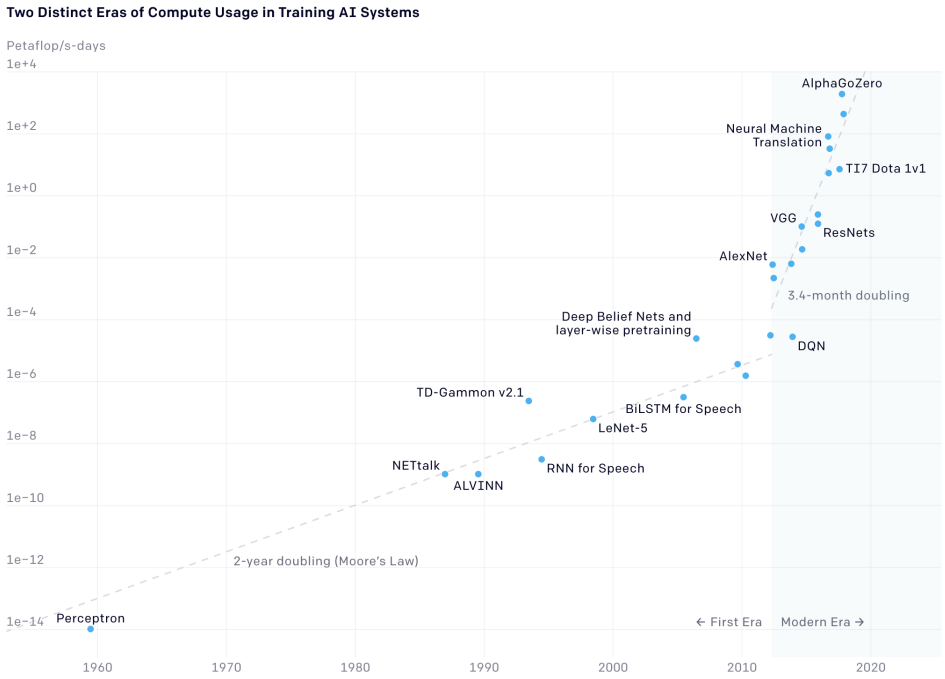
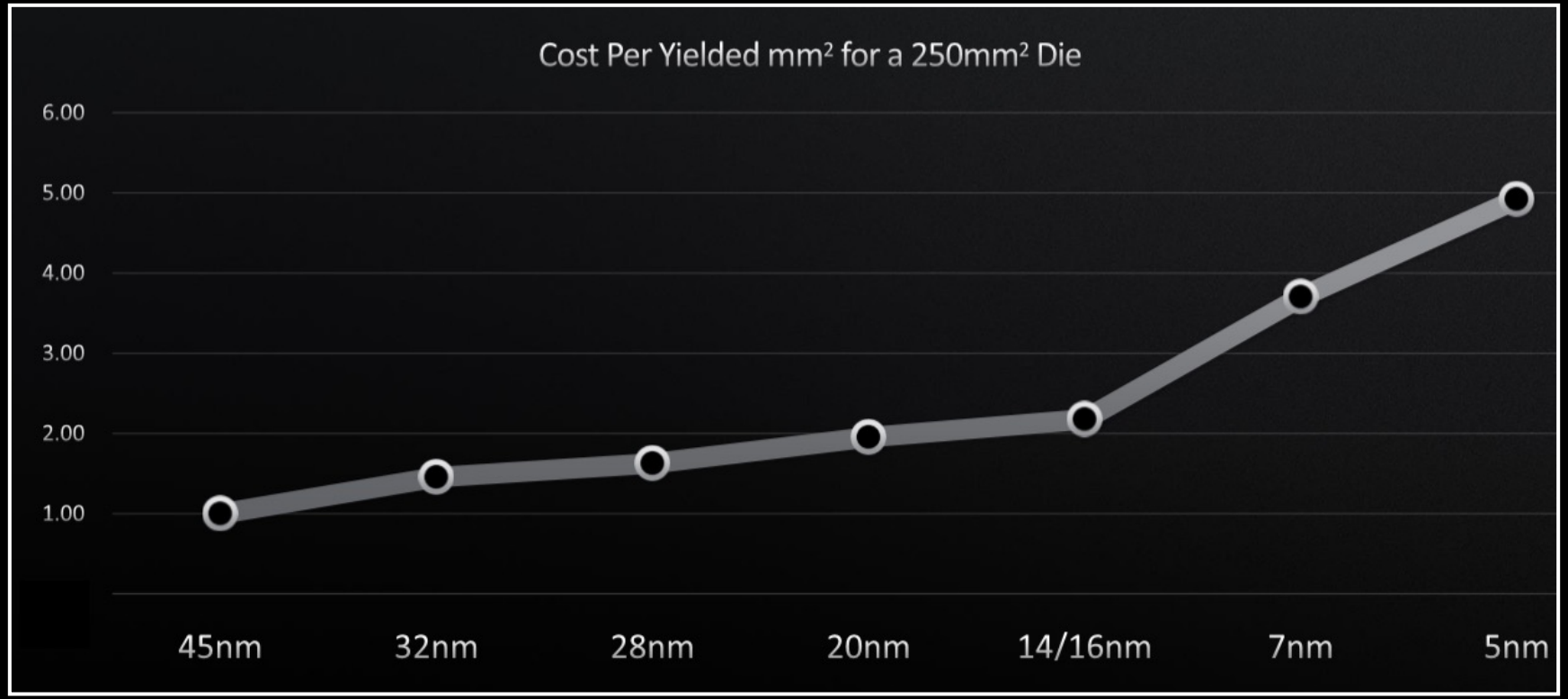
Next-Generation Deep-Learning Accelerators: From Hardware to System

Sophia Shao
ysshao@berkeley.edu
Electrical Engineering and Computer Sciences
Berkeley
UNIVERSITY OF CALIFORNIA

Growing Demand in Computing



OpenAI



Slowing Supply in Computing

AMD, HotChips, 2019

Growing Demand in Computing

Slowing Supply in Computing

Growing Demand in Computing

Slowing Supply in Computing

Domain-Specific Accelerators

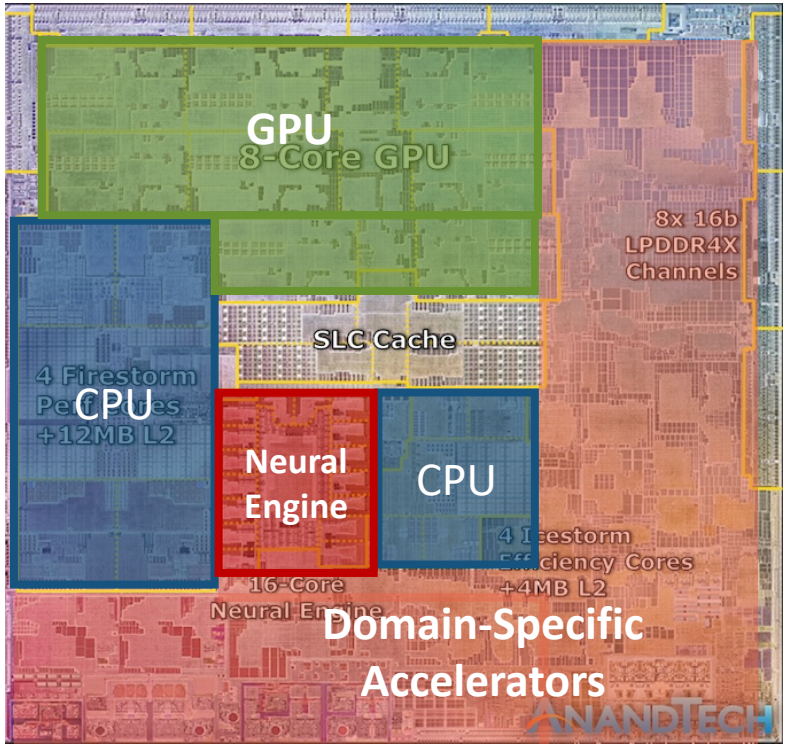
Growing Demand in Computing

Slowing Supply in Computing

Domain-Specific Accelerators

- Customized hardware designed for a domain of applications.

Apple M1 Chip
2020



* AnandTech

Full-Stack Optimization for DL Accelerators

Design of Accelerators	• Simba [MICRO'19 **Best Paper Award**, **CACM RH,** VLSI'20, JSSC'20 **Best Paper Award**]
Integration of Accelerators	• Chipyard [IEEE Micro'20] • Gemmini [DAC'21, **Best Paper Award**]
Scheduling of Accelerators	• CoSA [ISCA'21]

Full-Stack Optimization for DL Accelerators

Design of Accelerators	• Simba [MICRO'19 **Best Paper Award**, **CACM RH,** VLSI'20, JSSC'20 **Best Paper Award**]
Integration of Accelerators	• Chipyard [IEEE Micro'20] • Gemmini [DAC'21, **Best Paper Award**]
Scheduling of Accelerators	• CoSA [ISCA'21]

Scalable Inference Accelerators

Motivation

- Need for fast and efficient inference accelerators from mobile to datacenter.

Challenge

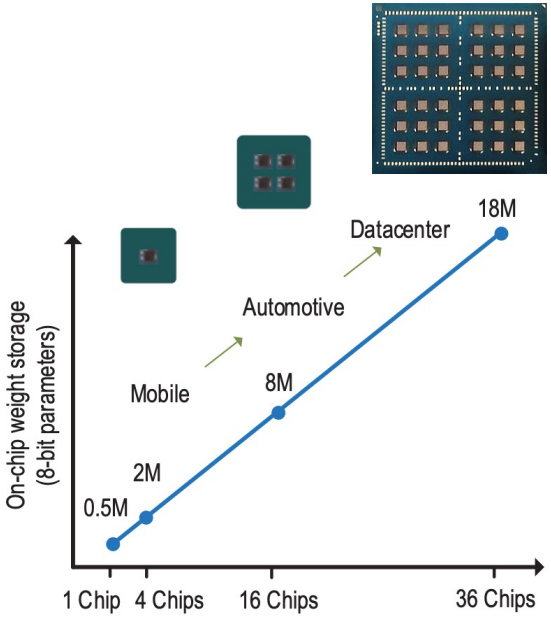
- High design cost of building unique hardware for each design target.

Opportunities

- Deep learning inference is intrinsically scalable with abundant parallelism.
- Recent advances in package-level integration for multi-chip-module-based designs.

The Multi-Chip-Module Approach

- Advantages:
  - Build systems larger than reticle limit
  - Smaller chips are cheaper to design
  - Smaller chips have higher yield
  - Faster time-to-market
- Challenges:
  - Area, energy, and latency for chip-to-chip communication



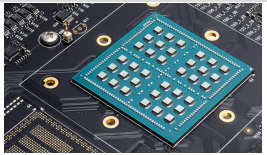
[VLSI'2019]

Simba: Scaling Inference with MCM-based Architecture

Best Paper Award at MICRO'2019, CACM Research Highlights

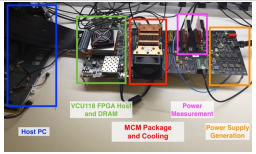
Simba Testchip:

- Package and chiplet architecture
- Processing element design
- Baseline uniform tiling across chiplets and PEs



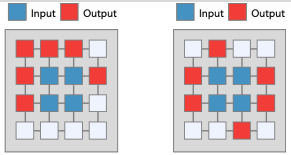
Simba Characterization:

- Comparison with GPUs
- NoP bandwidth sensitivity
- NoP latency sensitivity



Simba NoP-Aware Tiling:

- Non-uniform work partitioning
- Communication-aware data placement
- Cross-layer pipelining



[MICRO'2019]

Simba: Scalable MCM-Based Architecture

Package and chiplet spec

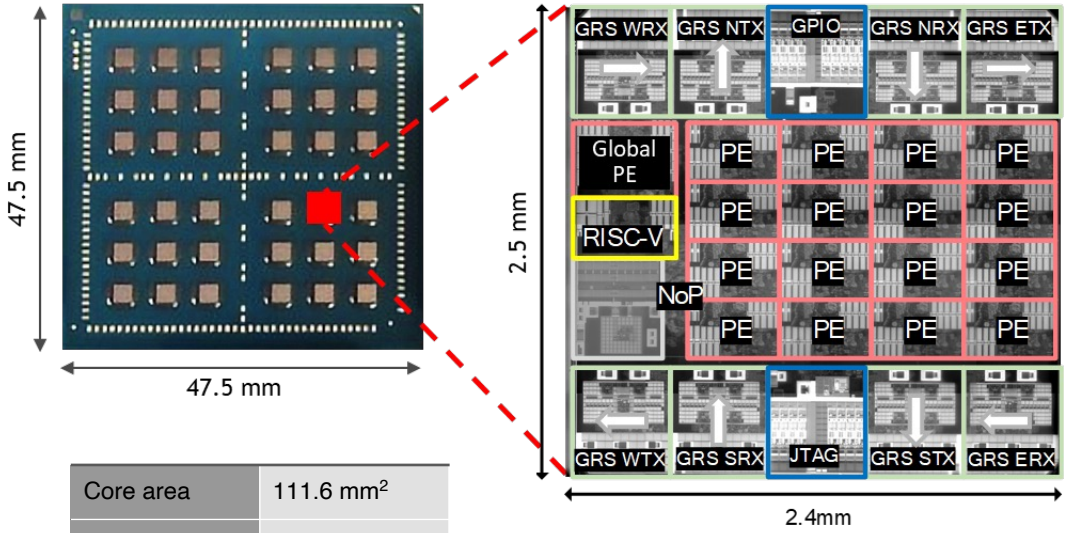
6mm^2 chiplet in TSMC 16nm
36 chiplets/package

Chip-to-chip interconnect

Ground-Referenced Signaling

Efficient compute tiles

128 TOPS
0.11 pJ/Op
8-bit integer datapath

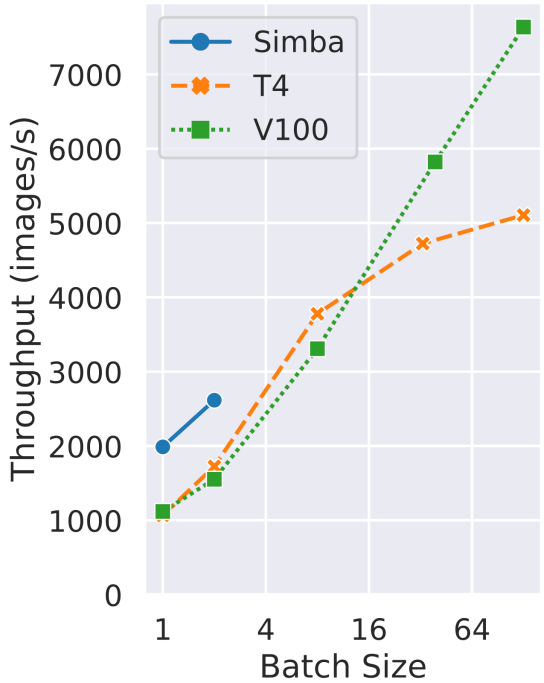


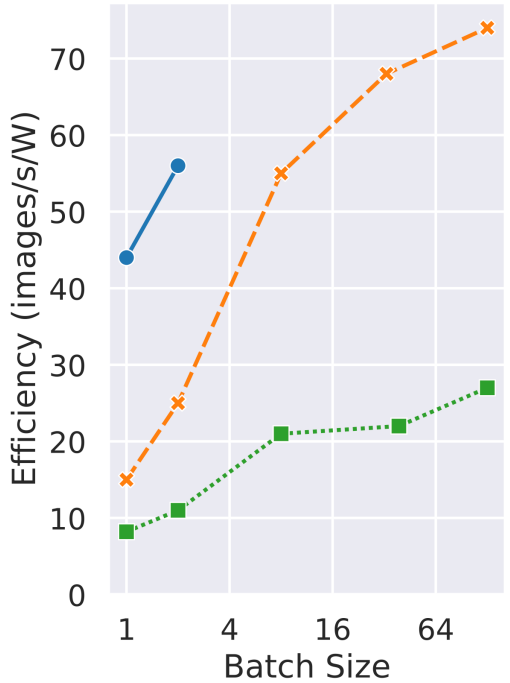
Core area	111.6 mm^2
Voltage	0.52-1.1 V
Frequency	0.48-1.8 GHz
SRAM	624KB/chip 23MB/package

[VLSI'2019]

Simba Characterization

- Comparison with GPUs running ResNet-50





[MICRO'2019]

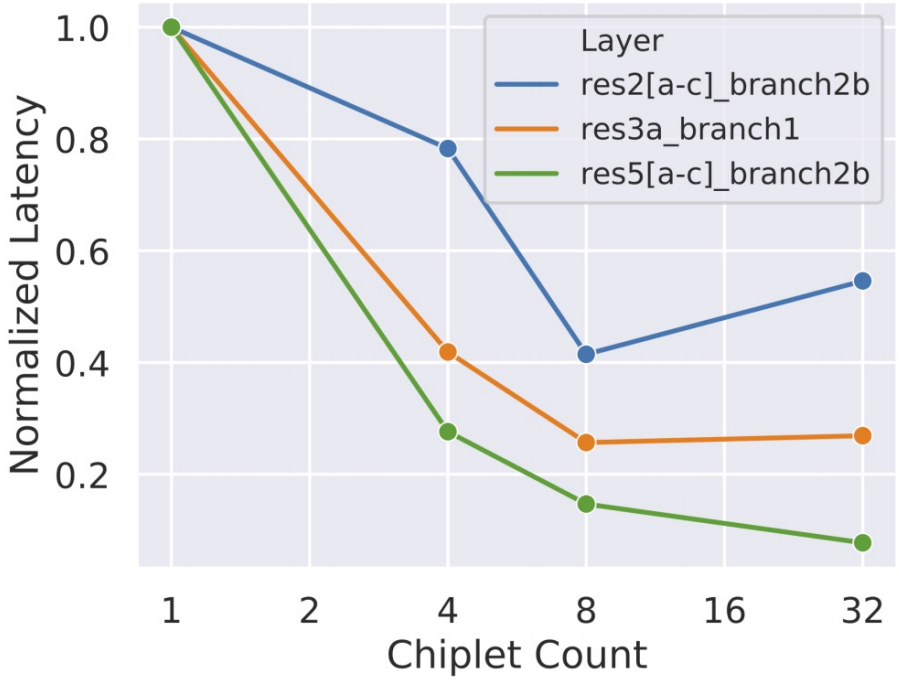
Simba Characterization

- Layer Sensitivity

- Running three ResNet-50 layers across different number of chiplets.

- Increasing the number of active chiplets does not always translate to performance gains.

- The cost of communication hinders the ability to exploit parallelism.

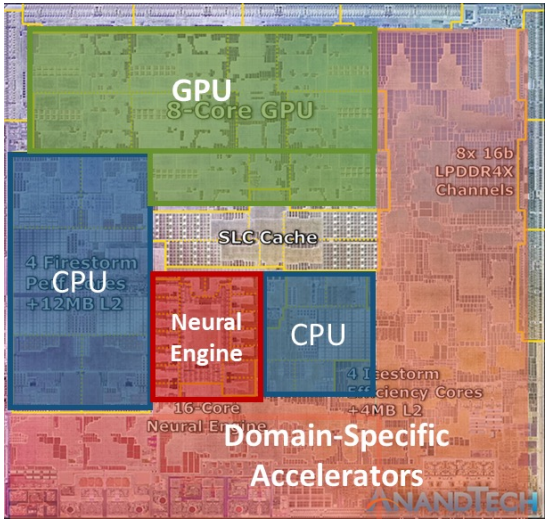


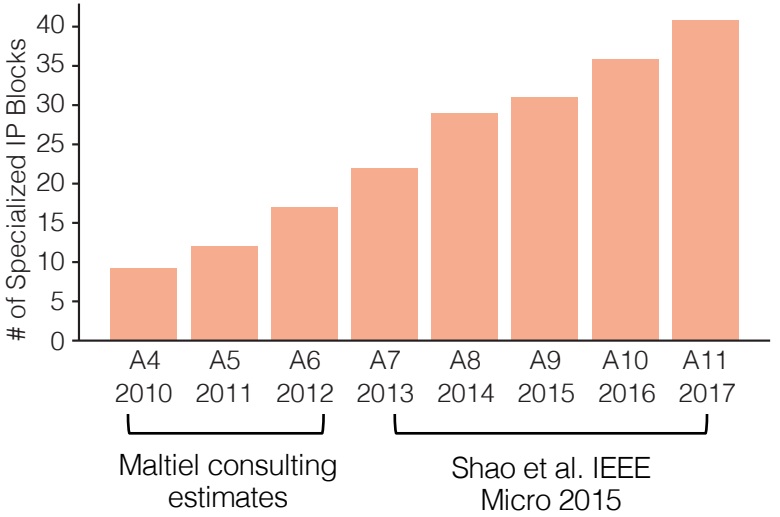
[MICRO'2019]

Full-Stack Optimization for DL Accelerators

Design of Accelerators	• Simba [MICRO'19 **Best Paper Award**, **CACM RH**, VLSI'20, JSSC'20 **Best Paper Award**]
Integration of Accelerators	• Chipyard [IEEE Micro'20] • Gemmini [DAC'21, **Best Paper Award**]
Scheduling of Accelerators	• CoSA [ISCA'21]

Accelerators don’t exist in isolation.

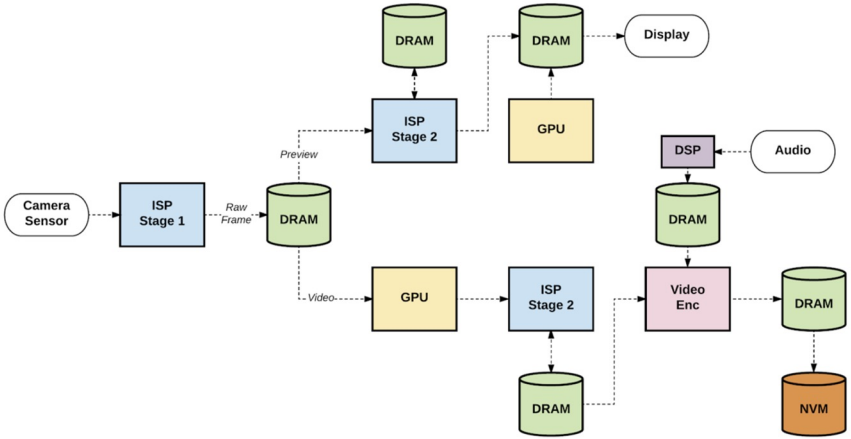




http://vlsiarch.eecs.harvard.edu/research/accelerators/die-photo-analysis/

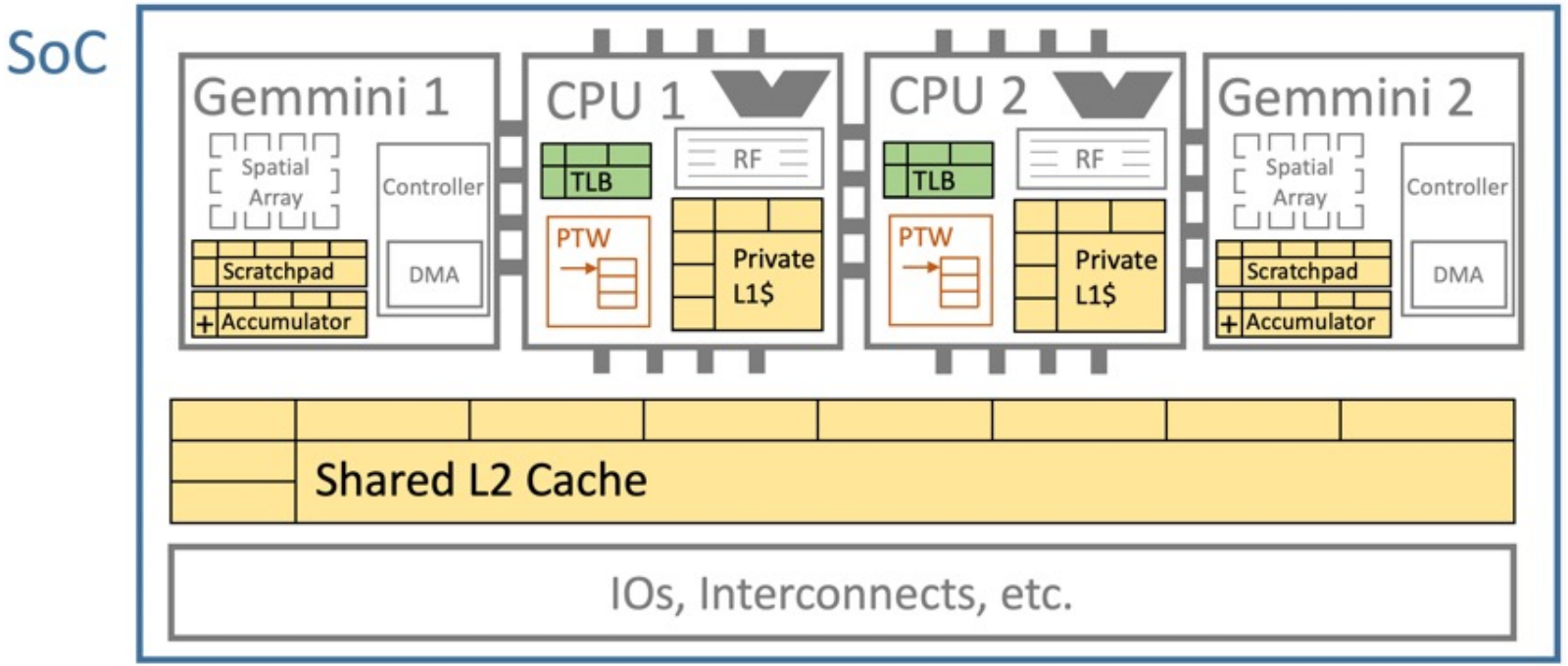
Mobile SoC Usecase

- Mainstream architecture has long focused on general-purpose CPUs and GPUs.
- In an SoC, multiple IP blocks are active at the same time and communicate frequently with each other.
- Example:
  - Recording a 4K video
  - Camera -> ISP
    - “Preview stream” for display
    - “Video stream” for storage
  - DRAM for data sharing

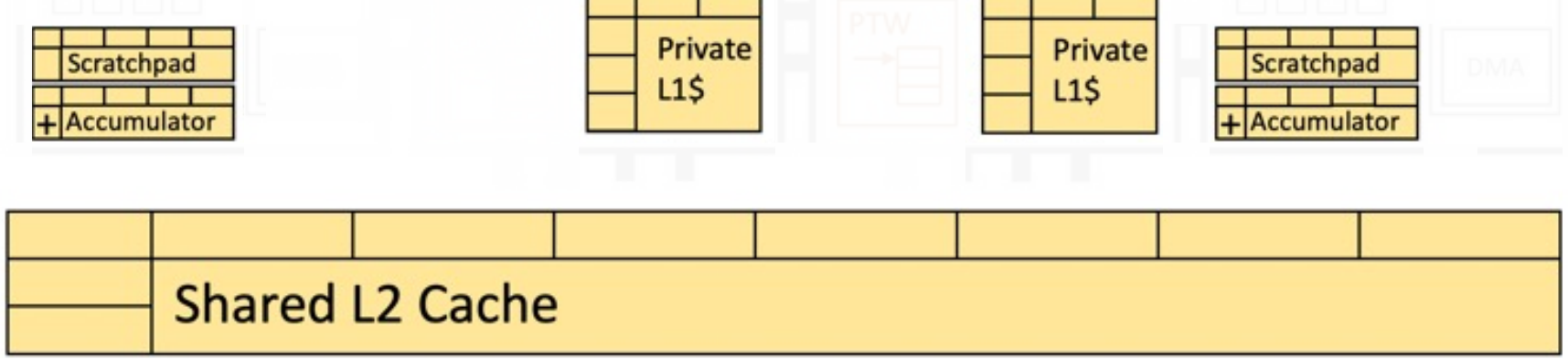


Two Billion Devices and Counting: An Industry Perspective on the State of Mobile Computer Architecture, IEEE Micro’2018

Full-System Visibility for DL Accelerators



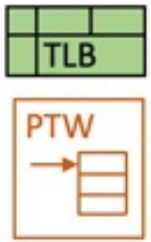
Full-System Visibility: Memory Hierarchy

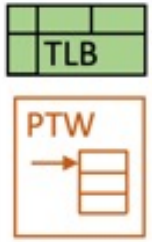


Performance Impacts

Resource contention, cache coherence, etc.

Full-System Visibility: Virtual Addresses

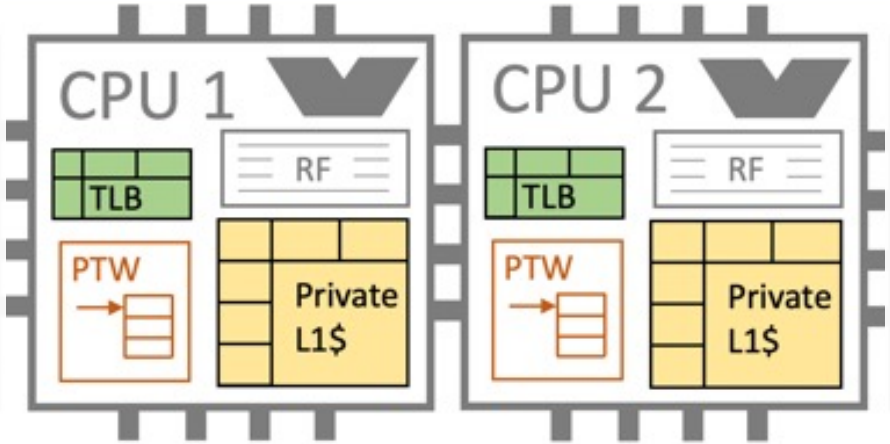




Performance Impacts

Page faults, TLB hits, etc.

Full-System Visibility: Host CPUs

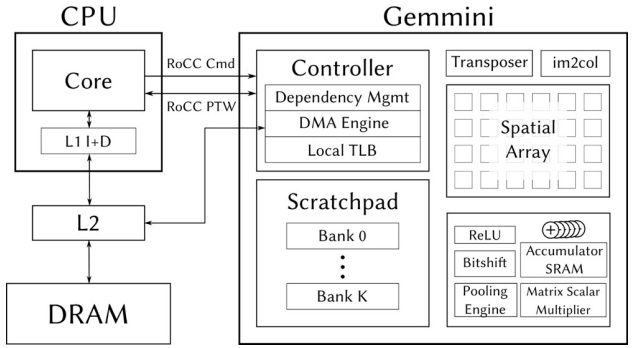


Performance Impacts

Unaccelerated kernels, etc.

Gemmini: Full-System Co-Design of Hardware Accelerators

- **Full-stack**
  - Includes OS
  - End-to-end workloads
  - "Multi-level" API
- **Full-SoC**
  - Host CPUs
  - Shared memory hierarchies
  - Virtual address translation

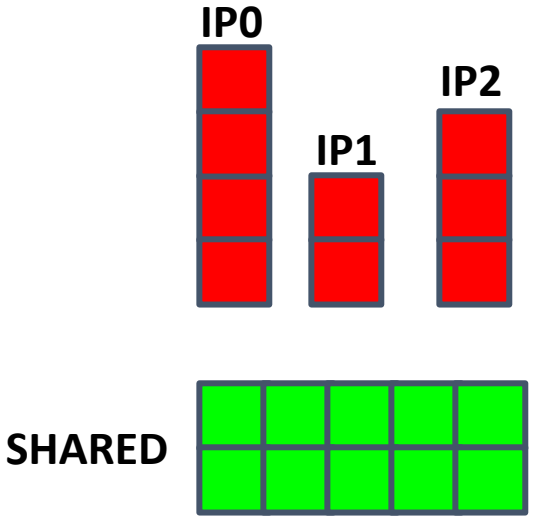


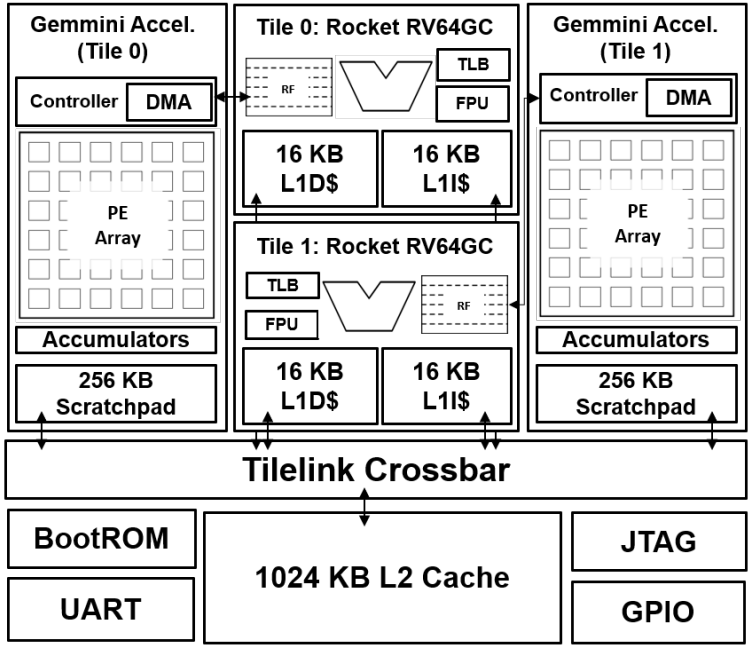
	Property	NVDLA	VTA	PolySA	DNNBuilder	MAGNet	DNNWeaver	MAERI	Gemmini
Hardware Architecture Template	Multiple Datatypes	Int/Float	Int	Int	Int	Int	Int	Int	Int/Float
	Multiple Dataflows	✗	✗	✓	✓	✓	✓	✓	✓
	Spatial Array	vector	vector	systolic	systolic	vector	vector	vector	vector/systolic
	Direct convolution	✓	✗	✗	✓	✓	✓	✓	✓
Programming Support	Software Ecosystem	Custom Compiler	TVM	Xilinx SDAccel	Caffe	C	Caffe	Custom Mapper	ONNX/C
	Hardware-Supported Virtual Memory	✗	✗	✗	✗	✗	✗	✗	✓
System Support	Full SoC	✗	✗	✗	✗	✗	✗	✗	✓
	OS Support	✓	✓	✗	✗	✗	✗	✗	✓

https://github.com/ucb-bar/gemmini

[DAC'2021 Best Paper Award]

Gemmini Case Study: Allocating on-chip SRAM



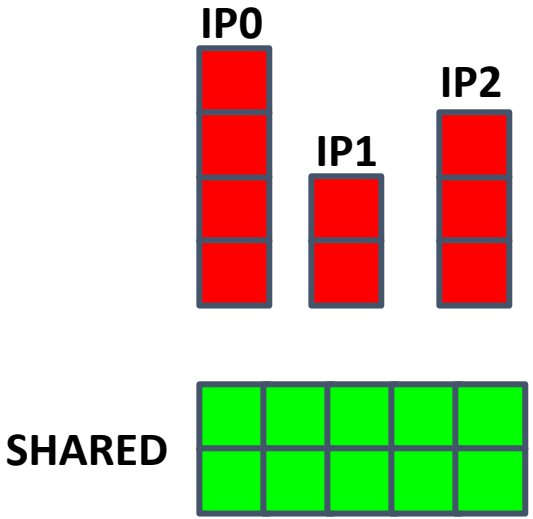


- **Where to allocated SRAM?**
  - Private within each IP
  - Shared

https://github.com/ucb-bar/gemmini

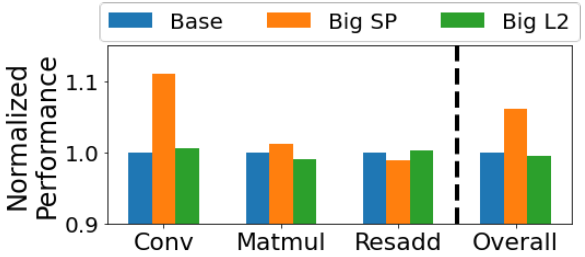
[DAC'2021 Best Paper Award]

Gemmini Case Study: Allocating on-chip SRAM



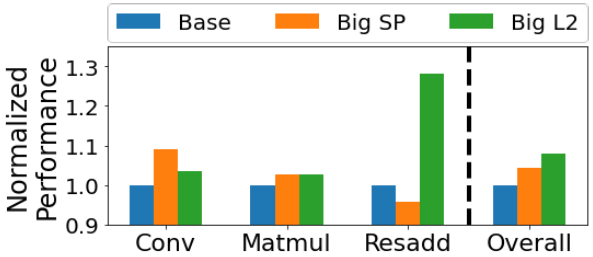
- **Where to allocated SRAM?**
  - Private within each IP
  - Shared

- **Application dependent.**



Single-Core SoC

- **SoC configuration dependent.**



Dual-Core SoC

https://github.com/ucb-bar/gemmini

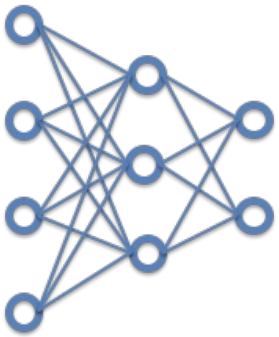
[DAC'2021 Best Paper Award]

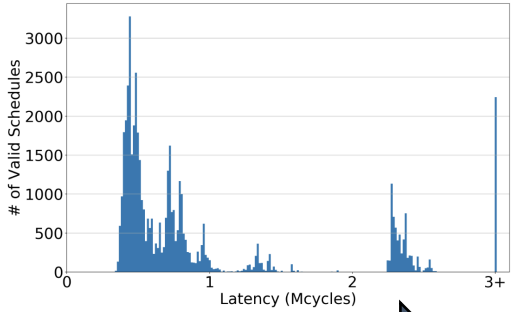
Full-Stack Optimization for DL Accelerators

Design of Accelerators	• Simba [MICRO'19 **Best Paper Award**, **CACM RH,** VLSI'20, JSSC'20 **Best Paper Award**]
Integration of Accelerators	• Chipyard [IEEE Micro'20] • Gemmini [DAC'21, **Best Paper Award**]
Scheduling of Accelerators	• CoSA [ISCA'21]

Large Space of Mapping Algorithms to ML Hardware

Algorithm

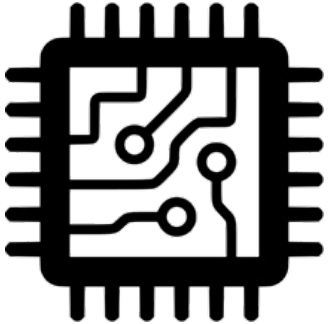




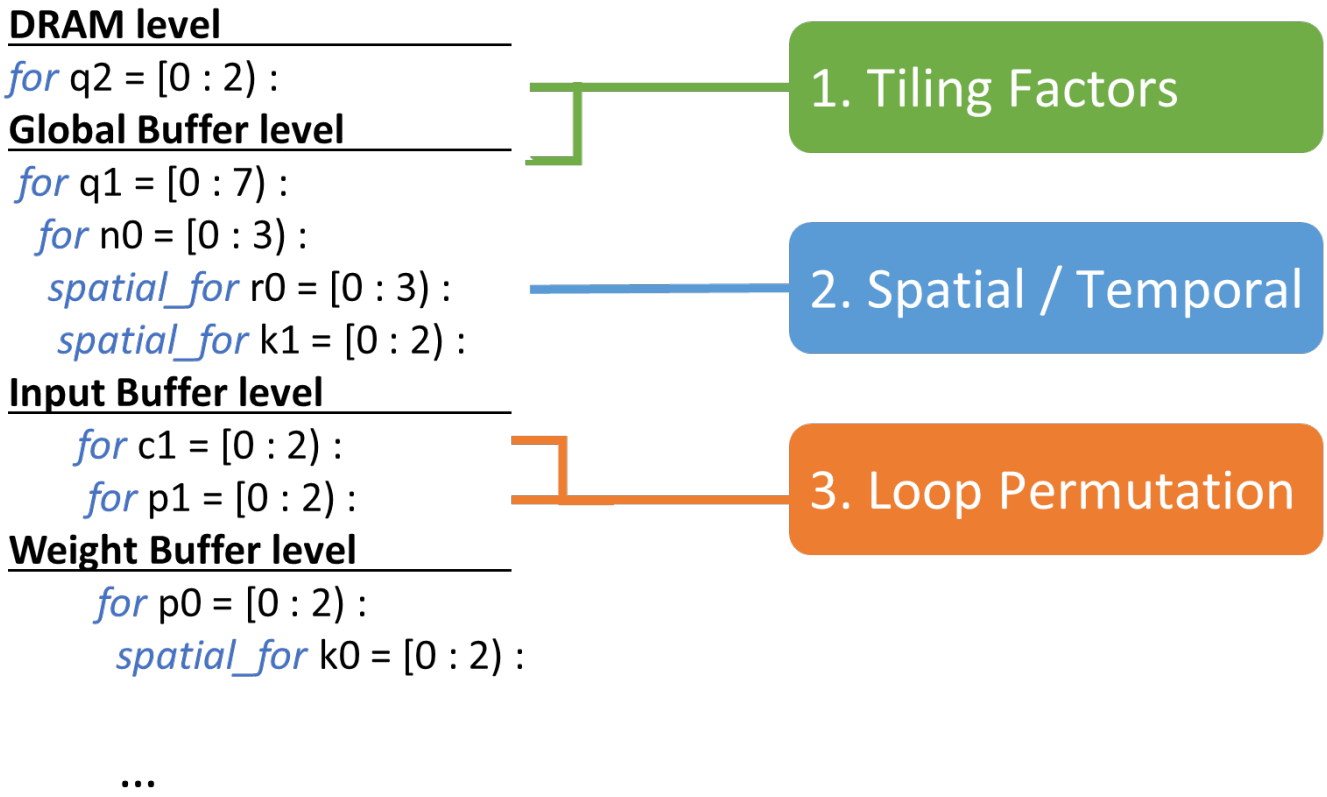
Scheduling

Scheduler	Search Algorithm
Brute-force approcahes:	
Timeloop	Brute-force & Random
dMazeRunner	Brute-force
Interstellar	Brute-force
Marvel	Decoupled Brute-force
Feedback-based Approaches:	
AutoTVM	ML-based Iteration
Halide	Beamsearch OpenTuner
FlexFlow	MCMC
Constrained Optimization Approaches:	
CoSA	**Mixed Integer Programming (MIP)**

Hardware

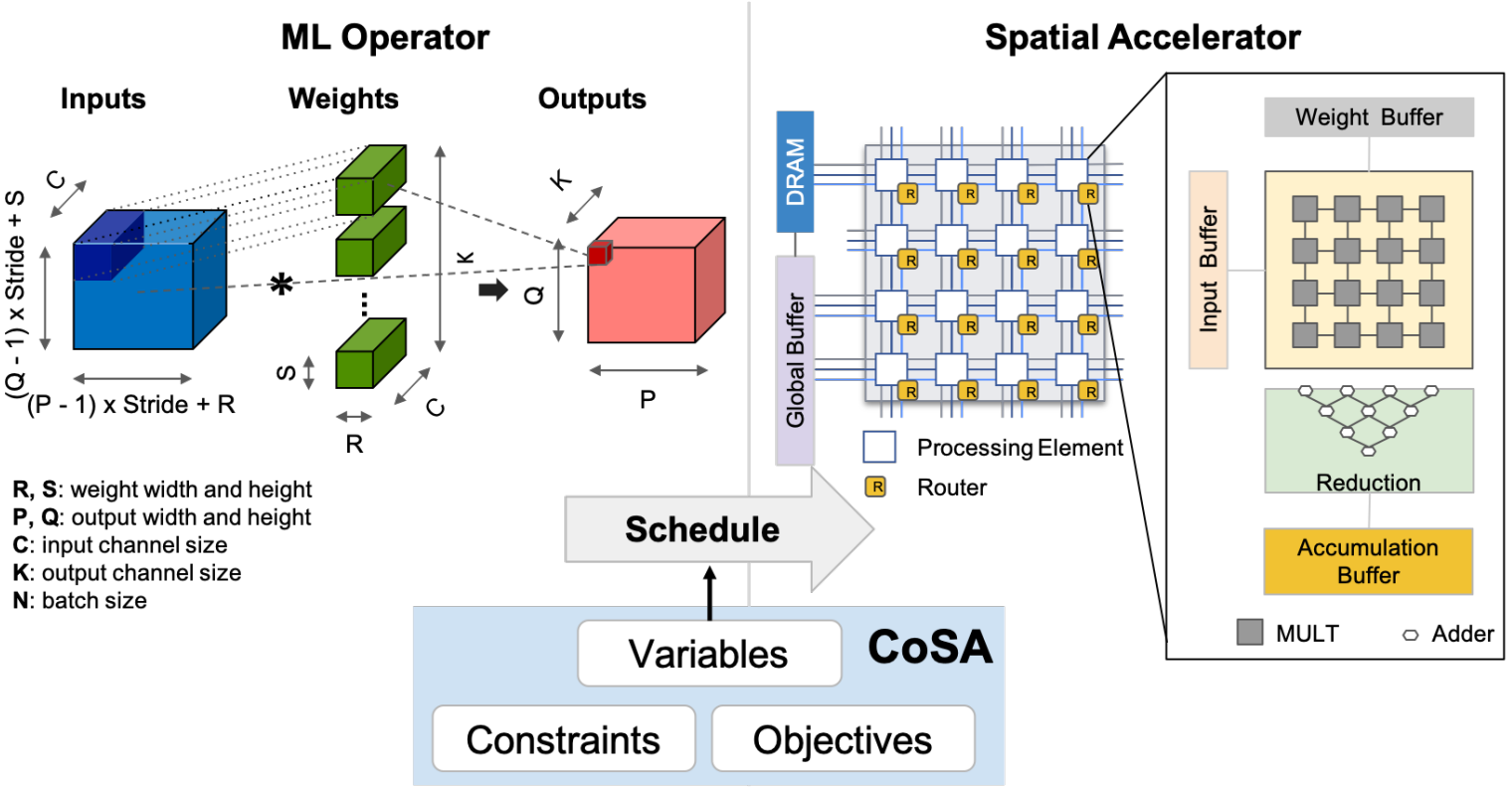


Navigating the Mapping Space



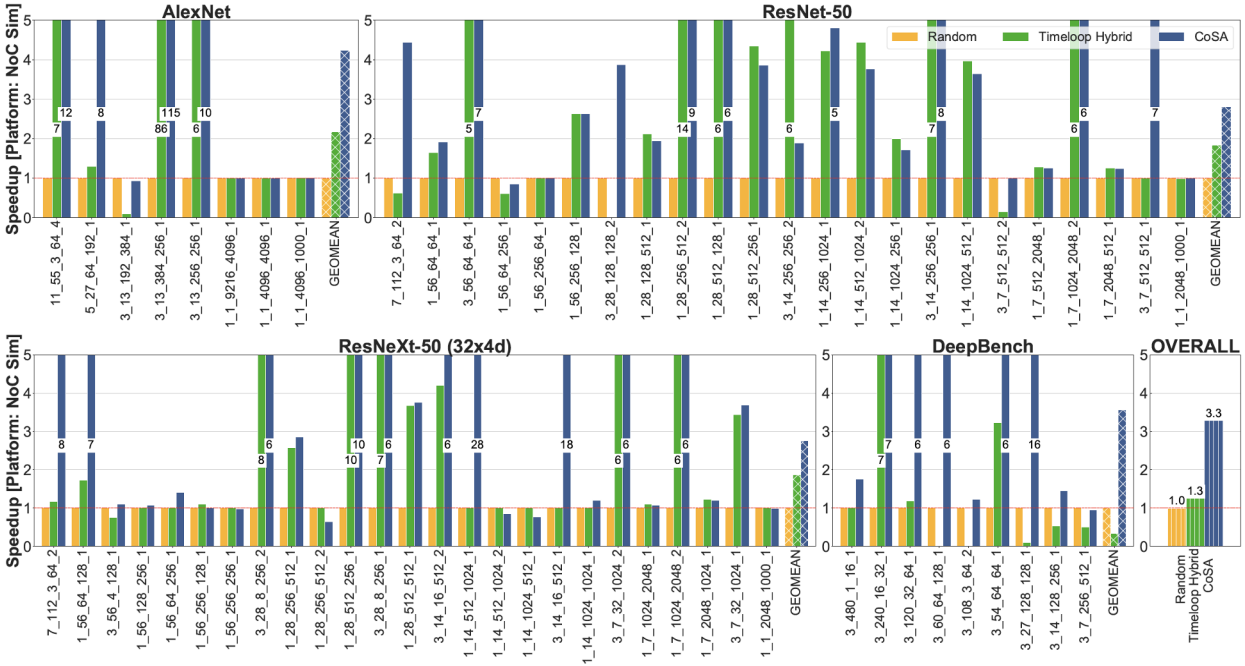
[ISCA'2021]

CoSA: Constrained-Optimization for Spatial Architecture



[ISCA'2021]

Results



	CoSA	Random (5×)	Timeloop Hybrid
Avg. Runtime / Layer	4.2s	4.6s	379.9s
Avg. Samples / Layer	1	20K	67M
Avg. Evaluations / Layer	1	5	16K

2.5x speedup compared to SoTA with 90x faster time-to-solution.

[ISCA'2021]

Acknowledgement

Hasan Genc

Jenny Huang

Seah Kim

- Collaborators from UC Berkeley and NVIDIA!
- Sponsored by DARPA, a Facebook Research Award, a Google Research Award, and ADEPT/SLICE industry sponsors!

Full-Stack Optimization for DL Accelerators

Design of Accelerators	• Simba [MICRO'19 **Best Paper Award**, **CACM RH,** VLSI'20, JSSC'20 **Best Paper Award**]
Integration of Accelerators	• Chipyard [IEEE Micro'20] • Gemmini [DAC'21, **Best Paper Award**]
Scheduling of Accelerators	• CoSA [ISCA'21]