CS 758: Advanced Topics in Computer Architecture

Lecture #15: Sparsity & Pruning

Professor Matthew D. Sinclair

Some of these slides were developed by Tushar Krishna at Georgia Tech
Slides enhanced by Matt Sinclair

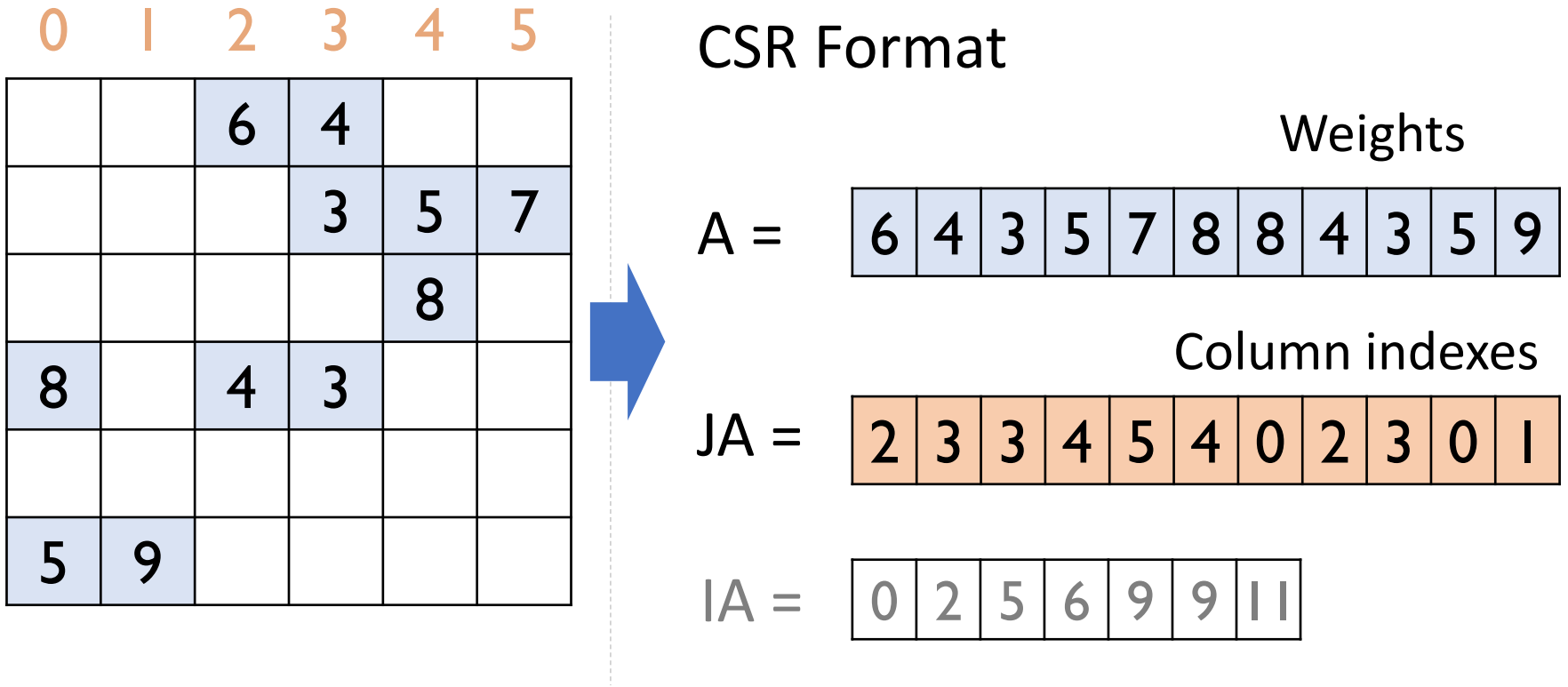
Announcements

- Project Progress Report due next Tuesday at 9 AM
- Updated course schedule

Is Pruning Always a Good Idea?

Drawbacks of Pruning

- Sparse format needs extra storage

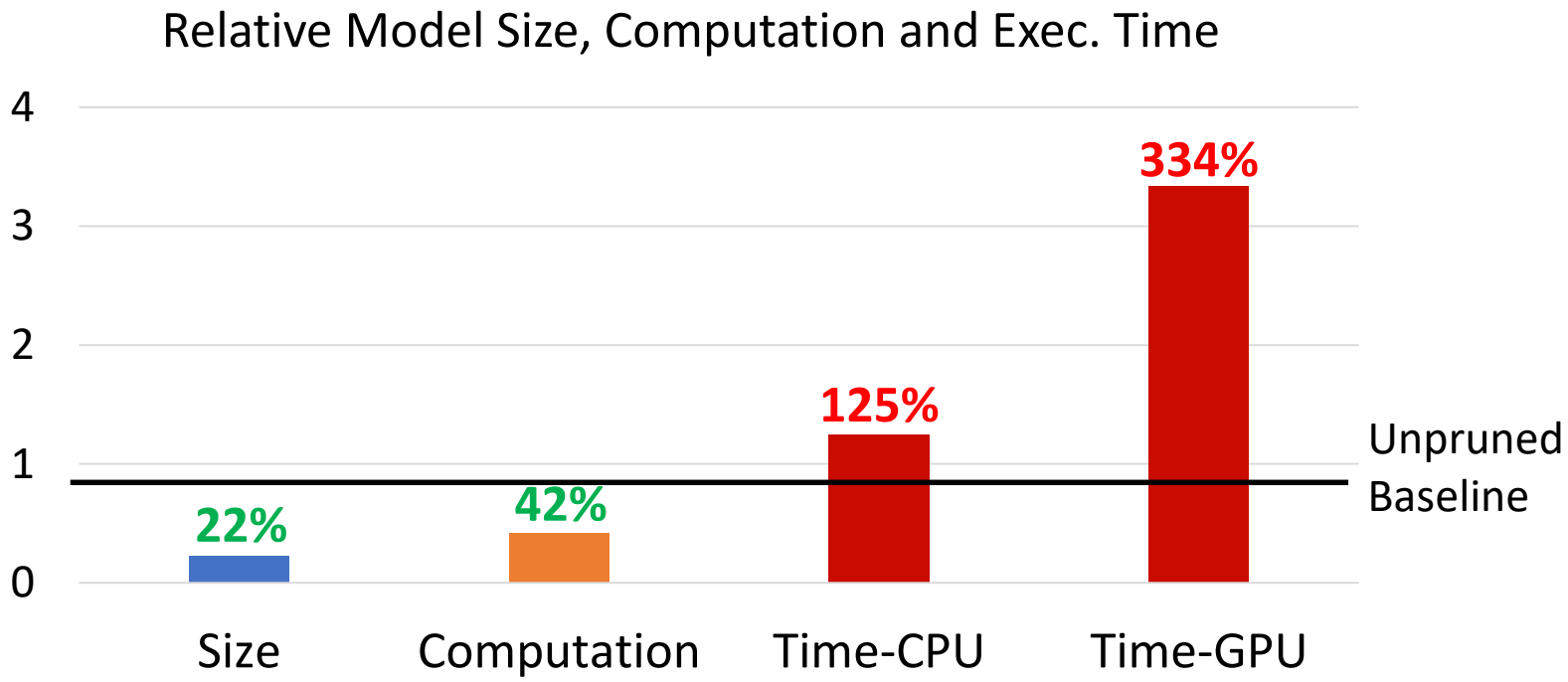


- One column index for each weight

Yu et al., "Scalpel: Customizing DNN Pruning to the Underlying Hardware Parallelism", ISCA 2017

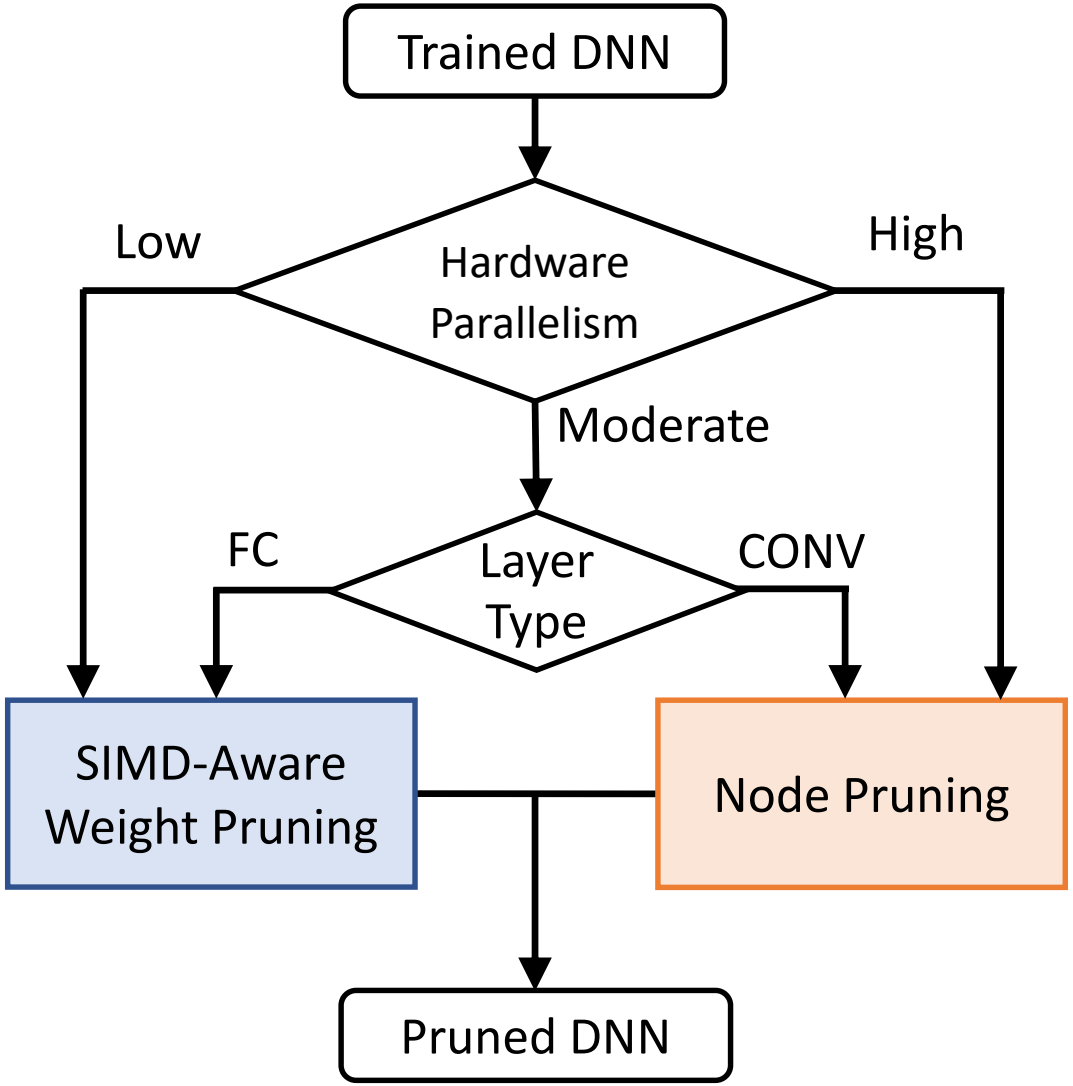
Drawbacks of Pruning

- Execution time increase
  - Computation reduction not fully utilized
  - Extra computation for decoding sparse format
- AlexNet *



Yu et al., "Scalpel: Customizing DNN Pruning to the Underlying Hardware Parallelism", ISCA 2017

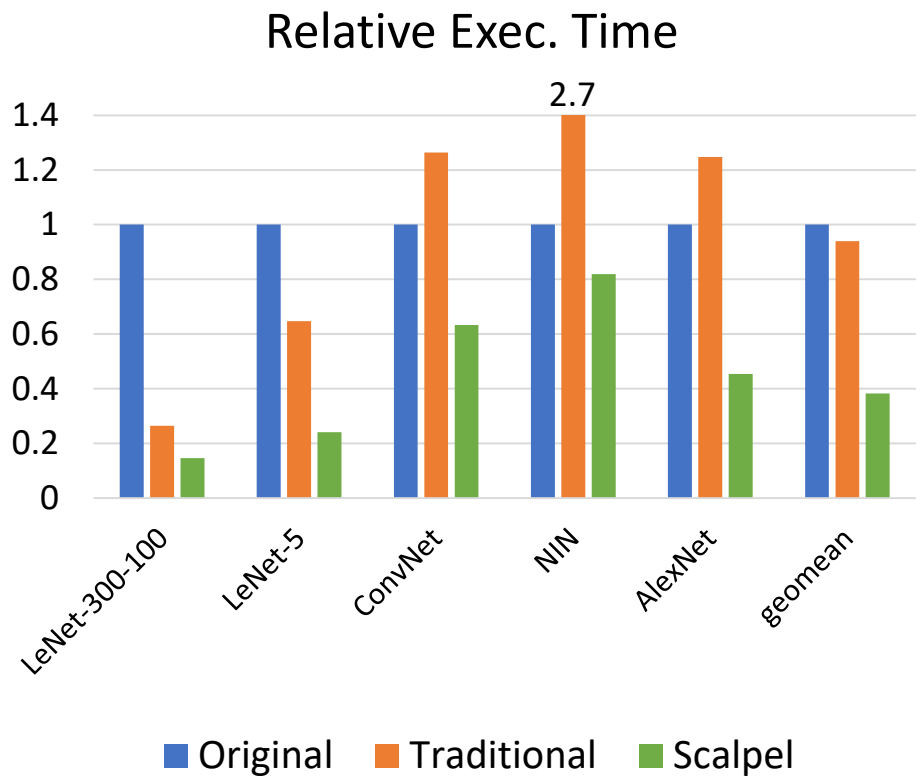
Scalpel [ISCA '17]

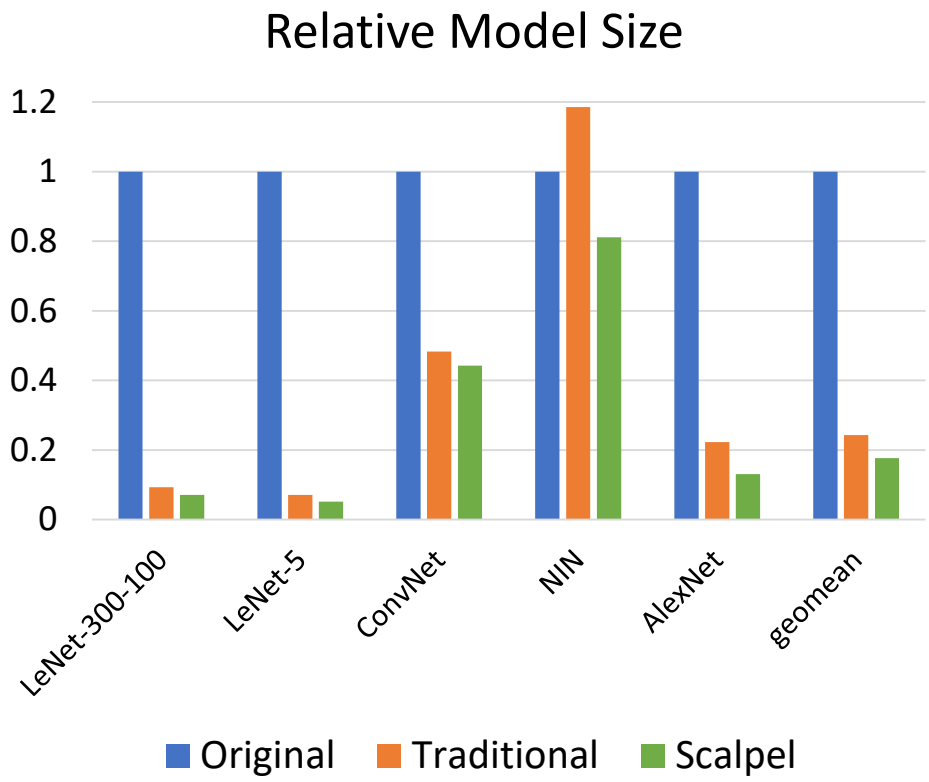


- Low parallelism - Micro.
  - No cache
  - Low storage (~100 KB)
- High parallelism - GPU
  - TLP
  - High bandwidth / long latency memory
- Moderate parallelism - CPU
  - ILP / MLP

Yu et al., "Scalpel: Customizing DNN Pruning to the Underlying Hardware Parallelism", ISCA 2017

Scalpel Results: Intel Core i7-6700 CPU





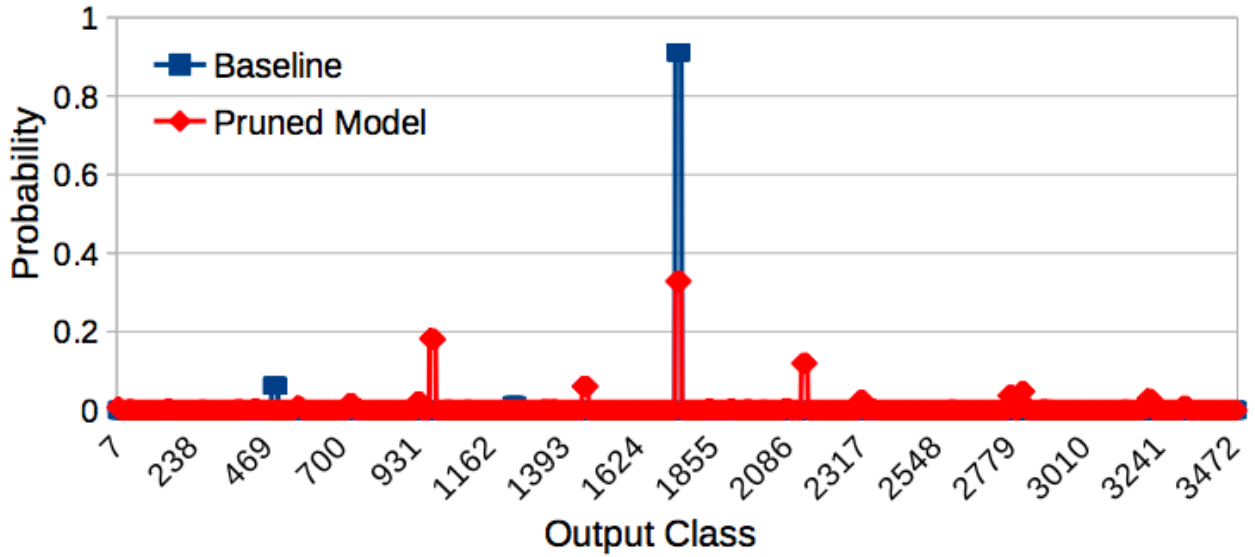
- 38% execution time
- 18% model size

Yu et al., "Scalpel: Customizing DNN Pruning to the Underlying Hardware Parallelism", ISCA 2017

The Dark Side of DNN Pruning [ISCA '18]

Side-Effect of DNN Pruning

- Lack of confidence in DNN classification
  - Speech network of acoustic modeling



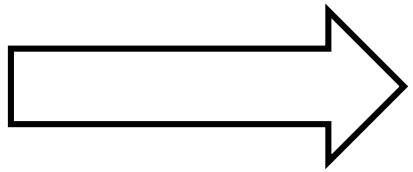
Yazdani et al., "The Dark Side of DNN Pruning", ISCA 2018

Compressed Data Formats

Bitmap

30	50	0	6
4	56	9	0
0	78	0	98
4	0	9	0

Matrix



1	1	0	1
1	1	1	0
0	1	0	1
1	0	1	0

Bitmap

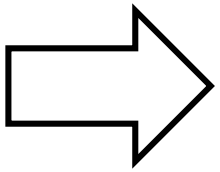
Pros
- Simplicity
- Manipulation requires bitwise operations

Cons
- Overhead is significant for low sparsity

Step

30	50	0	6
4	56	9	0
0	78	0	98
4	0	9	0

Matrix



30	50	6	4	56	9	78	98	4	9
1	2	1	1	1	3	2	1	2	

Step

Pros
- Compact representation

Cons
- Increased decoding complexity

Why do we care? Speedup due to sparsity:

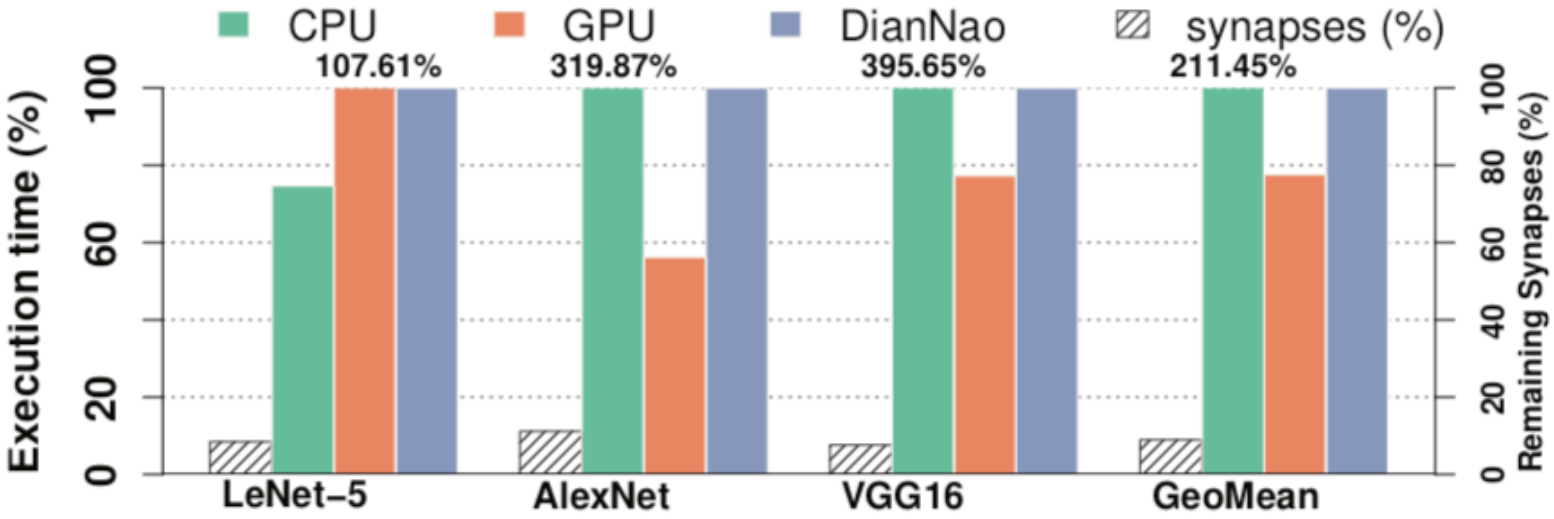


Fig. 3. The speedup of sparse NN vs. dense NN on CPU, GPU and DianNao.

- Conventional hardware, CPUs and GPUs cannot take advantage of sparsity.
- CPU slows **2x slowdown** on an average!!
- Dense accelerators do not offer any benefit.

Overall Cambricon-X Architecture

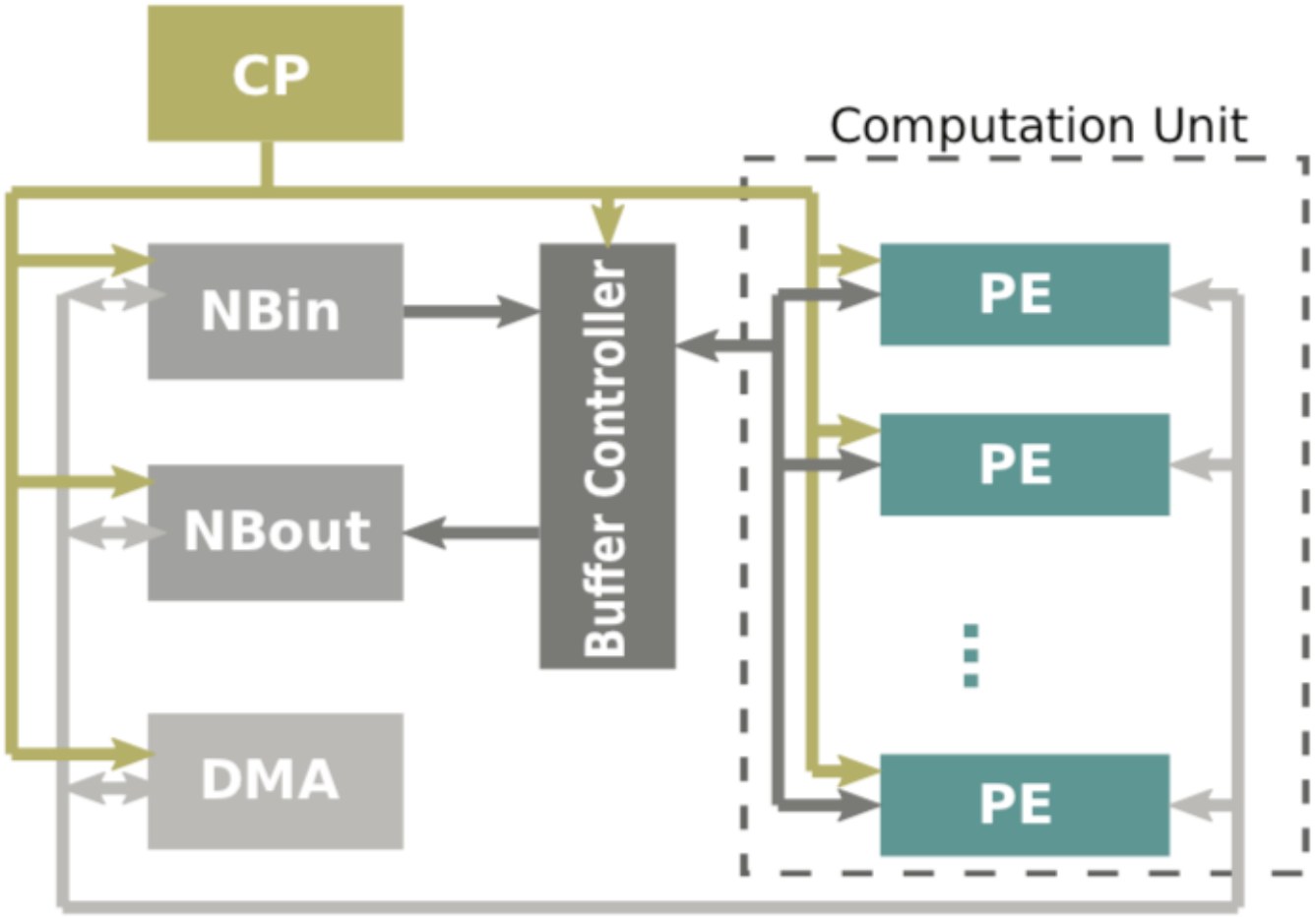


Fig. 4. Accelerator architecture.

Buffer Control

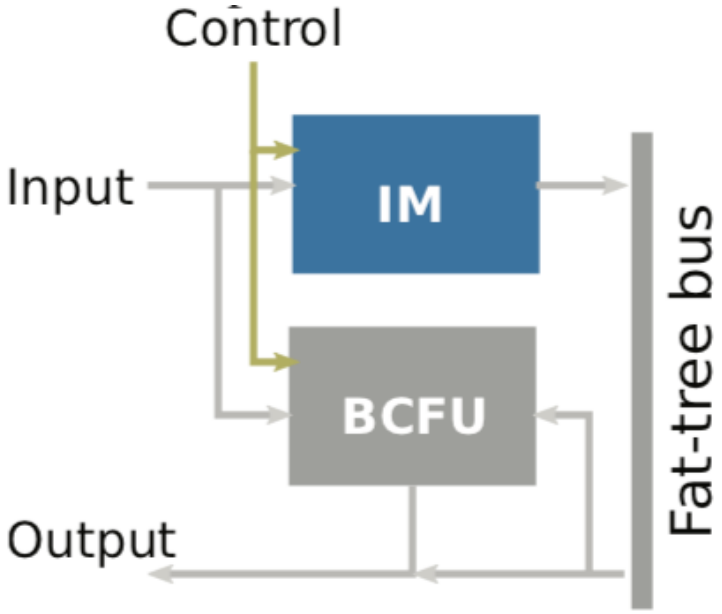


Fig. 8. The architecture of the buffer controller.

Data orchestration is critical
- Avoid reading zeros
- Pack operands to improve utilization

Indexing module (IM)
- Select inputs and pack

Buffer Control Functional Unit (BCFU)
- Additional processing

Buffer Controller

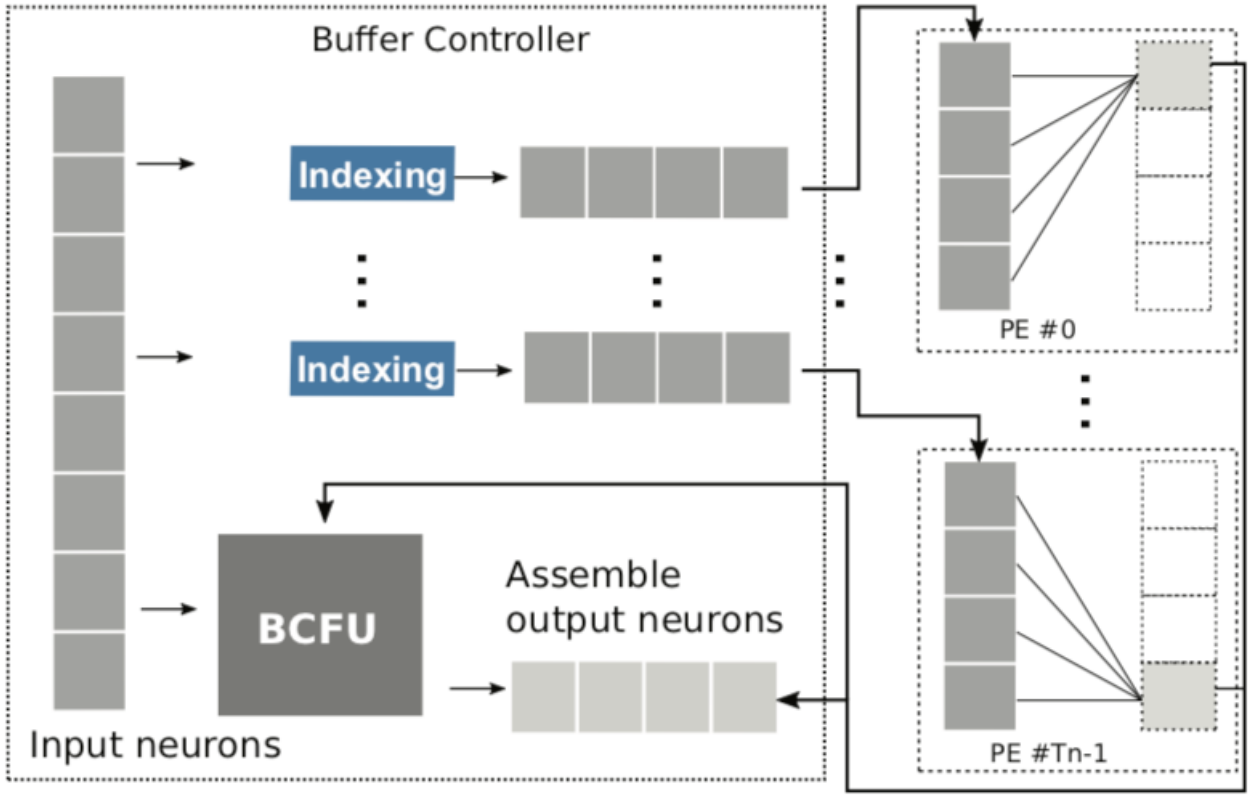


Fig. 5. Buffer controller architecture.

Separate buffers for each PE

Each buffer is independently indexed

Cambricon-X PE

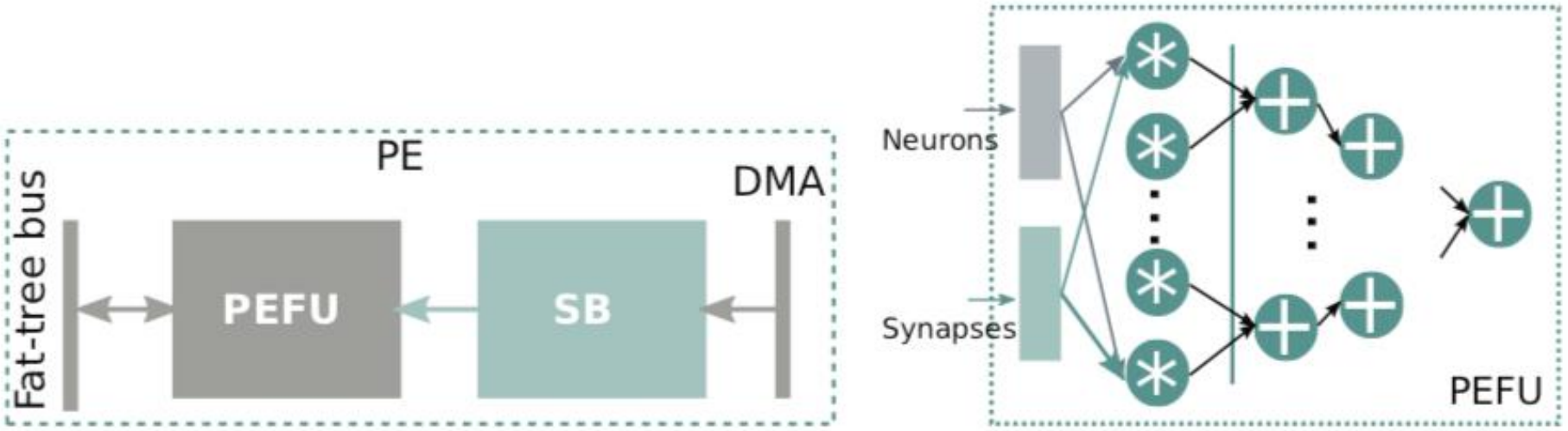


Fig. 6. (a) The architecture of the PE. (b) The architecture of the PEFU.

Speedup: Sparse over Dense

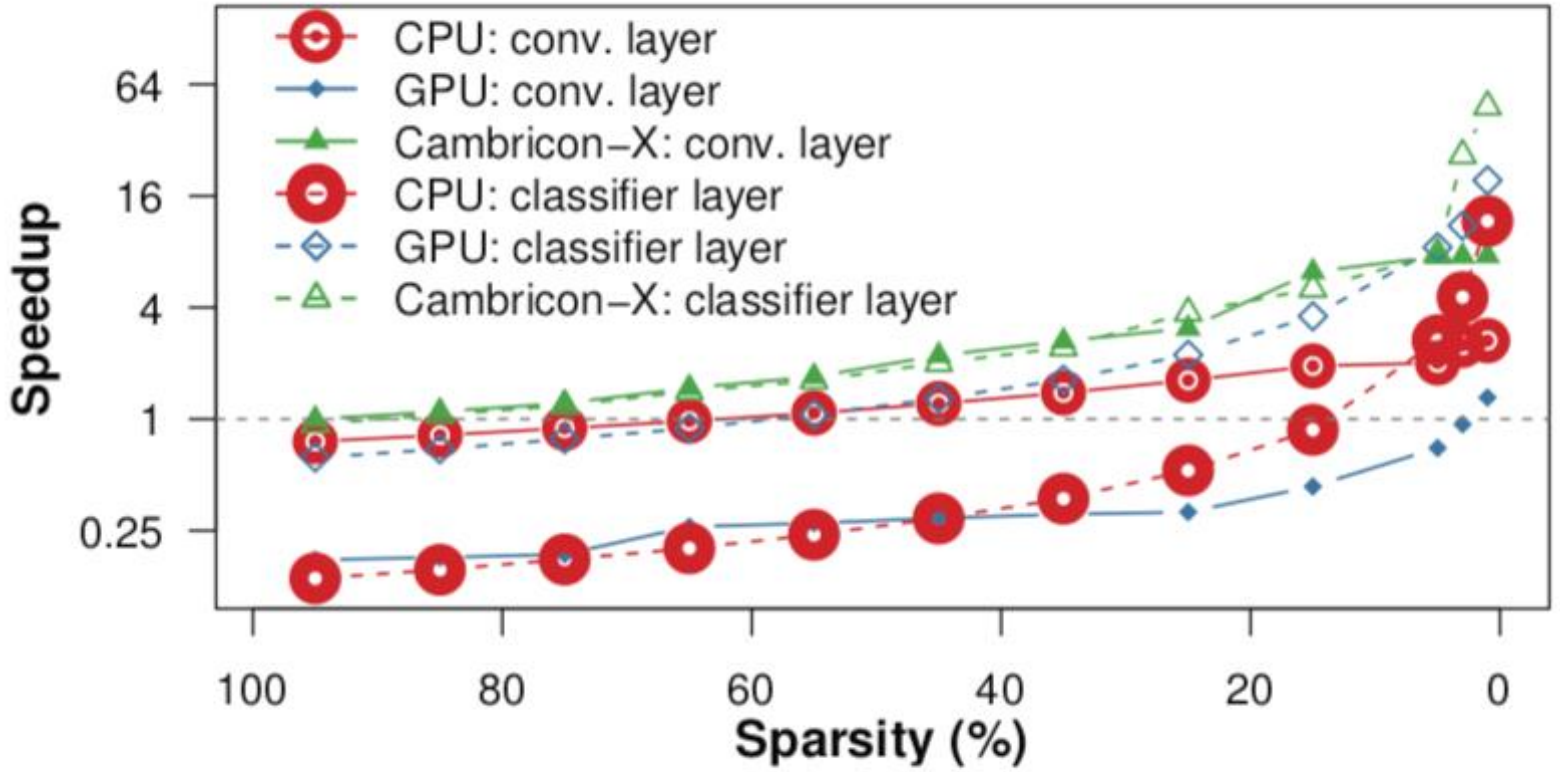


Fig. 19. Speedup of sparse layer over dense layer.

BACKUP

Breakdown of energy

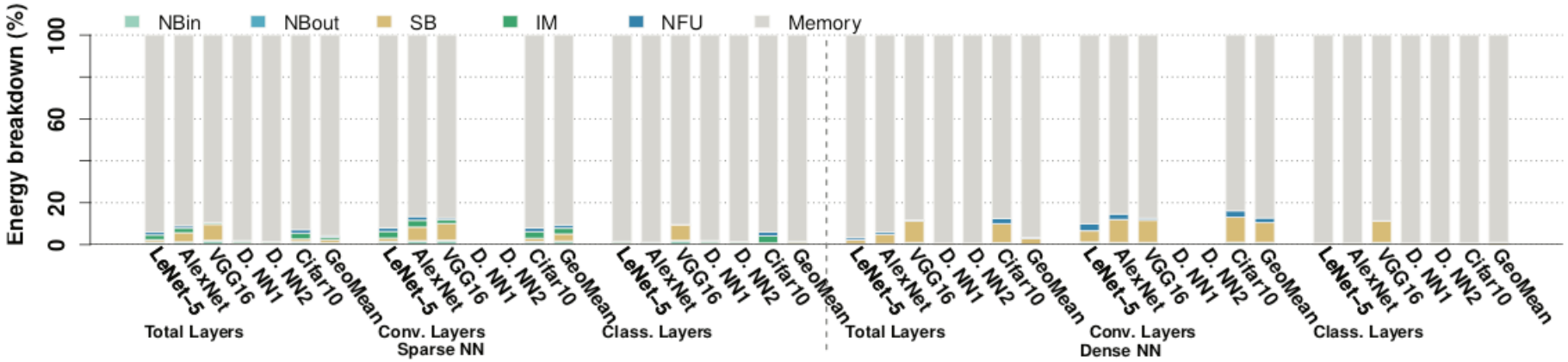


Fig. 18. Energy breakdown with memory accesses.