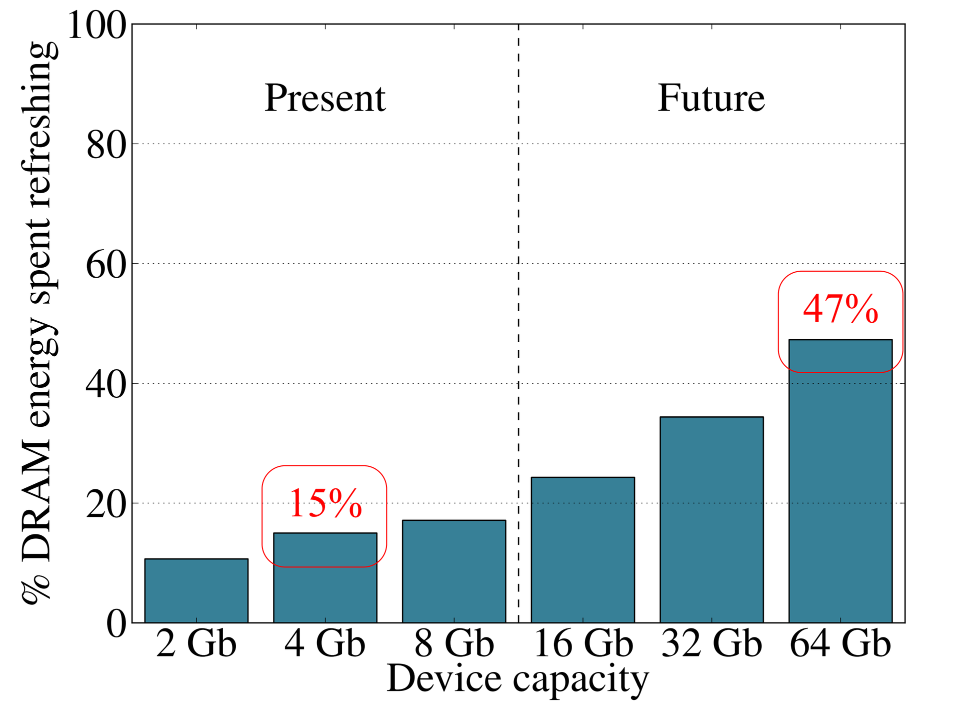
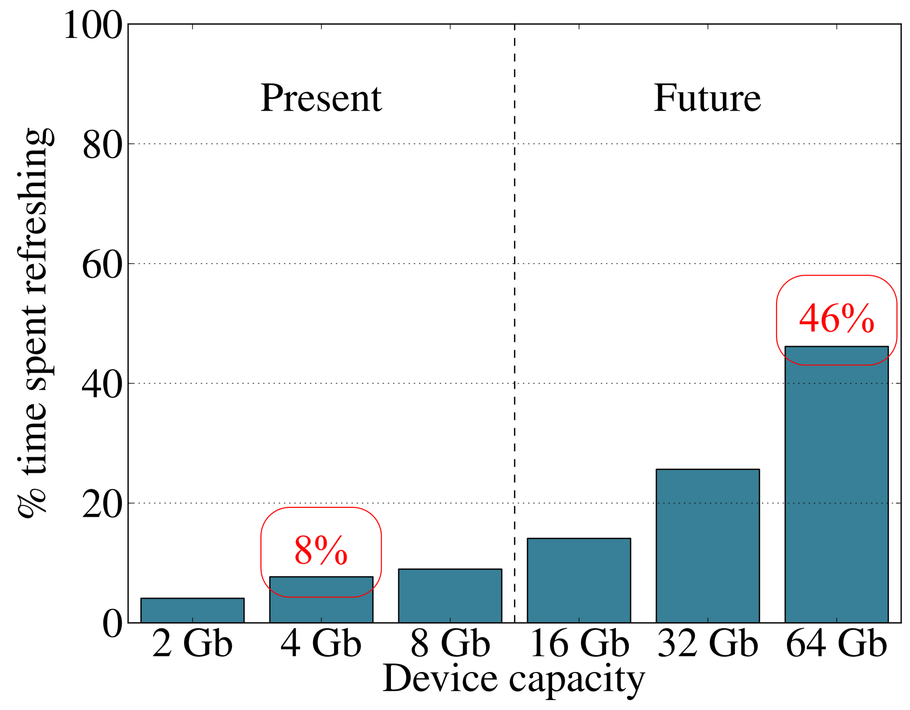
RAIDR: Retention-Aware Intelligent DRAM Refresh

Jamie Liu, Ben Jaiyen, Richard Veras, Onur Mutlu (Carnegie Mellon University)

Problem

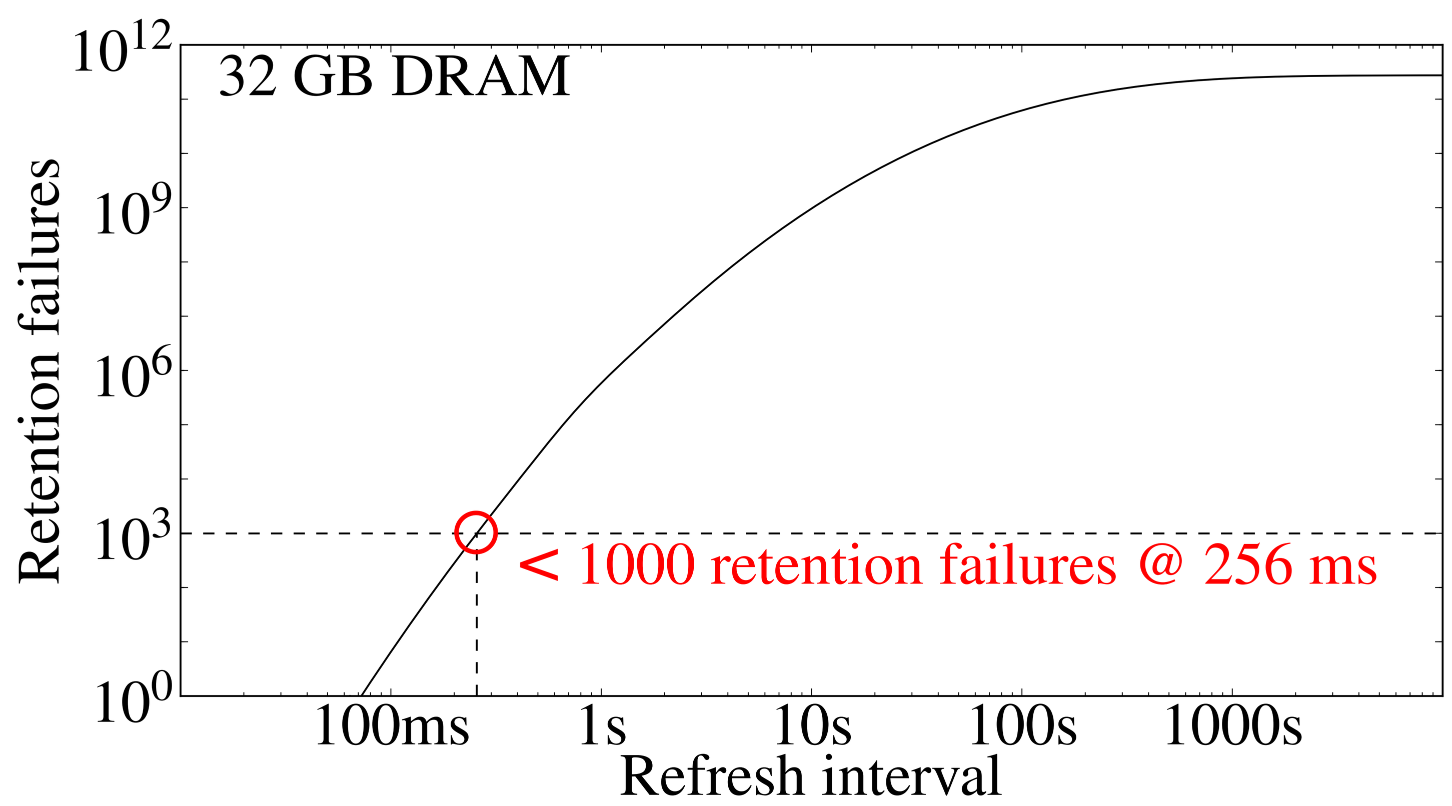
- DRAM requires periodic refresh to avoid data loss
- Refreshes interfere with memory accesses and waste energy
- Refresh overhead limits DRAM scaling





Key Observation and Idea

- Most DRAM cells can be refreshed infrequently without losing data
- All cells are refreshed at the same worst-case rate
- High refresh rate imposed by few weak cells
- Key idea: Refresh rows containing weak cells more frequently; refresh other rows less frequently

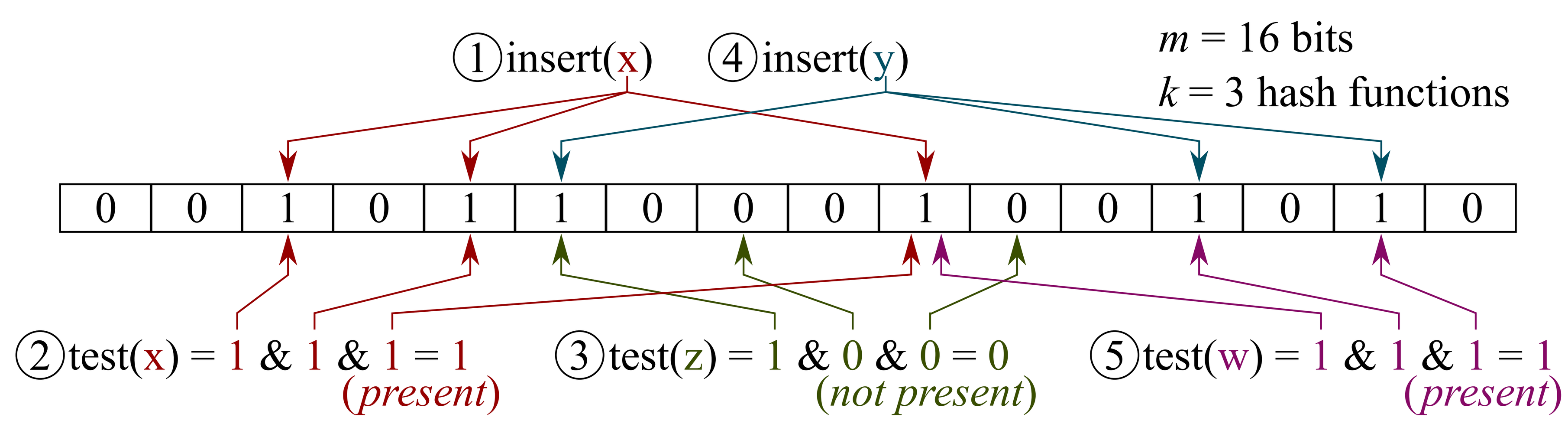


Our Mechanism: RAIDR

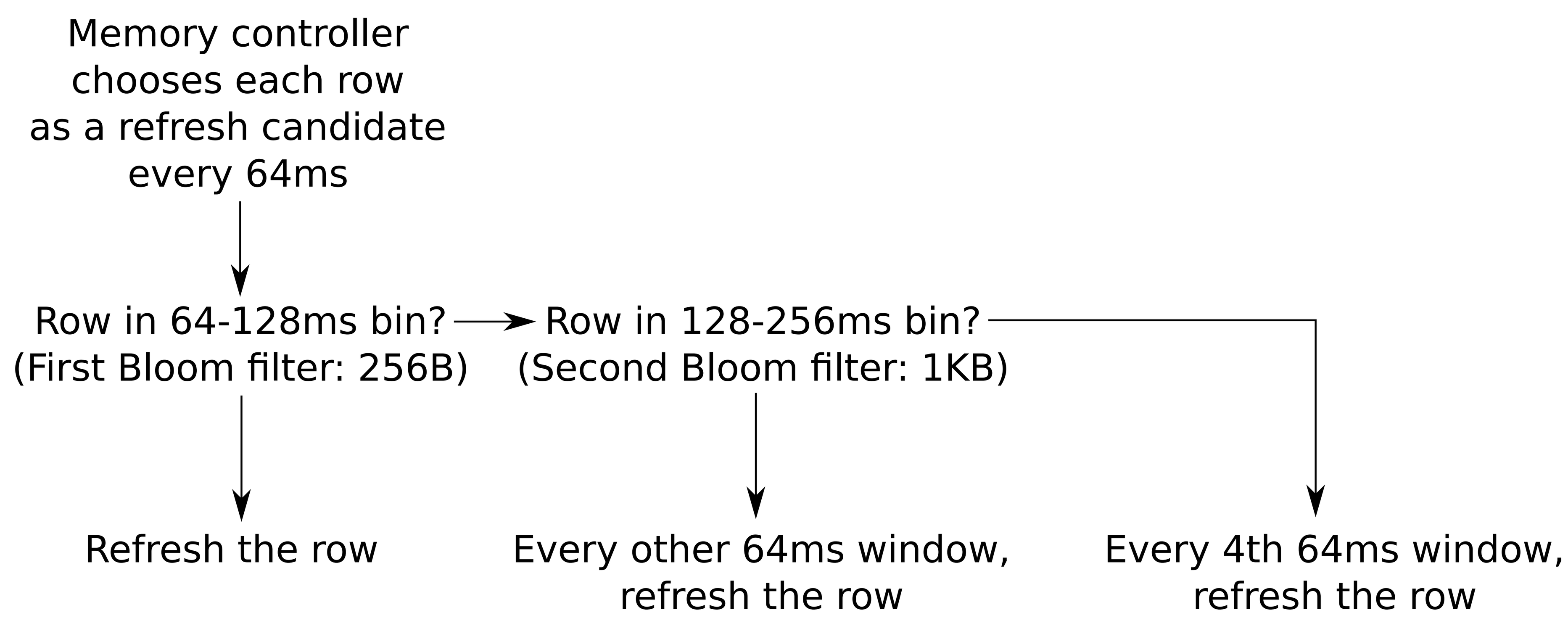
1. Profile retention times (how frequently each row needs to be refreshed to avoid losing data)

	Row 1	Row 2	Row 3
Initially	11111111...	11111111...	11111111...
After 64 ms	11111111...	11111111...	11111111...
After 128 ms	11011111... (64–128ms)	11111111...	11111111...
After 256 ms		11111011... (128–256ms)	11111111... (>256ms)

2. Group rows into different bins based on their retention time using Bloom filters for scalability and efficiency

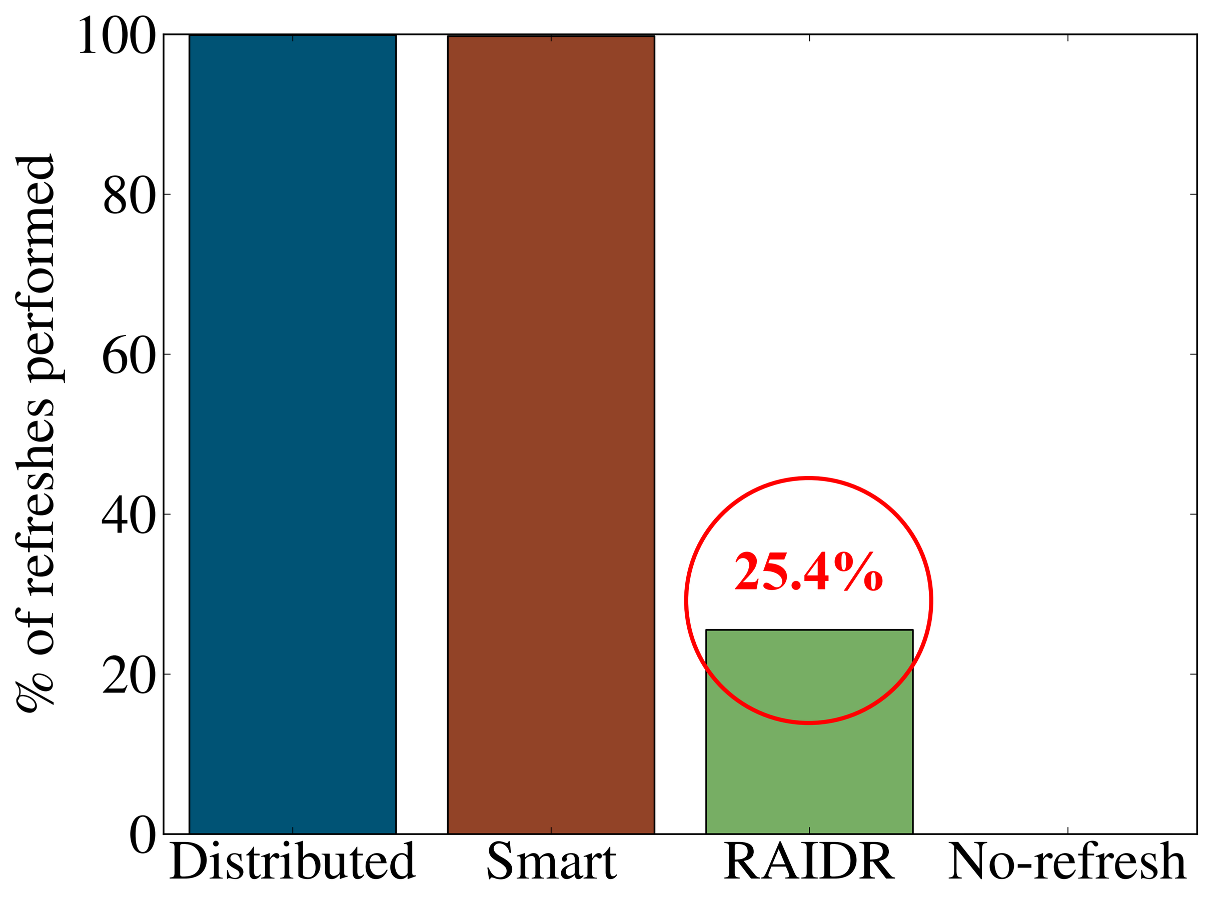


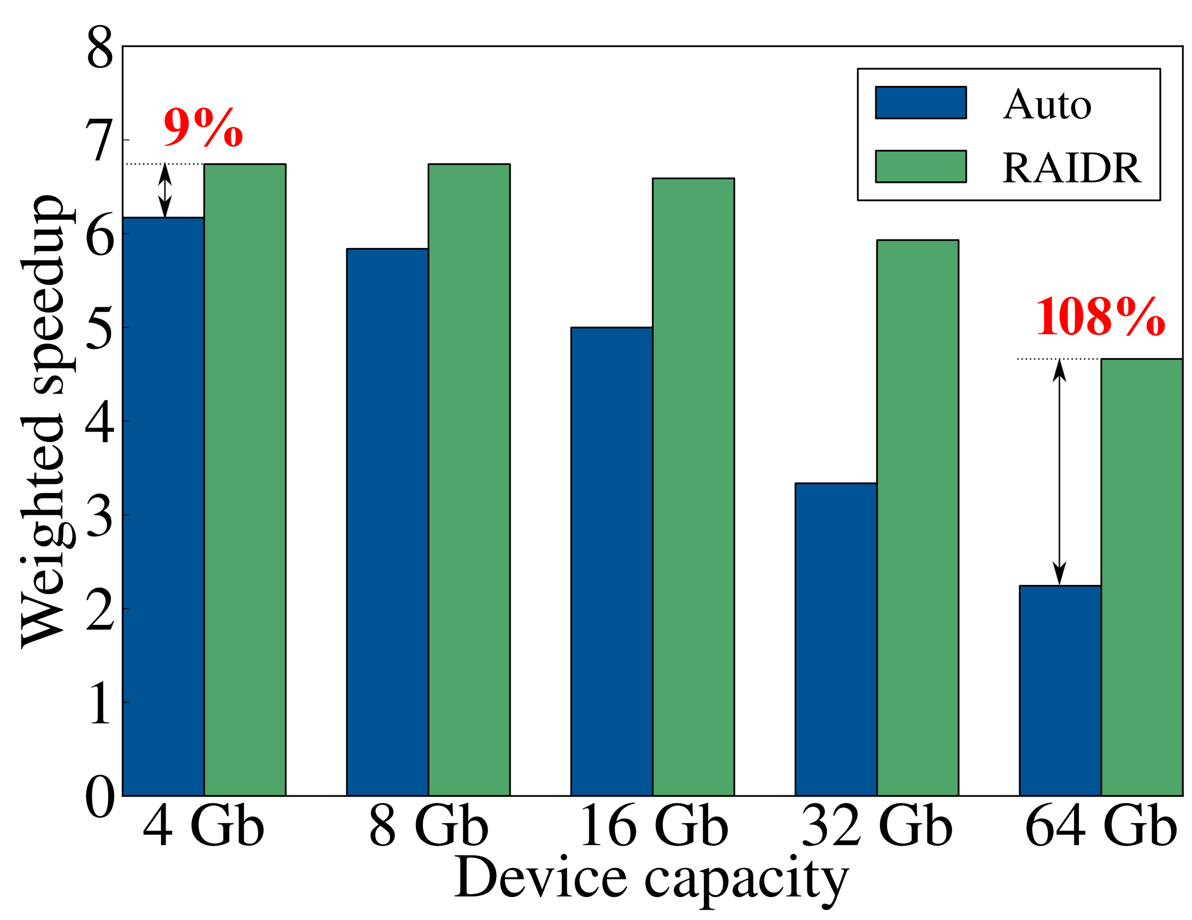
3. Refresh rows in different bins at different rates

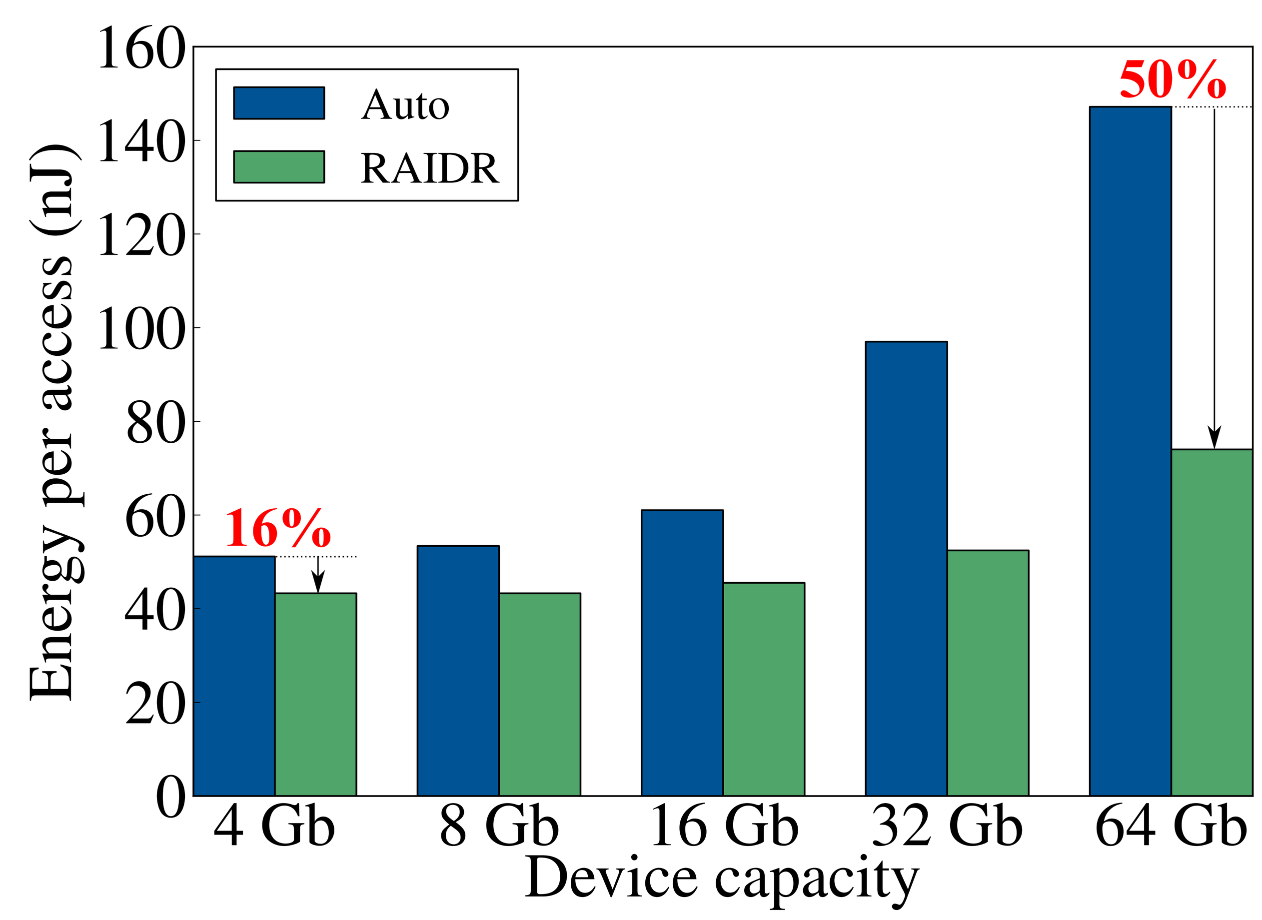


Results

- 1.5KB storage overhead in a 32GB DRAM system
- 74.6% refresh reduction resulting in 8.6% performance gain and 16.1% energy efficiency improvement on average
- Benefits increase as DRAM scales in density







Intel Science & Technology Center for Cloud Computing

Carnegie Mellon University · Georgia Tech · intel · Princeton University · UC Berkeley