Adaptive Granularity Memory Systems:
A Tradeoff between Storage Efficiency and Throughput

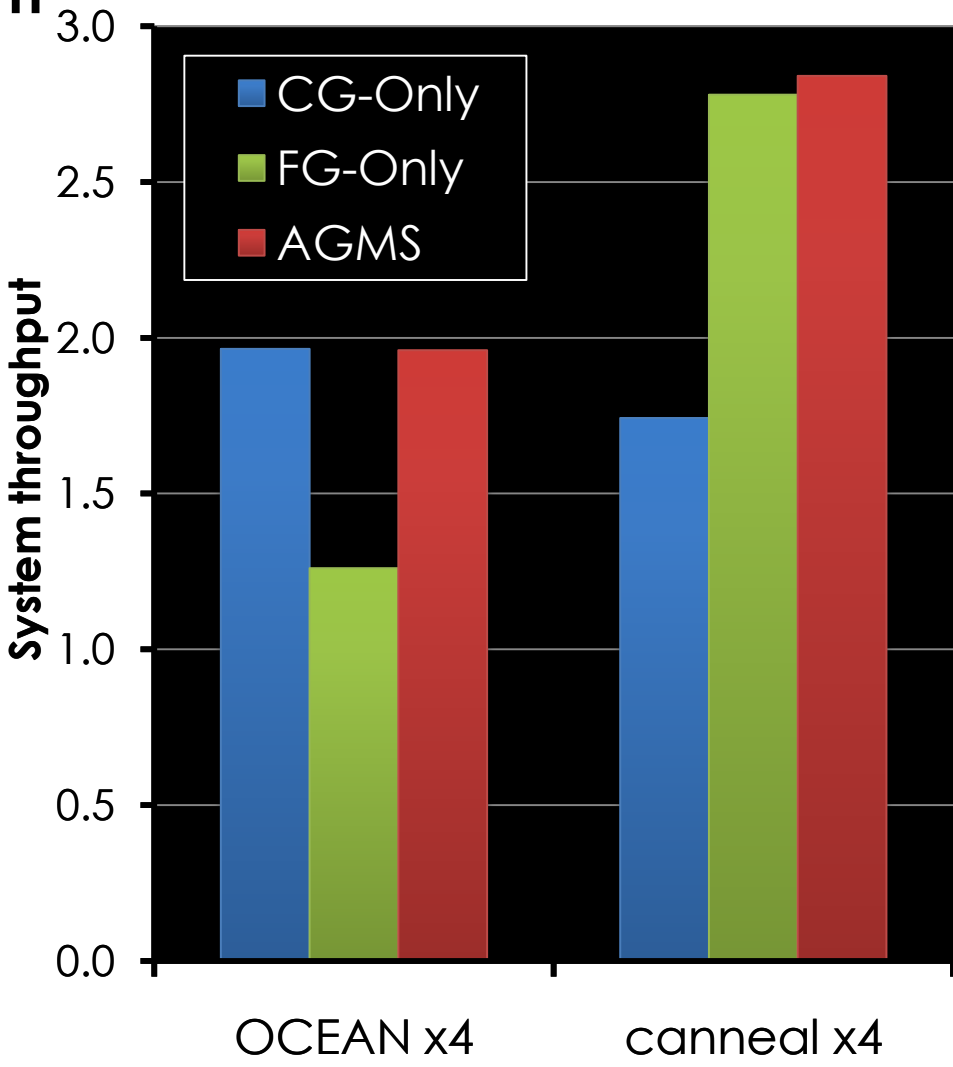
Doe Hyun Yoon
Min Kyu Jeong
Mattan Erez

The University of Texas at Austin
Electrical and Computer Engineering Dept.

THE UNIVERSITY OF TEXAS AT AUSTIN
UT ECE
ELECTRICAL & COMPUTER ENGINEERING

Adaptive Granularity

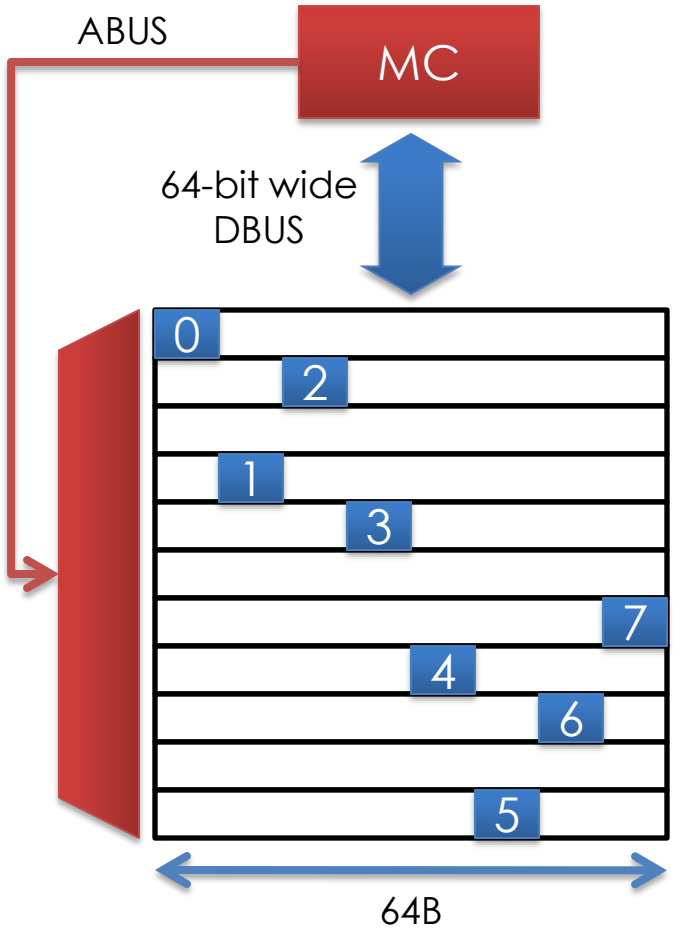
- *Coarse-grain-only* DRAM system
  - Lower ECC overhead
  - *Waste* BW when spatial locality is low
- *Fine-grain-only* DRAM system
  - Higher throughput when spatial locality is low
  - *Expensive*: higher ECC overhead / pin cost
- *Adaptive Granularity*
  - Take the best of CG-Only and FG-Only systems
  - Same or higher throughput across all applications
  - Throughput gain: 44% in 4-core, 85% in 8-core

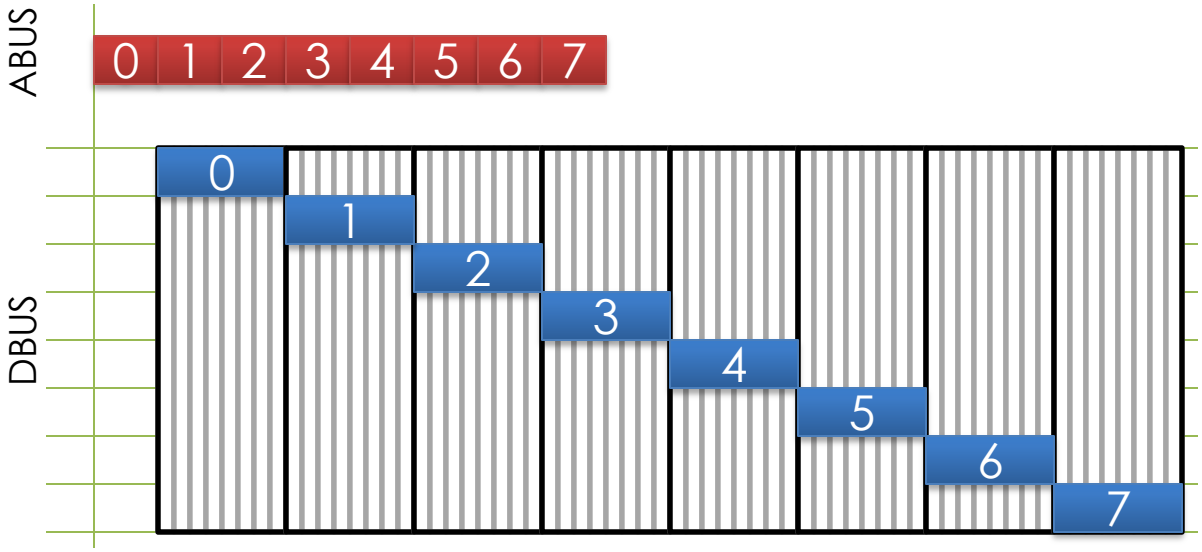


AGMS: MOTIVATION

How Does CG Waste BW?

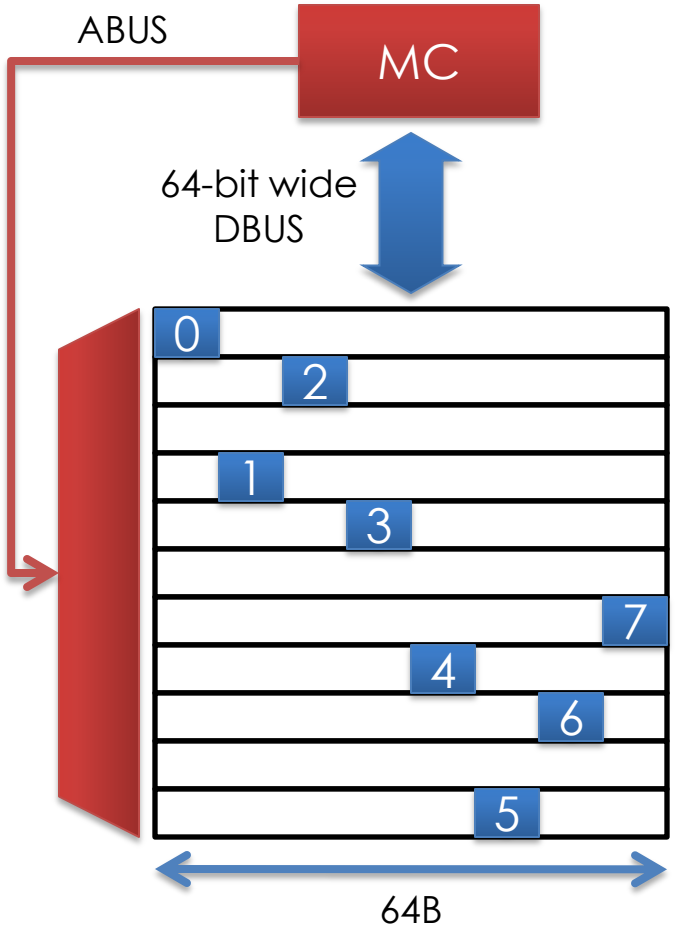
- **Transfer unnecessary data when spatial locality is low**

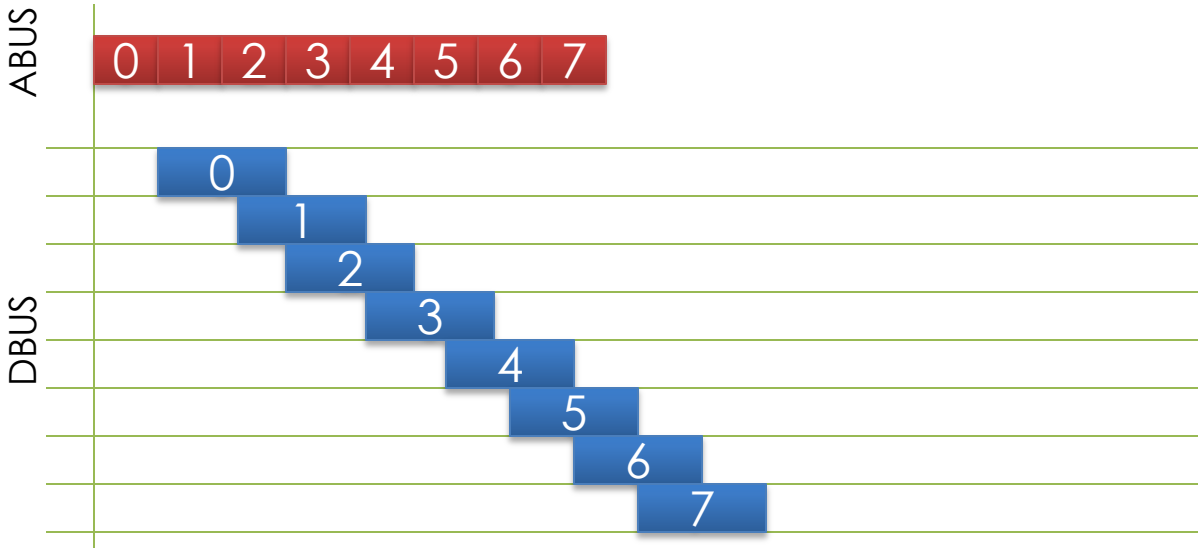




How Can FG Do Better?

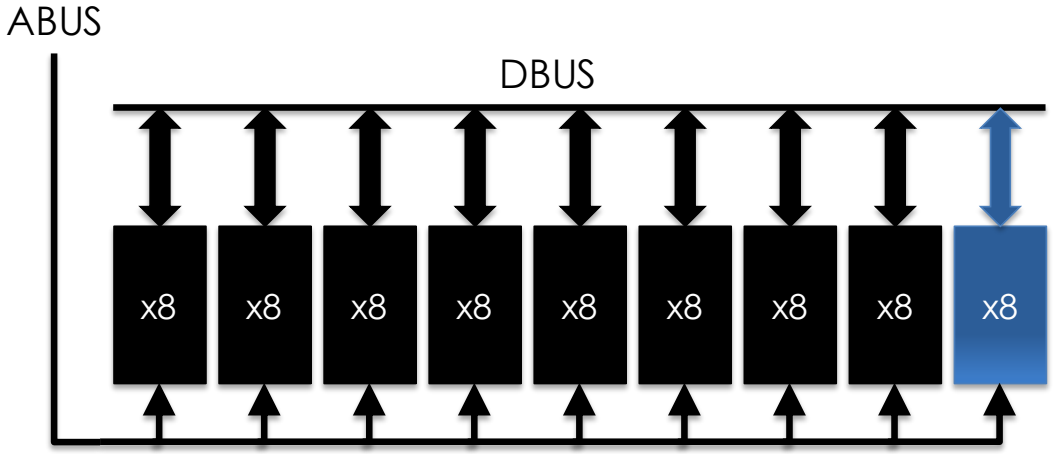
- Transfer only necessary data
- Higher throughput at lower power consumption



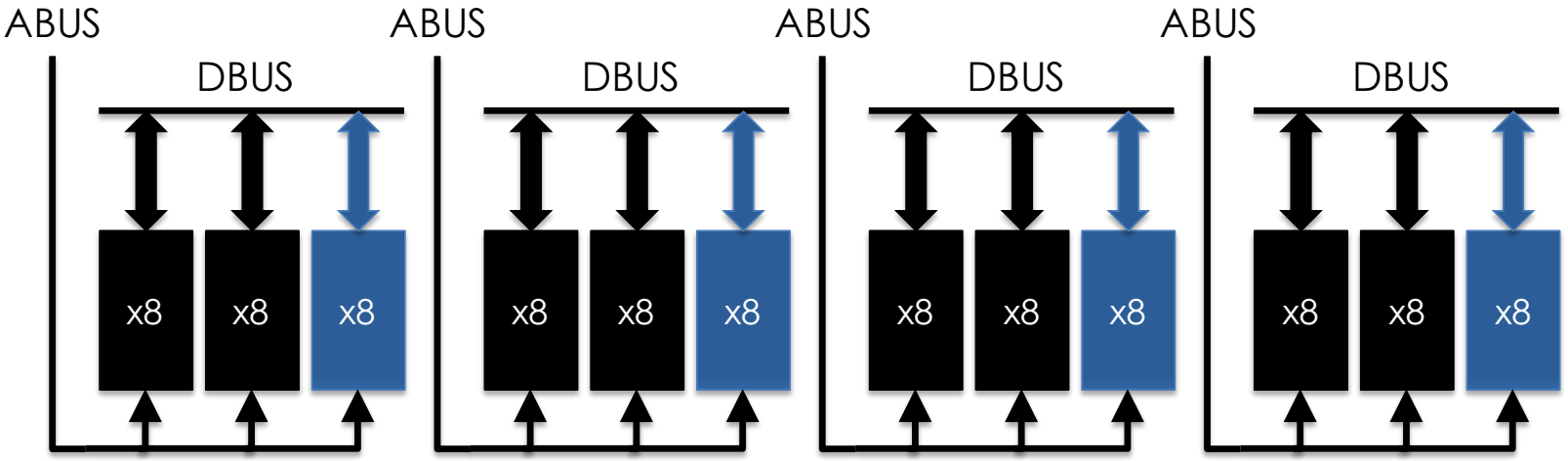


CG-Only and FG-Only Systems

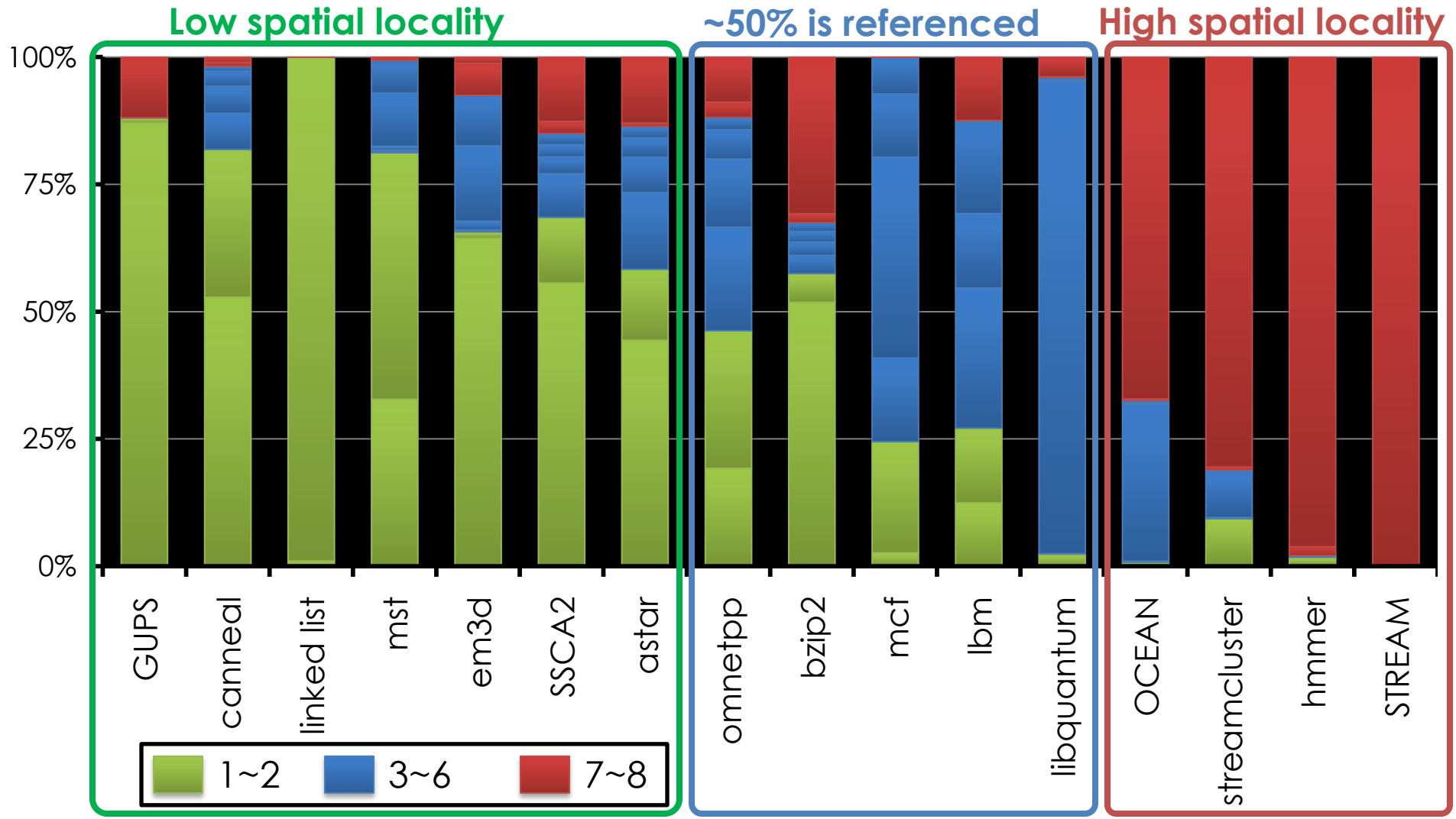
CG-Only: Wide DRAM channel



FG-Only: Many narrow channels



Spatial Locality in Real Apps



 PIN-based profiling with a 1MB cache with a 64B cache line

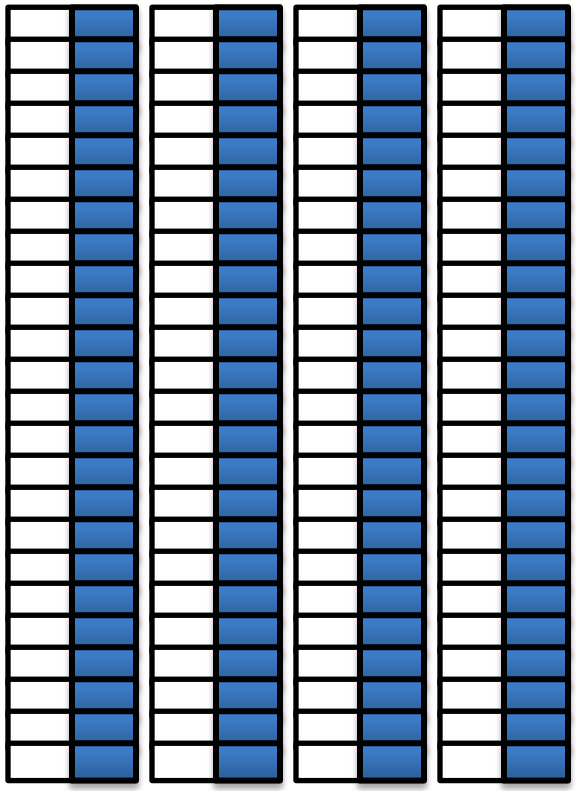
AGMS: DESIGN

CG, FG, and AGMS

CG-only

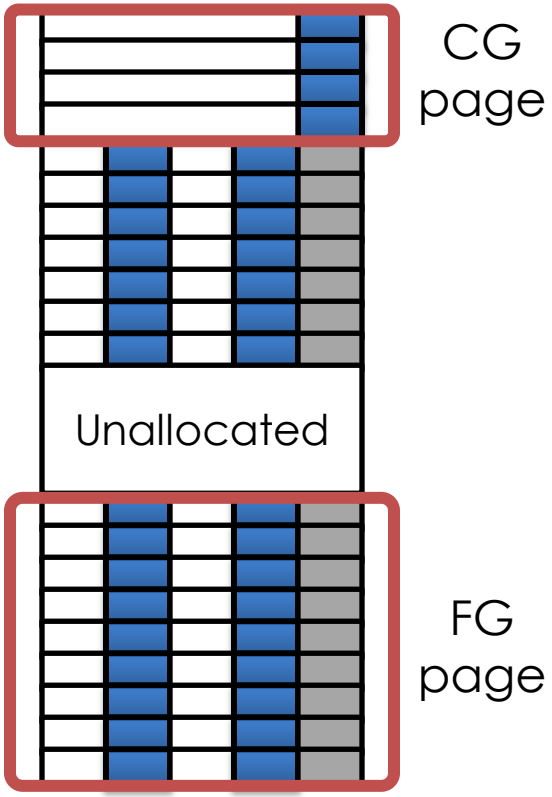
Wide channel
Low redundancy

FG-only



Many narrow channels
High-throughput when
low spatial locality

AGMS



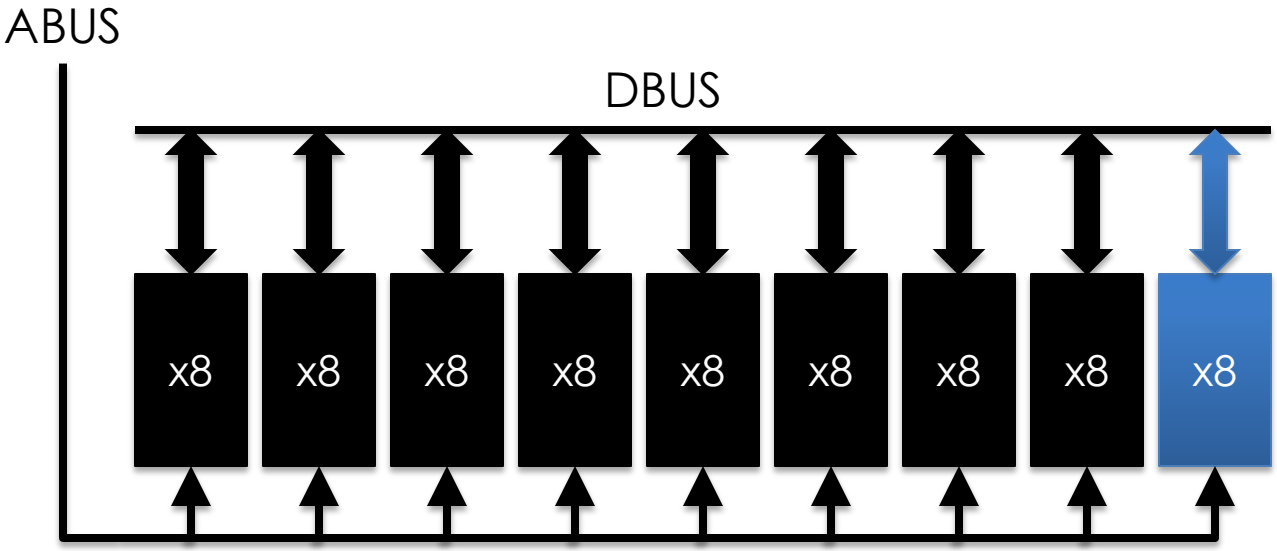
Support
both CG and FG

Challenges in AGMS Design

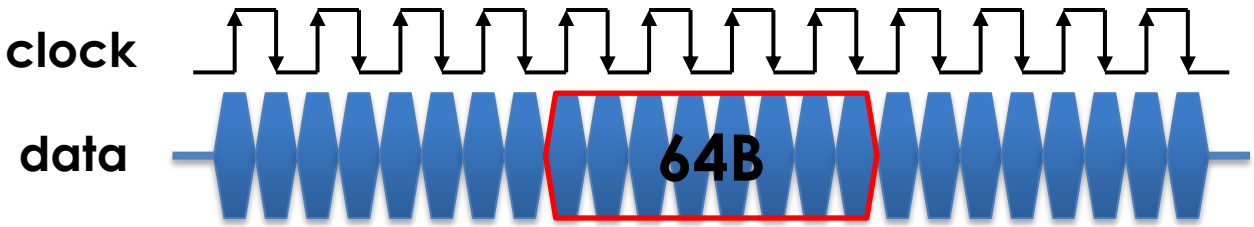
- How to enable fine-grained access?
  - Current memory hierarchy is tuned for coarse-grained access
    - DRAM minimum access granularity: 64B
    - Large cache line size: 64B or larger
- How to identify access granularity?
  - Current programming interface cannot identify per-page access granularity

Typical DDR3 System

- **72-bit wide channel: 64 bit data and 8 bit ECC**



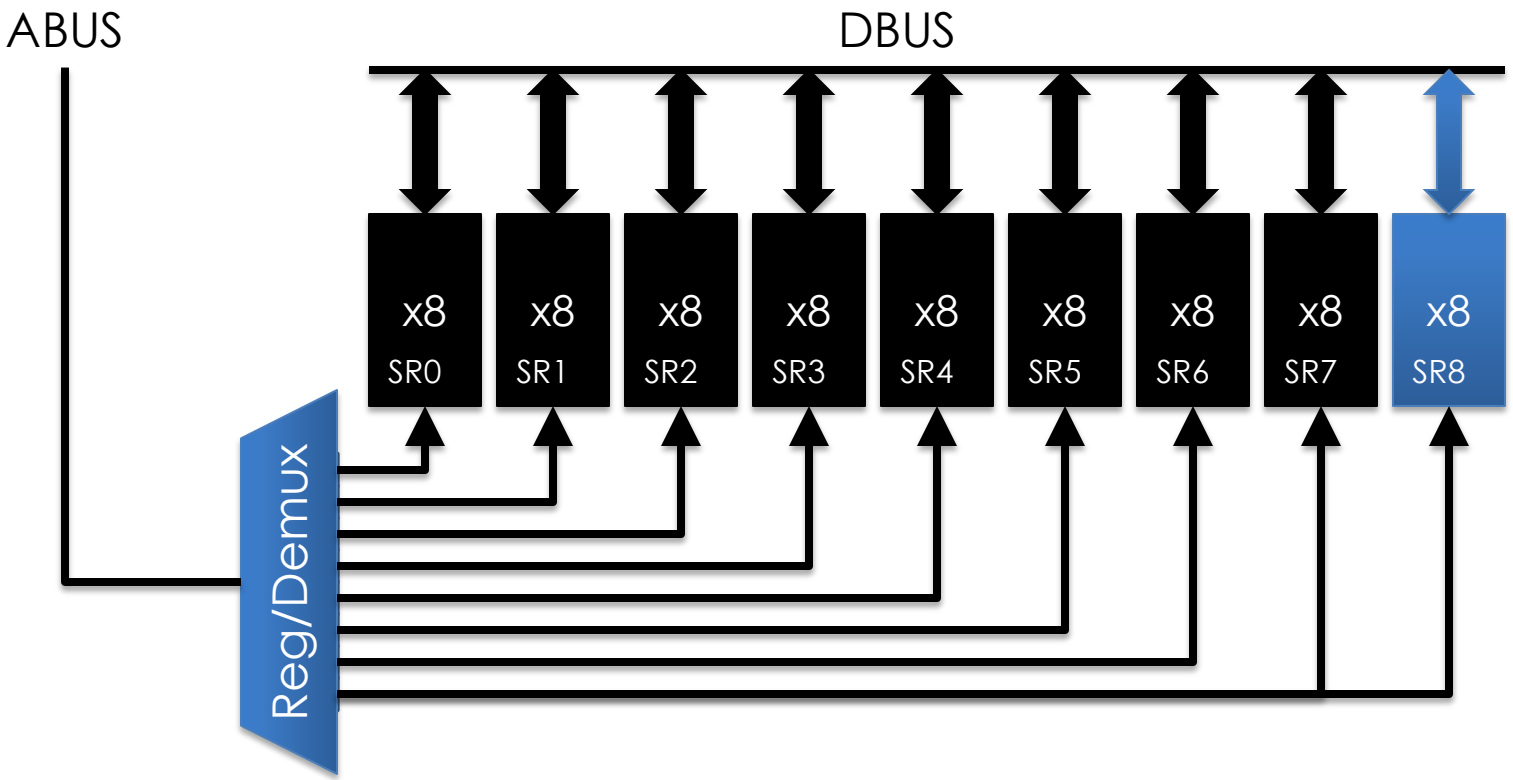
- **Access granularity = 64B (64 bit x 8 burst)**



Sub-Ranked DRAM System

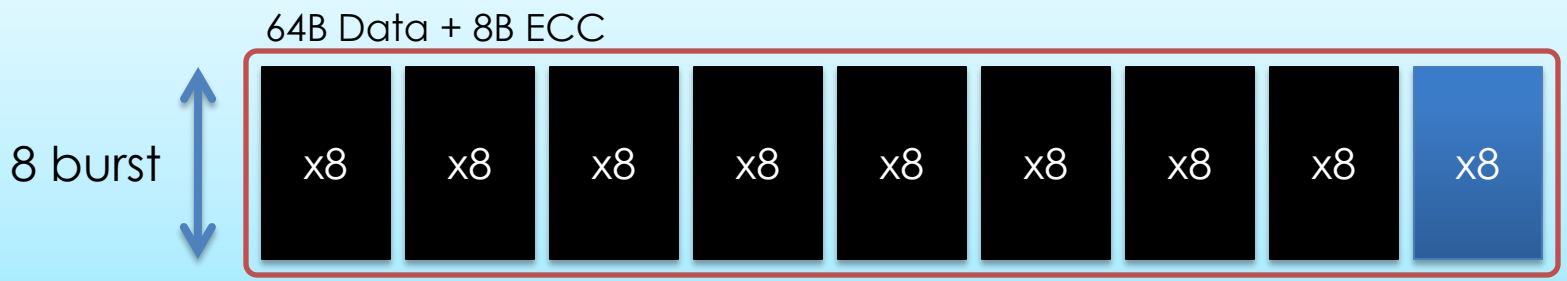
[Threaded module, Mini-rank, MC-DIMM, S/G DIMM]

- Independently control individual DRAM chip
- Access granularity = 8B (8 bit x 8 burst)

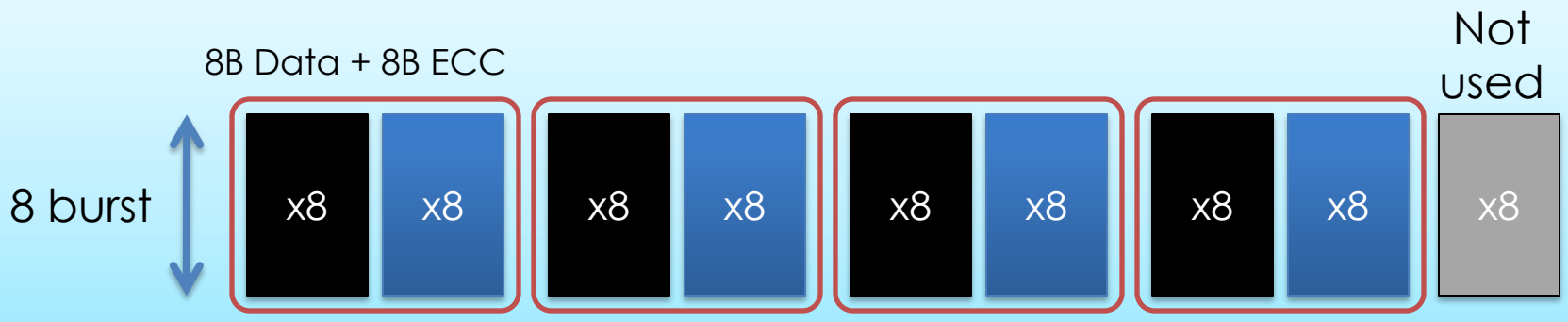


Data Layout

Coarse-grained

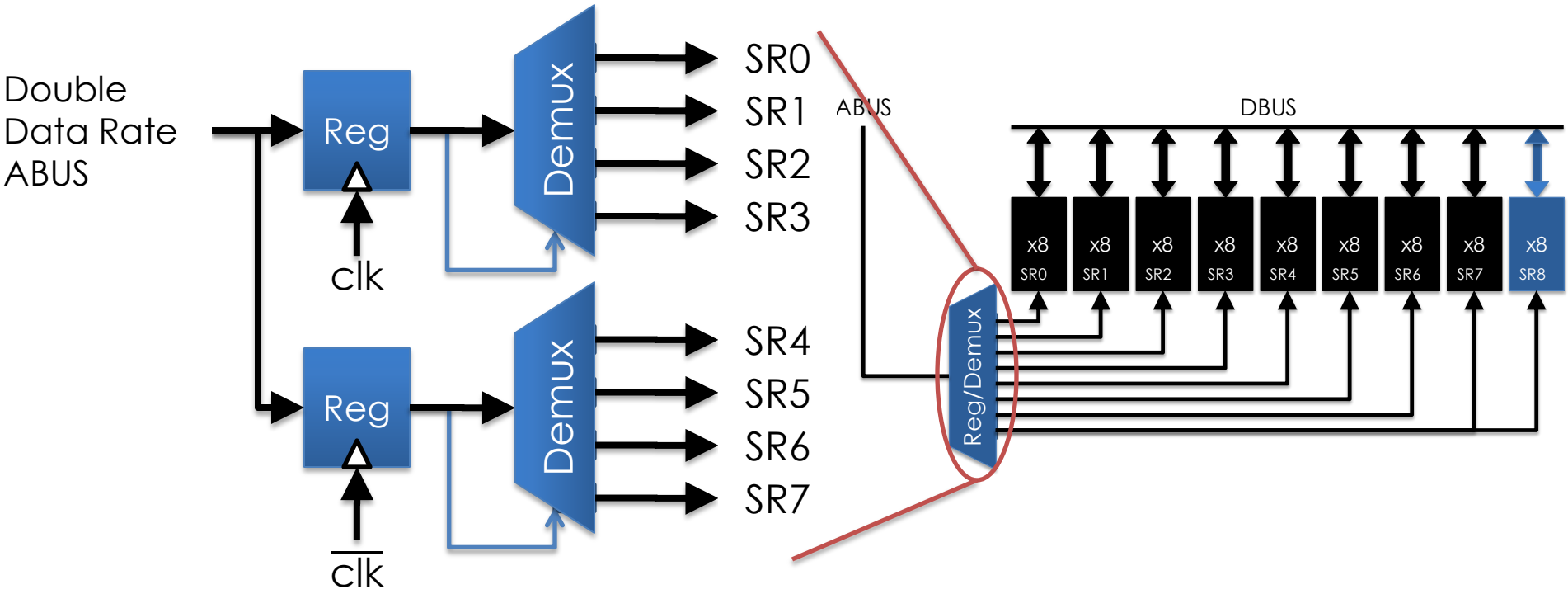


Fine-grained



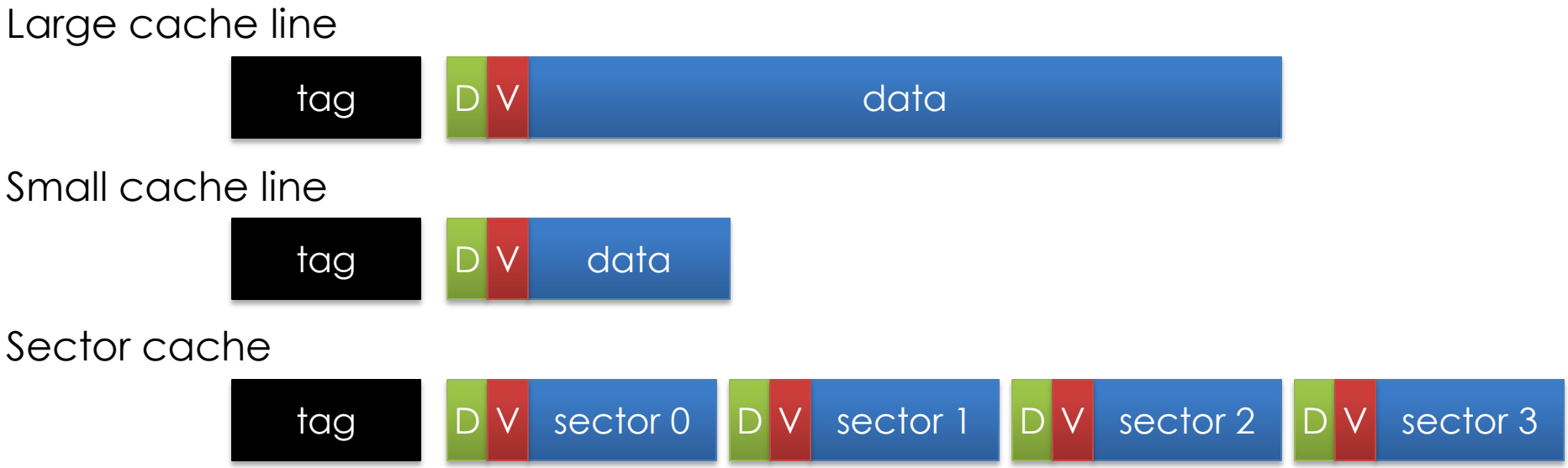
Higher ABUS BW for FG

- 2x ABUS with double data rate ABUS signaling
  - High-end systems already employ faster signaling
    - FB-DIMM, Buffer on board in Nehalem-EX & Power7



Sector Cache

- **Caches also need to manage fine-grained data**
- **We use a simple *sector* cache**
  - **Allow partially valid cache lines**
  - **Low tag overhead**



Application/OS/Runtime

- Augment virtual memory (VM)
  - 1-bit in Page Table Entry (PTE) identifies granularity per page
- Programmers annotate granularity hint in the program
  - e.g., malloc(size, granularity)
- OS/runtime may override granularity decisions dynamically

Not a focus of this study

- Off-line profiler
  - Determine preferred access granularity per page
    - FG (8B) or CG(64B)
  - Emulate *expert* programmers
  - More details in the paper

AGMS Design Summary

Programmer annotation

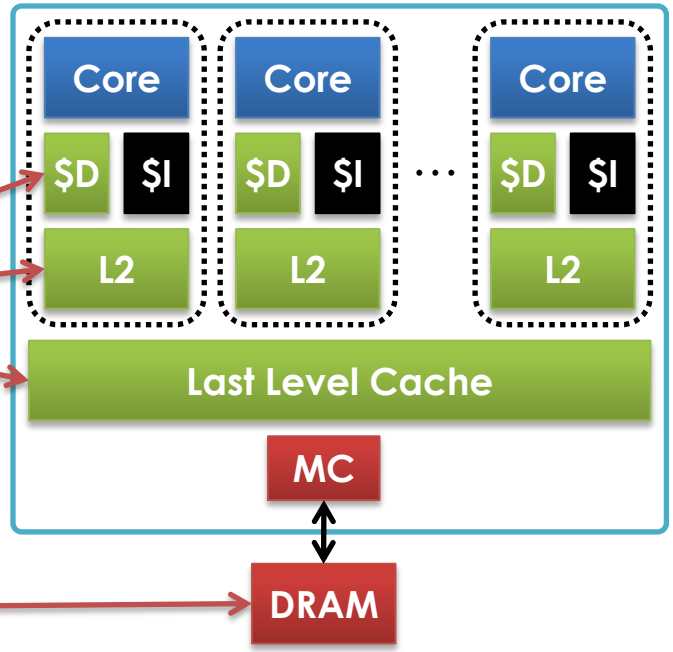
Static/Dynamic mgmt

Currently, profiler-guided static mgmt only

Granularity info in PTE

Sector cache
(8 8B sectors per line)

Sub-Ranked DRAM w/ 2X ABUS
Support both CG and FG



AGMS: EVALUATION

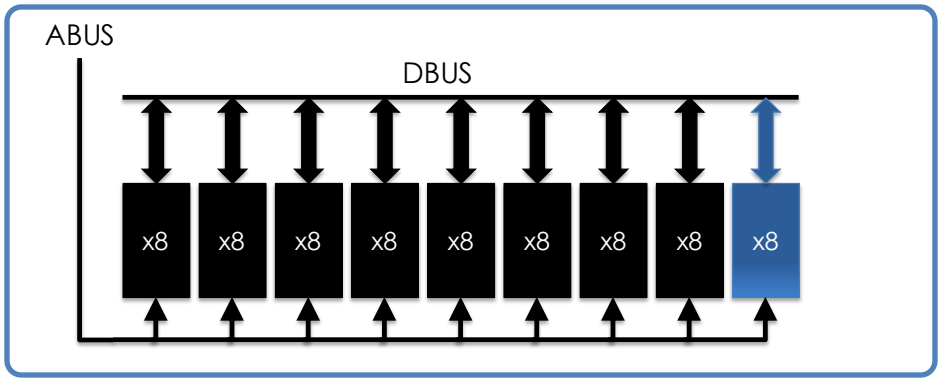
Evaluation

- **Zesto simulator**
  - 4 or 8 out-of-order x86 cores
  - Private cache: 32kB L1 I/D, 128kB & 256kB unifiedL2
  - Last-level shared cache: 4MB (4 core) or 8MB (8 core)
  - Detailed DRAM model: sub-ranked with 2x ABUS

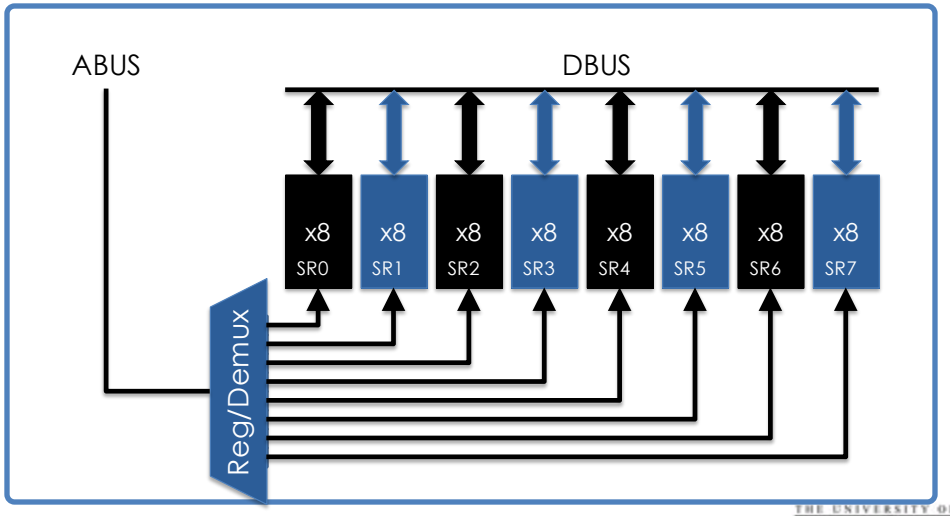
- **Workloads**
  - Memory-intensive apps with low spatial locality
  - SPEC, Olden, PARSEC, SPLASH2, HPCS, and μ-benchmarks

Configurations

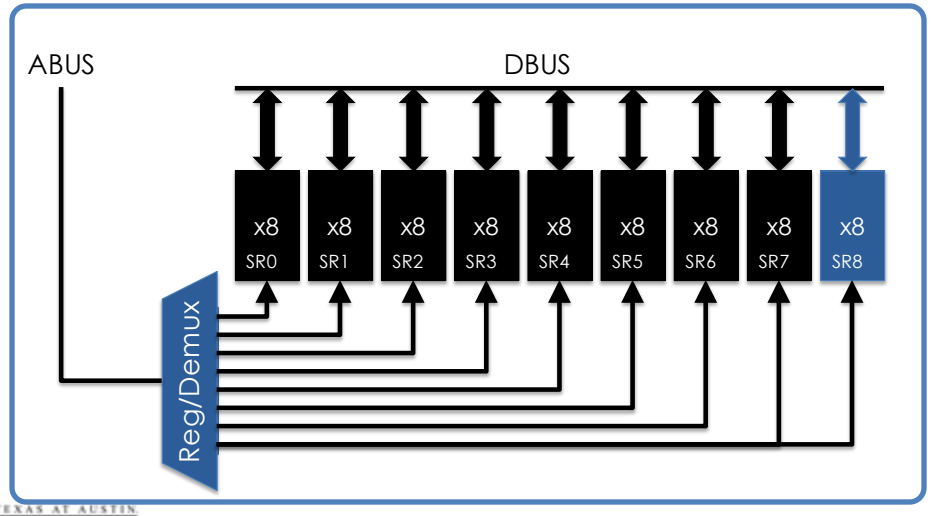
CG+ECC



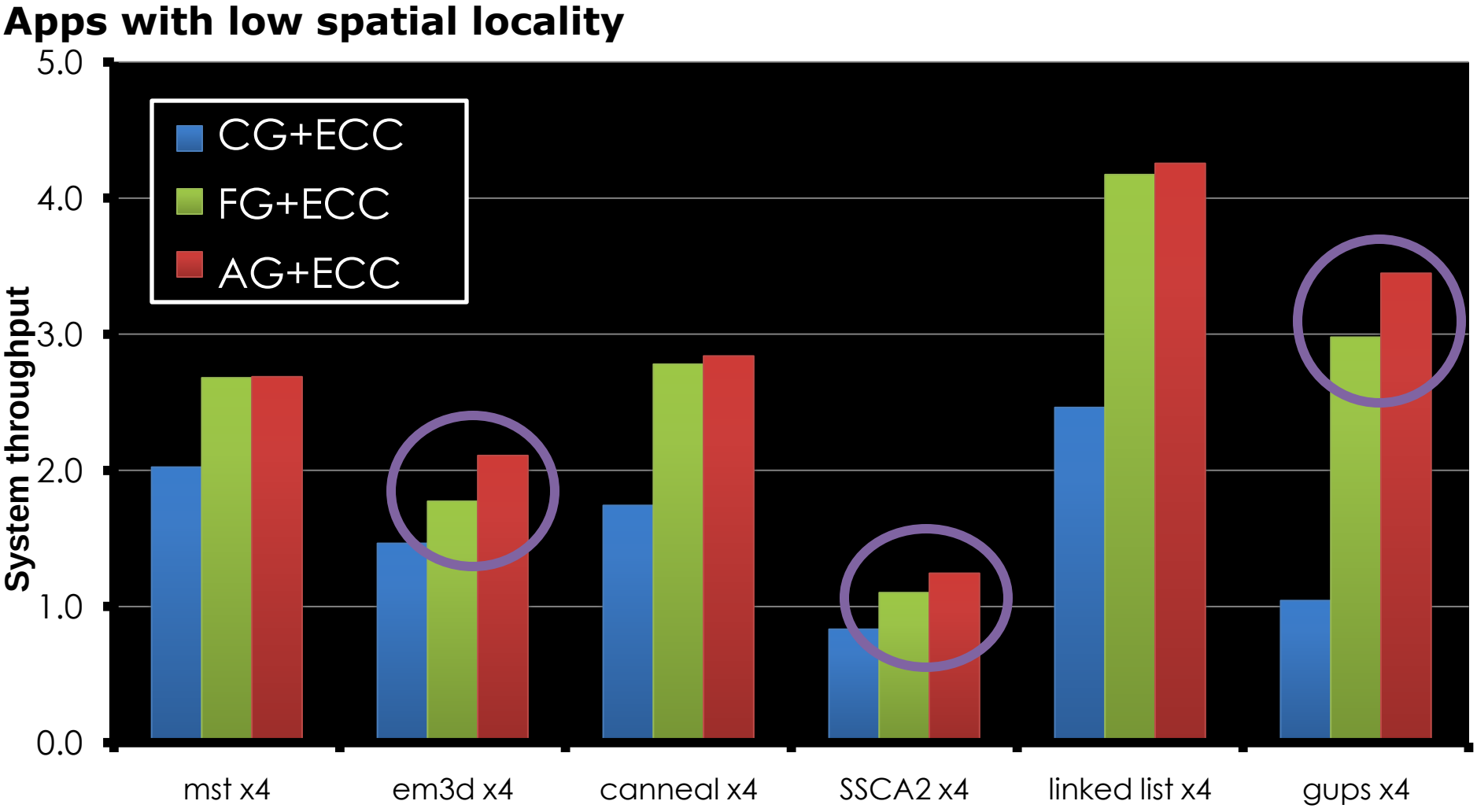
FG+ECC



AG+ECC

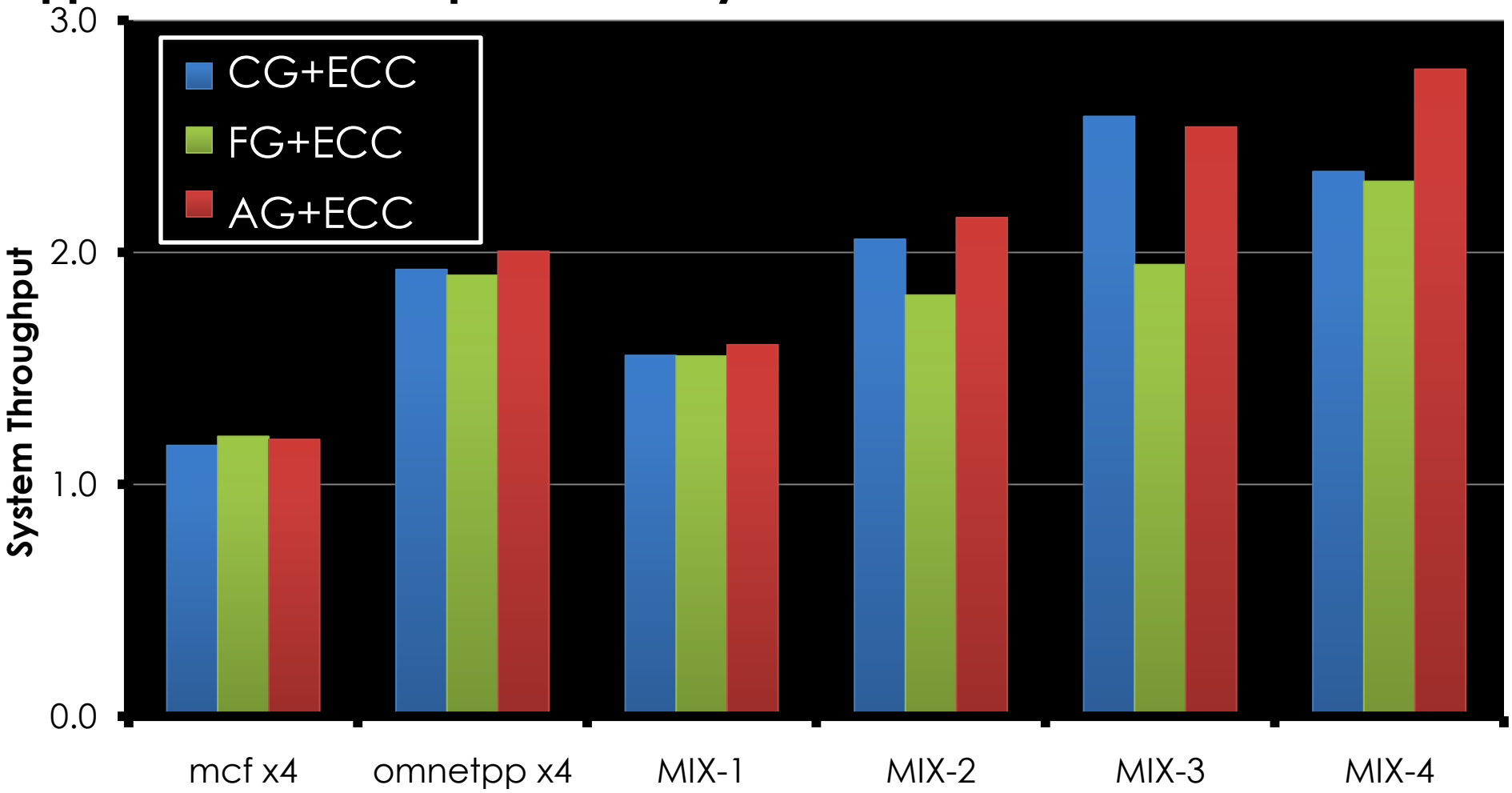


4-Core Results



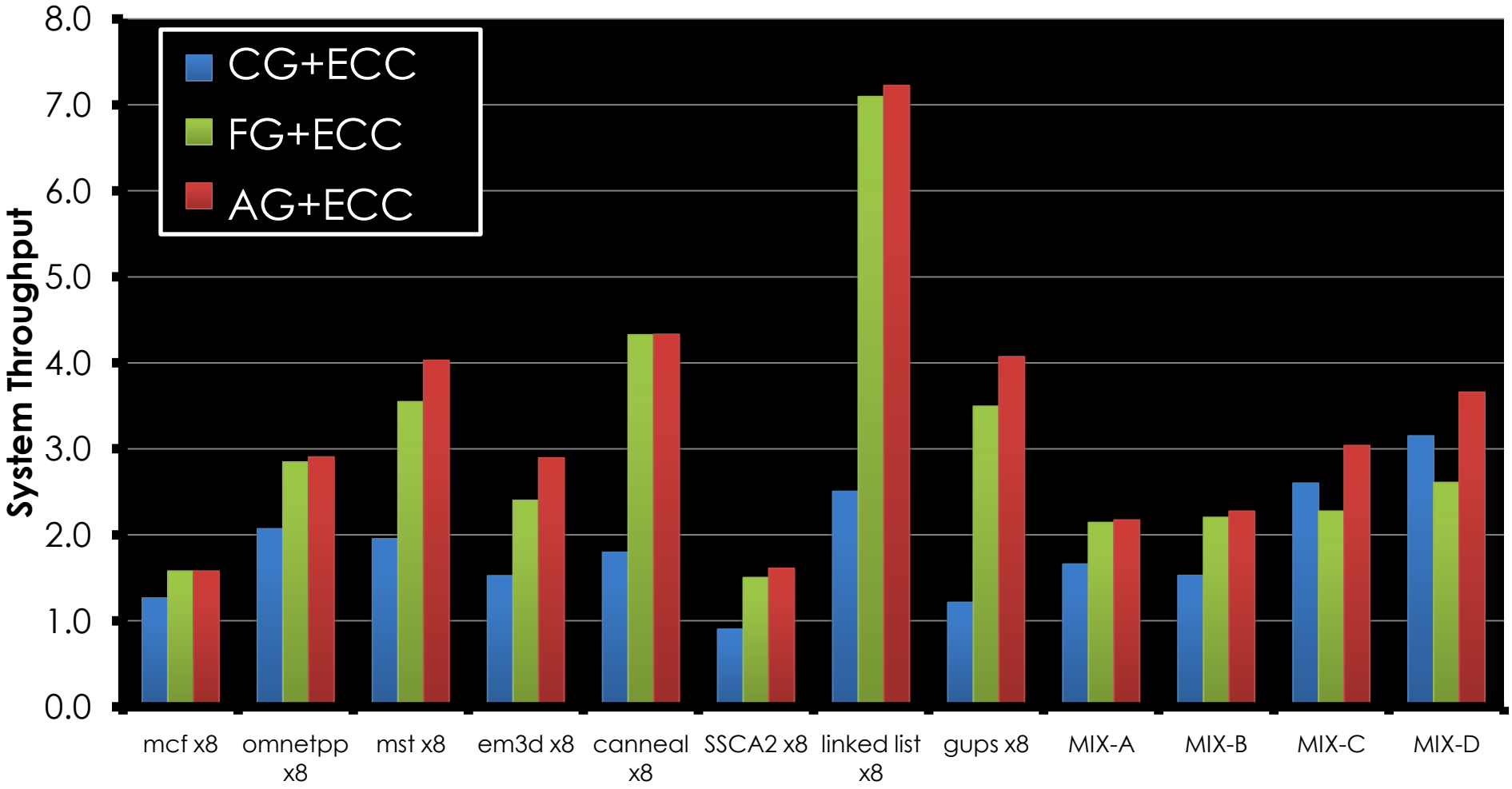
4-Core Results

Apps with medium spatial locality & mixed



MIX-1: mcf, omnetpp, mcf, omnetpp
MIX-2: SSCA2, lbm, astar, SSCA2
MIX-3: libquantum, hmmer, mst, mcf
MIX-4: SSCA2, linked list, mst, hmmer

8-Core Results



MIX-A: mcf x4, omnetpp x4 MIX-B: SSCA2 x2, mcf x2, omnetpp x2, mst x2	MIX-C: SSCA, mcf, omnetpp, mst, astar, hmmer, lbm, bzip2 MIX-D:libquantum x2, hmmer x2, mst x2, mcf x2

Conclusions

- Enable an optimized tradeoff between storage overhead and throughput & power
- Throughput improvements
  - 44% in 4-core, 85% in 8-core
- Reduce DRAM power and off-chip traffic
- Improve power efficiency
- Results for non-ECC systems in the paper

Adaptive Granularity Memory Systems:
A Tradeoff between Storage Efficiency and Throughput

Doe Hyun Yoon
Min Kyu Jeong
Mattan Erez

The University of Texas at Austin
Electrical and Computer Engineering Dept.

THE UNIVERSITY OF TEXAS AT AUSTIN
UT ECE
ELECTRICAL & COMPUTER ENGINEERING