# GraphR: Accelerating Graph Processing Using ReRAM

Linghao Song*, Youwei Zhuo#,
Xuehai Qian#, Hai Li*, Yiran Chen*

*Duke University
#University of Southern California

CEI
cei.pratt.duke.edu



ALCHEM
alchem.usc.edu

# Graph Processing

- To understand relationships in a group of nodes

- A wide range of application domains

  —Bioinformatics, Social Networks, Cyber Security, Data Mining…

- Classic algorithms:

  —Sparse Matrix Vector Multiplication (SpMV)

  —Single Source Shortest Path (SSSP)

  —Page Rank

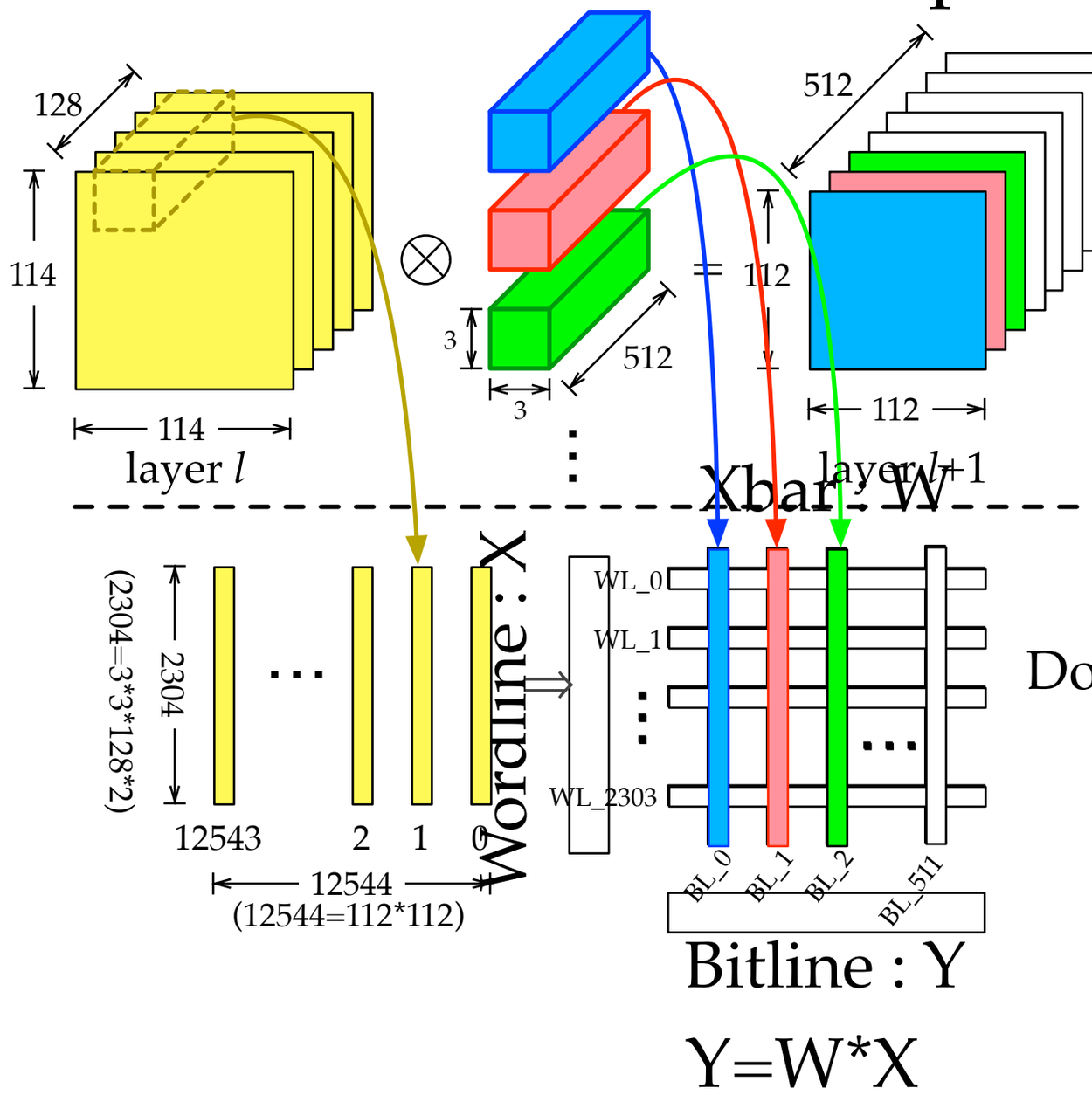
(en.wikiquote.org)

# The Need for Graph Processing Accelerators

- Graph processing algorithms:
  - Generate **random access**
  - Require **high memory bandwidth**
- Good target for hardware acceleration
  - Tesseract (**ISCA'15**): HMC+Inorder-Cores
  - Graphicionado (**MICRO'16**): dedicated memory accessing module
  - Energy Efficient Architecture for Graph (**ISCA'16**): asynchronous execution
- These accelerators are based on:
  - Vertex-centric processing model
  - Conventional CMOS technology

# ReRAM Based Acceleration

- ReRAM Xbar for Matrix-Vector Multiplication

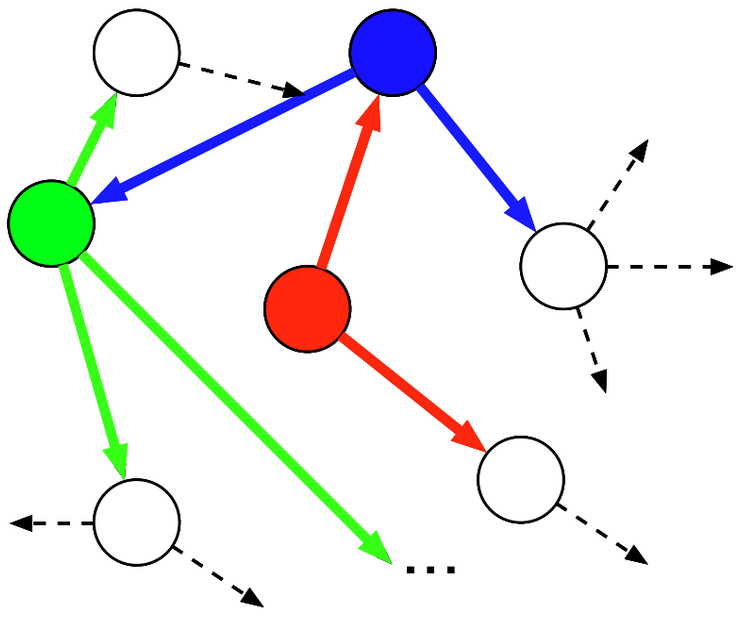



(PipeLayer **HPCA'17**)

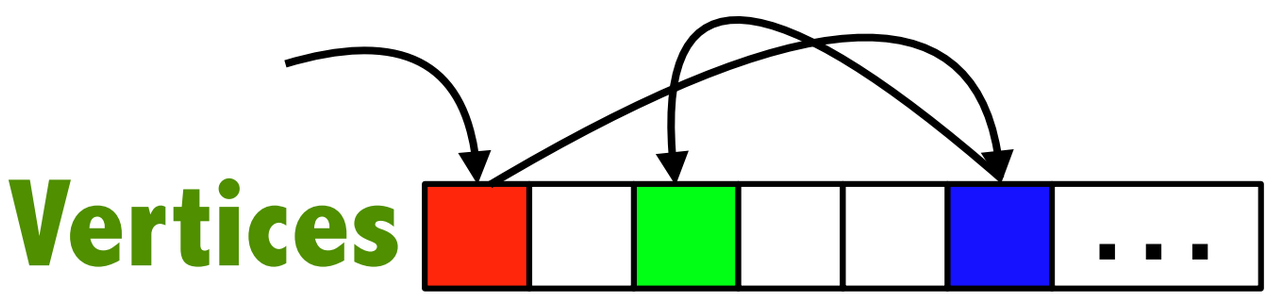
(PRIME **ISCA'16**,
ISAAC **ISCA'16**,
Dot-Product Engine **DAC'16**,
RENO **DAC'15**)

# Graph Processing in Action

- **Vertex-centric** processing model

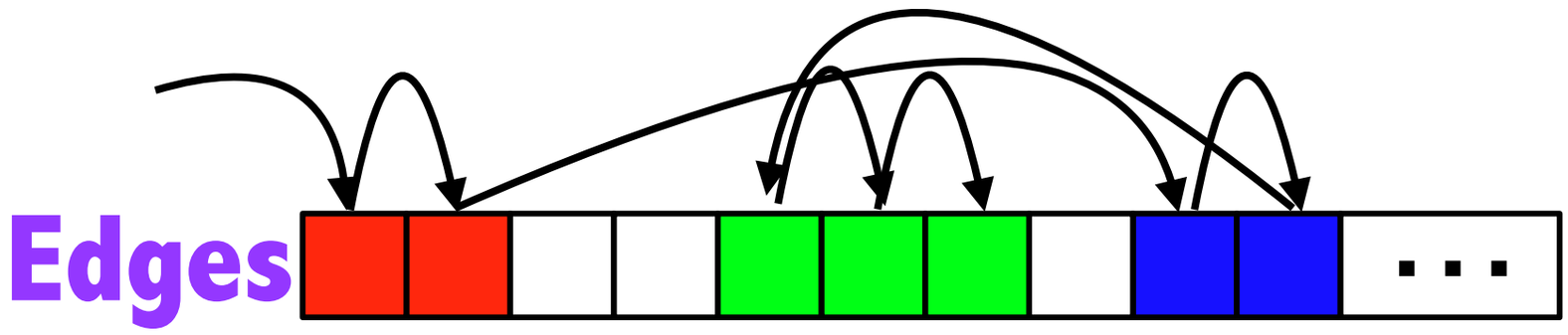


**random** access




High memory **bandwidth** — little computation on the randomly fetched data

global **random** access




CEI

cei.pratt.duke.edu

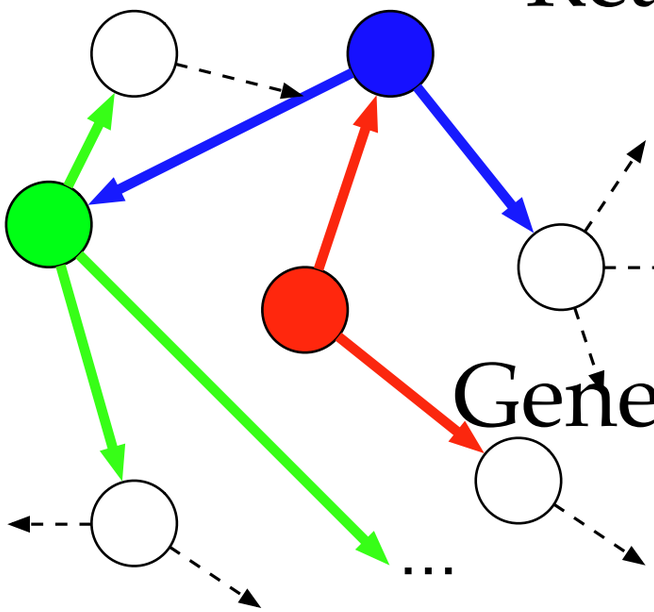
ALCHEM

alchem.usc.edu

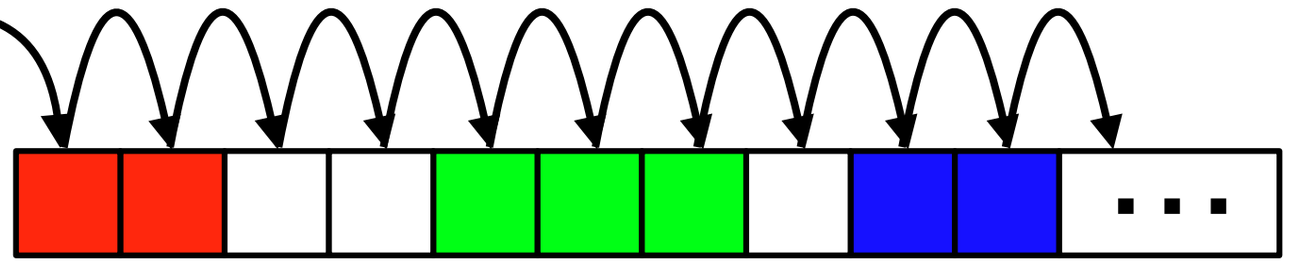
# Graph Processing in Action

- **Edge-centric** Processing Model

(X-Stream SOSP'13)



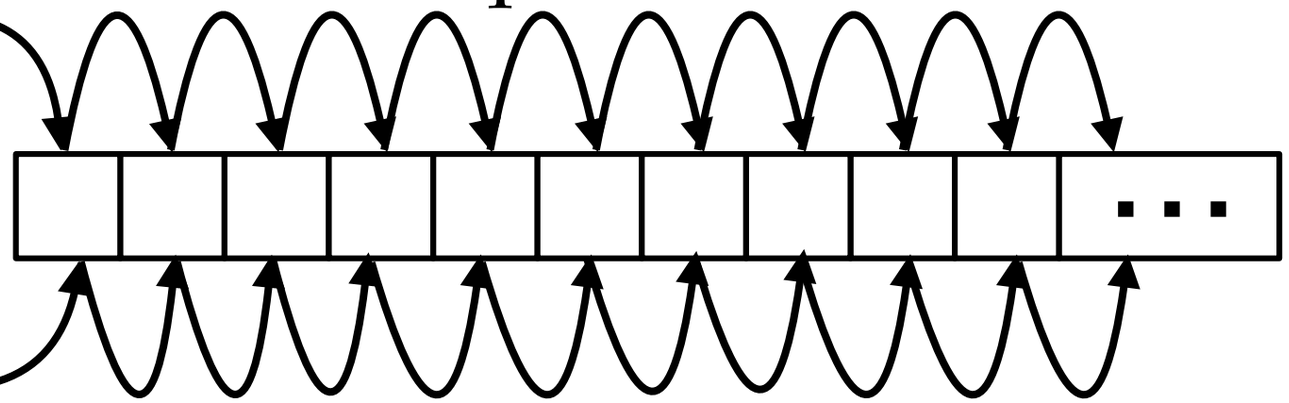

Read & Process




**Edges**

sequential write

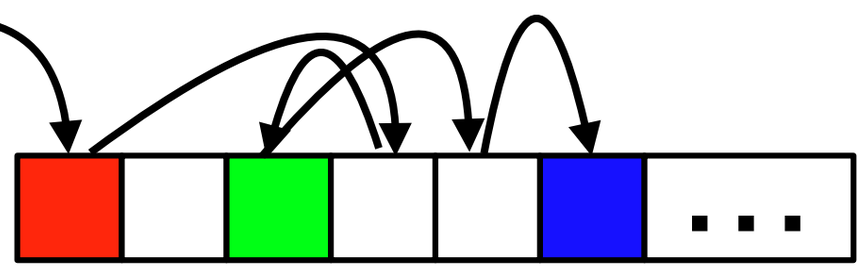
Generate Updates




sequential read

Sequential edge access.
Random vertex access.

Renew




**Vertices**

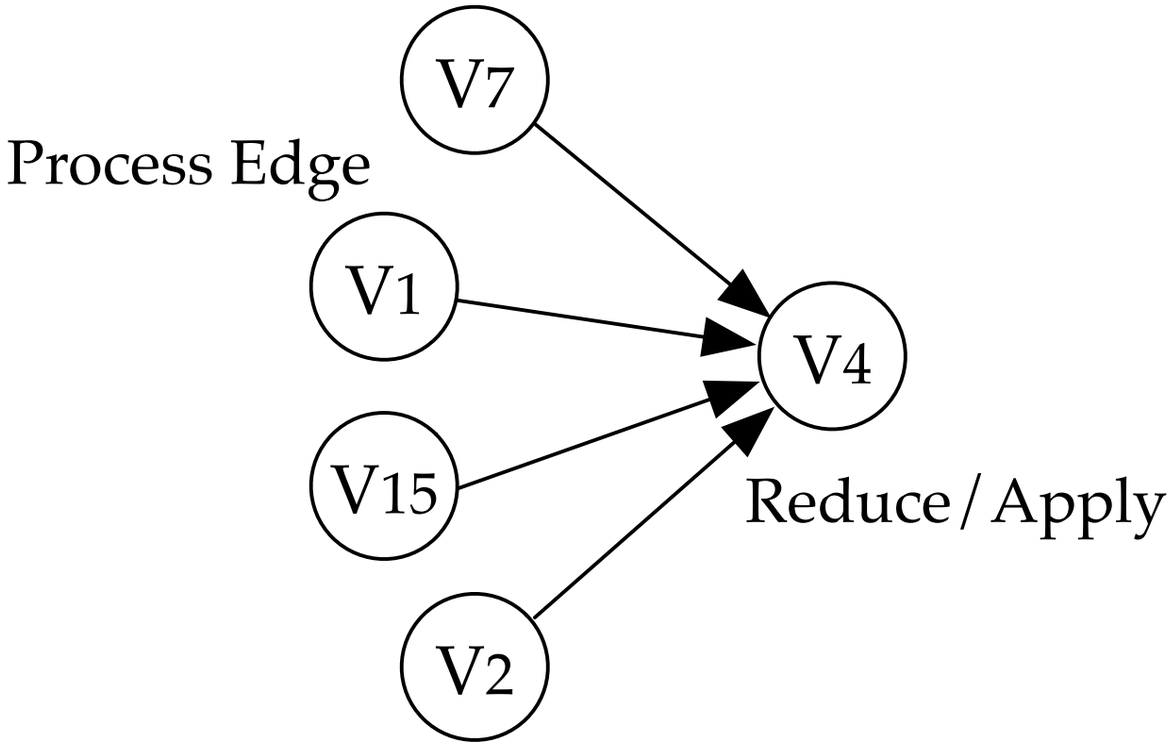
**CEI**

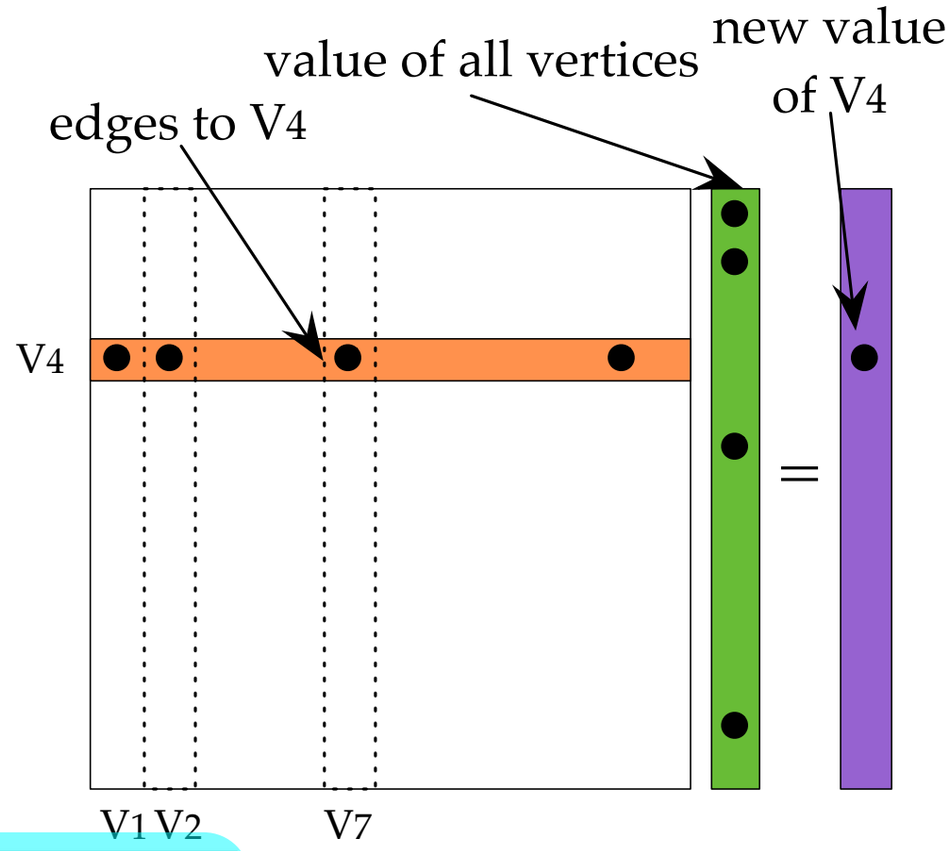
cei.pratt.duke.edu

**ALCHEM**

alchem.usc.edu

# **GraphR**: Graph Processing with ReRAM Xbar




But, WAIT!
A Xbar with
a size of V-by-V?
The matrix is **sparse**.

**ReRAM**
**Crossbar (CB)**
*perform SpMV in analog manner*

# Storage and Computation Efficien

Store:

Compressed Format

(row,col,val)

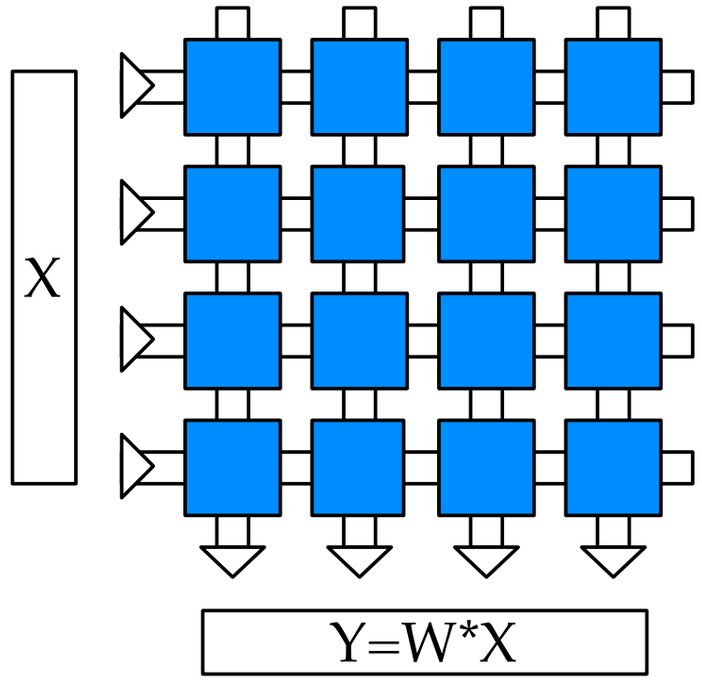
| |
|---|
| (0,2,3) |
| (0,3,8) |
| (1,2,7) |
| (2,0,1) |
| (3,1,4) |
| (3,3,2) |

Coordinate List

COO

Computation:

ReRAM Xbar SpMV




Storage Efficiency

GraphR

Computation Efficiency

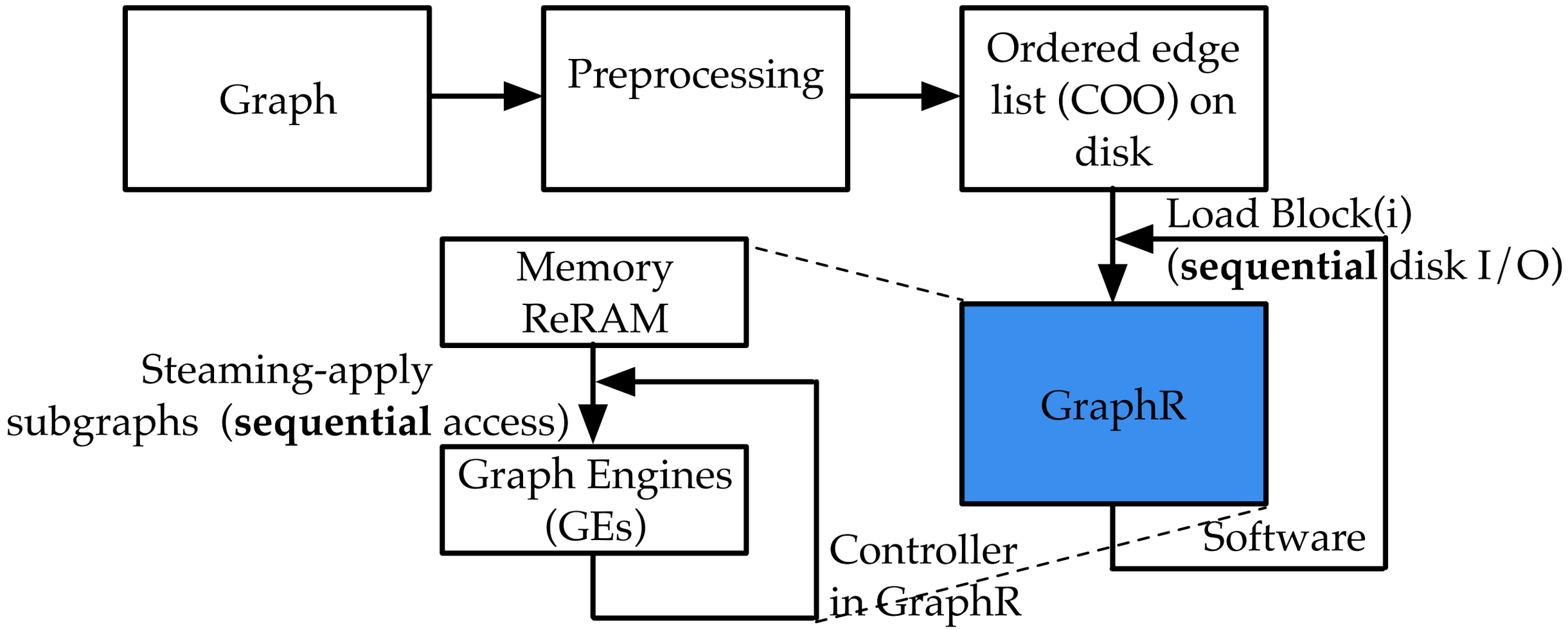
CEI

cei.pratt.duke.edu

ALCHEM

alchem.usc.edu

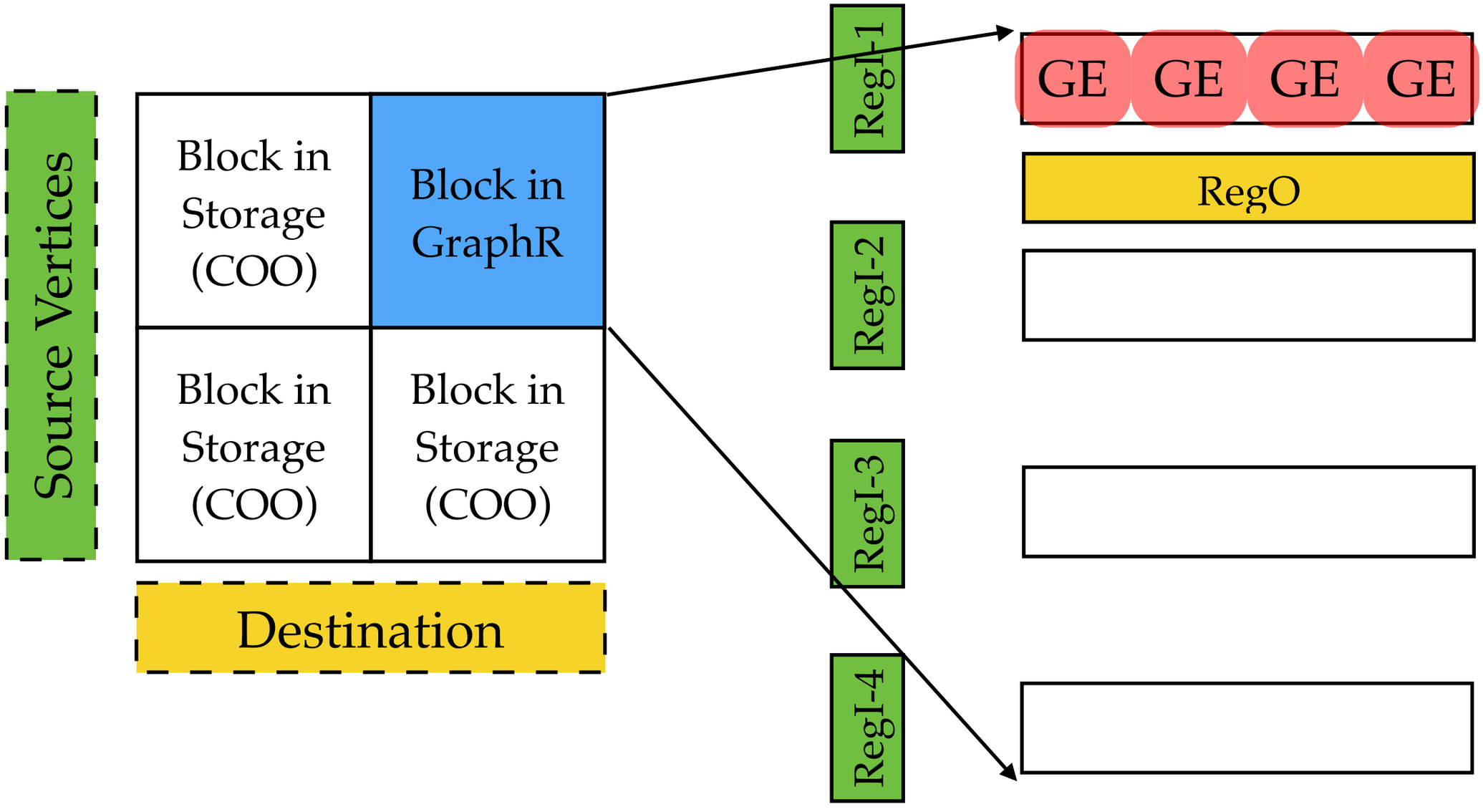
# GraphR Overview




CEI
cei.pratt.duke.edu

ALCHEM
alchem.usc.edu

# Stream-Apply Execution



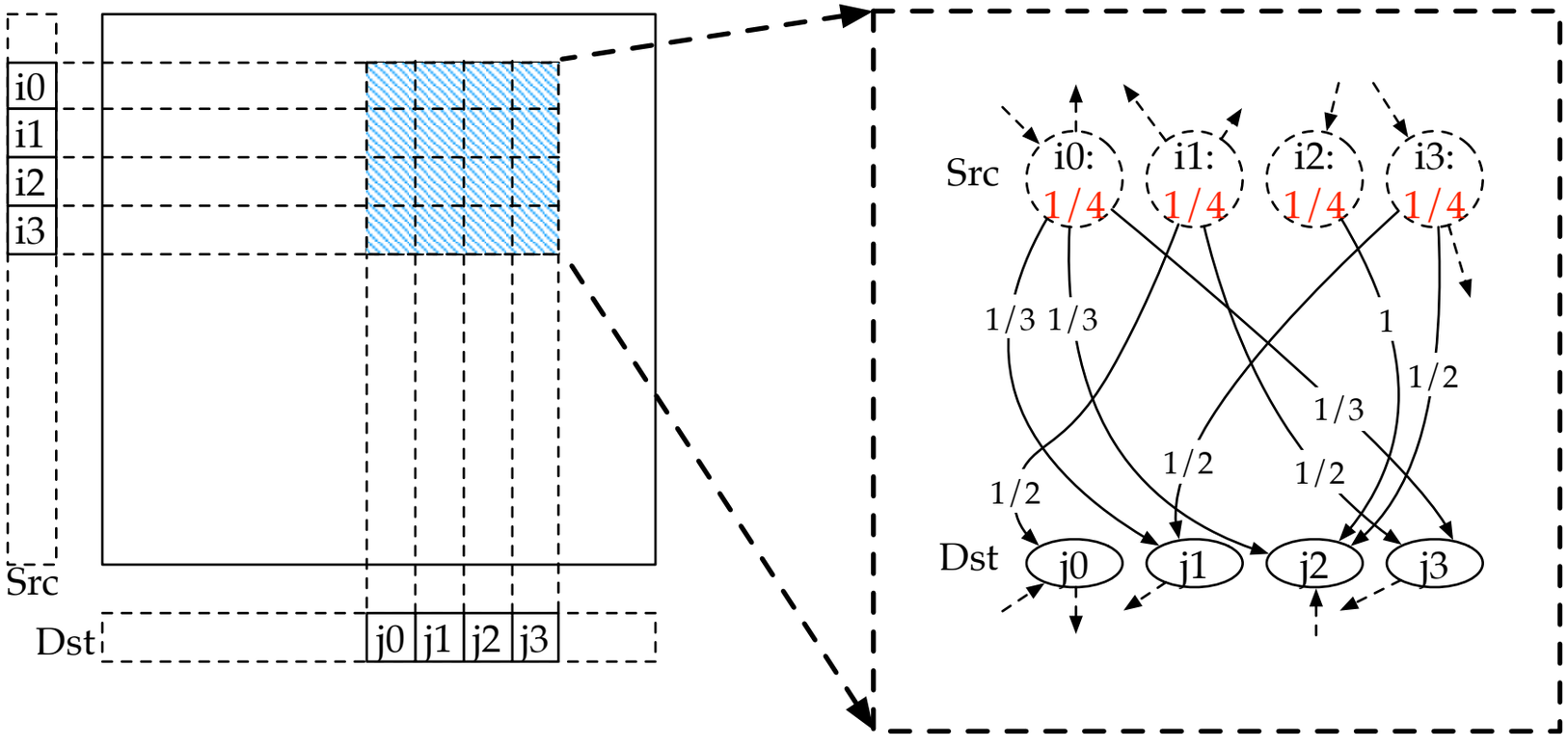
# Graph Engine (GE) Processing Patterns

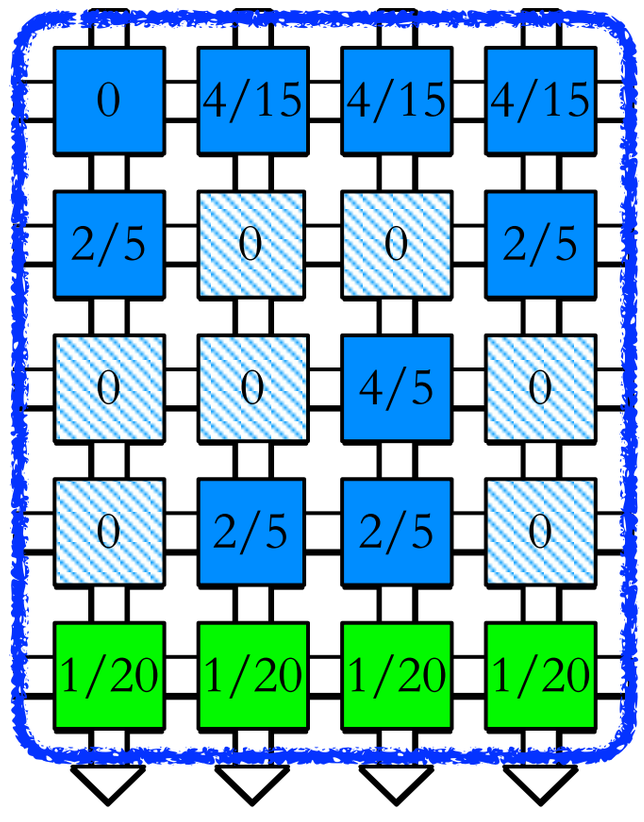
- Different algorithms achieve different parallelism when mapped to Xbars
- Assuming  an N×N Xbar
- **Parallel Multiply-Accumulate (MAC)**
  - Performing $N^2$ multiplications and $N^2$ additions in parallel
- **Parallel Add-op**
  - Performing N additions and N ops (can be defined) in parallel

# Parallel MAC

- Performing $N^2$ multiplications and $N^2$ additions in parallel

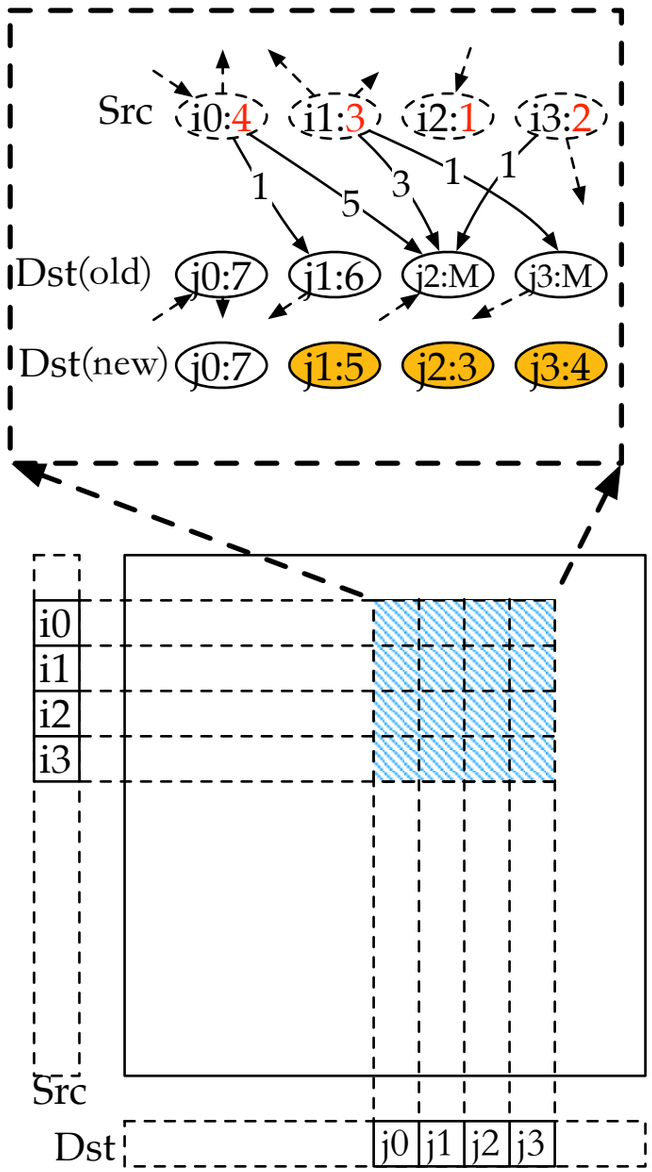
16 MULT , 16 ADD

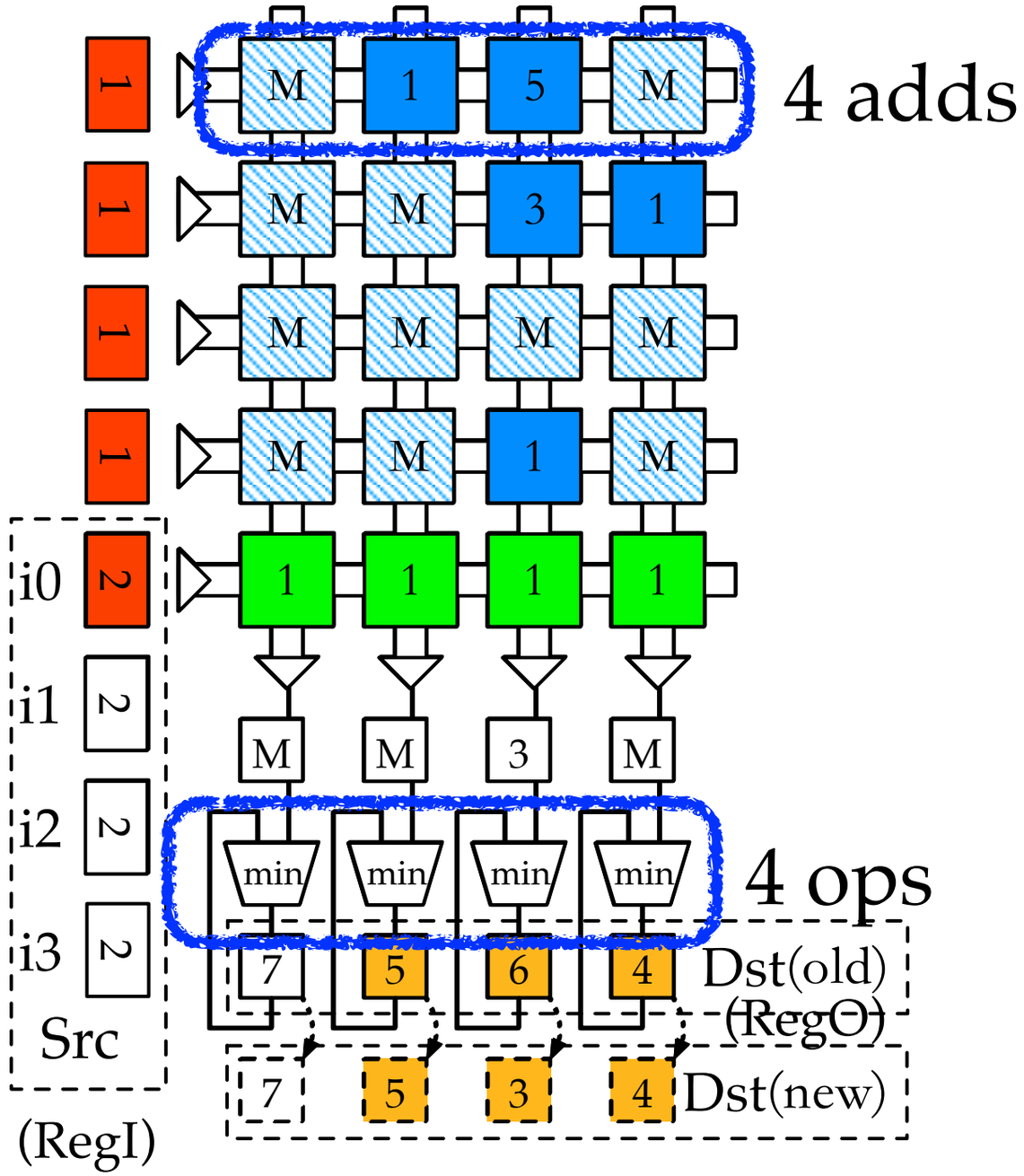


# Parallel Add-op

- Performing N additions and N ops (can be defined) in parallel




CEI

cei.pratt.duke.edu

ALCHEM

alchem.usc.edu

# Also in the paper …

- Graph dataset preprocessing method
- Hardware components in GraphR
- Detailed comparison to other accelerators (Table 1)

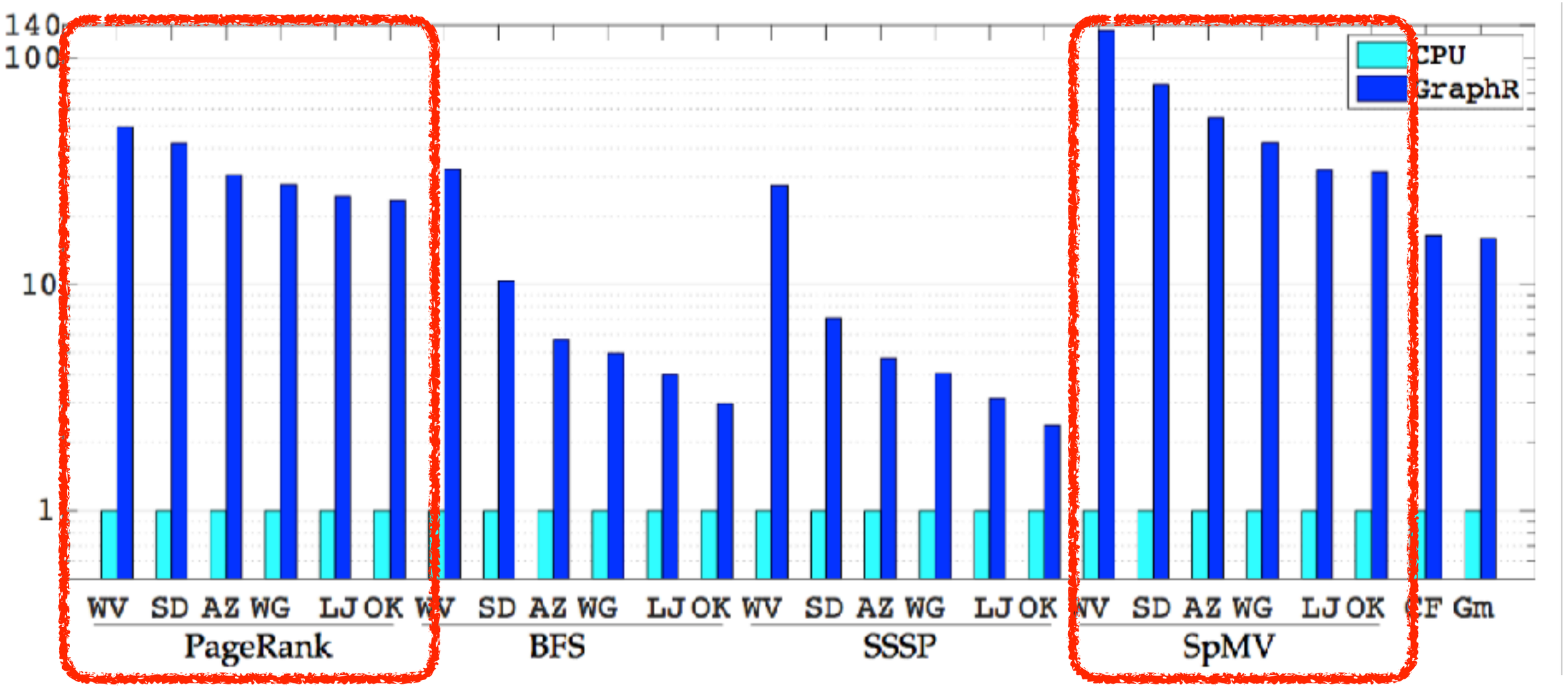
# Evaluation

- Evaluation Setup
  - Data Sets

| Dataset | # Vertices | #Edges |
|---|---|---|
| WikiVote(WV) [32] | 7.0K | 103K |
| Slashdot(SD) [32] | 82K | 948K |
| Amazon(AZ) [32] | 262K | 1.2M |
| WebGoogle(WG) [32] | 0.88M | 5.1M |
| LiveJournal(LJ) [32] | 4.8M | 69M |
| Orkut(OK) [51] | 3.0M | 106M |
| Netflix(NF) [8] | 480K users, 17.8K movies | 99M |

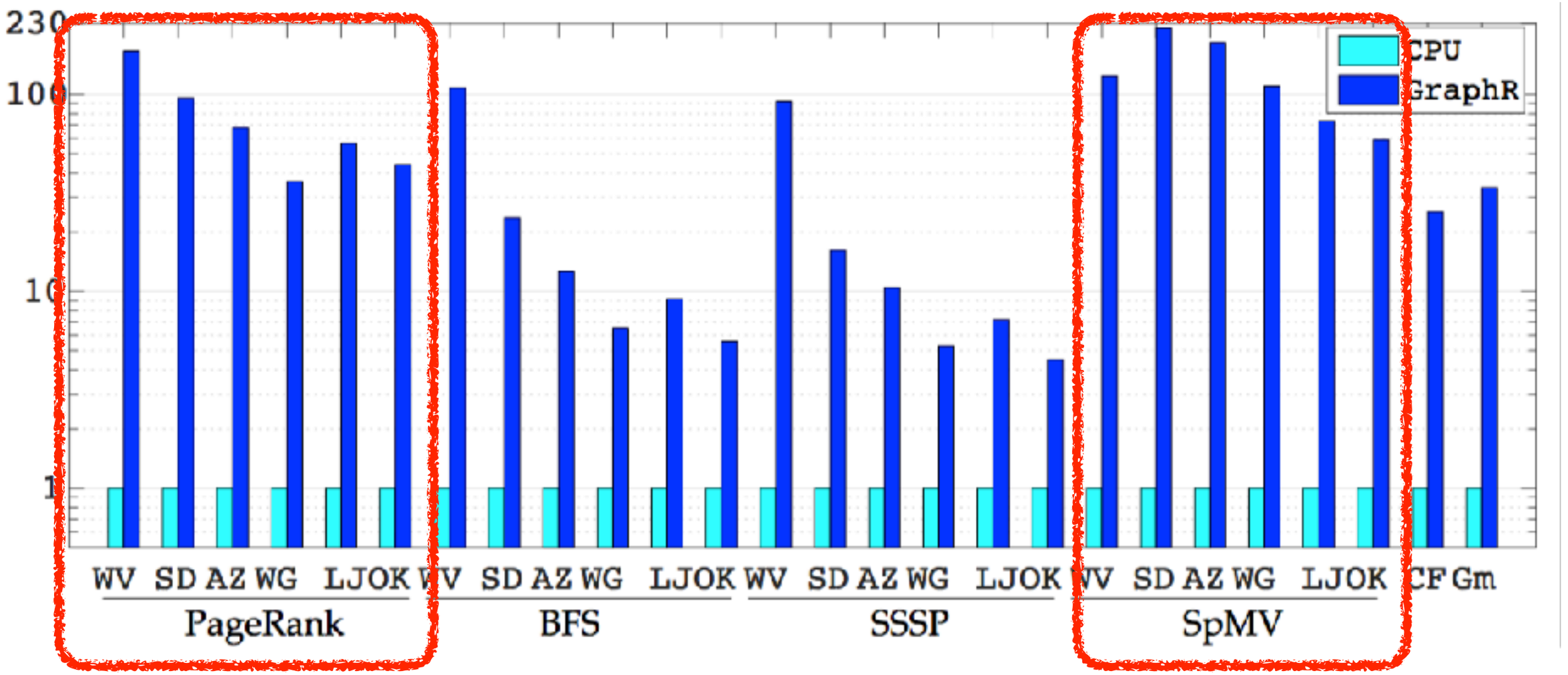
  - Applications: PageRank, BFS, SSSP, SpMV
  - CPU: Intel Xeon E5-2630 V3
  - GPU: NVIDIA Tesla K40c
  - GraphR: 8-8 Xbar, 32 Xbars/GE, 64 GEs

# CPU Comparison: Performance



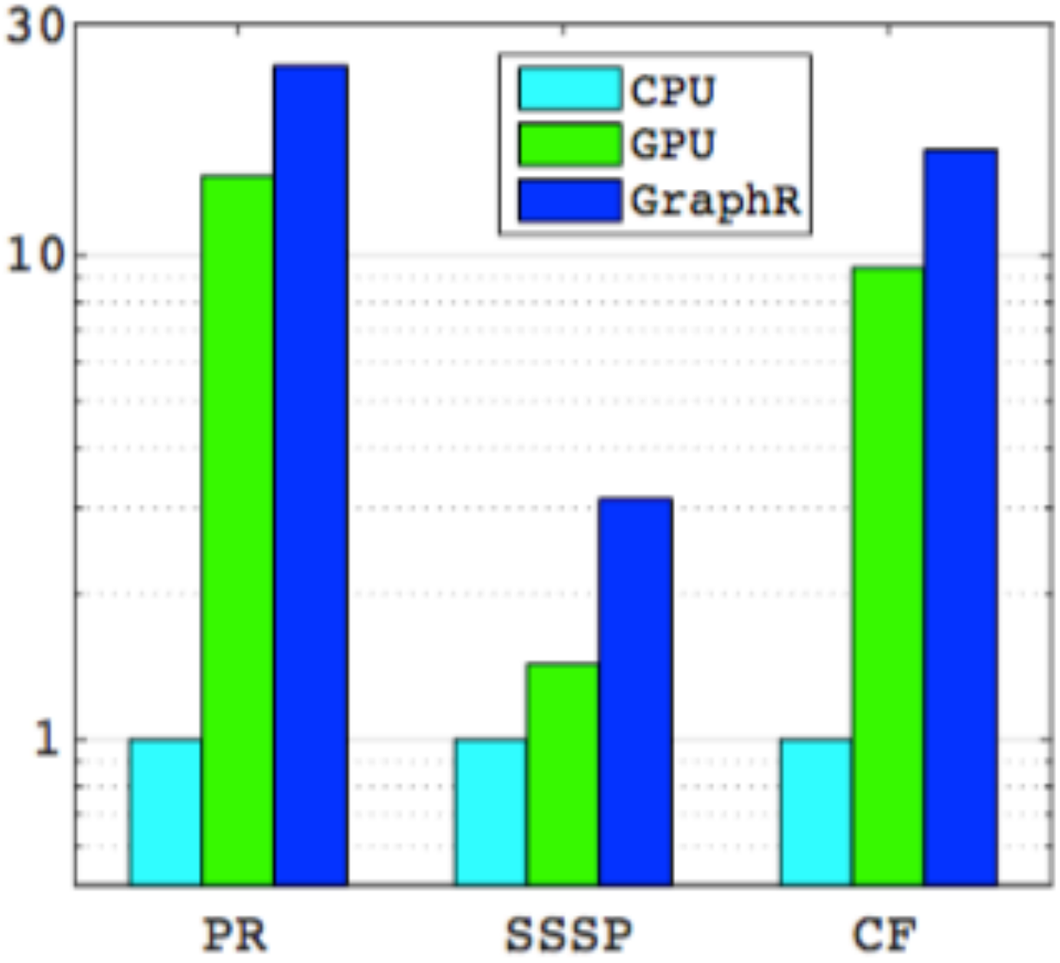

- Gmean: Performance 16.01x
- SpMV, PageRank > BFS, SSSP
  - Parallel MAC leads to higher speedup

# CPU Comparison: Energy Efficiency

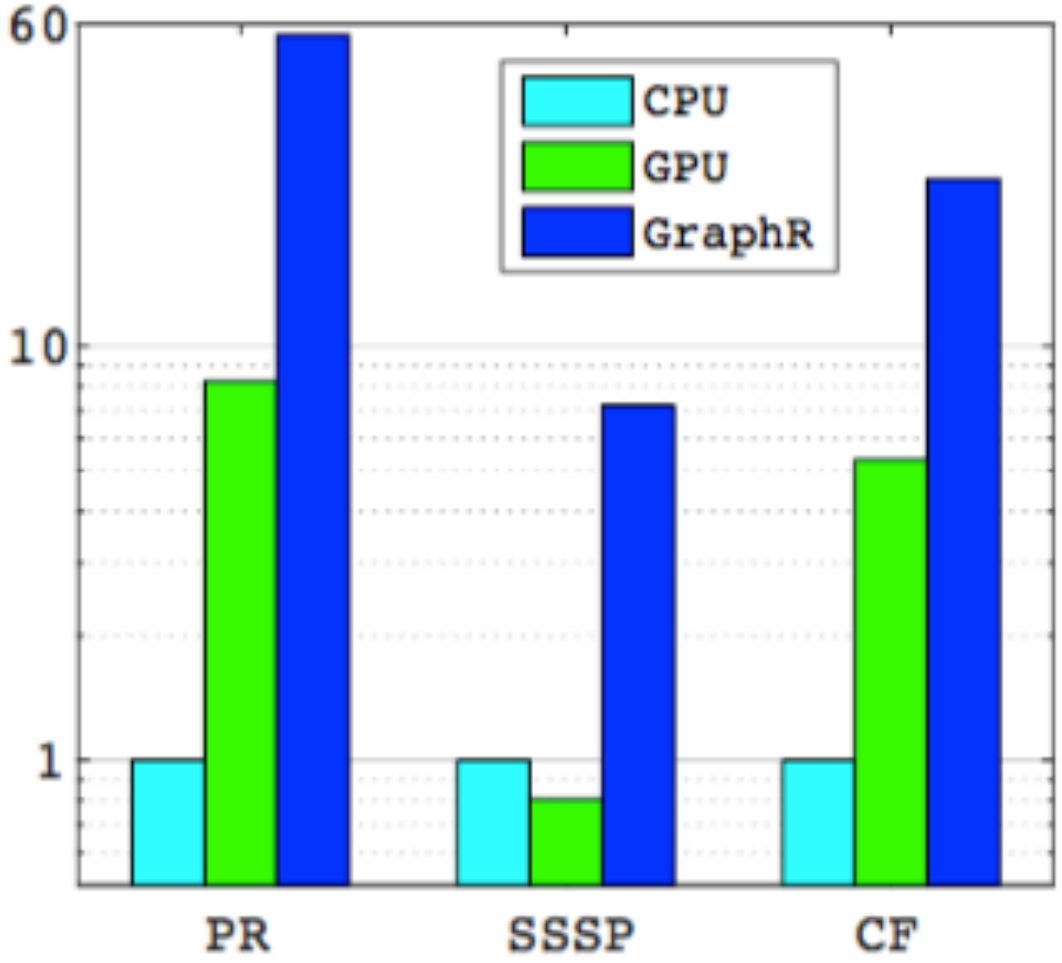



- Energy Efficiency 33.82x
- SpMV, PageRank > BFS, SSSP
  - Parallel MAC leads to higher energy efficiency

# GPU Comparison



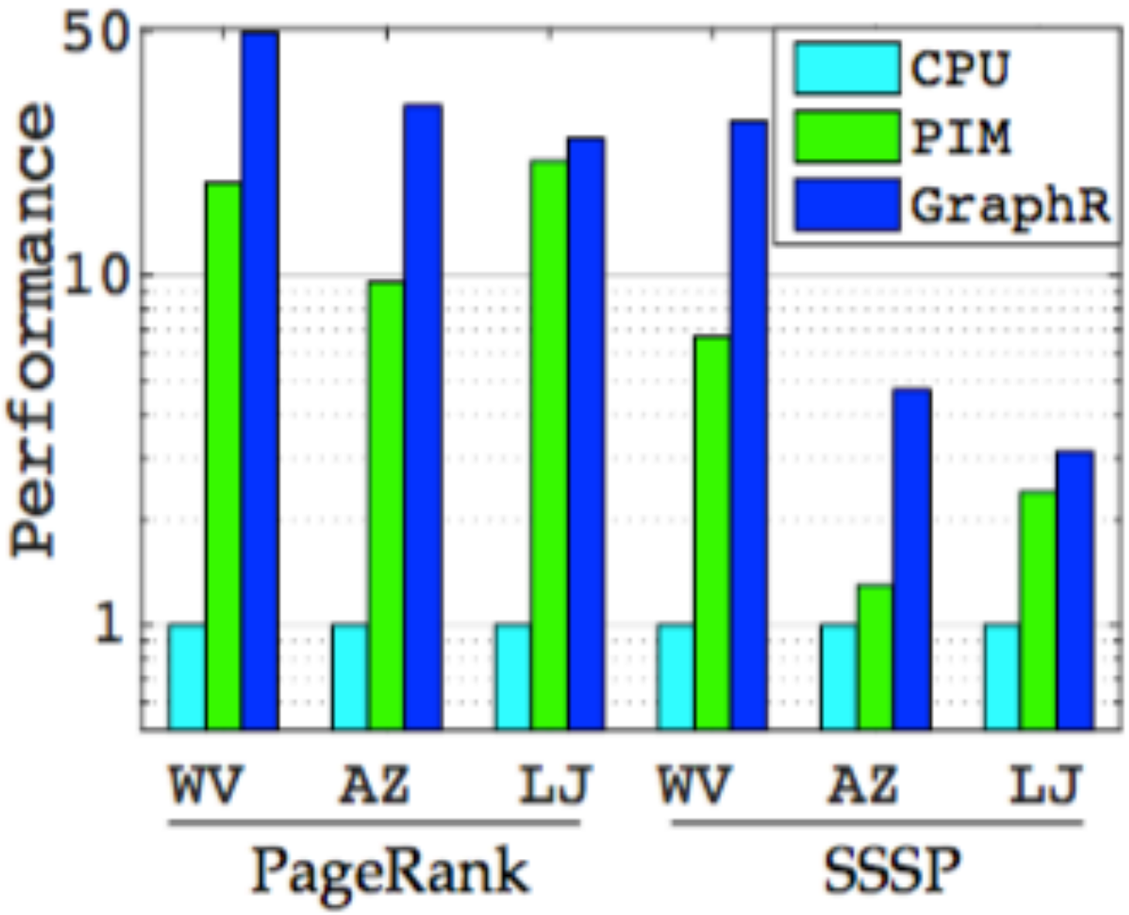

(a) Performance



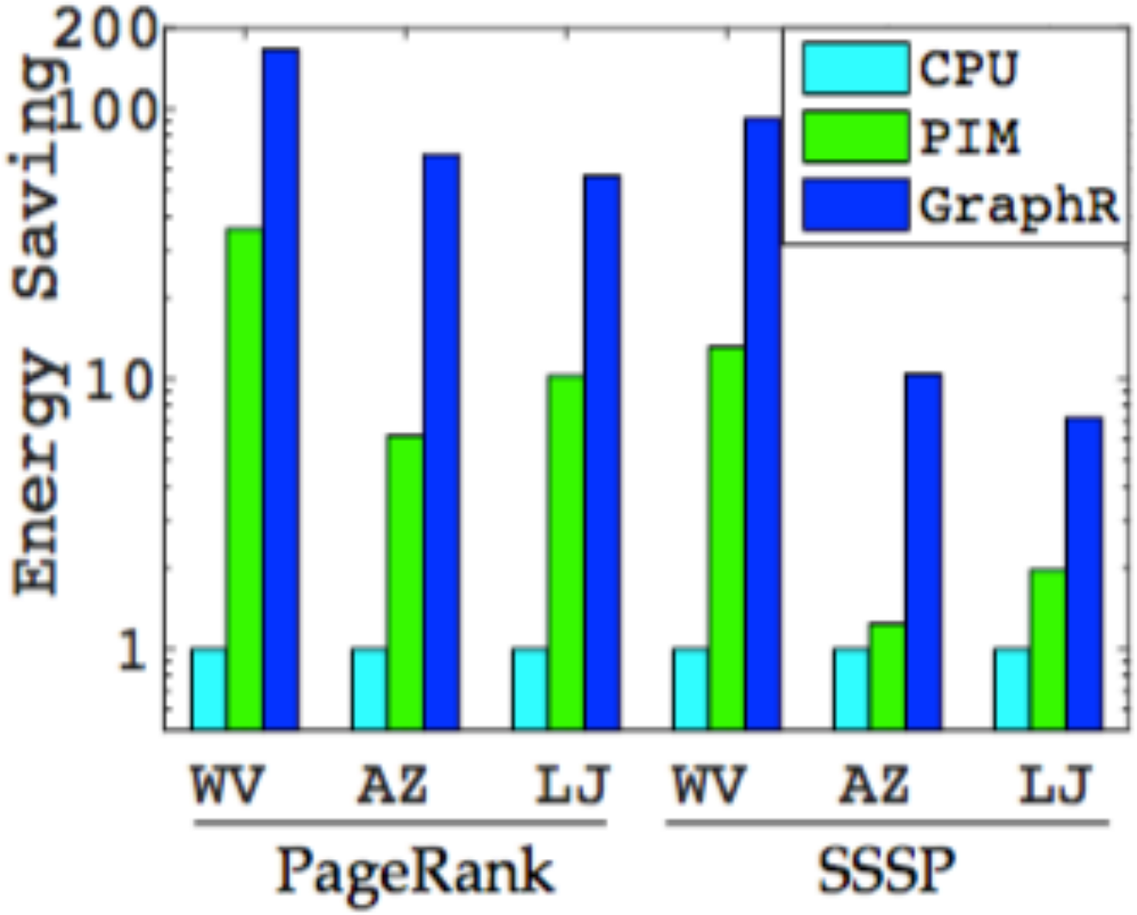

(b) Energy Saving

- Speedup: 1.69× to 2.19×
- Energy Efficiency: 4.77× to 8.91×

# Accelerator Comparison




(a) Performance




(b) Energy Saving

- Speedup: 1.16× to 4.12×
- Energy Efficiency: 3.67× to 10.96×

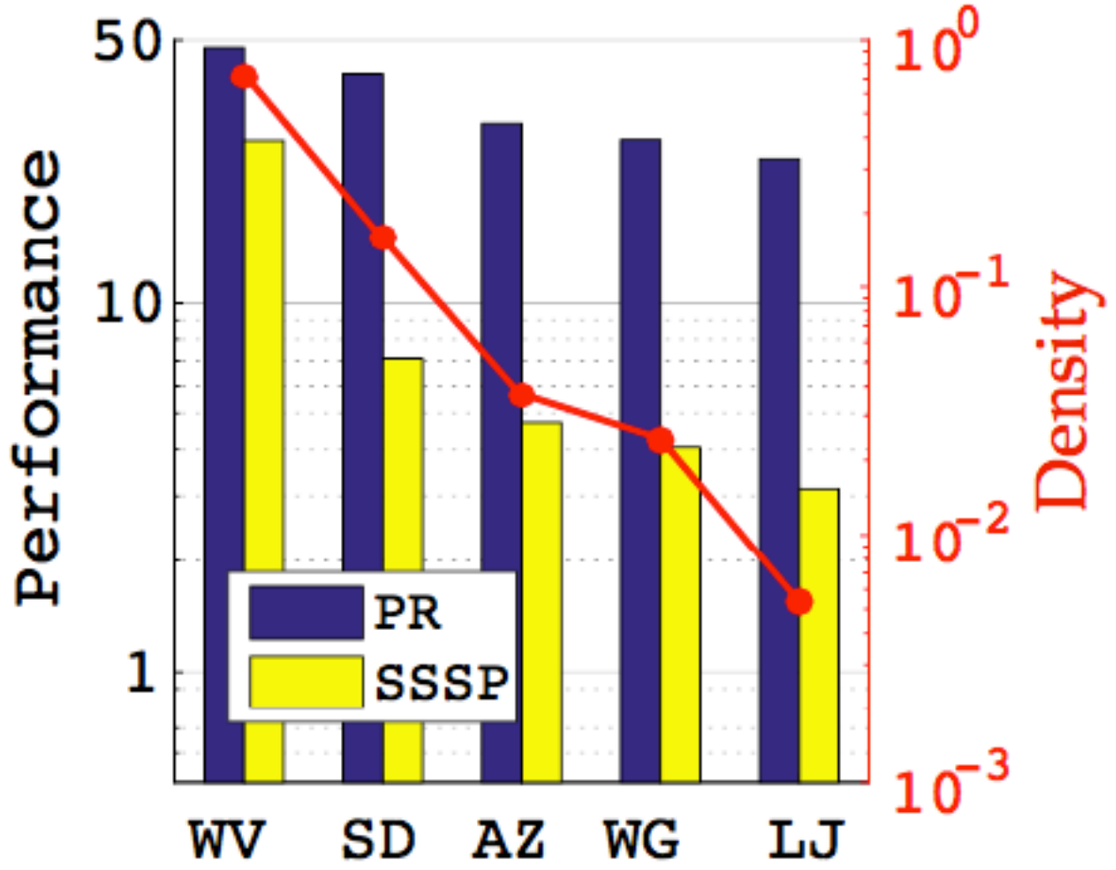
CEI

cei.pratt.duke.edu

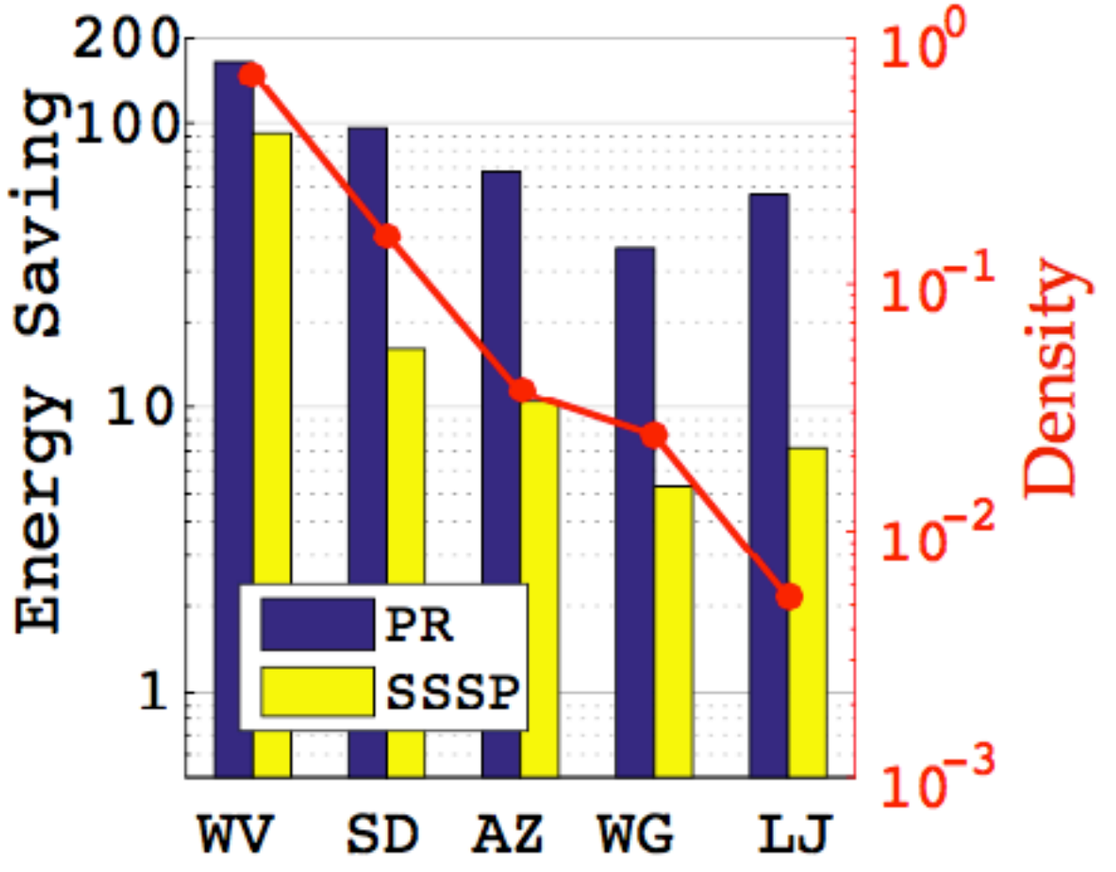
ALCHEM

alchem.usc.edu

# Sensitivity to Density




(a) Performance




(b) Energy Saving

- Density↑ -> Speedup & Energy Efficiency↑
  - Achieving greater parallelism

# Conclusion

- We propose **GraphR**:
  - A graph processing accelerator based on ReRAM
- Key Insights/Results:
  - ReRAM based SpMV for processing in graph engine
  - Stream-apply execution
  - Parallel MAC and Add-Op patterns
  - 16.01x performance gain and 33.82 in energy efficiency

# GraphR: Accelerating Graph Processing Using ReRAM

Linghao Song*, Youwei Zhuo#,
Xuehai Qian#, Hai Li*, Yiran Chen*

*Duke University
#University of Southern California

CEI
cei.pratt.duke.edu



ALCHEM
alchem.usc.edu