LDPC Codes for Flash Channel

Xinde Hu
xhu@stec-inc.com
Outline

• LDPC codes for NAND Flash – Introduction
• LDPC-based flash channel for enterprise SSDs – Challenges and Solutions
• Flexible and efficient implementation of LDPC-code for Flash channel
• Conclusions
Introduction to Low-Density Parity Check Codes

• Near capacity error correcting performance
• Iterative BP decoding algorithm requires soft information (LLRs) to reach maximum error correcting capability

Challenges

• Performance cannot be characterized theoretically (large simulation required)
• Code design to avoid error floors
• Low complexity implementation of LDPC decoder
LLRs from NAND Flash

• Each cell is binned by applying multiple read threshold levels
• Different reliability values (LLRs) are assigned to cells in different bins

Key challenges
• Optimal read level settings change as flash memory ages
• Negative read levels
LDPC Codes for Enterprise Class SSDs

• Enterprise class SSDs
  • Longer endurance requirements
  • Low power consumption per IOPS
  • High IOPS/throughput
  • More stringent data reliability requirements

• LDPC code suitable for enterprise-grade SSDs
  • Minimize code overhead
  • LDPC code design to avoid error floors
  • Efficient LDPC encoder/decoder implementation
  • Intensive emulation/validation processes
- Phenomenon associated with iterative decoding-based ECC – Turbo code, LDPC code
- Failure rate steadily decreases as the signal condition (SNR) improves
- After a certain point, the failure rate does not fall as quickly – reaching a “floor”
• **The error floor problem**
  - Certain sub-structure in the code called “trapping set” is the root cause
  - Trapping set will lead to error floor when iterative decoding is used
  - Under iterative decoding, trapping sets could stuck in a “trap” with a high probability even when the signal-to-noise ratio is high

• The error floor is also significantly affected by codeword length, column weight, and decoder precisions
  - Shorter codes have higher error floors
  - Low precision decoders have higher error floors
Error Floor of LDPC Codes – the solution

• Set key parameters
  • Longer LDPC codewords
  • High precision decoders

• Design LDPC code to avoid high error floors

• Trapping set based H matrix design
  • Find key trapping set structures within the QC-LDPC framework (coderate, column weight, etc.)
  • Derive mathematical formula representing the key trapping set structures
  • Use progressive design process when designing H matrix, examining the formula in every step
  • This way, the key trapping sets are minimized in the code
• Long simulation is required to verify that the error floor is below requirements.
Implementation of LDPC Based Flash Channel

• Efficient decoder design
  • Quasi-cyclic type of code reduces decoding complexity without losing error correcting performance
  • This also enables highly parallelized decoding processes to boost decoder throughput and reduce data latency

• Different code rates are required
  • Different Flash memory designs require different coderates
    • Newer generation of flash memory requires lower coderate to hit the UBER limit
  • Different SSD endurance requirements translate to different coderate requirements
    • Optimal tradeoff between SSD overheads and ECC performance
Implementation of LDPC Based Flash Channel - Variable Coderate Solution

- **Zero-padding solution – Good and bad**
  - No additional encoder and decoder hardware needed
  - If the number of zeros padded is too large, the error correcting performance will suffer
  - Throughput (in terms of host payload) will decrease when zero-padding is applied

- **The verdict**
  - Zero-padding alone is not an optimal variable rate solution
Implementation of LDPC Based Flash Channel
- Variable Coderate Solution

• Code rate range from 0.75 ~0.95
• High rate LDPC codes to support SLC-based SSDs
• Low rate LDPC codes to support TLC flash types or SSDs with high endurance requirements
• Multi-rate LDPC decoder with minimal added hardware cost
• Overall throughput variation between different codes is less than 10%
Conclusion

• LDPC code is THE answer for future flash channels
• LDPC codes need soft-information from the flash
  • This can be obtained by multiple reads
• Design LDPC code to avoid high error floors
  • Trapping set based approach
  • Long simulation to confirm that the codes meet the requirements
• Make LDPC code fit in the SSD big picture
  • Low complexity design
  • Variable coderate solution for different flash memory chips/SSDs
Thank You

xhu@stec-inc.com