Cambricon-D: Full-Network Differential Acceleration for Diffusion Models

Group 18
Athul John Kurian
Avinash Singh
Shubham Kumar
Shubham Santosh Kumar

BACKGROUND

- Diffusion models have become increasingly important in image generation tasks
- Common challenges faced – computational redundancy and inefficient hardware usage
- Cambricon-D addresses these issues using convolution and ReLU operators

KEY INNOVATIONS

- Sign-mask dataflow
- Outlier-aware PE design

IMPLEMENTATION AND CONTRIBUTIONS

- A simplified Cambricon-D model focusing on convolution and ReLU operators.
- A baseline model for comparison.

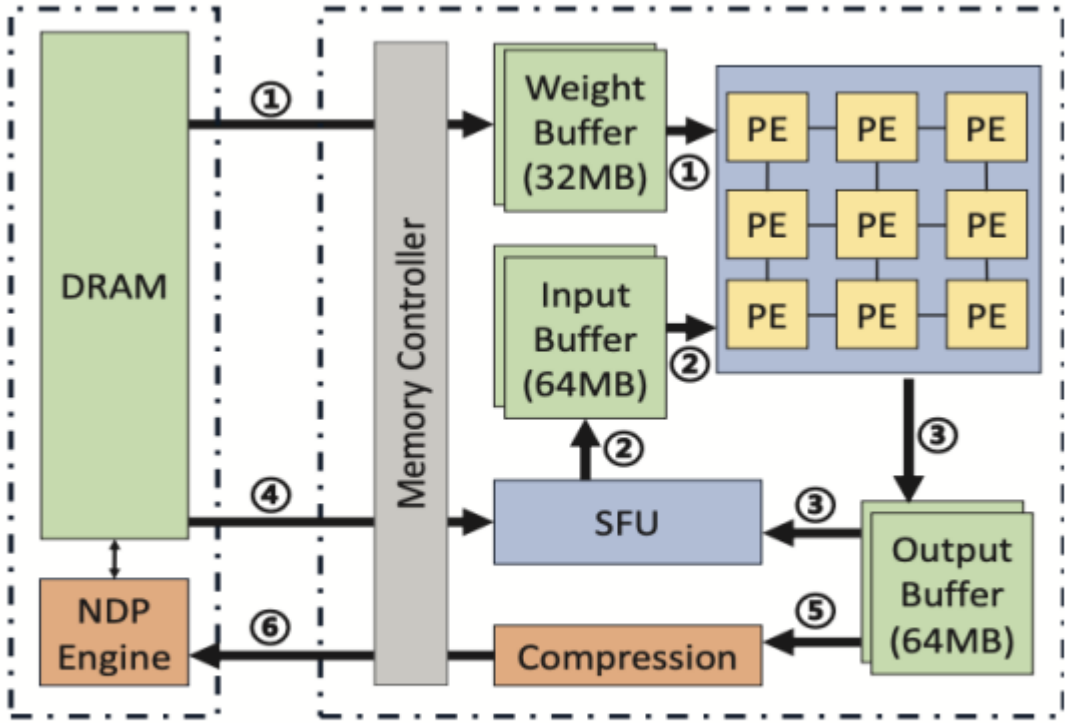


Fig. 1. Overall architecture of Cambricon-D

CHALLENGES FACED

- Simulation Difficulties (using Scalesim)
- Adapting to Simplified Model (convolution and ReLU)

EVALUATION METHODOLOGY

- Sample Input Model
- GUID128
- GUID512

RESULTS (Sample Input Model)

Number of iterations captured on **Cambricon-D** for a **sample input model**

```
C:\Users\athul\Desktop\TAMU\Fall 2024\CSCE-614\Project\Cambricon-D>python cambriconD.py
Total main iterations (over spatial locations): 16384
Total output channel iterations: 1048576
Total quantization operations: 27870912
Total multiplier operations: 55741824
Total iterations per tile: 2097152
```

RESULTS (Sample Input Model)

Number of iterations captured on the **baseline systolic array** for a **sample input model**

```
C:\Users\athul\Desktop\TAMU\Fall 2024\CSCE-614\Project\Cambricon-D>python baseline.py
Total Cycles for Computation: 16384
Total Memory Access Cycles: 49152
Total Cycles for Simulation: 2473984
Memory Access Time: 174.76 ns
```

Speedup of Sample Model for Cambricon-D over Systolic Array Prototype

Sample Model

- Total cycles for simulation in Systolic Array: 2473984
- Total iterations per tile in Cambricon-D: 2097152
- Speedup = $\frac{2473984}{2097152}$ = 1.1796875

RESULTS (GUID 128 and GUID 512)

Number of iterations captured on **Cambricon-D** for prototypes of **GUID 128** and **GUID 512** models as inputs

```
root@LAPTOP-P9AM5HG8:/mnt/c/CSCE 614/Cambricon-D-Group-18/CambriconD# python3 cambriconD.py

Running convolution for GUID 128...
Total main iterations (over spatial locations): 4096
Total output channel iterations: 524288
Total quantization operations: 13716864
Total multiplier operations: 27433728
Total iterations per tile: 1048576
Activation memory accesses: 4096
Weight memory accesses: 524288

Running convolution for GUID 512...
Total main iterations (over spatial locations): 16384
Total output channel iterations: 8388608
Total quantization operations: 222967296
Total multiplier operations: 445934592
Total iterations per tile: 16777216
Activation memory accesses: 16384
Weight memory accesses: 8388608
```

RESULTS (GUID 128 and GUID 512)

Number of iterations captured on the **baseline systolic array** for prototypes of **GUID 128** and **GUID 512** models as inputs

```
C:\Users\athul\Desktop\TAMU\Fall 2024\CSCE-614\Project\Cambricon-D>python baseline.py

GUID 128 Results:
Matrix Dimension: 128x128
Total Cycles for Computation: 16384
Total Memory Access Cycles: 49152
Total Cycles for Simulation: 1982464
Memory Access Time: 174.76 ns

GUID 512 Results:
Matrix Dimension: 512x512
Total Cycles for Computation: 262144
Total Memory Access Cycles: 786432
Total Cycles for Simulation: 31719424
Memory Access Time: 2796.20 ns
```

Speedup of Cambricon-D over Systolic Array Prototype

GUID 128

- Total cycles for simulation in Systolic Array: 1982464
- Total iterations per tile in Cambricon-D: 1048576
- Speedup = $\frac{1982464}{1048576}$ = 1.890625

GUID 512

- Total cycles for simulation in Systolic Array: 31719424
- Total iterations per tile in Cambricon-D: 16777216
- Speedup = $\frac{31719424}{16777216}$ = 1.890625

Memory Access Cycles for Cambricon-D and Systolic Array Prototype

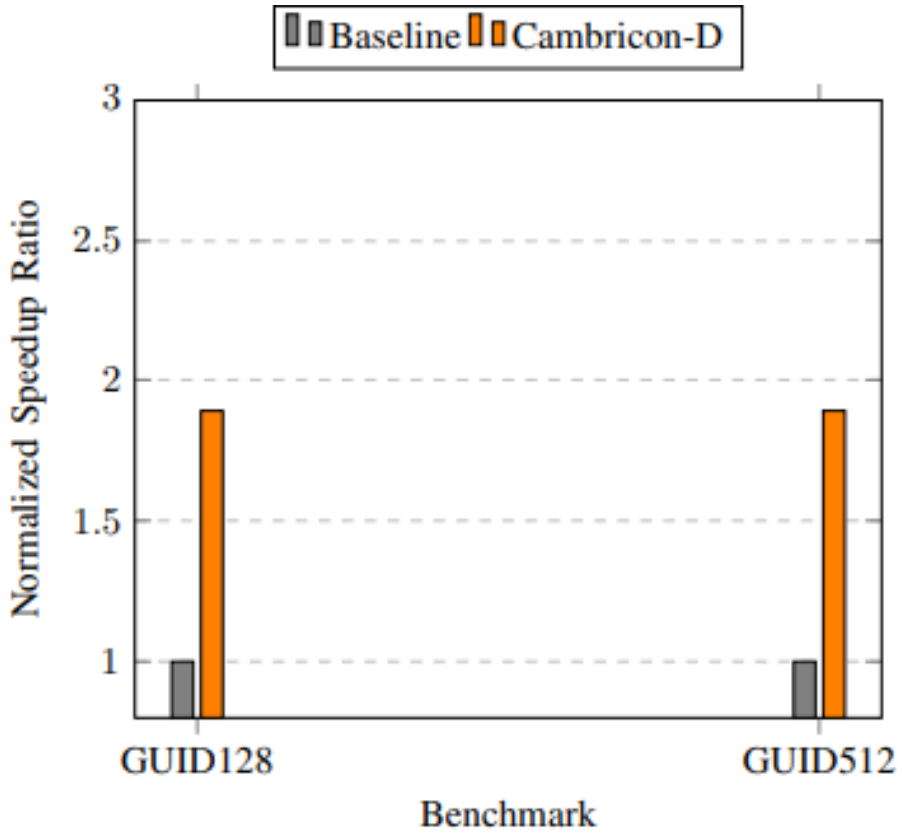
GUID 128

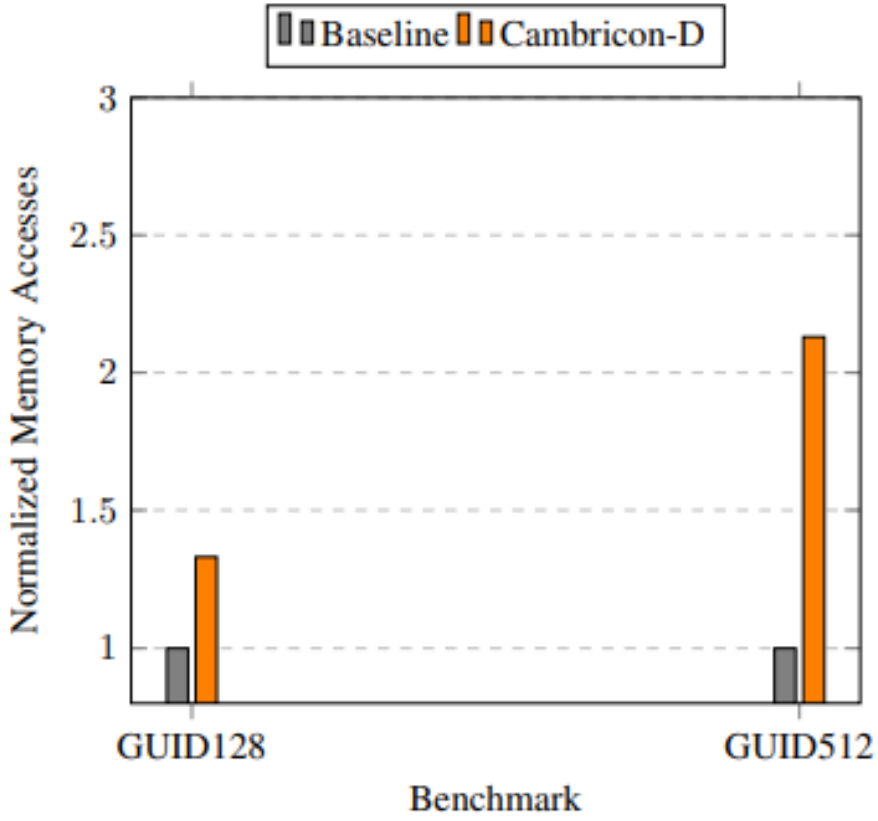
- Total memory access cycles in Systolic Array: 393216
- Weight memory accesses in Cambricon-D: 524288
- Activation memory accesses in Cambricon-D: 4096
- Memory Access Ratio = $\frac{524288+4096}{393216}$ = 1.34375

GUID 512

- Total memory access cycles in Systolic Array: 3932160
- Weight memory accesses in Cambricon-D: 8388608
- Activation memory accesses in Cambricon-D: 16384
- Memory Access Ratio = $\frac{8388608+16384}{3932160}$ = 2.1375

RESULTS


Baseline
Cambricon-D
Normalized Speedup Ratio
3
2.5
2
1.5
1
GUID128
GUID512
Benchmark


Baseline
Cambricon-D
Normalized Memory Accesses
3
2.5
2
1.5
1
GUID128
GUID512
Benchmark

CONCLUSION AND FUTURE WORK

- Implement additional operators to more closely match the actual Cambricon-D design
- Explore more advanced simulation tools for more accurate performance modeling
- Investigate the impact of different input sizes and model configurations on performance

THANK YOU!