Power Optimized 5G NR Low Density Parity Check Decoder for Satellite Communications

UCLA Samueli School of Engineering
SUMMER UNDERGRADUATE RESEARCH PROGRAM

Jessie Chan, Brandon Kam, Arshnoor Dhaliwal, Jingchao Luo, Dr. Richard Wesel

Electrical and Computer Engineering Department | University of California, Los Angeles

Introduction and Background

Satellites need **reliable** and **fast** communications to meet requirements set forth by the Space Development Agency (SDA) standard [1]

Why LDPC

- **Problem:** Noise over the communication channel can corrupt transmitted data
- **Low Density Parity Check (LDPC)** is an error correcting code applied to the transmitted data in order to recover noisy messages using a decoder

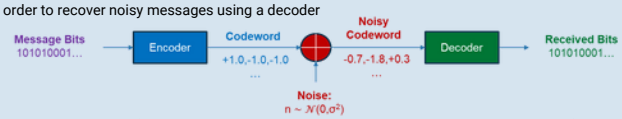


Fig 1. Diagram of typical communication link with noise and an LDPC error correcting code. The encoder transforms the message into a **codeword** that will be used by the decoder to recover the original message.

LDPC Codes

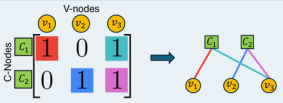


Fig 2. Graphical representation of parity check matrix. **V-nodes** are matrix columns while **C-nodes** are matrix rows. A 1 represents a connection between nodes.

- **Parity Check**: Method where additional bits called **parity bits** are added to determine integrity of original message
- Specific parity checks are defined by a **parity check matrix**
- Parity check matrix can be represented by a graph
- **V-nodes** represent codeword bits and **C-nodes** represent parity checks

- **Message passing algorithm**: Decoder works by passing messages back and forth in between **C-nodes** and **V-nodes**
- C-nodes typically uses the **Min-Sum algorithm** to process information from V-nodes [2]

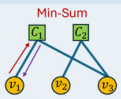


Fig 3. Decoding algorithm with c-nodes and v-nodes

FPGA

Fig 4. The ZCU106 FPGA board used for the experiment

- LDPC decoders are typically implemented on **customizable chips made up of logic tables** called **Field-Programmable Gate Array (FPGA)**
- More flexible than digital circuits
- Runs faster than general purpose processors

Current Limitations

- The SDA standard uses the **5G NR parity check matrix**
- **The 5G NR Parity Check Matrix** is divided into **384x384 circulants** (shifted identity matrices) [3]
- Circulant property allows decoders to perform **384 parallel computations**
- However, this over parallelization **occupies too much chip area** and **large chip area consumes too much power**

$$\begin{matrix}1&0&0\\0&1&0\\0&0&1\end{matrix} \rightarrow \begin{matrix}0&0&1\\1&0&0\\0&1&0\end{matrix}$$

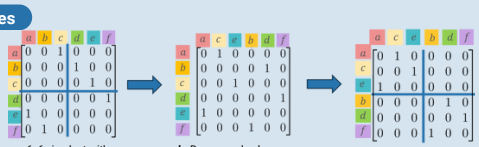
Fig 5. An example of a circulant

Objective

Design a smaller and more power efficient LDPC decoder for the 5GNR standard on an FPGA that maintains the target data rate of 2.05 Gigabits per second (Gbps)

Materials and Methods

Design Changes



a. 6x6 circulant with shift 2 **b.** Rearranged columns **c.** Rearranged rows. Forms a 2x2 of 3x3 subcirculants

Fig 6. An example of splitting a 6x6 circulant into smaller 3x3 sub-circulants

Circulant Splitting

- Split 384x384 circulants into 32x32 sub-circulants to **reduce parallelization and chip area**
- Smaller chip area **reduces power consumption** at the **cost of lower data rate**
- **Fig 6a.** shows that splitting a circulant into smaller matrices doesn't result in sub-circulants
- **Fig 6b.** and **Fig 6c.** show how columns and rows are rearranged to create sub-circulants
- Reordered columns means **decoder must reorder codeword bits**

Other Optimizations

- **Pipelining** splits the decoding process into stages, increases throughput
- **Min-Star** algorithm is used instead of the Min-Sum algorithm, which gives better decoding performance at the cost of a slight increase in computational complexity

Experimental Setup

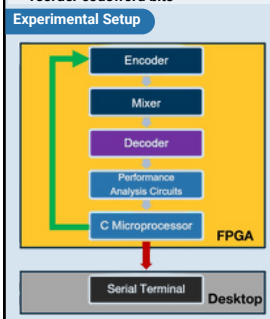


Fig 7. Flow diagram of experimental setup

1) 4800 codewords are randomly generated from the **encoder** and sent to the **mixer**

2) **Mixer** adds noise to codeword and produces probabilistic representations for each bit

3) **Decoder** takes in **noisy codeword** and attempts to output correct message bits

4) **Performance Analysis Circuits** keep track of bit and codeword errors based on output of decoder

5) **C Microprocessor** collects bit and frame error data and repeats steps 1-5 until 75 codeword errors are detected

6) Data collected by the **C Microprocessor** is displayed via **Serial Terminal** on a desktop

Parameters

Changed

- **COMP bits:** Number of bits used to represent V-nodes
- **MSG bits:** Number of bits used for Min-Star
- **Max iterations:** Max times messages are sent in decoding
- **Code rate:** Ratio of number of message bits to total number of bits sent
- **SNR(Signal to Noise ratio):** How much noise is added to the signal

Measured

- **Frame error rate (FER):** How often a codeword is incorrect. Measures **decoding performance**
- **Bit error rate (BER):** How often a bit is incorrect. Measures **decoding performance**
- **Data rate:** How fast the decoder outputs bits. Measures **speed**
- **Cores:** Number of decoders that can fit on a single FPGA board. Measures **chip area**

Results and Discussion

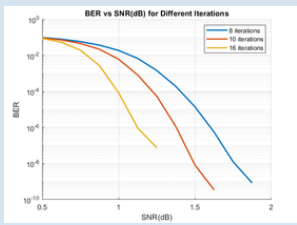


Fig 8a. Performance as a function of max iterations, keeping COMP = 10, MSG = 8, and code rate = 0.5. The best performing curve is 16 max iterations.

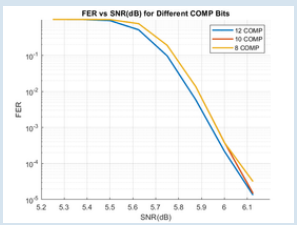


Fig 8b. Performance as a function of COMP bits, with max iterations = 8, MSG = 7, and code rate = 0.8462. The best performing curve is 12 COMP bits.

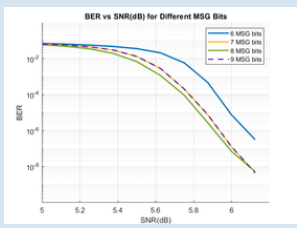


Fig 8c. Performance as a function of number of MSG bits, with max iterations = 8, COMP = 10, and code rate = 0.8462. The best performing curve is 8 MSG bits.

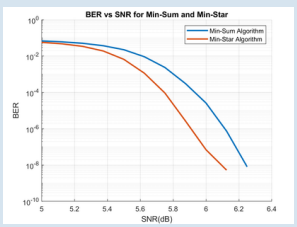


Fig 8d. Decoding performance of Min-Sum vs Min-Star, with max iterations = 8, COMP = 10, MSG = 8, and code rate = 0.8462. The best performing curve is Min-Star.

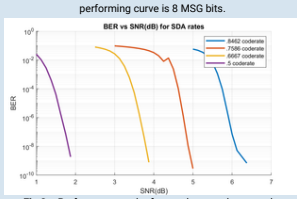


Fig 8e. Performance at the four code rates that must be supported by the satellite with parameters set at COMP = 10, MSG = 8, and max iterations = 8.

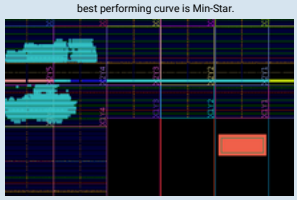


Fig 9. Space occupied by a decoder with COMP = 10, MSG = 8, and one core on an FPGA board. The light blue indicates space occupied by the decoder.

- Increasing iterations improves **decoding performance**, but decreases **speed**
- Increasing COMP and MSG bits improves **decoding performance** with diminishing returns at the cost of **chip area**
- The parameters with the best tradeoff of **decoding performance**, **speed**, and **chip area** is **8 max iterations, 10 COMP bits, and 8 MSG bits**
- A single FPGA board can fit **17 cores**
- The optimized decoder runs at the target data rate of **2.05 Gbps** using the optimized parameters
- Min-Star **improves decoding performance by 0.2 dB of SNR** over Min-Sum

Conclusion

- Splitting the parity matrix into smaller parts allows the decoder to **occupy less space on the FPGA.** The decreased chip area **lowers the decoder's power consumption at the cost of lower data rate per core.** However, we can place more cores on a single FPGA to recover the data rate
- Pipelining increases data rate of optimized decoder
- Using Min-Star over Min-Sum **improves decoding performance by 0.2 dB of SNR**
- The optimal parameters for the decoder are **10 COMP, 8 MSG, and 8 max iterations**. With these parameters, a single FPGA board can fit **17 cores** to achieve **the required data rate of 2.05 Gbps**
- With our decoder, satellites can meet the data rate and reliability requirements of the SDA standard

Future Work

- **Modify stopping condition** of decoding process by either **checking fewer parity bits** or using a **cyclic redundancy check (CRC)**, an error detection method, on the message bits
- Further reduce memory and FPGA resource utilization to fit more decoders on FPGA
- Test data rate and **record power consumption** of optimized decoder on satellite

References

[1] Space Development Agency, "Optical Communications Terminal (OCT) Standard Version 3.1.0", 2023, pp.53
[2] AMD, "LDPC Encoder Decoder v1.0 Product Brief (PB052)," 2022, pp. 3
[3] 3GPP, "5G; NR; Multiplexing and channel coding (3GPP TS 38.212 version 15.2.0 Release 15)," 2018, pp. 16-18

Acknowledgements

We would like to thank our PI Dr. Richard Wesel for his continued support and guidance throughout the program. This research is supported by CACI and National Science Foundation (NSF) grants CCF-1955660 and CCF-2008918. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect views of NSF nor CACI.