Using Rate-Adaptive LDPC Codes to Maximize the Capacity of SSDs

Stephen Bates, PhD
Technical Director
PMC-Sierra Inc.

stephen.bates@pmcs.com

Outline

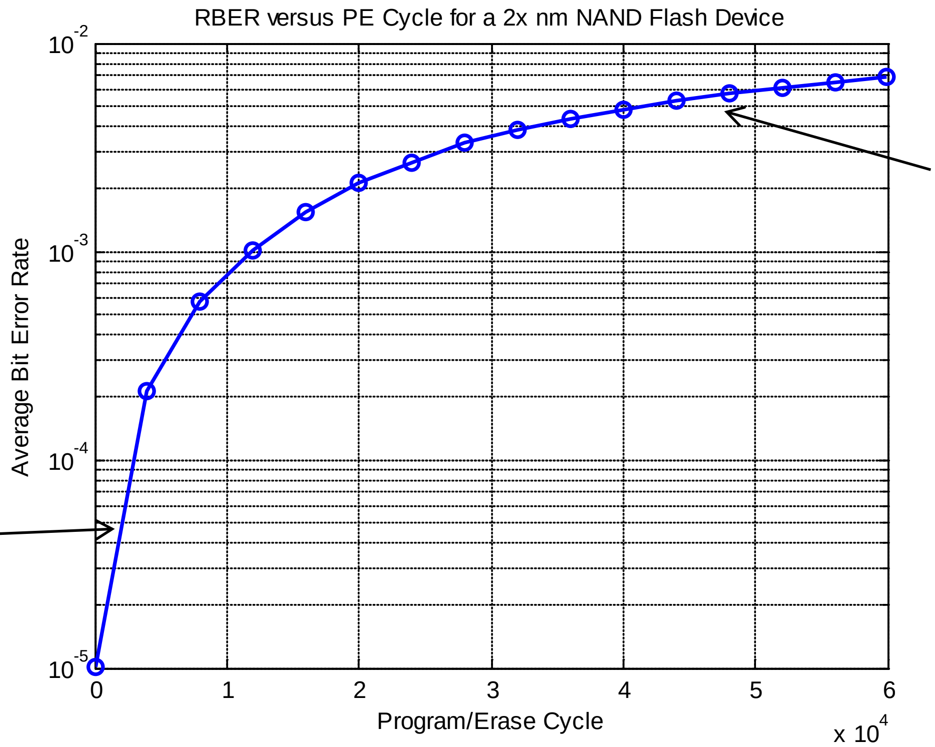
- The vagaries of NAND flash
- Rate-Adaptive LDPC codes for SSDs
  - Reduced cost per user GB
  - Reduced cost per PetaByte written
  - Consistent low latency reads
  - Variable random write performance
- Conclusions

The vagaries of NAND flash

- As we all know, the Raw BER of a flash page increases as the block it resides in undergoes PE cycles
- In addition, at any given PE cycles, there is a spread of RBER across all the pages on the drive at that PE cycle
- Both these facts imply rate-adaptive codes make sense

Map the ECC overhead to the RBER of the page

The vagaries of NAND flash

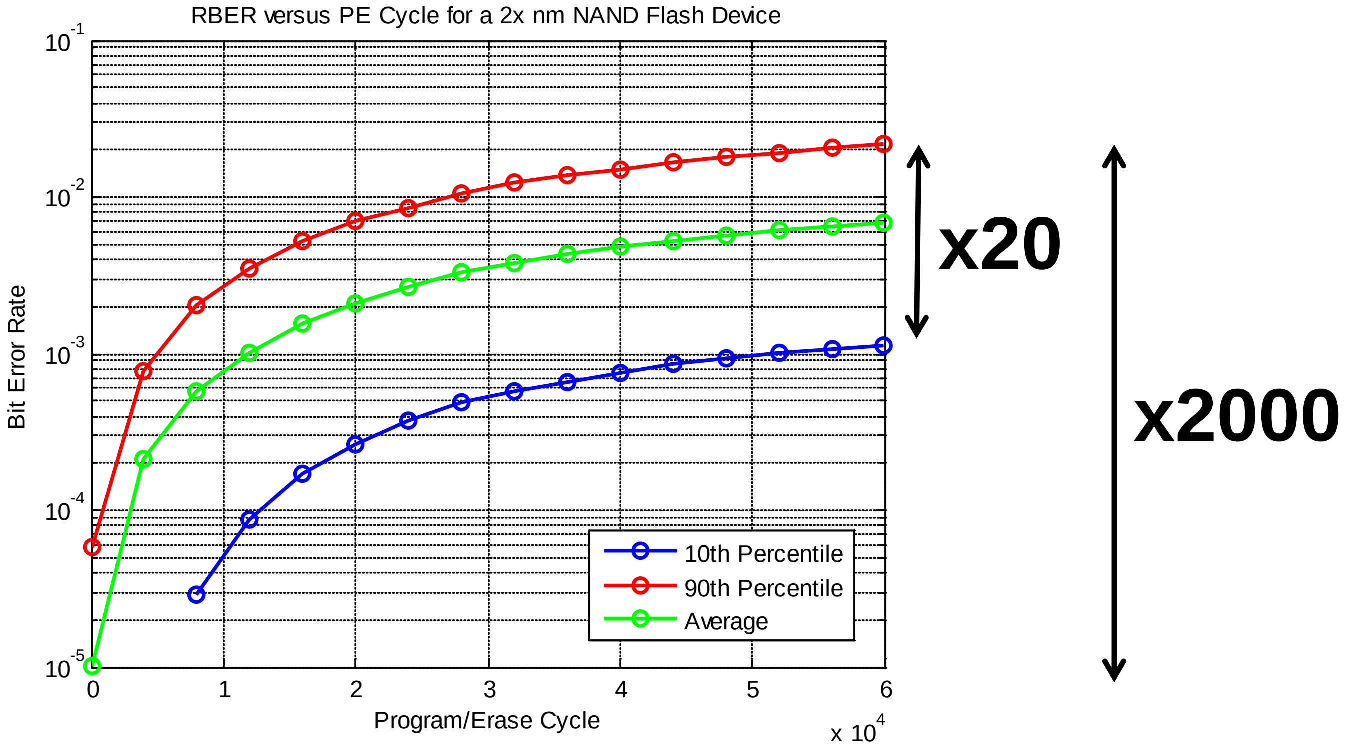


Onerous ECC requirements here.

Minimal ECC requirements here.

RBER of a page increases with PE cycle
Using the same ECC throughout is very inefficient!

The vagaries of NAND flash



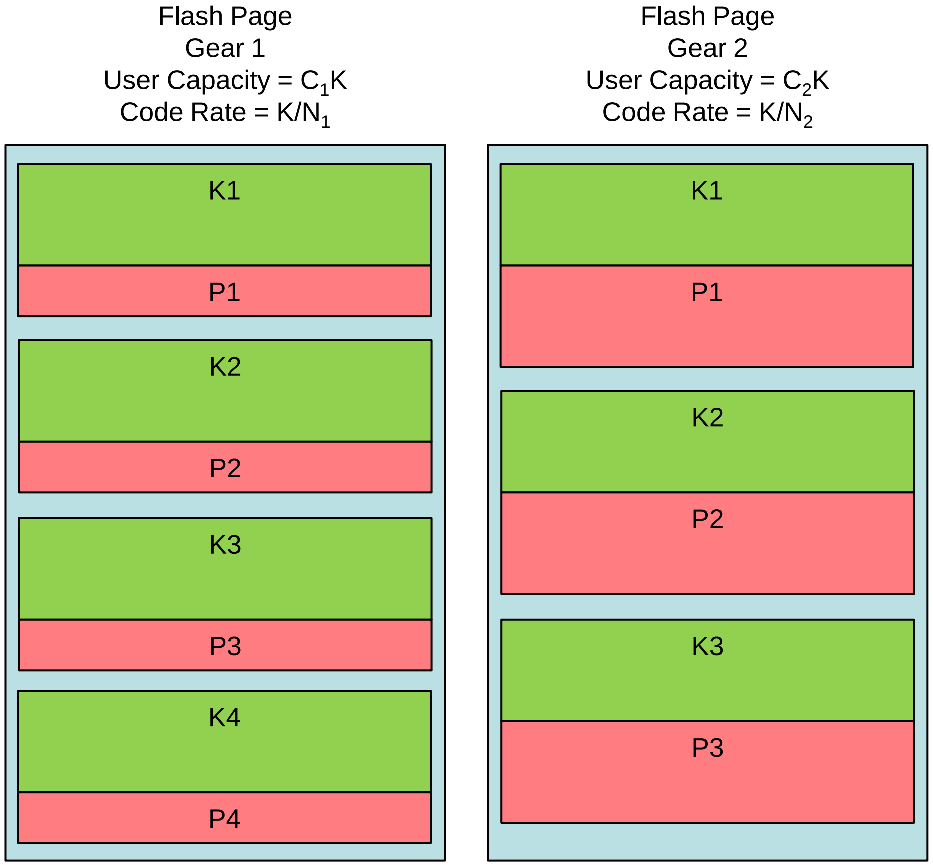
RBER of pages as same PE cycle very different
Using the same ECC on all pages is very inefficient!

The vagaries of NAND flash

- A 10 Random Fills Per Day SSD, constructed with 28% OP needs to achieve about 45000 PE cycles
  - At End Of Life the 90th percentile RBER is about 0.018
  - At End Of Life the 10th percentile RBER is about 0.001
  - At Beginning of Life the 90th percentile is about 0.0005
- Using the same ECC for all these cases sacrifices capacity
- When RBER is low use a high rate code
- When RBER is high use a lower rate code

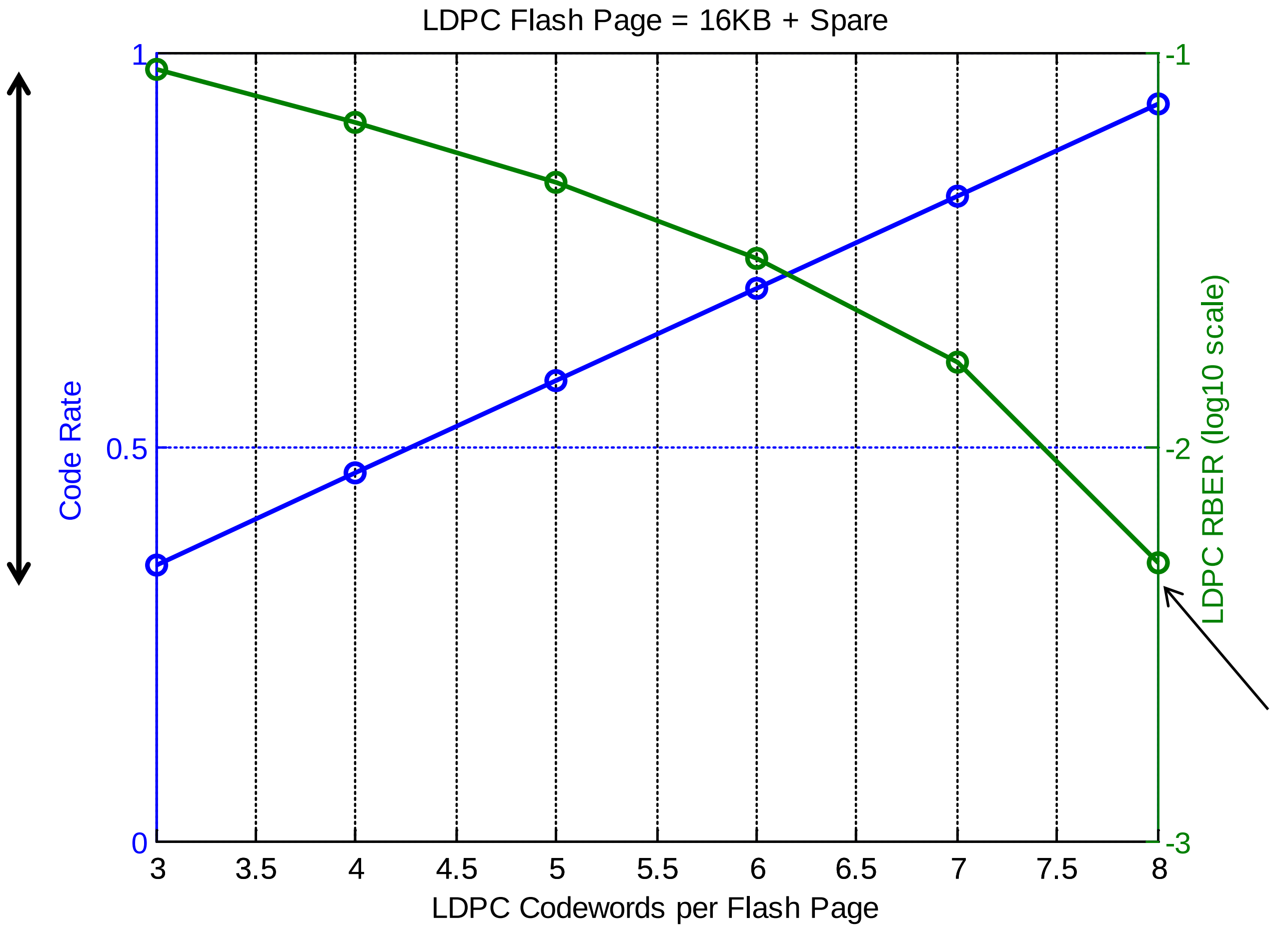
Rate-Adaptive LDPC codes

- Quasi-Cyclic LDPC codes are very amenable to supporting different rates
- In this example we keep the payload (K) the same and vary the codeword size (N) and codewords per Flash page (C)
- In reality more codewords make sense to allow for finer granularity of R
- You can also straddle codewords across physical pages though this complicates things



ECC Performance and User Capacity

Wide range of RBER can be accommodated.



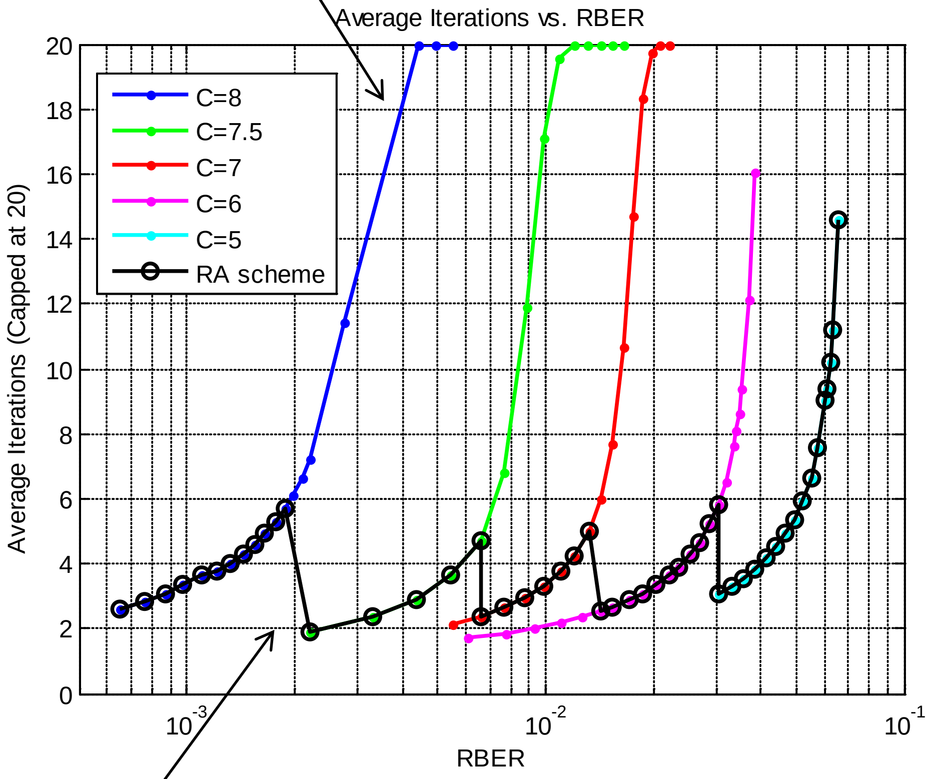
LDPC codeword size is ~2KB which is quite reasonable for LDPC.

Shorter LDPC codewords struggle at rates > 0.8.

Decode Latency

- For LDPC decoders decode latency is typically small until the waterfall is reached, then quickly rises
- For non-RA schemes this implies latency on old or “bad” pages can be high
- For RA schemes the latency can be kept small across more pages for longer

Non-RA solutions are obliged to follow this curve. Latency rises with RBER

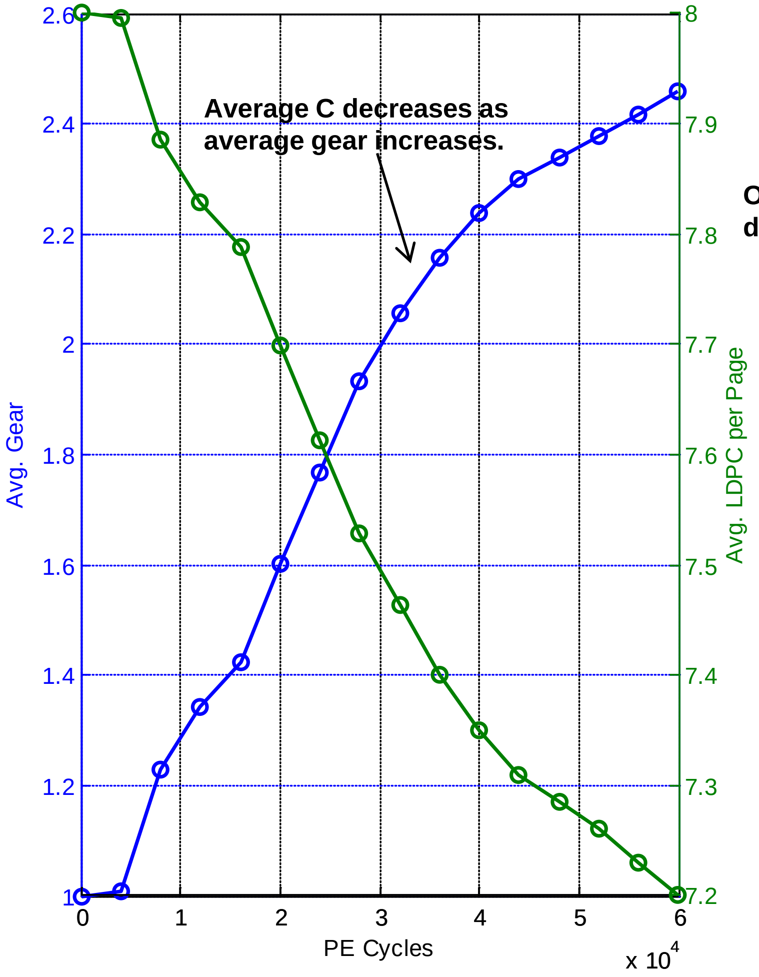


RA schemes can be configured to follow the black curve ensuring latency stays bounded for longer.

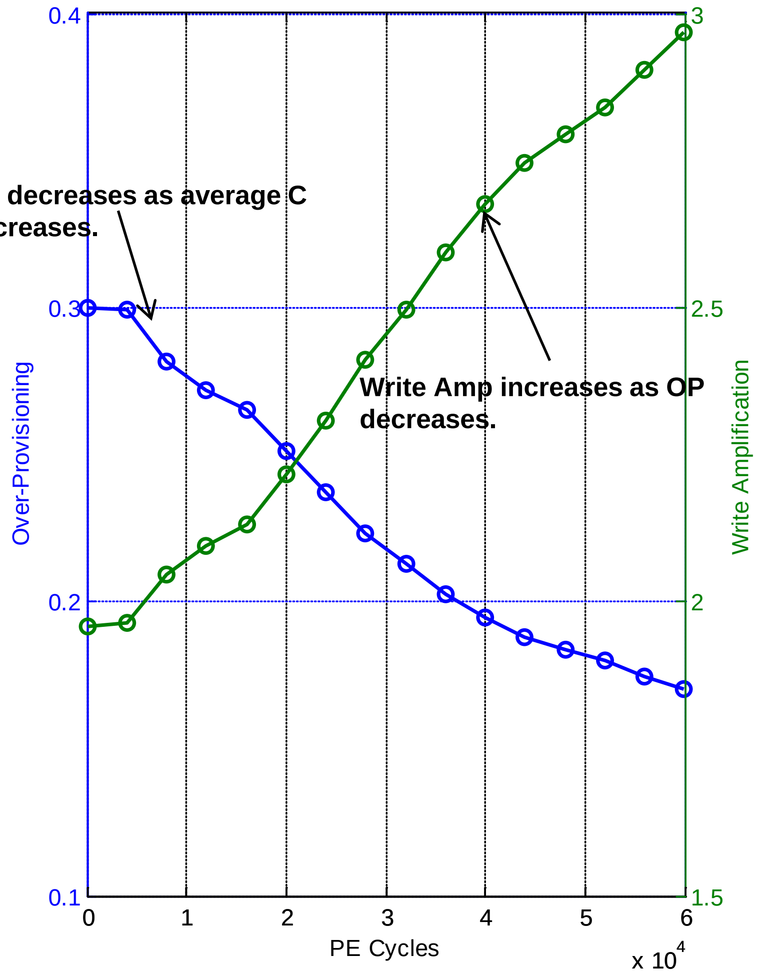
Random Write Performance

- As the drive ages, pages change gear and the capacity of the drive diminishes
- This drop in capacity eats into the over-provisioning (OP) and, as predicted by the Lambert equation, causes the write amplification to increase
- This effect can be hidden from the host if required
- Only an issue for random writes. Sequential write performance is not affected

Random Write Performance



Average C decreases as average gear increases.



OP decreases as average C decreases.

Write Amp increases as OP decreases.

Conclusions

- In NAND flash:
  - Page RBER increases with PE cycle
  - RBER varies among pages at the same PE cycle
  - The RBER spread across age and pages is several orders of magnitude (1e-4 to 1e-2)
- QC LDPC codes are very amenable to RA implementation
- Rate-Adaptive LDPC codes match the RBER to the ECC parity and enable:
  - Improved drive capacity
  - More consistent low read latency