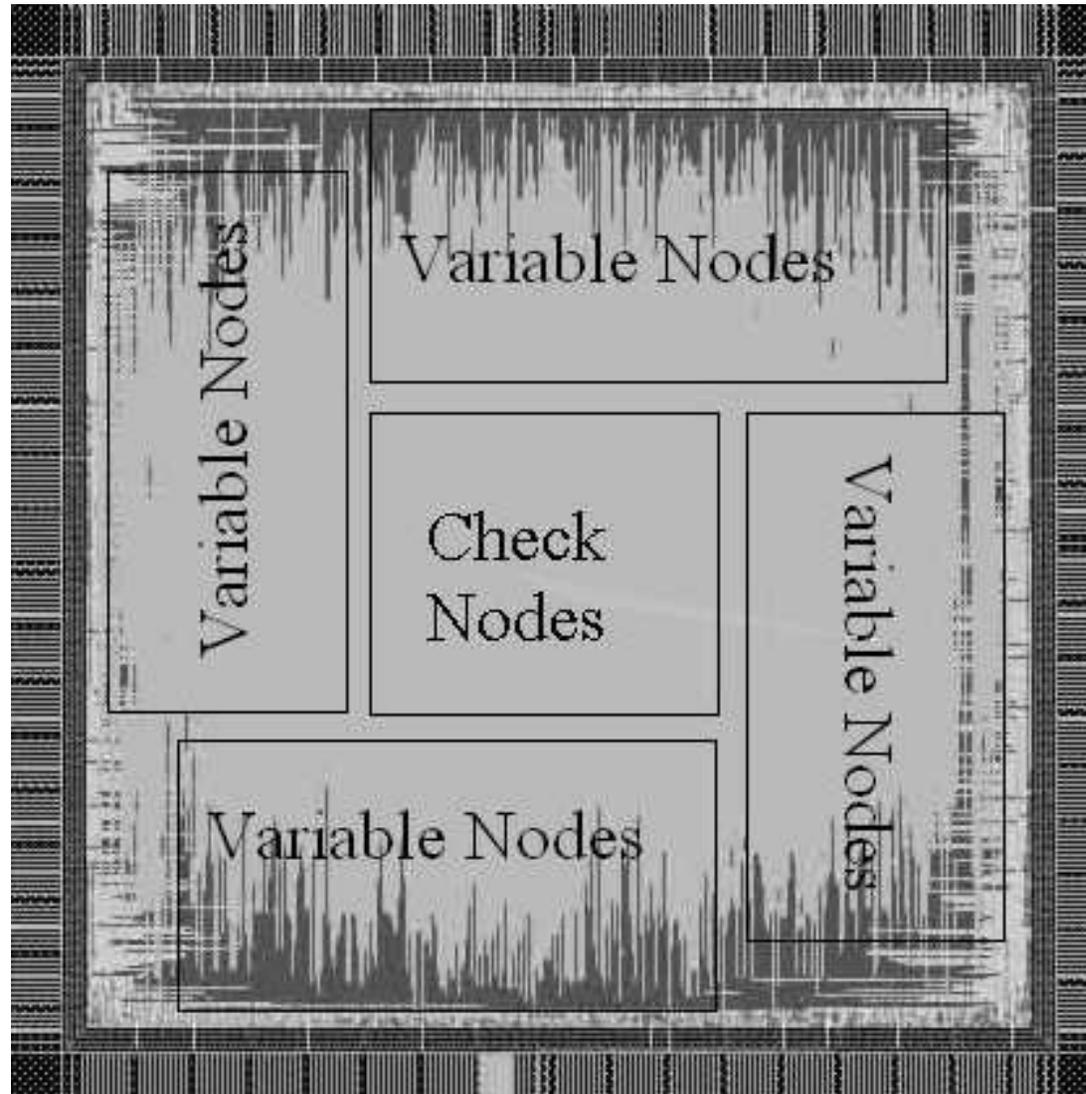


Top-level design



- 32 64-bit Serial-In-Parallel_Out shift registers at the input
- 32 64-bit Paralle-In-Serial_Out shift registers at the output

Decoder layout



Simulation Results

	This work	[Blanksby02]
Process	CMOSP18-6M	CMOSP16-5M
Code type	RS-LDPC (6,32) regular	Irregular
Code rate	0.84	0.5
Code length	2048	1024
Decoding algorithm	Hard decision MP	4-bit MP
# of iterations/frame	32	64
Clock frequency	100MHz	64MHz
Throughput	3.2 Gbit/sec	1 Gbit/sec
Die size	17.2 mm ²	52.5 mm ²

Conclusion

- High throughput LDPC decoders are in demand for several applications
- Parallel LDPC decoders suffer from routing congestion and interconnect delay
- Half-broadcasting reduces the check-to-variable global wirelength by more than 40%, hence reducing congestion.
- Column reordering reduces the maximum Manhattan distance between two neighboring nodes by 37%, hence reducing interconnect delay

References

- A. J. Blanksby and C. J. Howland, “690 mW 1Gb/sec 1024-b rate-1/2 low density parity check decoder”, IEEE journal of Solid State Circuits, vol. 37, no. 3, March 2002.
- P. Urard, et al., “A 135 Mbps DVB-S2 compliant Codec based on 64,800b LDPC and BCH codes”, International Solid-State Circuits Conference, San Francisco, CA, Feb 2005.
- M. M. Mansour, N. R. Shanbhag, “High-throughput LDPC decoders”, Transactions on VLSI systems, vol. 11, no. 6, Dec 2003.