# A Case for Toggle-Aware Compression for GPU Systems

**Gennady Pekhimenko**,
Nandita Vijaykumar,
Onur Mutlu, Todd C. Mowry

Evgeny Bolotin,
Stephen W. Keckler

SAFARI Carnegie Mellon

NVIDIA

# Executive Summary

**Data compression** is a known technique to decrease the bandwidth pressure

**Observation:** Compression significantly increases the energy cost of communication by increasing the number of bit toggles (bit flips)

**Our approach**: ***Toggle-Aware Compression***

- Energy Control (EC): send compressed data only when it is beneficial
- Metadata Consolidation (MC): consolidates metadata bits to reduce the bit toggle count

**Key results**: 2.2X increase in bit toggles reduced to only 1.1X with most of the performance benefits preserved

# Performance and Energy Efficiency

*Energy efficiency*

**Applications today are *data-intensive***

MEMCACHED

amazon DynamoDB


Memory Caching

Databases

Graphics

# Computation vs. Communication

**Modern memory systems are** ***bandwidth constrained***

**Data movement is very costly**

*Sources:*
*Bill Dally (NVIDIA/Stanford)*
*Kayvon Fatahalian (CMU)*

- Integer operation: **~1 pJ**
- Floating operation: **~20 pJ**
- Low-power memory access: **~1200 pJ**

**Implication**

- Transfer **less** or keep data **near processing units**

# Potential for Data Compression

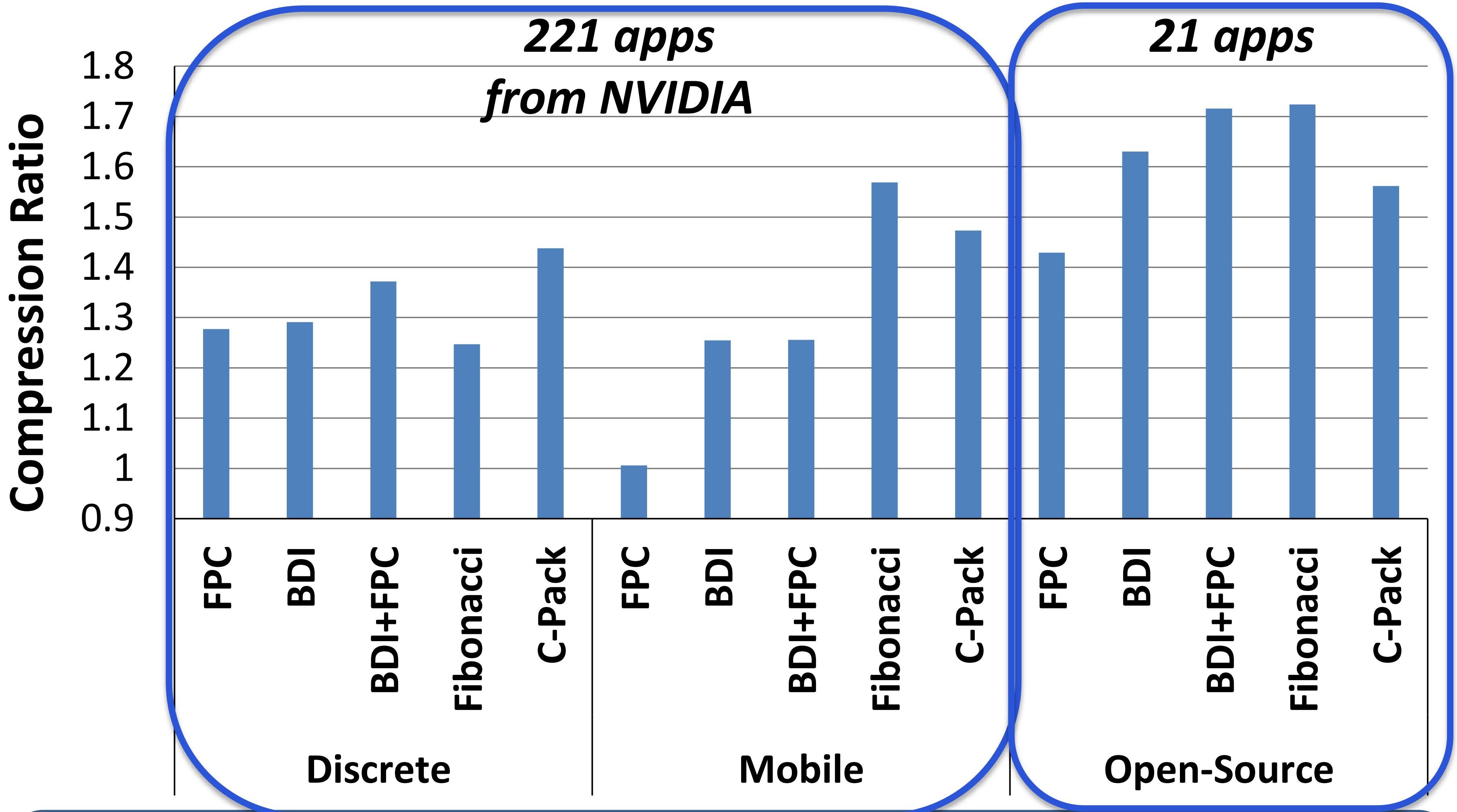
Significant redundancy in memory transfers:

| | | | | |
|---|---|---|---|---|
| 0x00000000 | 0x0000000B | 0x00000003 | 0x00000004 | … |

How can we exploit this redundancy?

- **Bandwidth compression**
  - Provides effect of a **higher** effective bandwidth *without increasing the number of wires or raising the frequency*

# Bandwidth Compression for GPUs




**Compression is effective in providing higher bandwidth**

# Common Wisdom about Compression

**A new problem:**

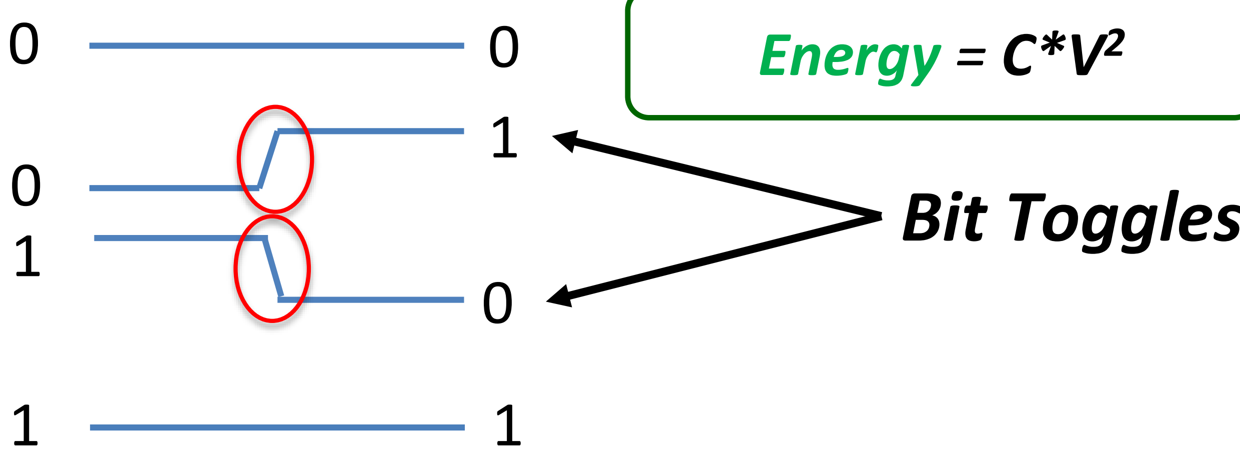
**Significant increase in the bit toggle count (# bit flips), despite less bits sent**

# What is a Bit Toggle?

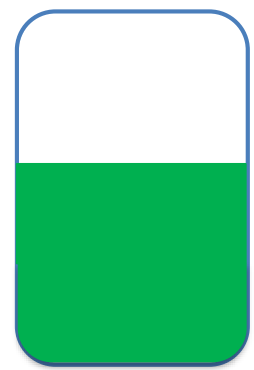
**How energy is spent in data transfers:**

**Previous data:** 0011 **New data:** 0101

***Energy*** $= C^*V^2$




***Energy:***



**Energy of data transfers (e.g., NoC, DRAM) is proportional to the bit toggle count**

# Excessive Number of Bit Toggles

*Uncompressed Cache Line*

0x00003A00 0x8001D000 | 0x00003A01 0x8001D008 | ...

***Flit 0***

**XOR**

***Flit 1***

**=**

000000010....01000

***# Toggles = 2***

*Compressed Cache Line (FPC)*

0x5 0x3A00 0x7 8001D000 0x5 0x3A01 0x7 8001D008 ...

5 3A00 7 8001D000 5 1D

***Flit 0***

**XOR**

1 01 7 8001D008 5 3A02 1

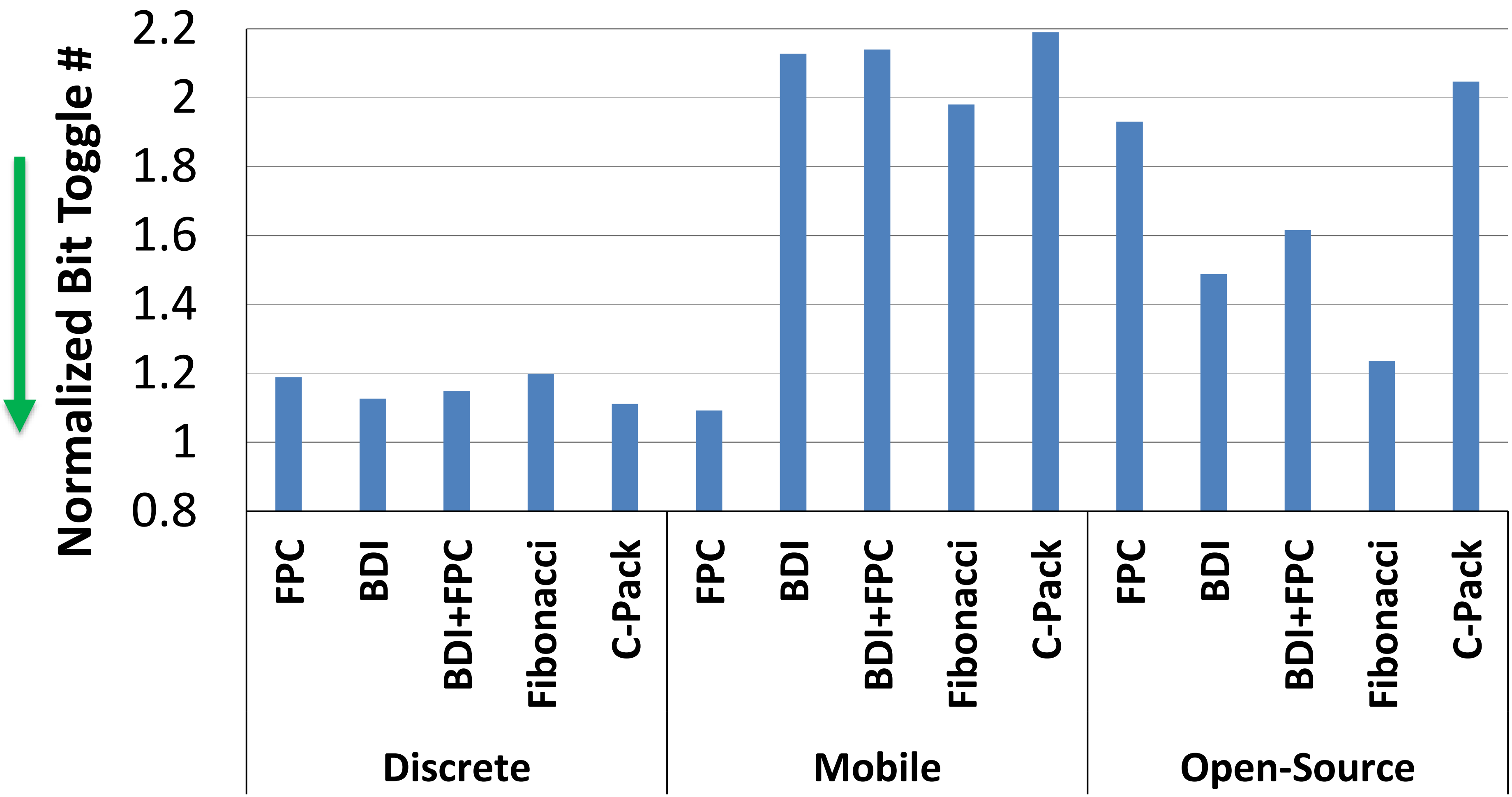
***Flit 1***

**=**

001001111 ... 110100011000

***# Toggles = 31***

# Effect of Compression on Bit Toggles

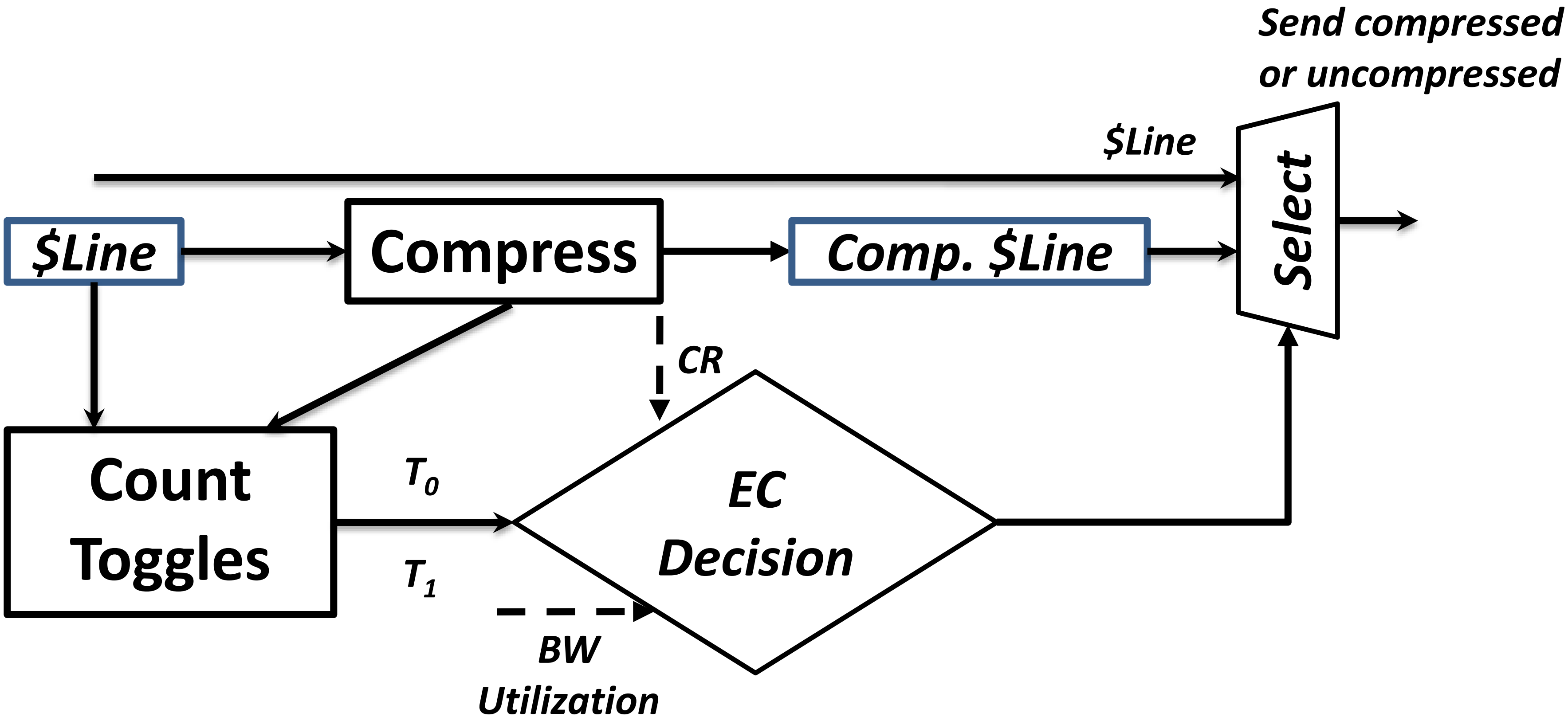



**Compression significantly increases bit toggle count**

# Outline

- Motivation
- Key Observations
- Toggle-Aware Compression:
  - Energy Control (EC)
  - Metadata Consolidation (MC)
- Evaluation
- Conclusion

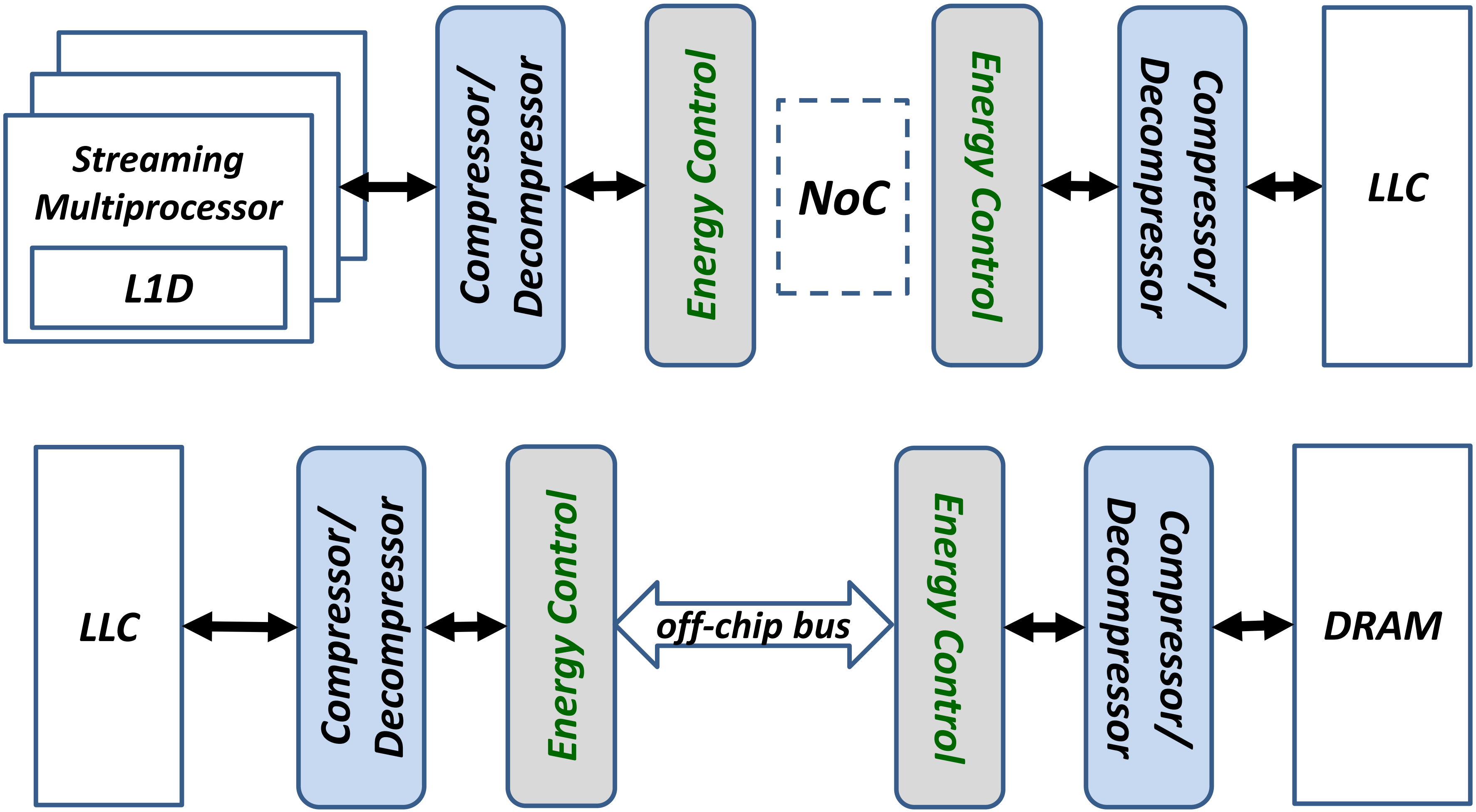
# Energy Control Decision Flow



# How to Make the EC Decision?

- **Energy**
  – Battery life
- **Energy X Delay**
  – Balance performance and energy
- **Energy X Delay$^2$**
  – Fixed power with voltage scaling
- Energy: **~ Toggle #**, Delay ~ **1/(Comp. Ratio)**
  – When **bandwidth utilization (BU)** is high (>50%) use 1 / (1 - BU) coefficient

# EC in the System

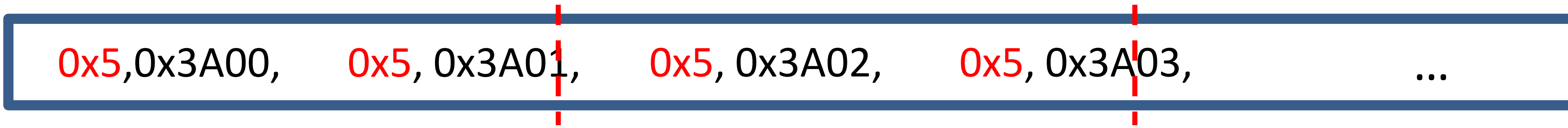


# Energy Control Summary

- ***Bit toggle count***: compressed vs. uncompressed
- Use a heuristic (*Energy X Delay* *or* *Energy X Delay*$^2$ metric) to estimate the trade-off
- Take ***bandwidth utilization*** into account
- Throttle compression when it is **not** beneficial

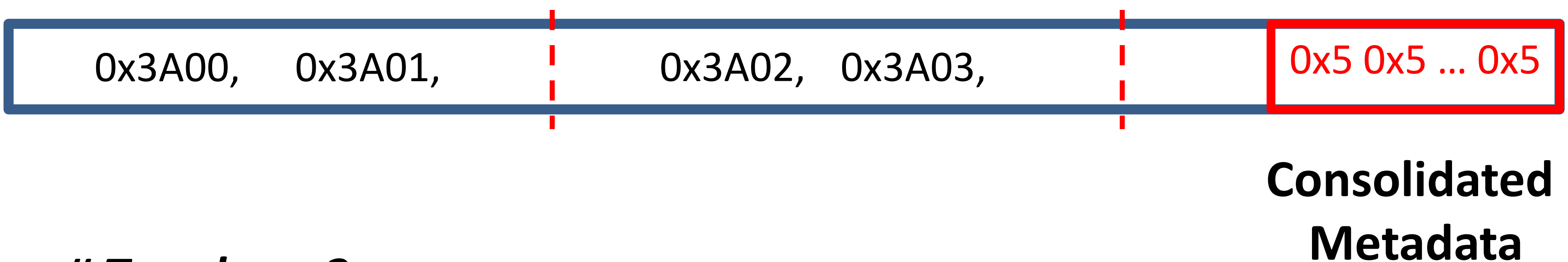
# Metadata Consolidation

Compressed Cache Line with FPC, 4-byte flits




***# Toggles = 18***

Toggle-aware FPC: all metadata **consolidated**




***# Toggles = 2***

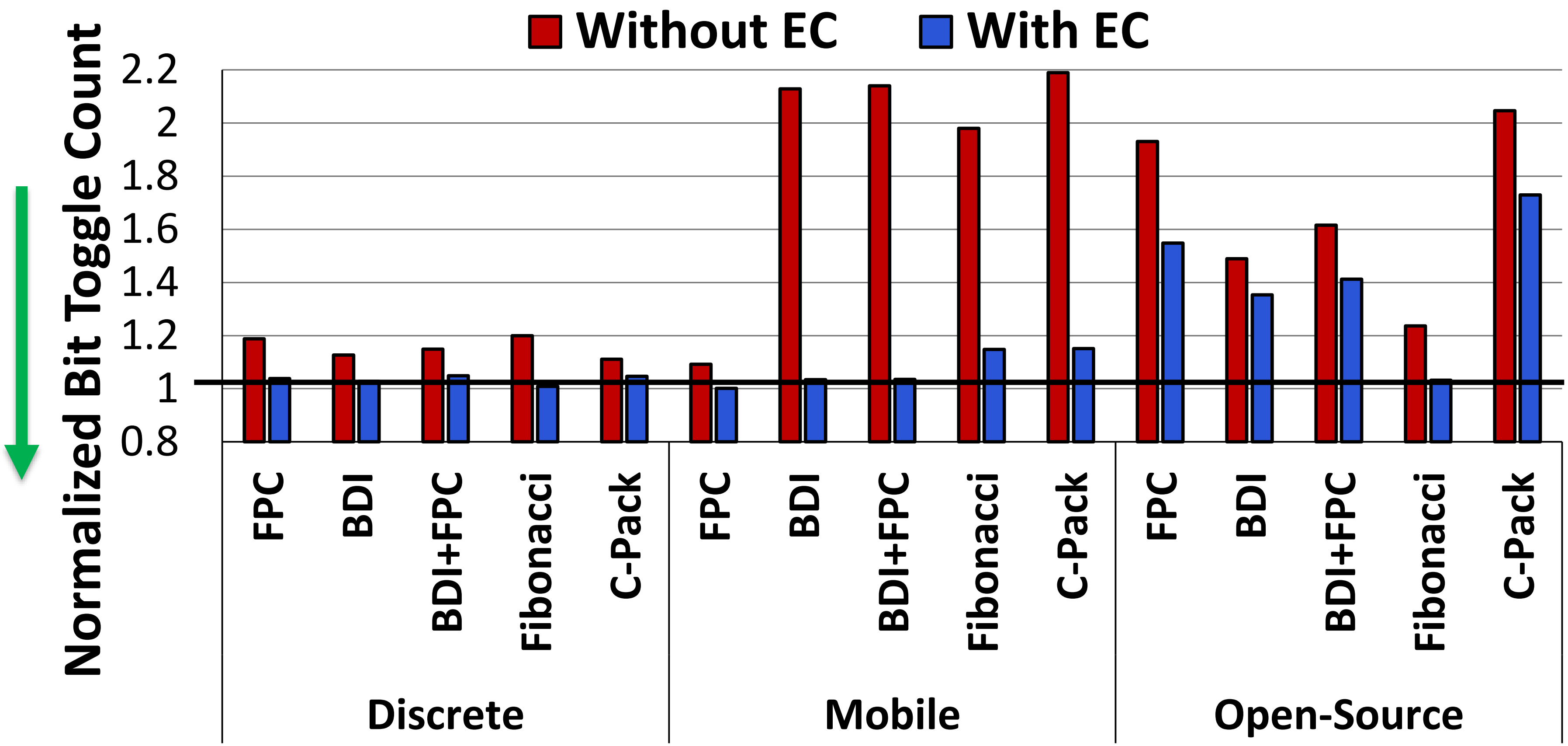
# Outline

- Motivation
- Key Observations
- Toggle-Aware Compression:
  - Energy Control (EC)
  - Metadata Consolidation (MC)
- Evaluation
- Conclusion

# Methodology

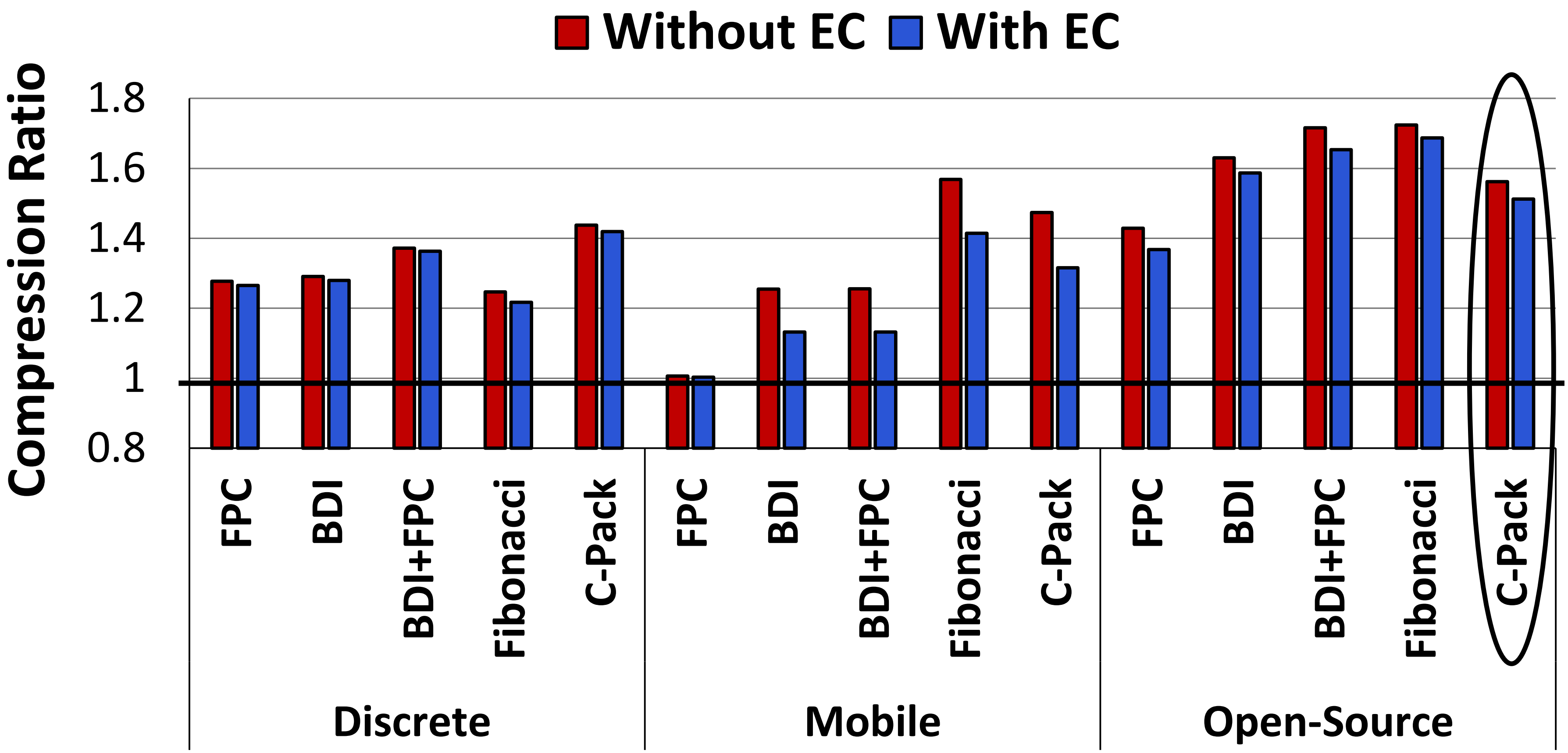
- **Simulator**: GPGPU-Sim 3.2.x and in-house simulator
- **Workloads**:
  - NVIDIA apps (discrete and mobile): **221 apps**
  - Open-source (Lonestar, Rodinia, MapReduce): **21 apps**
- **System parameters (Fermi)**:
  - 15 SMs, 32 threads/warp
  - 48 warps/SM, 32768 registers, 32KB Shared Memory
  - Core: 1.4GHz, GTO scheduler, 2 schedulers/SM
  - Memory: 177.4GB/s BW, GDDR5
  - Cache: L1 - 16KB; L2 - 768KB

# Effect of EC on Bit Toggle Count



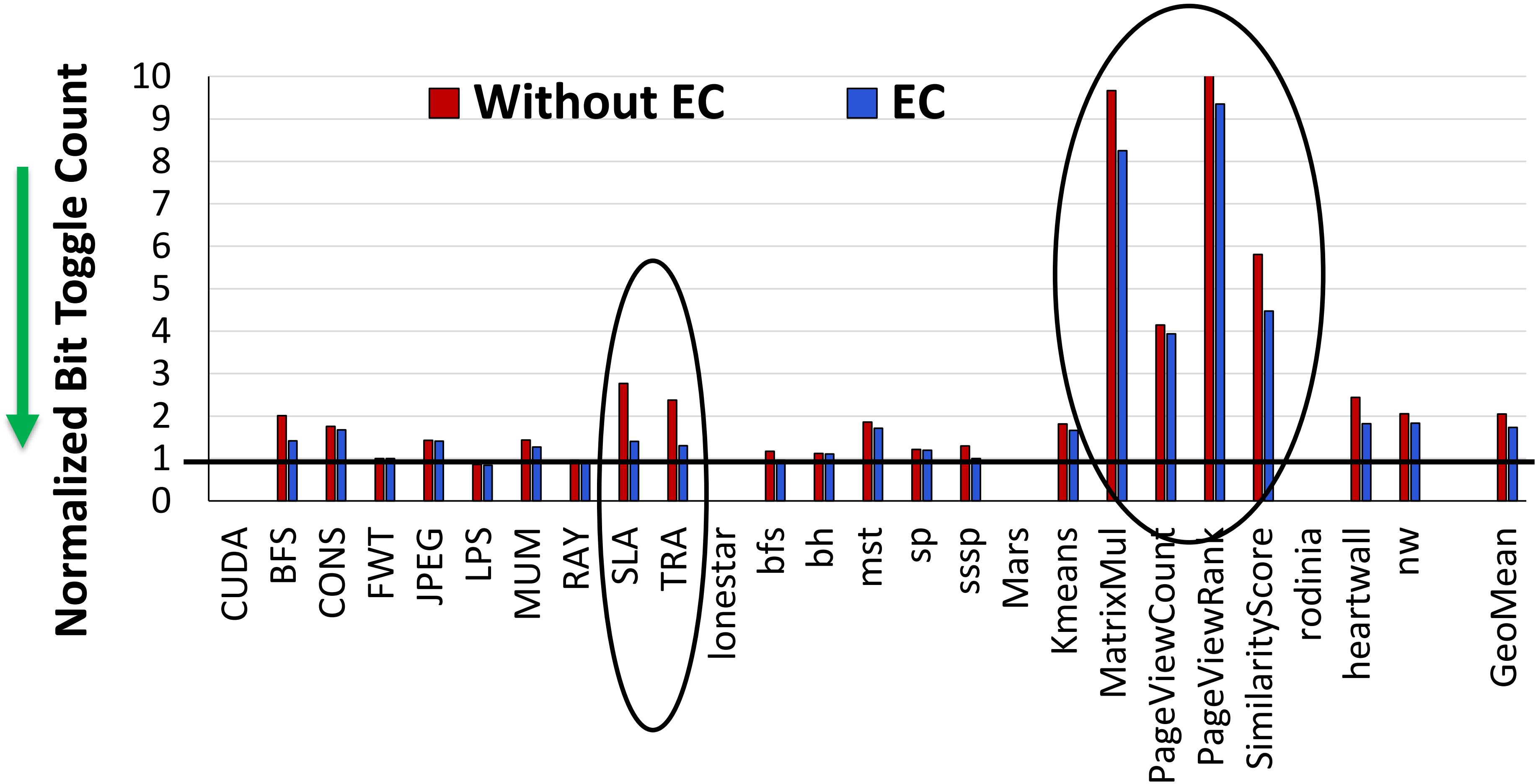

- ✓ EC significantly reduces the bit toggle count
- ✓ Works for different compression algorithms

# Effect of EC on Compression Ratio



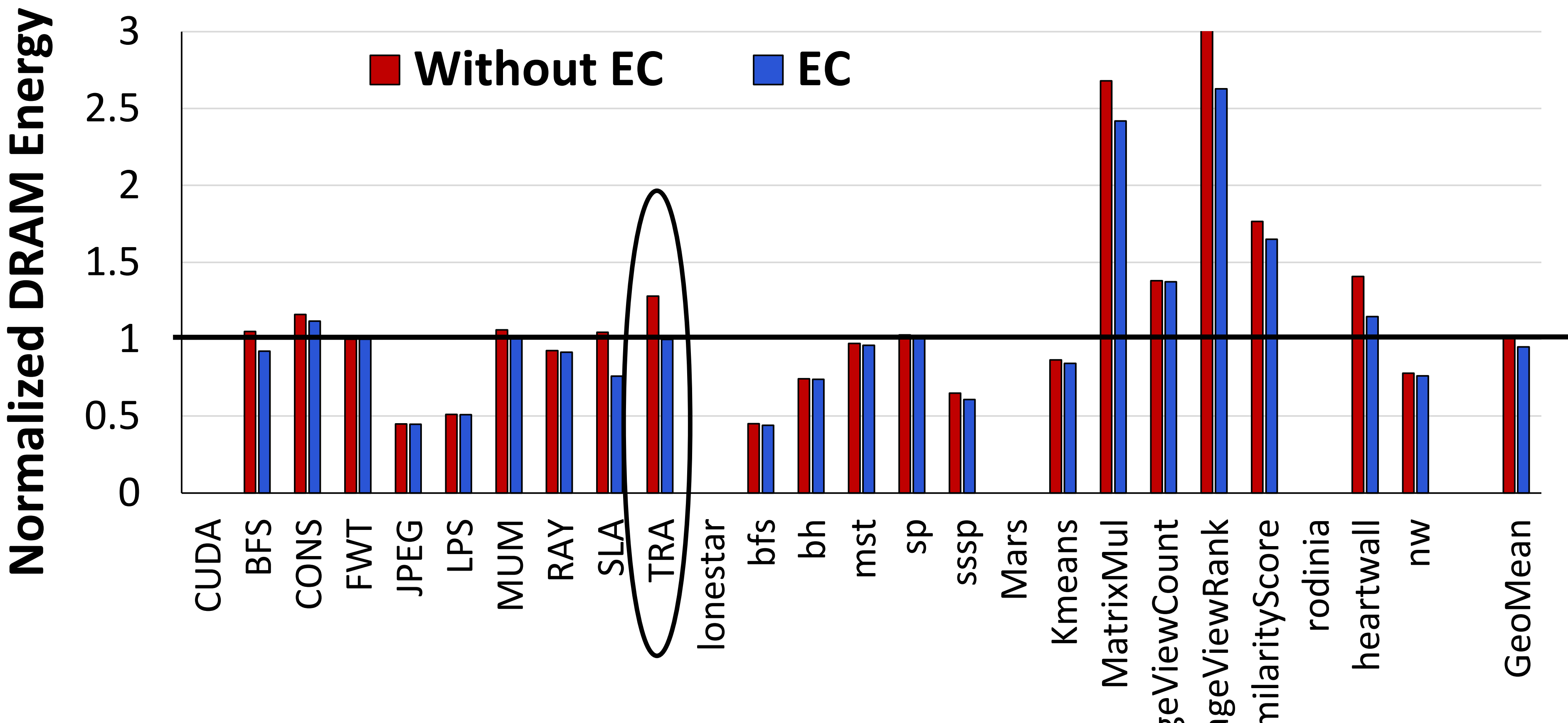

**EC preserves most of the benefits of compression**

# Bit Toggles for C-Pack Algorithm



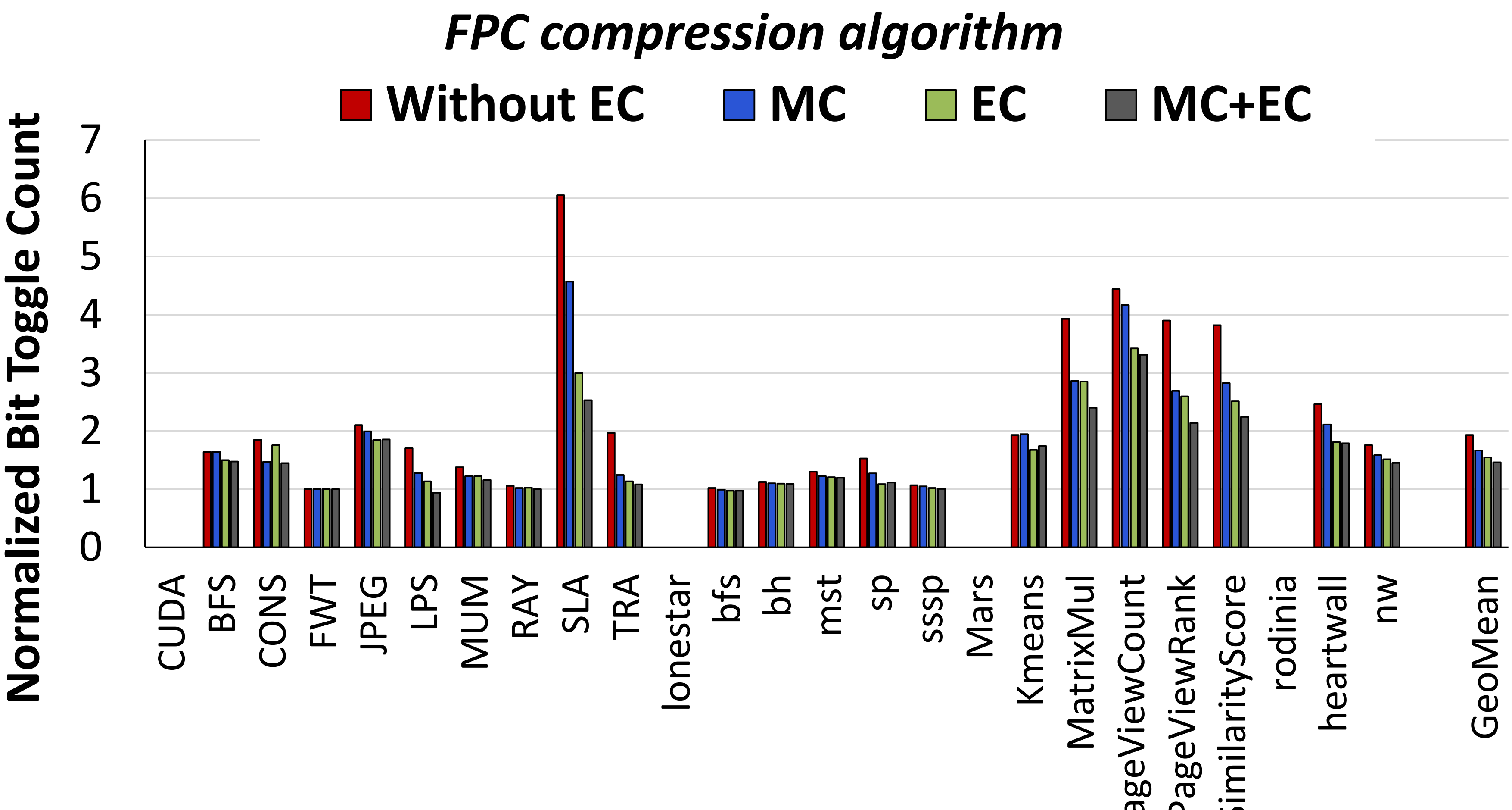

**Different tradeoffs for different applications**

# DRAM Energy for C-Pack




**7% average DRAM energy reduction, up to 28% for TRA**

# Effect of Metadata Consolidation (MC)




**MC is effective in reducing the bit toggle count**
**But less effective than EC**

# Other Results in the Paper

- On-chip interconnect results
  - Higher impact of bit toggles on the interconnect energy, but lower overall energy impact
- Data bus inversion (DBI)
  - EC and MC benefits are independent on whether DBI encoding is used
- Complexity estimation
  - Energy and latency
- Analyzing different EC decision functions
  - Energy x Delay vs. Energy x Delay$^2$

# Conclusion

**Data compression** is a known technique to decrease the bandwidth pressure

**Observation:** Compression significantly increases the energy cost of communication by increasing the number of bit toggles (bit flips)

**Our approach**: ***Toggle-Aware Compression***

- Energy Control (EC): send compressed data only when it is beneficial
- Metadata Consolidation (MC): consolidate metadata bits to reduce the bit toggle count

**Key results**: 2.2X increase in bit toggles reduced to only 1.1X with most of the performance benefits preserved

# A Case for Toggle-Aware Compression for GPU Systems

**Gennady Pekhimenko**,
Nandita Vijaykumar,
Onur Mutlu, Todd C. Mowry

Evgeny Bolotin,
Stephen W. Keckler

SAFARI Carnegie Mellon

NVIDIA