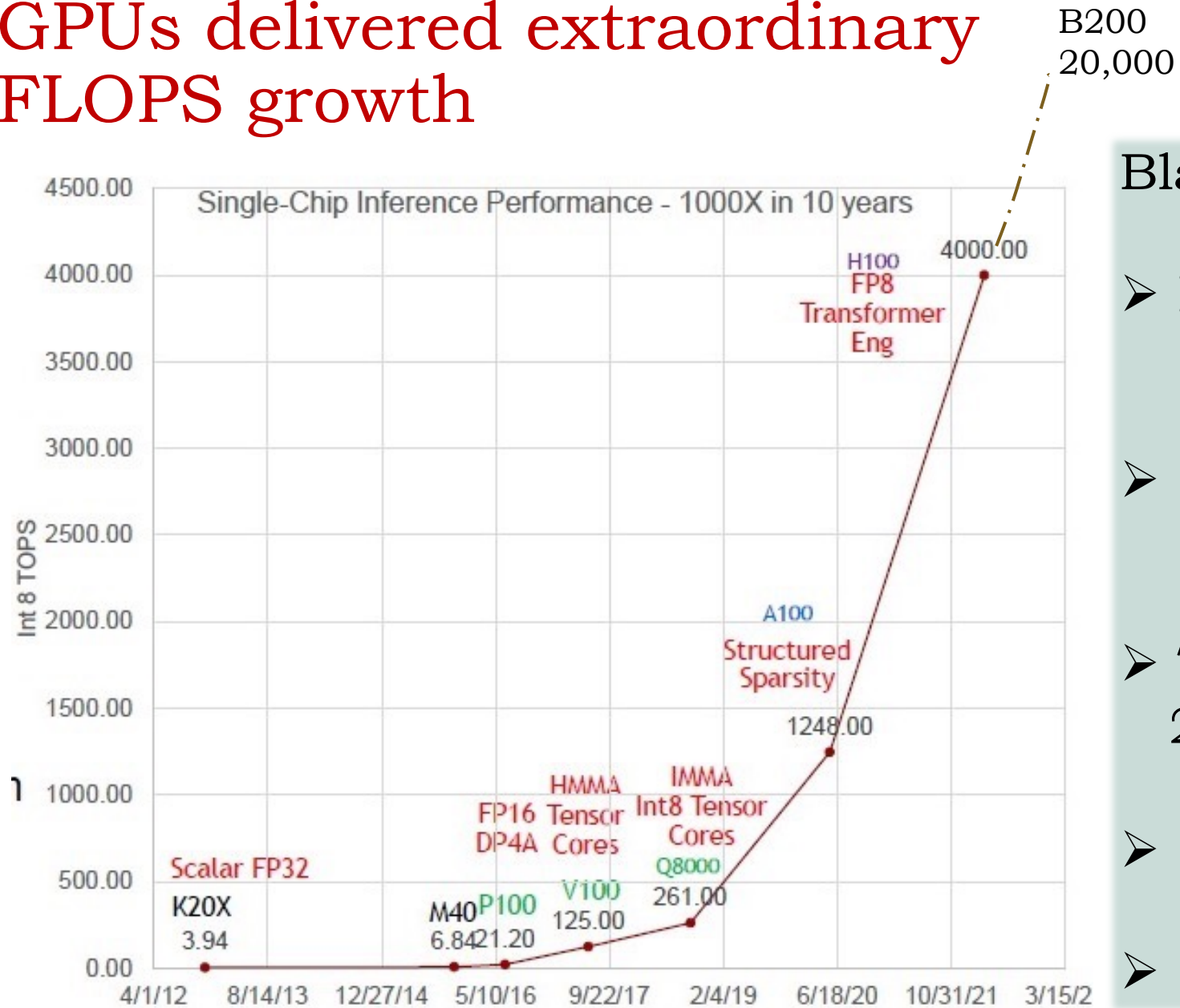


From FLOPS to Intelligence per Joule: The Case for Adaptive Dataflow

Tulika Mitra
School of Computing
National University of Singapore

<https://ecolab.comp.nus.edu.sg/>

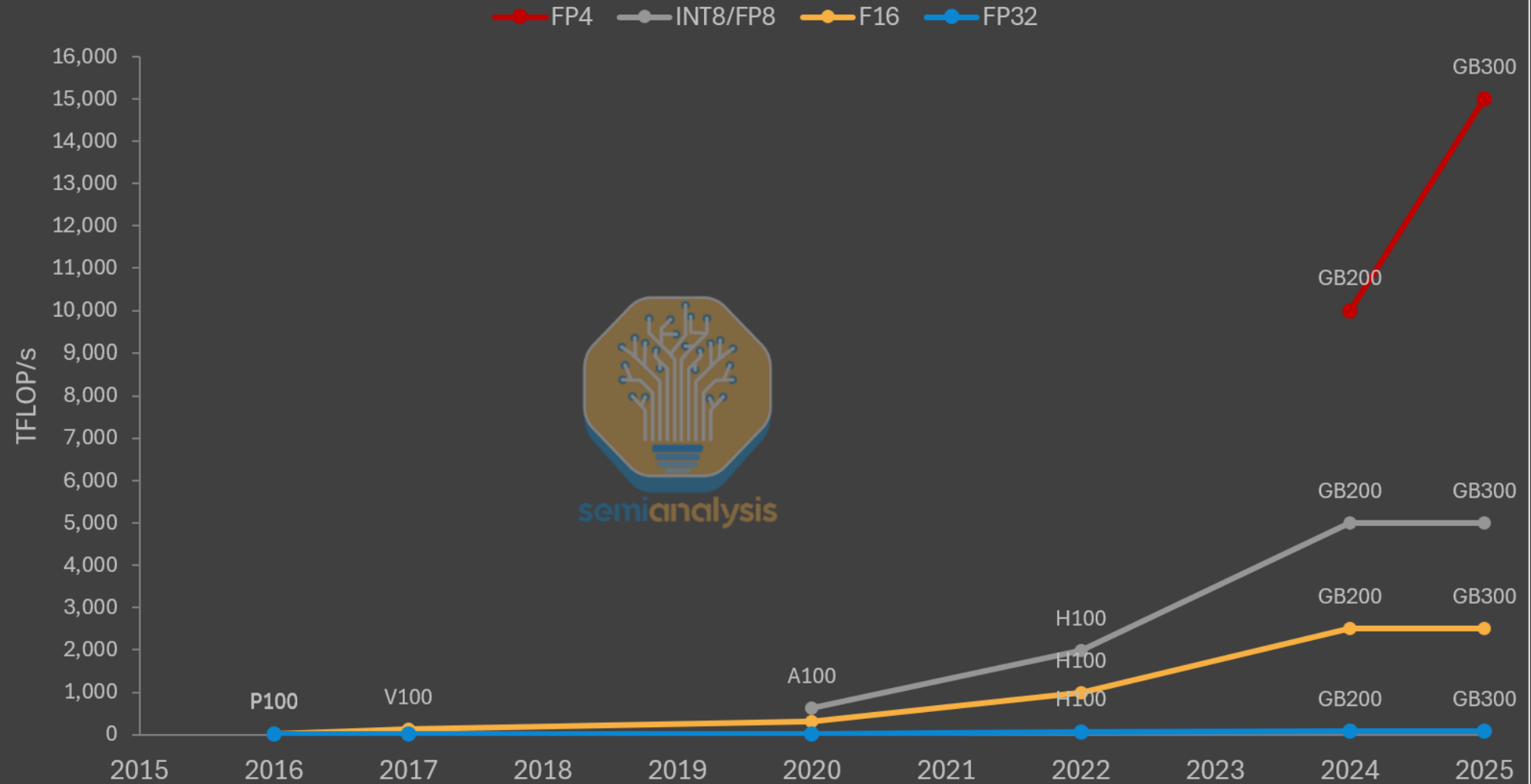
GPUs delivered extraordinary FLOPS growth



Blackwell gains w.r.t. K20X

- Number Representation (~32x)
FP32, FP16, Int8, FP4
- Complex Instructions (~12.5x)
DP4, HMMA, IMMA
- Technology node (~3x)
28nm, 16nm, 7nm, 5nm, 4nm
- Structured Sparsity 2x
- Die Size 2x

NVIDIA Single GPU Dense Throughput



From FLOPS to Intelligence per Joule

- **FLOPS:** What's the raw compute capability of the hardware?
- Hardware needs storage, data movement, control, and scheduling to turn raw compute into useful AI computation
- More FLOPS $\neq$ Proportionally more AI capability
- As arithmetic becomes cheaper, non-compute costs dominate
- **Intelligence/Joule:**
How much useful AI capability can we deliver at required quality and latency per unit energy?

**Sparsity is becoming a key path
to higher Intelligence/Joule**

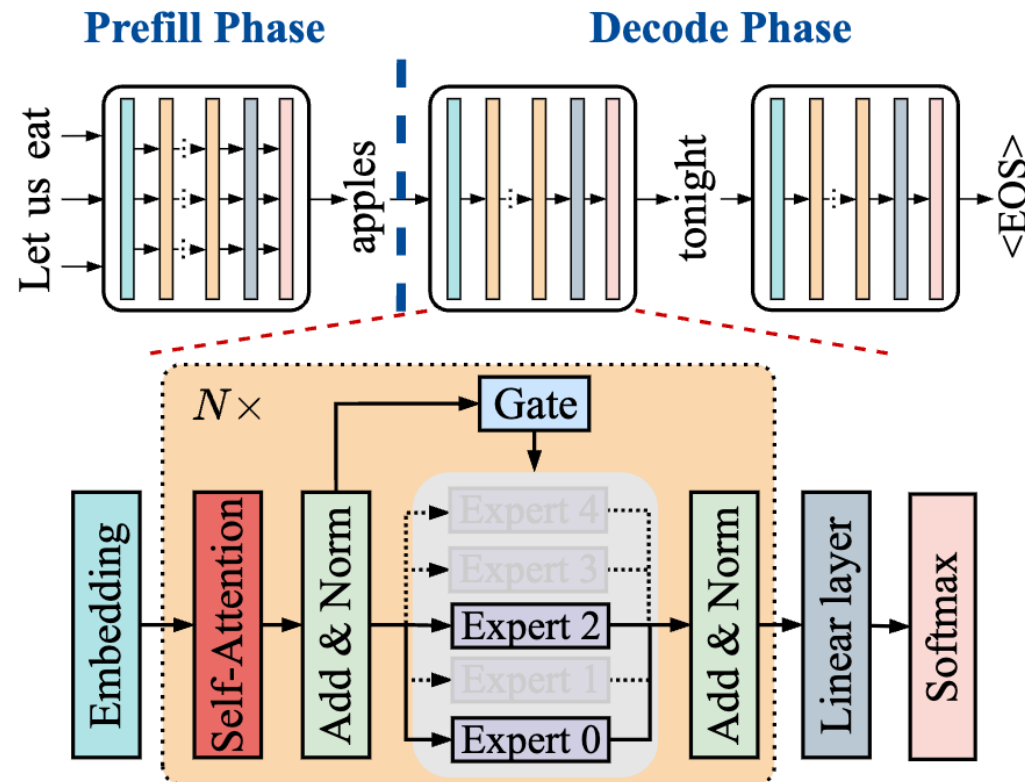
Sparsity in LLM Workloads

- **Mixture of Experts (MoE):**

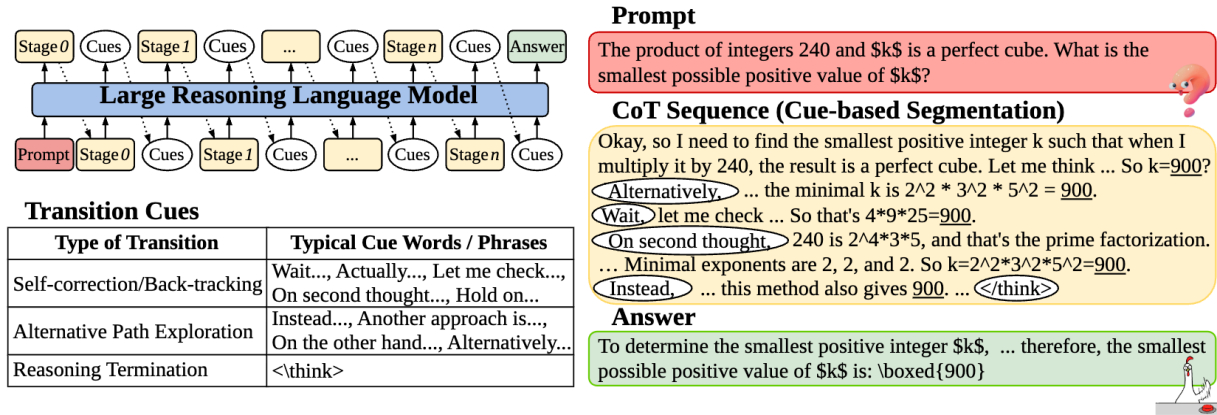
Increased model capacity without proportional compute cost

A subset of experts activated per token:

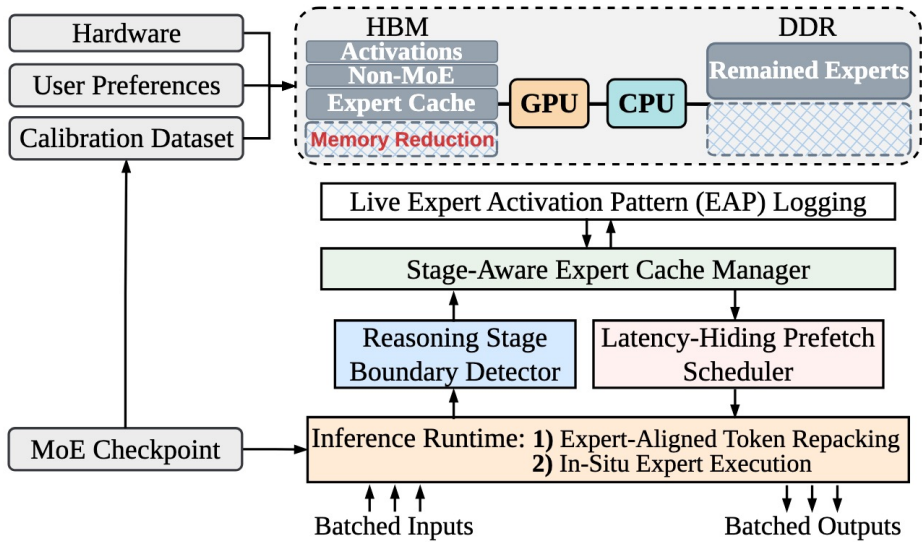
~5.5% active parameters in Deepseek-v3 671B



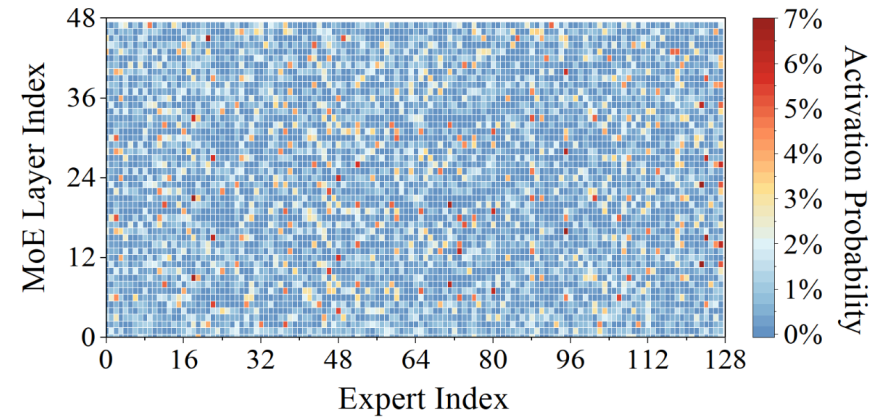
Exploiting Reasoning Boundaries for Expert Caching



Reasoning stage boundary detection



Stage aware CPU-GPU expert cache manager



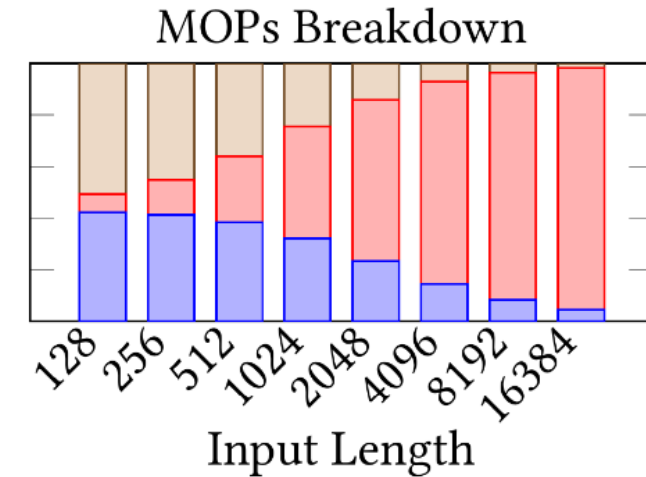
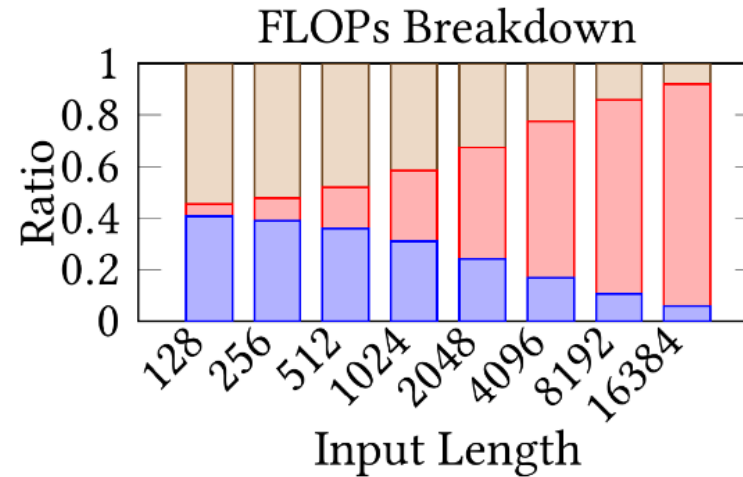
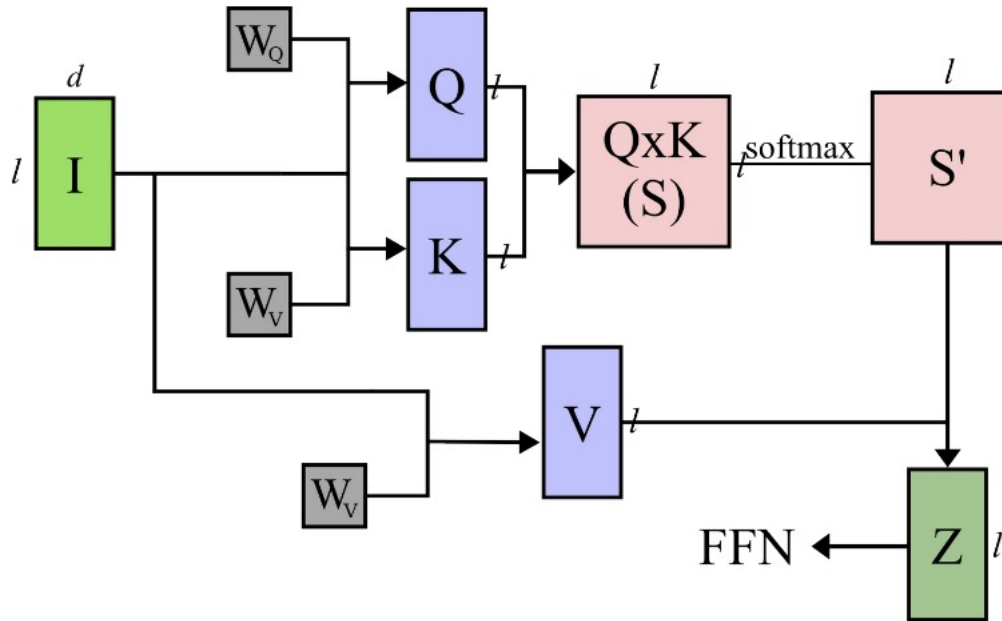
Sparse Expert Activation (Qwen-3-30B)

	GPU	CPU
Processor	NVIDIA A100	$2 \times$ Intel Xeon Gold 6326 (16 physical cores, 2.9GHz)
Memory	80 GB HBM	512 GB DDR
Bandwidth	1935 GB/s	–
Interface	PCIe 4.0 (64 GB/s)	

- ✓ 1.6x speedup over SOTA in real system
- ✓ No impact on accuracy

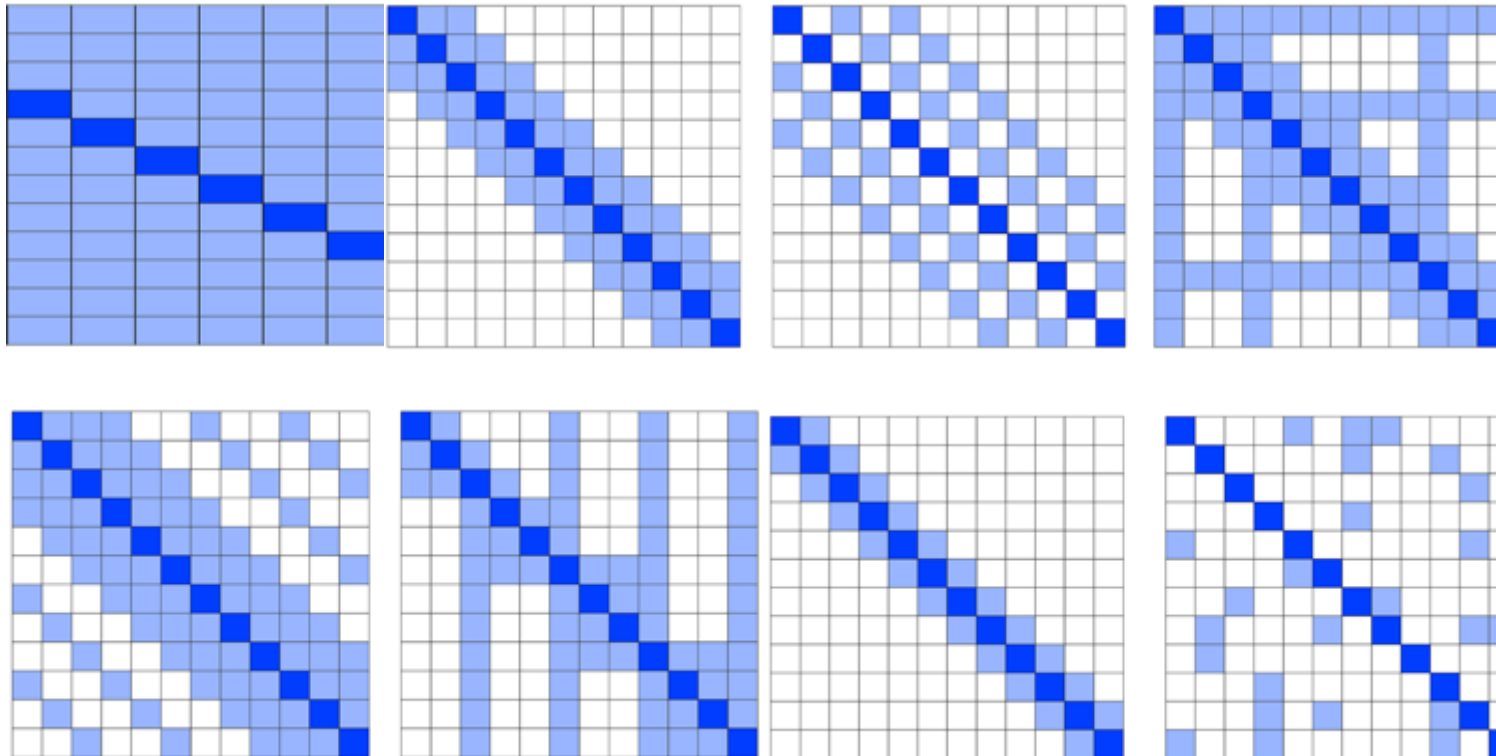
Attention: Poor Scalability with Context Length

Attention scales poorly with context length, requiring quadratic operations and memory for n tokens



Sparsity in LLM Workloads

- **Mixture of Experts (MoE):**
Increased model capacity without proportional compute cost
- **Structured sparsity in attention:** Sparse attention patterns (e.g., Mistral's sliding window, DeepSeek native sparse attention)



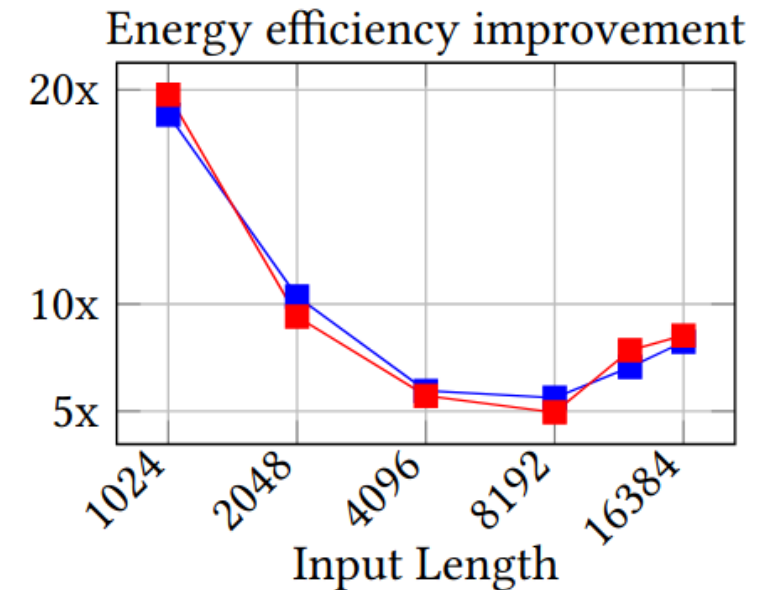
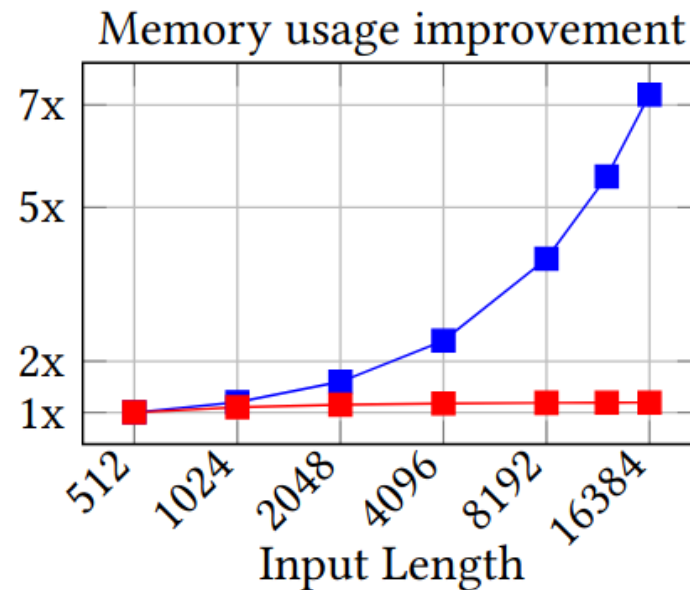
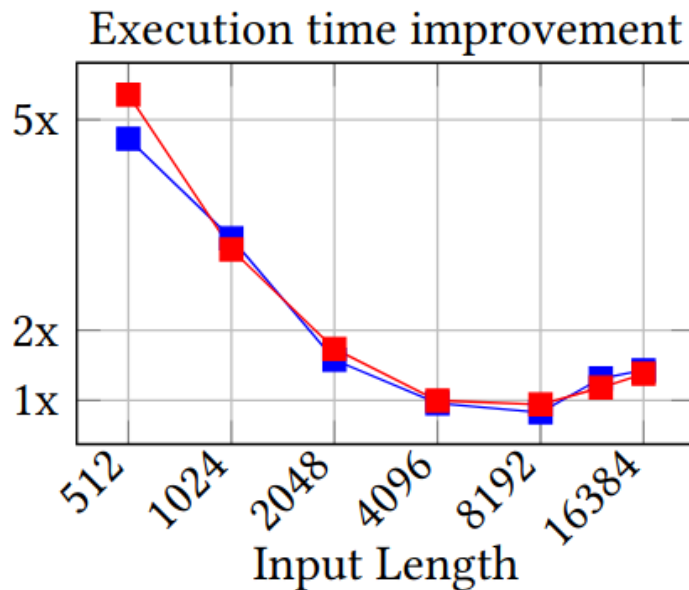
FPGA vs. GPU implementations

Single-batch:

Xilinx U55C (16nm) vs. Nvidia A100 (7nm)

FPGA Sliding-Window Attention achieves 15x energy-efficiency over GPU

- Improved execution time on worse technology node
- High energy efficiency gain



—■— SWAT vs. A100 Dense (both FP32) —■— SWAT vs. A100 Sliding Chunks (both FP32)

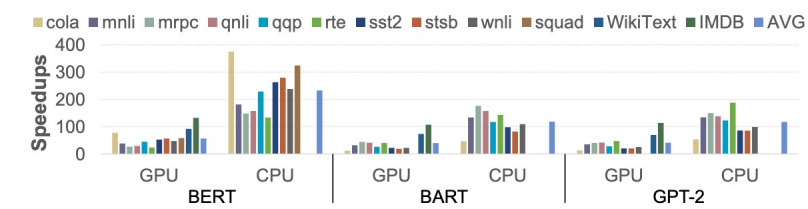
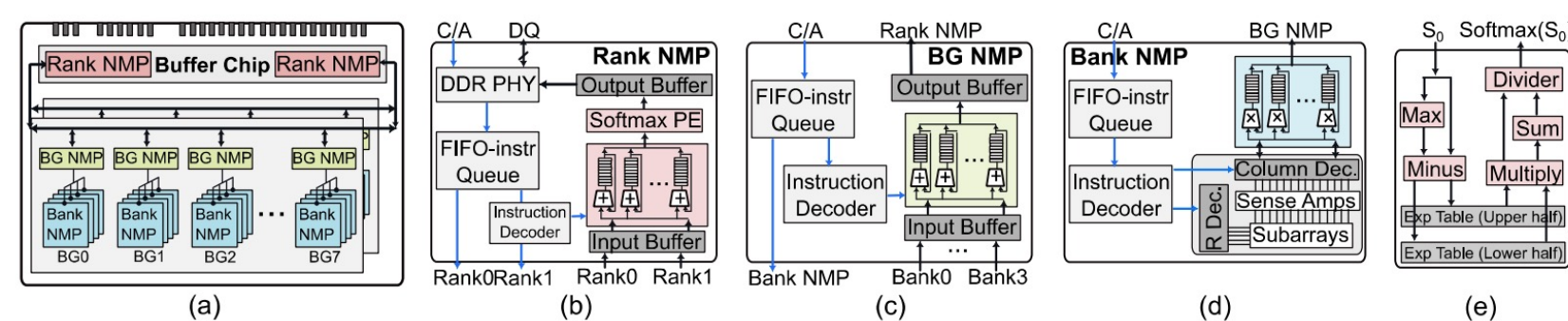


Fig. 8. The speedup improvement of SADIMM over GPU and CPU on BERT, BART, and GPT-2 benchmarks

SADIMM: Sparse Attention Accelerator with DIMM-based Near-memory Processing

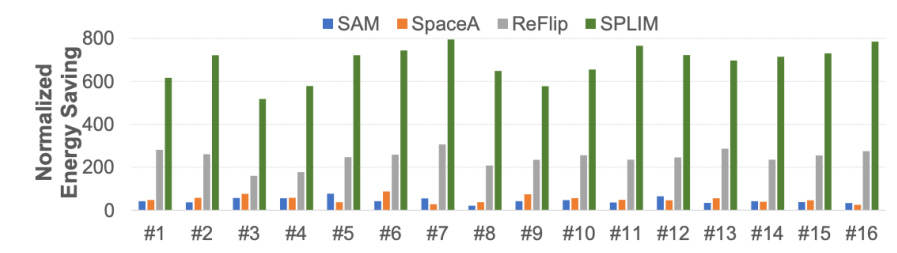
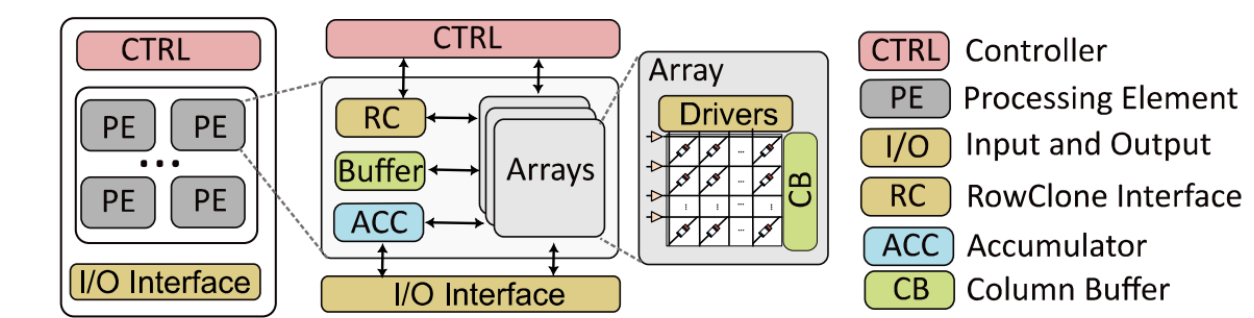
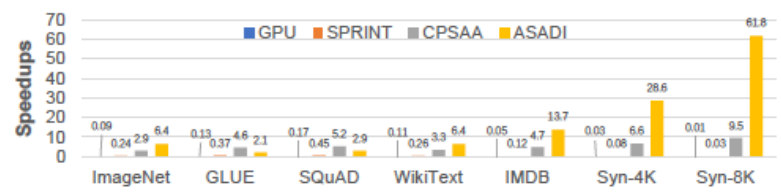
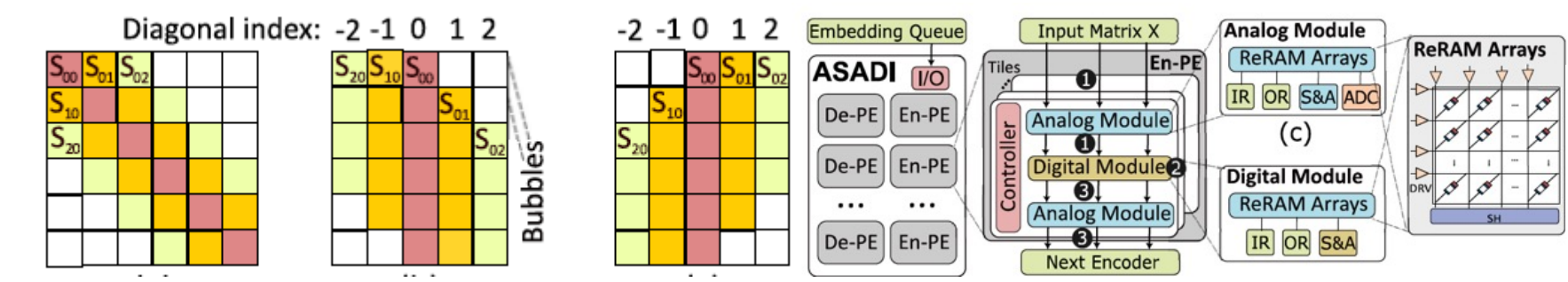


Fig. 15. Energy saving comparison between GPU baseline, SAM, SpaceA, ReFlip, and SPLIM (normalized to GPU)

SPLIM: Unstructured sparse matrix operations on structured in-situ computing

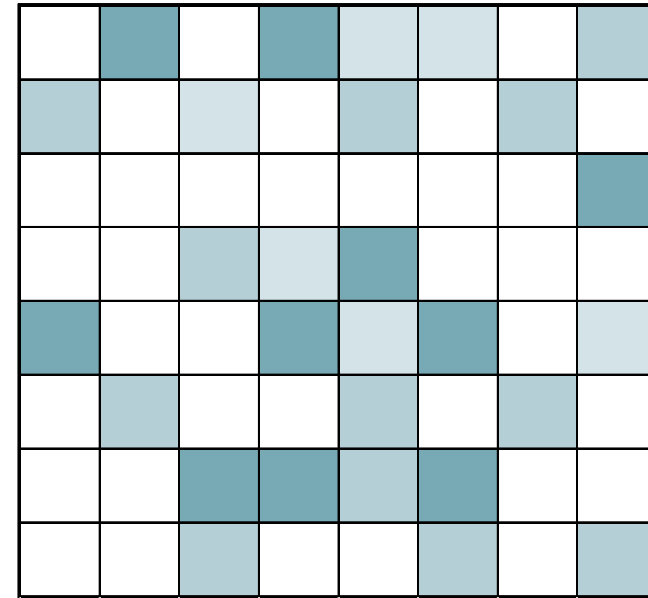
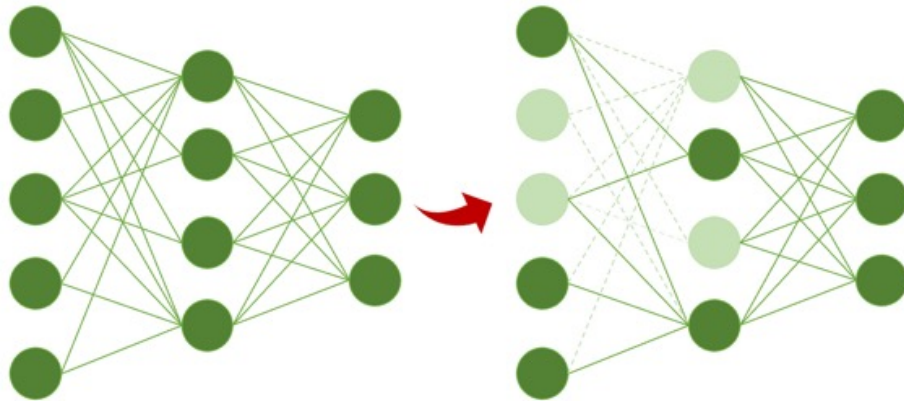


ASADI: In-memory computing for sparse attention

SADIMM: Accelerating Sparse Attention using DIMM-based Near-memory Processing. IEEE Transactions on Computers 2025
 SPLIM: Bridging the Gap Between Unstructured SpGEMM and Structured In-situ Computing. IEEE TCAD 2025
 ASADI: Accelerating Sparse Attention using Diagonal-based In-situ Computing. HPCA 2024

Sparsity in LLM Workloads

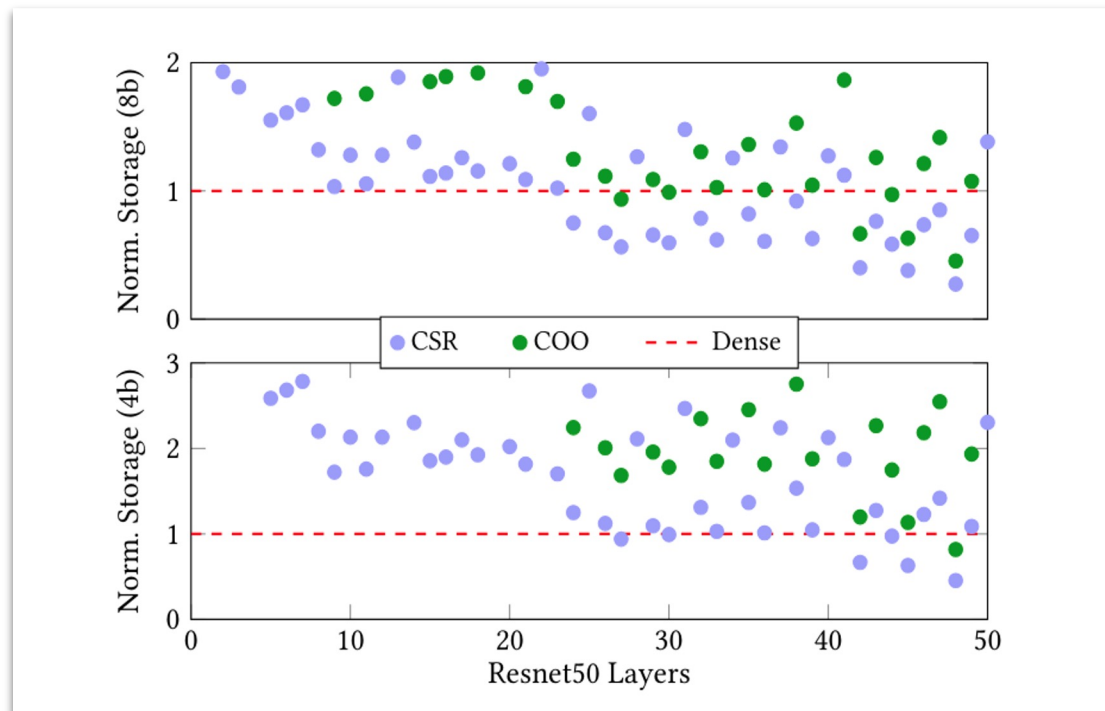
- **Mixture of Experts (MoE):**
Increased model capacity without proportional compute cost
- **Structured sparsity in attention:** Sparse attention patterns (e.g., Mistral's sliding window, DeepSeek native sparse attention)
- **Unstructured Sparsity:** Individual parameters becoming zero via pruning, activation, or training-time sparsity constraints



Sparsity creates opportunity for efficiency...
Can hardware realize it?

Unstructured Sparsity: Metadata Cost

At 4-bit or 8-bit quantization, sparse matrix representation formats used by hardware accelerators (CSR, COO) require more storage than Dense format due to meta-data overhead!!



Dense

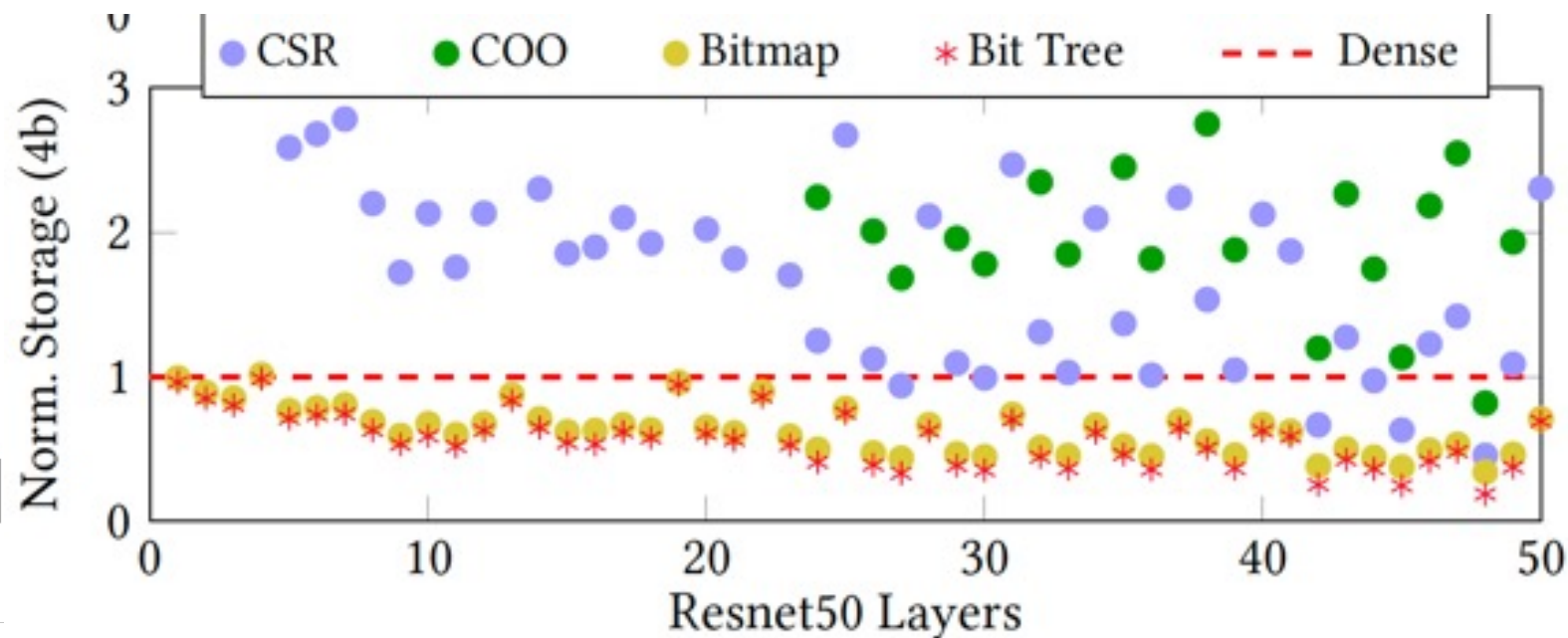
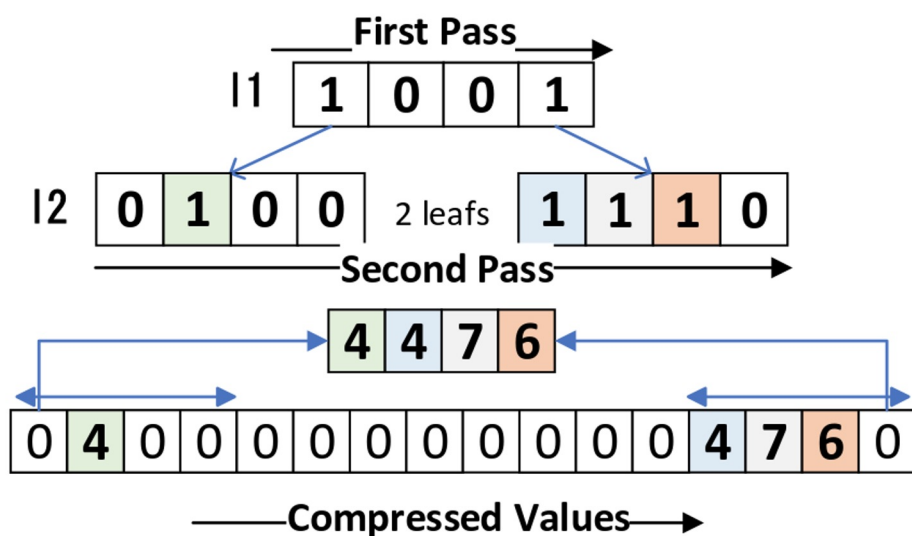
a	b	c		
		d		
	e	f		g
				h
	i		j	

CSR format

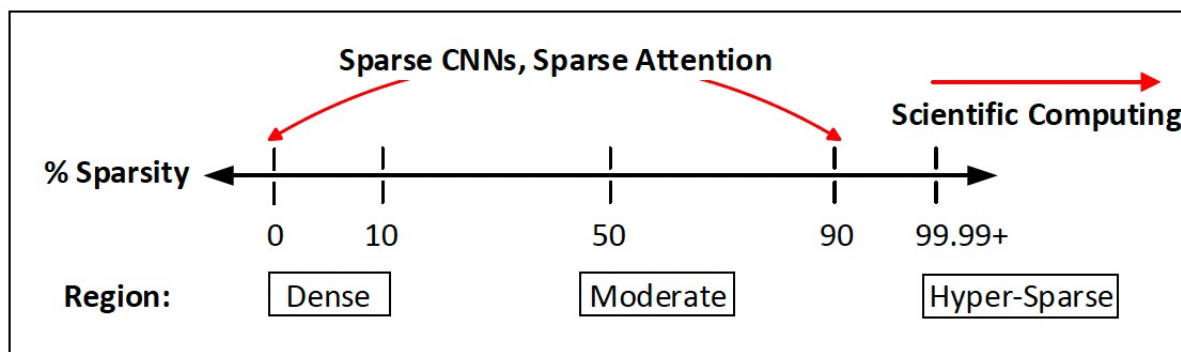
rpt	0	3	4	7	8	10				
col	0	1	2	2	1	2	4	4	1	3
val	a	b	c	d	e	f	g	h	i	j

Sparse storage formats lead to inflation not compression!
Bitmap decoding is costly due to long runlength of zeros

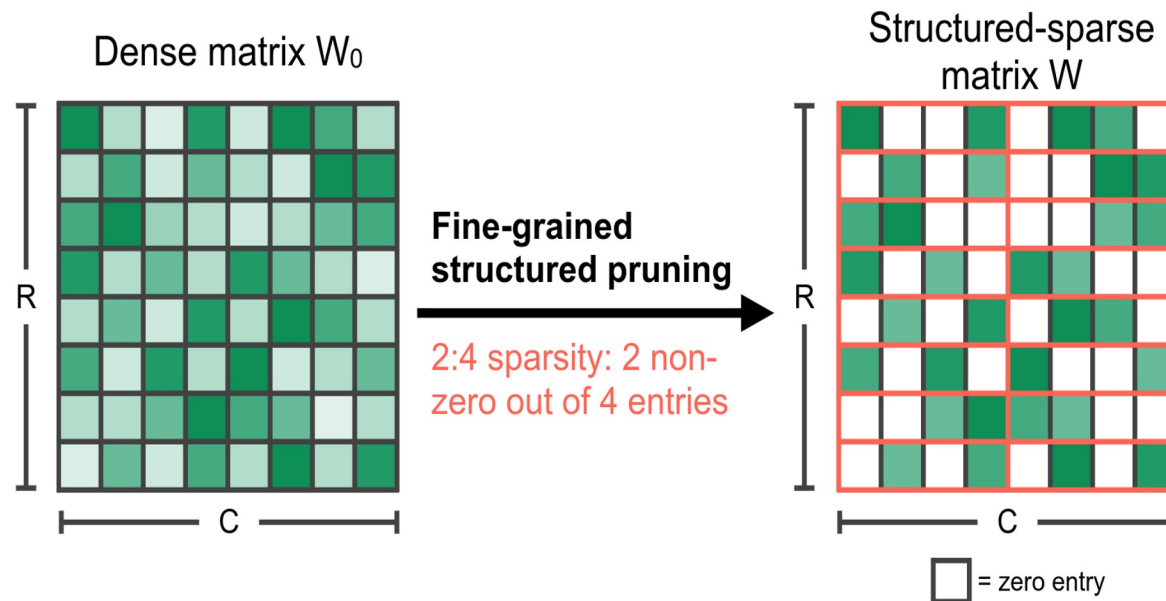
Bit-Tree Format Reduces Metadata Overhead



- ✓ Storage: Same as ideal bitmap
- ✓ Fast Decoding: N-passes over N-level Bit-Tree
- ✓ Handles all sparsity level



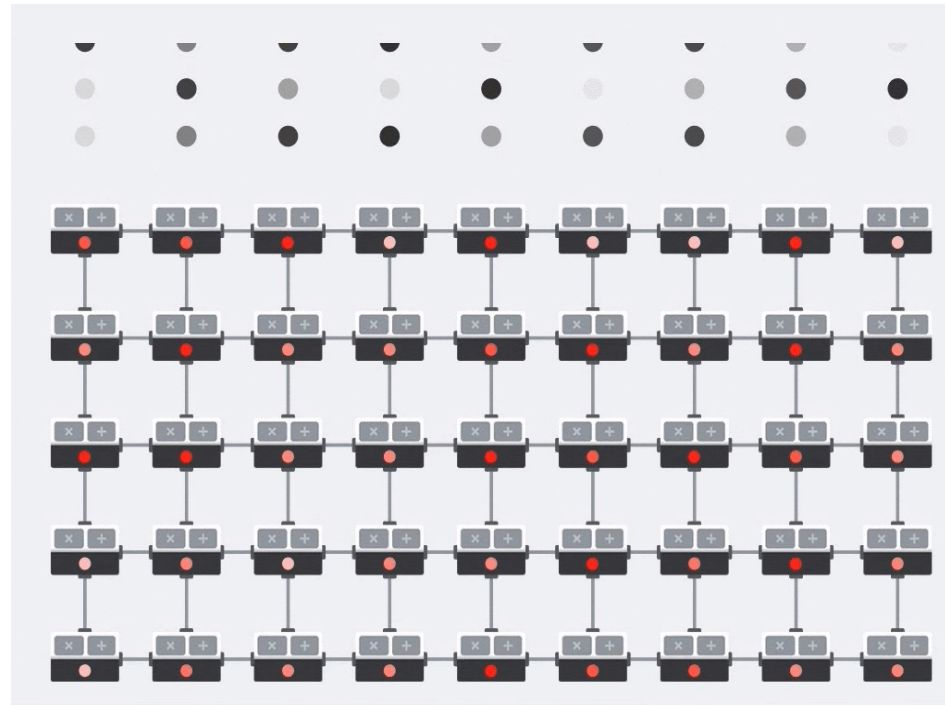
GPUs Adopted Restricted Structured Sparsity



GPUs have limited support for fine-grained unstructured sparsity

Hardware-Algorithm Gap on Systolic Array

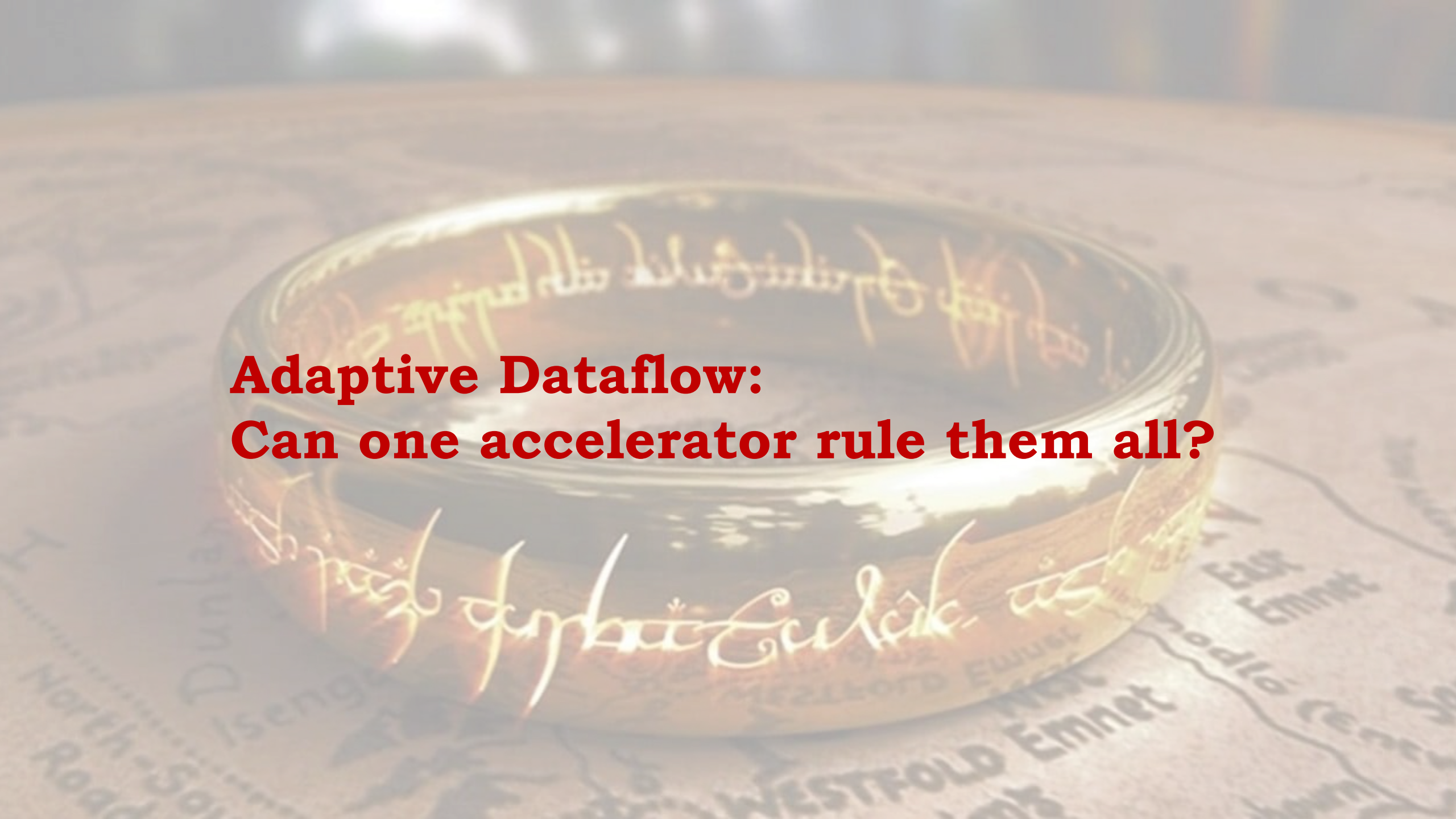
- Hardware needs structured dataflow, but AI creates unstructured sparsity
- Scattered non-zeros cannot match fixed hardware dataflow



Google TPU Matrix Multiply Unit (MXU) Dataflow

From FLOPS to Intelligence per Joule

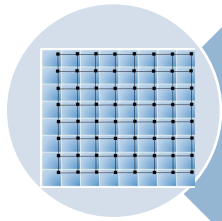
- Arithmetic is becoming cheap
- AI models are becoming sparse, irregular, and dynamic
- Bottleneck shifts from arithmetic to data movement and control
- Today's hardware is fundamentally mismatched with emerging sparse workloads
- Future accelerator must adapt its dataflow to the model to achieve high Intelligence/Joule
- Hardware must follow algorithm, not the other way around

A golden ring with a large, clear, oval-shaped gemstone is positioned in the center of the frame. The ring is resting on a parchment map that features various place names and geographical lines. The map is slightly out of focus, with the ring being the primary subject. The text is overlaid on the ring and the map.

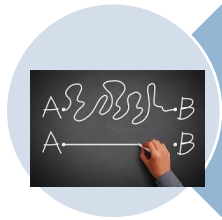
**Adaptive Dataflow:
Can one accelerator rule them all?**

Ideal Dataflow Accelerator

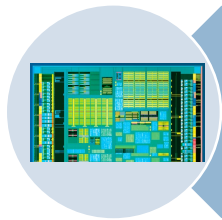
- Software (compiler) instantiates the dataflow corresponding to different applications onto the same hardware
- Should appear as easily programmable to software engineers
- Performance, energy-efficiency closer to domain-specific accelerators



Embrace Parallelism



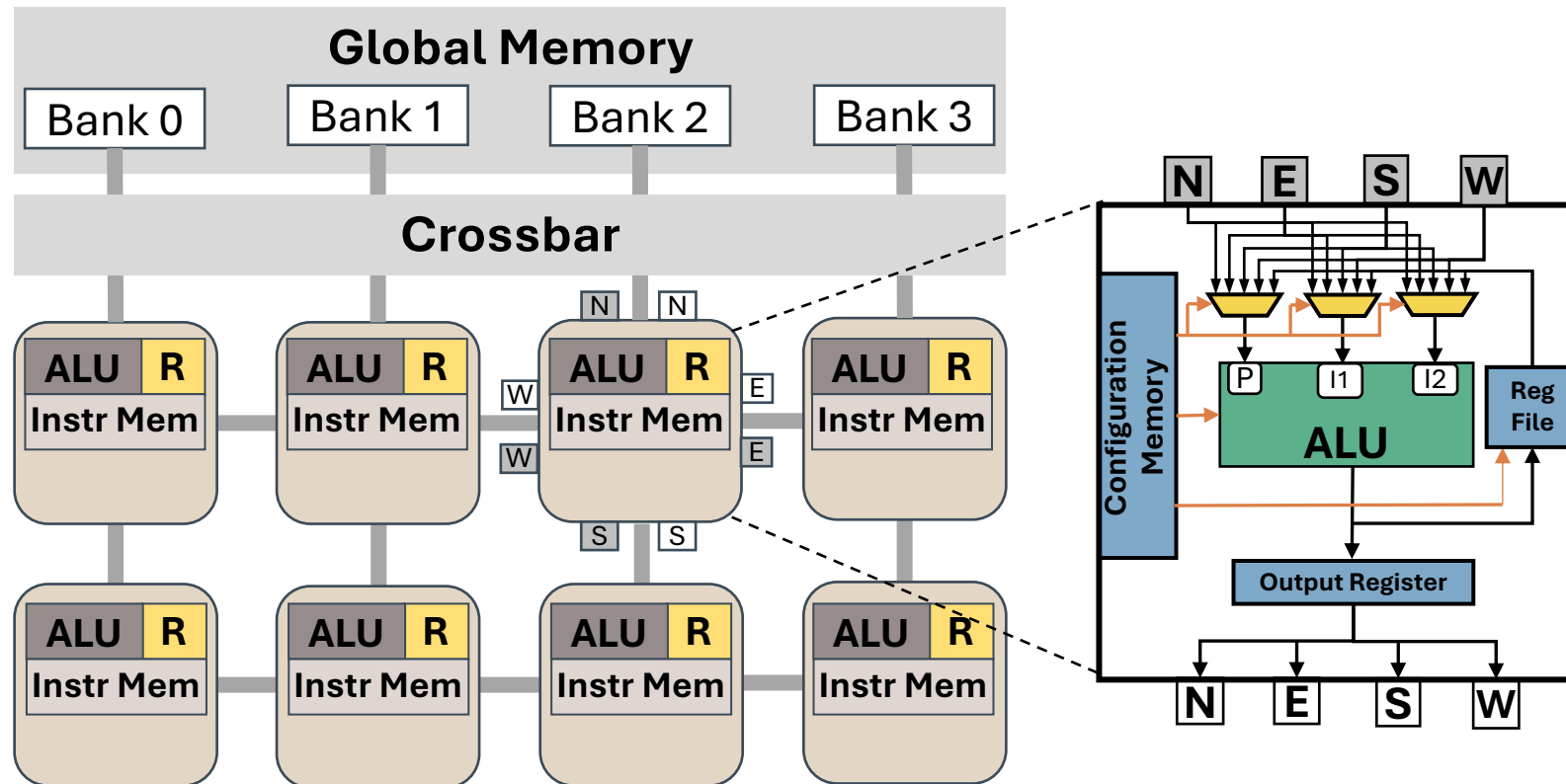
Keep Hardware Simple



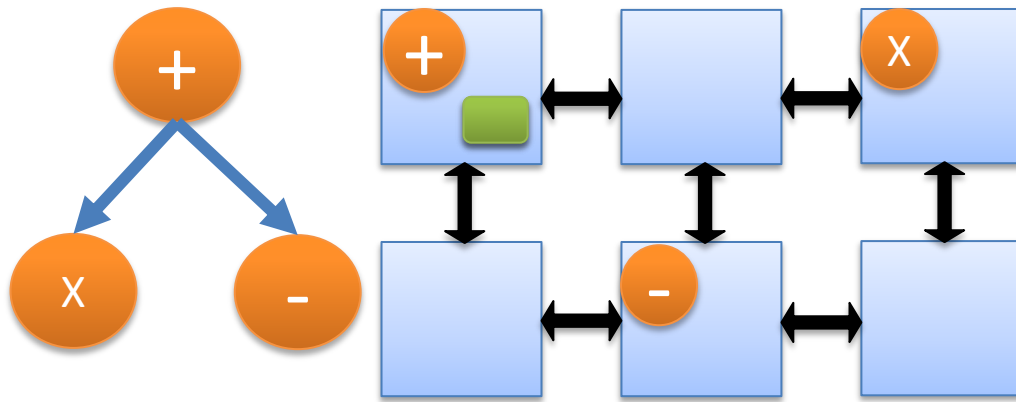
Expose to Software

CGRA Accelerator: Reconfigurable Dataflow

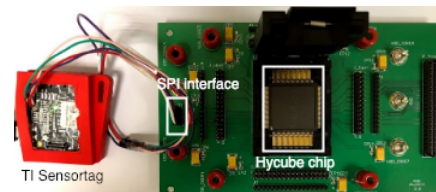
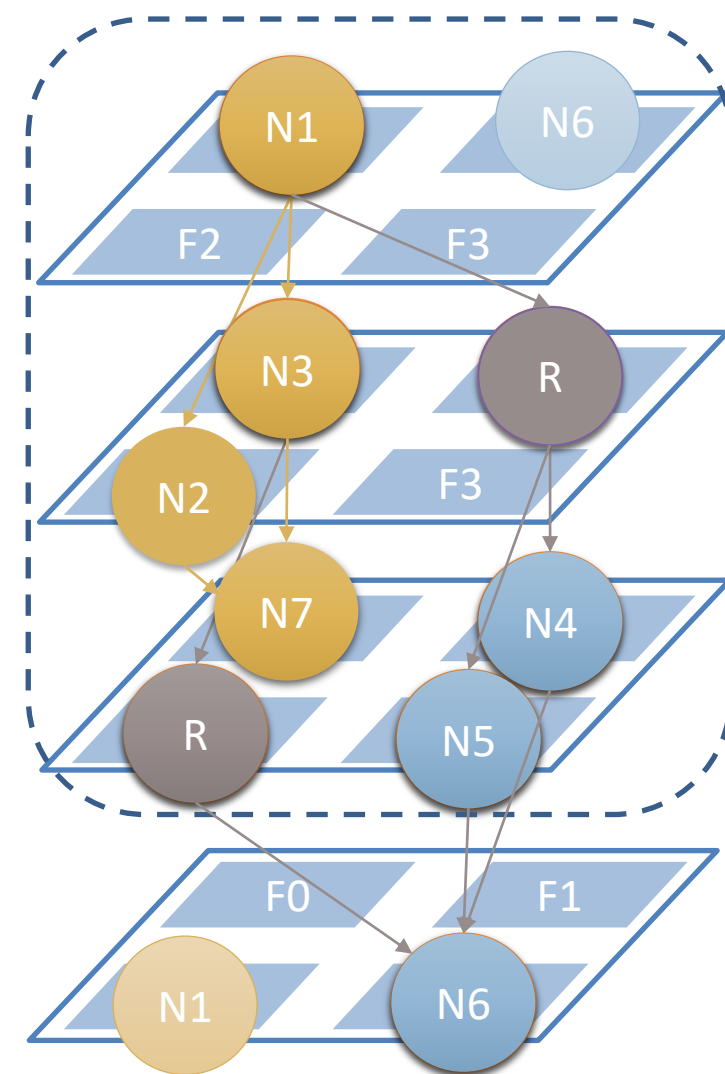
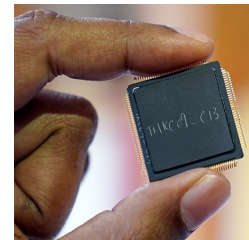
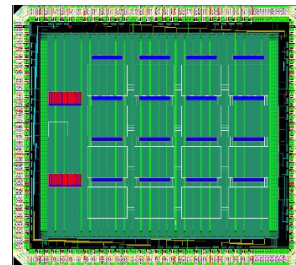
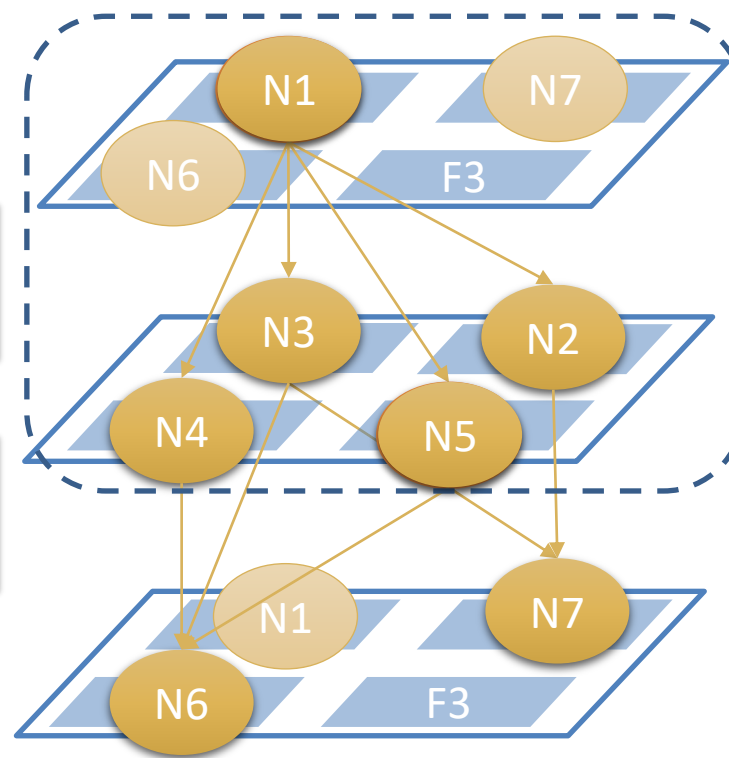
- **Parallelism:** Array of simple processing elements (PE)
- **PE operations and routing can be configured per cycle**
- **Exposed to software:** Compiler sets the configuration according to dataflow
- **Simple hardware and hence low power:** During execution, PE array and switches simply follow the pre-defined configurations



HyCUBE CGRA @ NUS

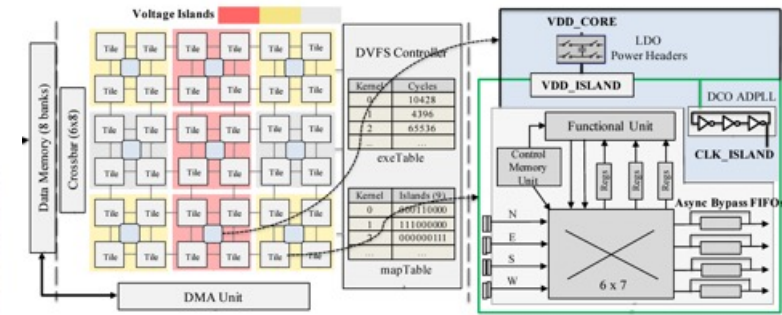
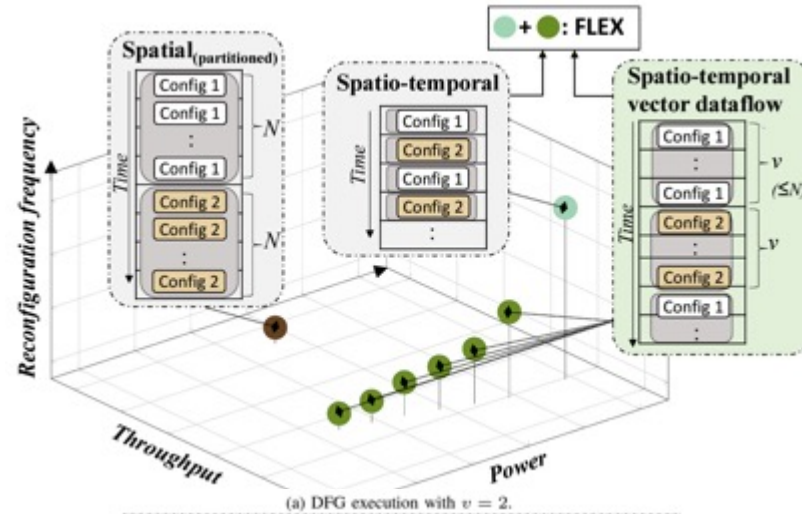
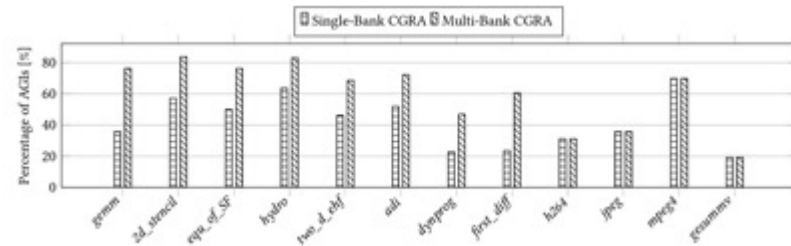


- Reconfigurable single-cycle multi-hop interconnect
- Creates large dynamic neighborhood reachable within single-cycle through asynchronous bypassing of intermediate PEs

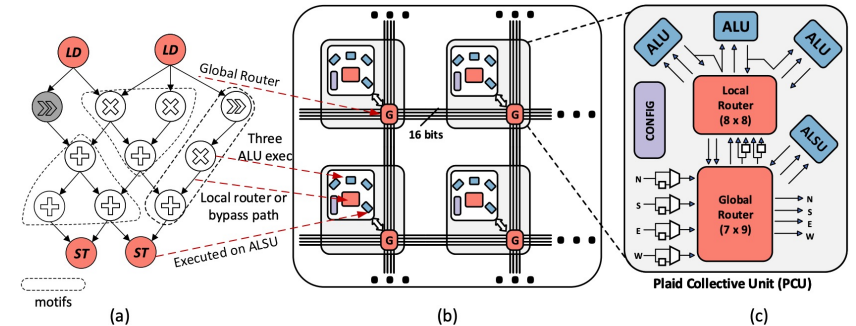
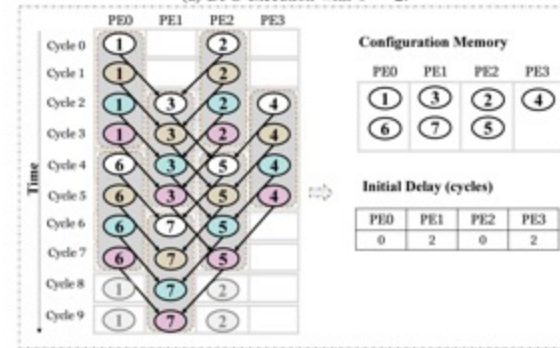


22

Improving CGRA Efficiency



ICED:
DVFS-aware CGRA
1.32x energy-efficiency over
CGRA at same performance



CASCADE
Decoupled Access-Execute CGRA
3x speedup, 2.2x performance/watt
over iso-area CGRA

FLEX:
Spatio-temporal vector dataflow
45% less energy. 1.9x power efficiency
at same performance over CGRA

Plaid:
Reduced routing provision
43% less power
46% less area

CASCADE: High Throughput Data Streaming via Decoupled Access/Execute CGRA. CASES 2019

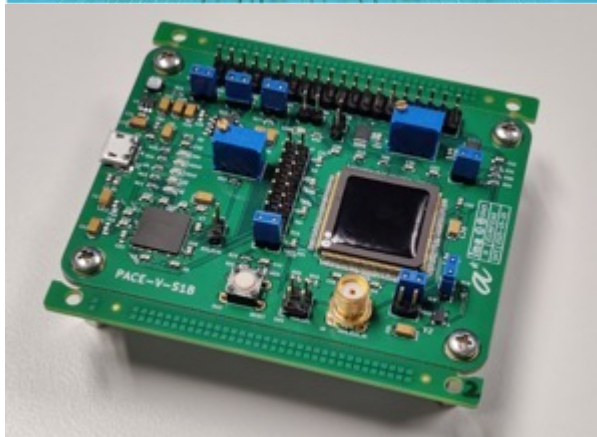
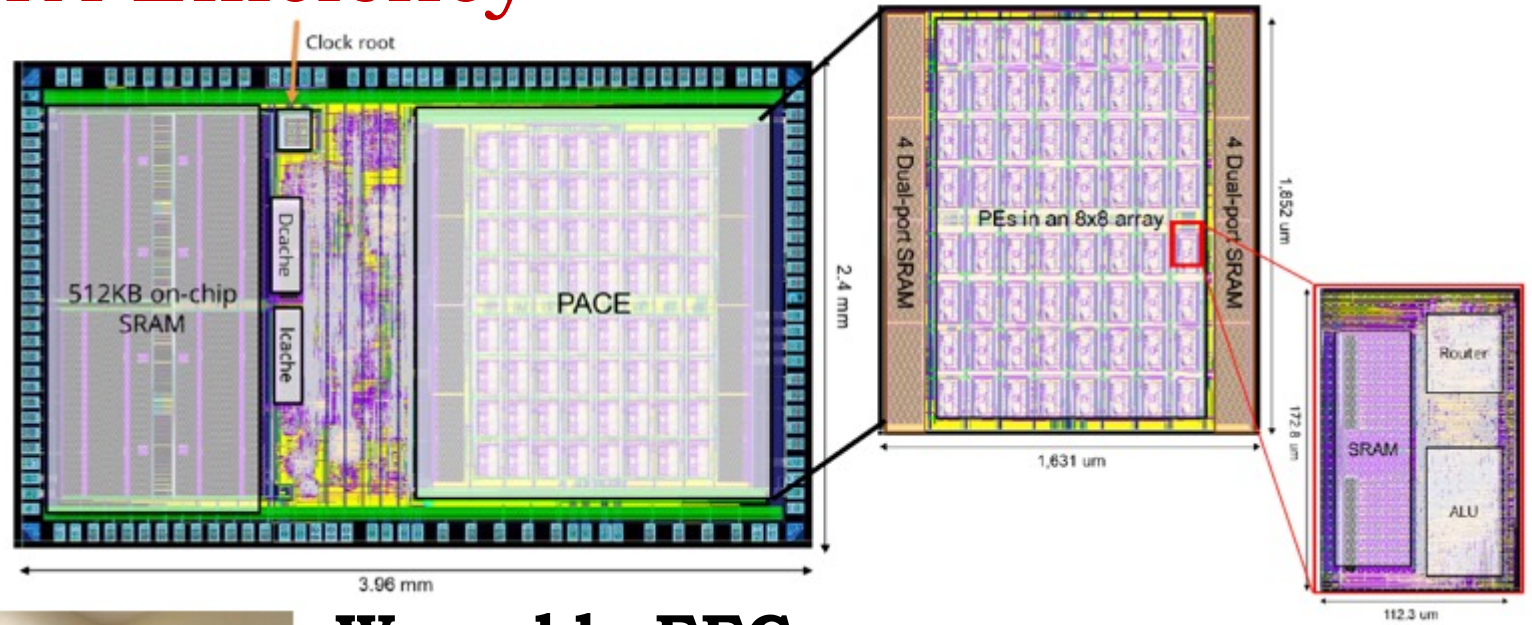
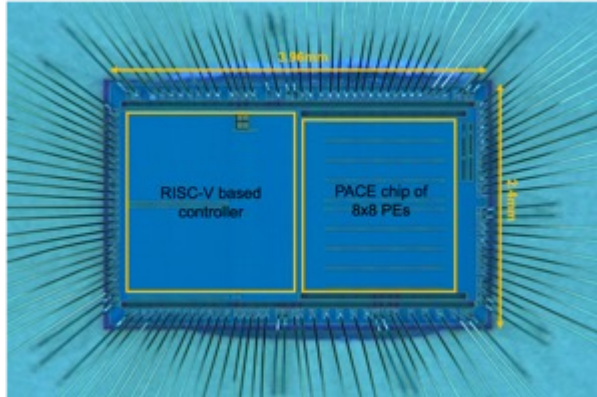
FLEX : Introducing FLEXible Execution on CGRA with Spatio-Temporal Vector Dataflow. ICCAD 2023

ICED: An Integrated CGRA Framework Enabling DVFS-Aware Acceleration. MICRO 2024

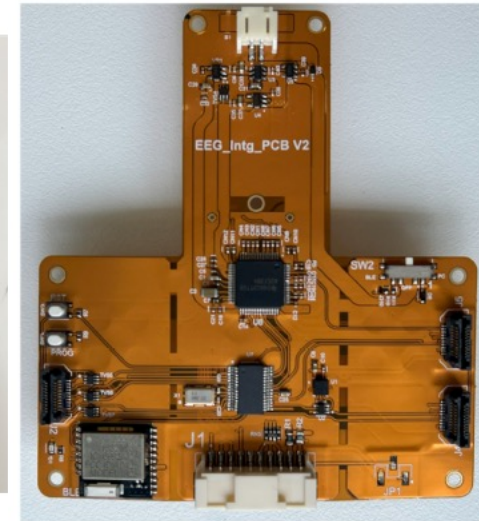
Plaid: Enhancing CGRA Efficiency Through Aligned Compute and Communication Provisioning. ASPLOS 2025

NUS PACE CGRA: SOTA Efficiency

8x8 CGRA: 1.1 mW at 10MHz
582 GOPS/W at 0.45V, 40nm ULP
Estimated 1 TOPS/W at 22nm

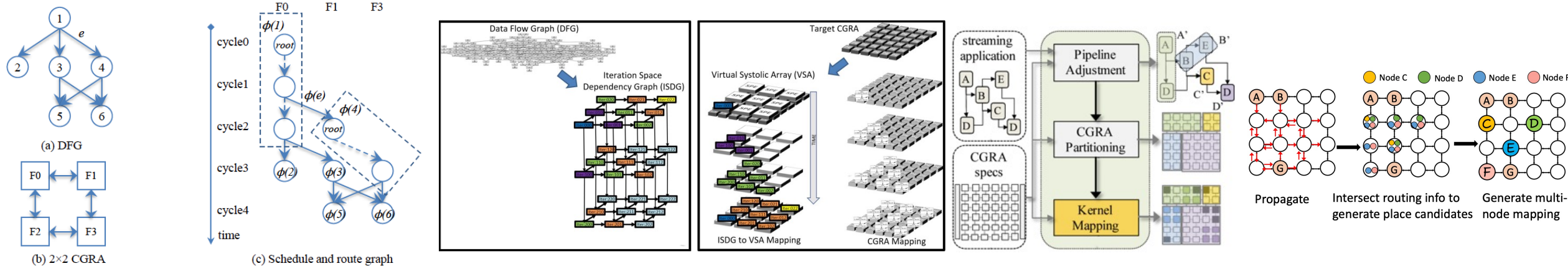


Wearable EEG



Dataflow Synthesis: Improving CGRA Programmability

- Graph Minor (2012): Theoretically optimal based on graph minor modeling
- ChordMap (2020): Streaming dataflow application synthesis
- HiMap (2021), Panorama (2022): Scalable mapping through hierarchical abstractions
- LISA (2022): Automated compiler synthesis using GNN
- Rewire (2025): Consolidated routing



Graph Minor Approach for Application Mapping on CGRAs. Best paper award FPT 2012 TRETS 2014

ChordMap: Automated Mapping of Streaming Applications onto CGRA. TCAD 2021

HiMap: Fast and Scalable High-Quality Mapping on CGRA via Hierarchical Abstraction. DATE 2021 TCAD 2022

Panorama: Divide-and-Conquer Approach for Mapping Complex Loop Kernels on CGRA. DAC 2022

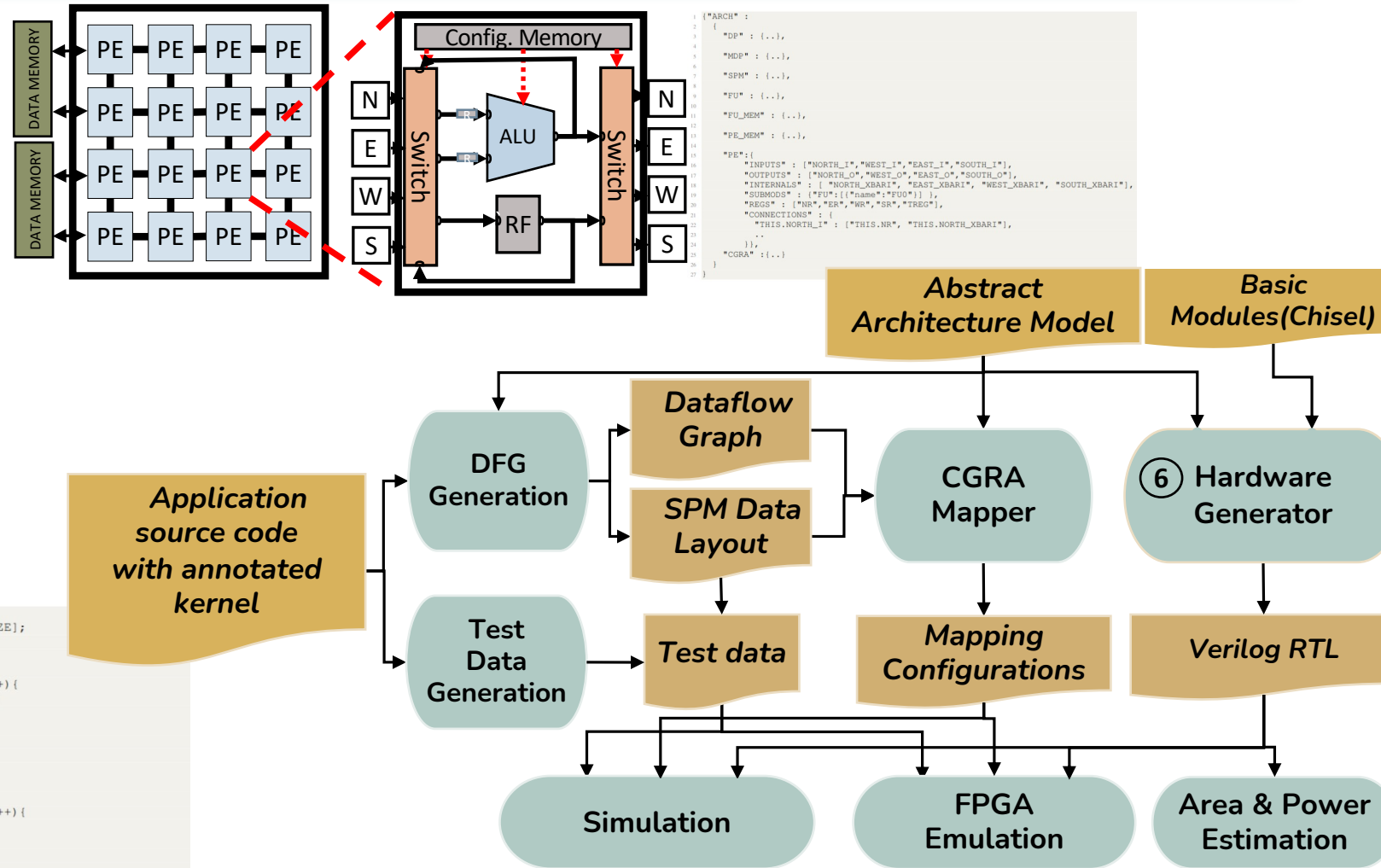
LISA: Graph Neural Network based Portable Mapping on Spatial Accelerators. HPCA 2022

Rewire: Advancing CGRA Mapping Through a Consolidated Routing Paradigm. DAC 2025

NUS Morpher Open-Source CGRA Toolchain

Our position paper at ICCAD 2025:

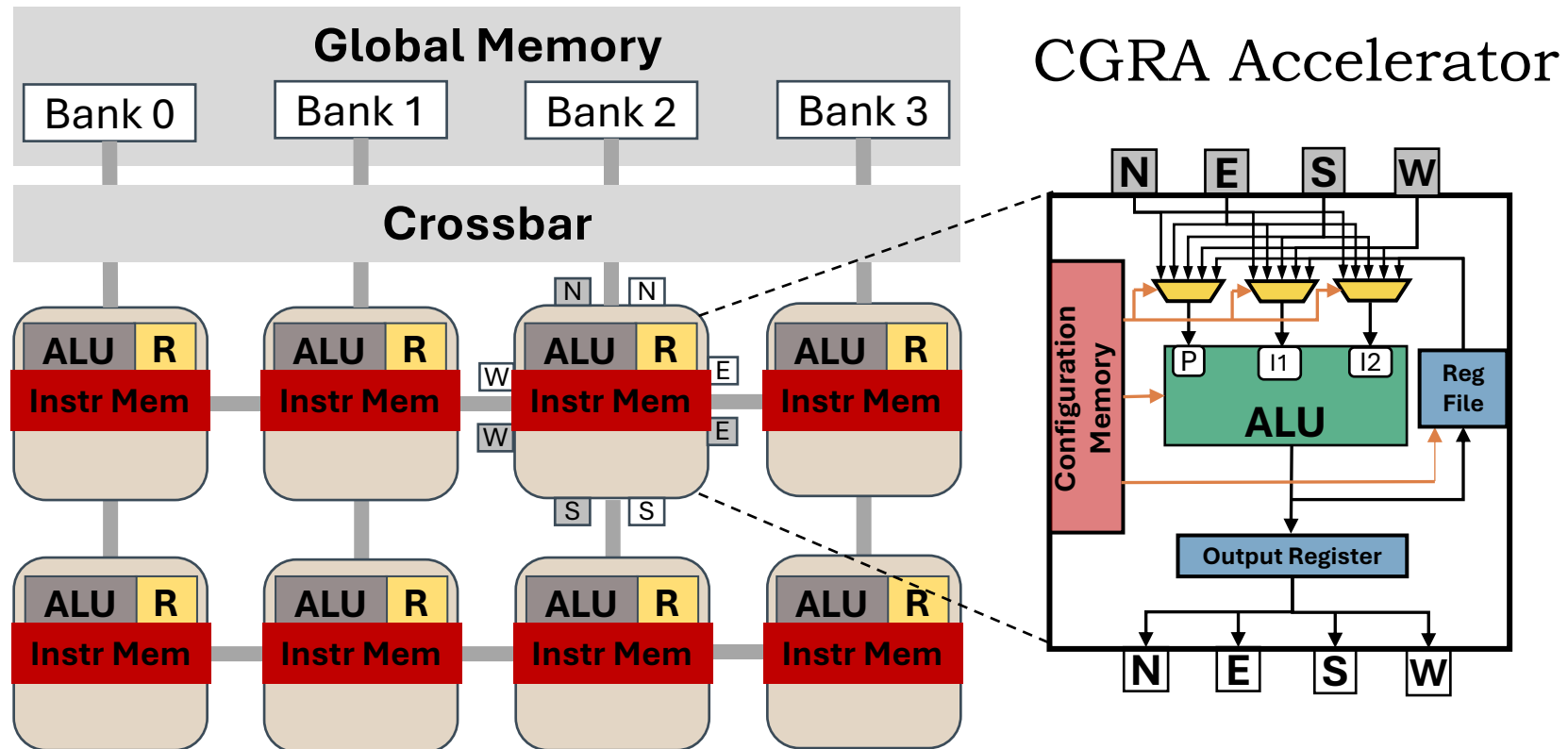
Building an Open CGRA Ecosystem for Agile Innovation



SCAN ME

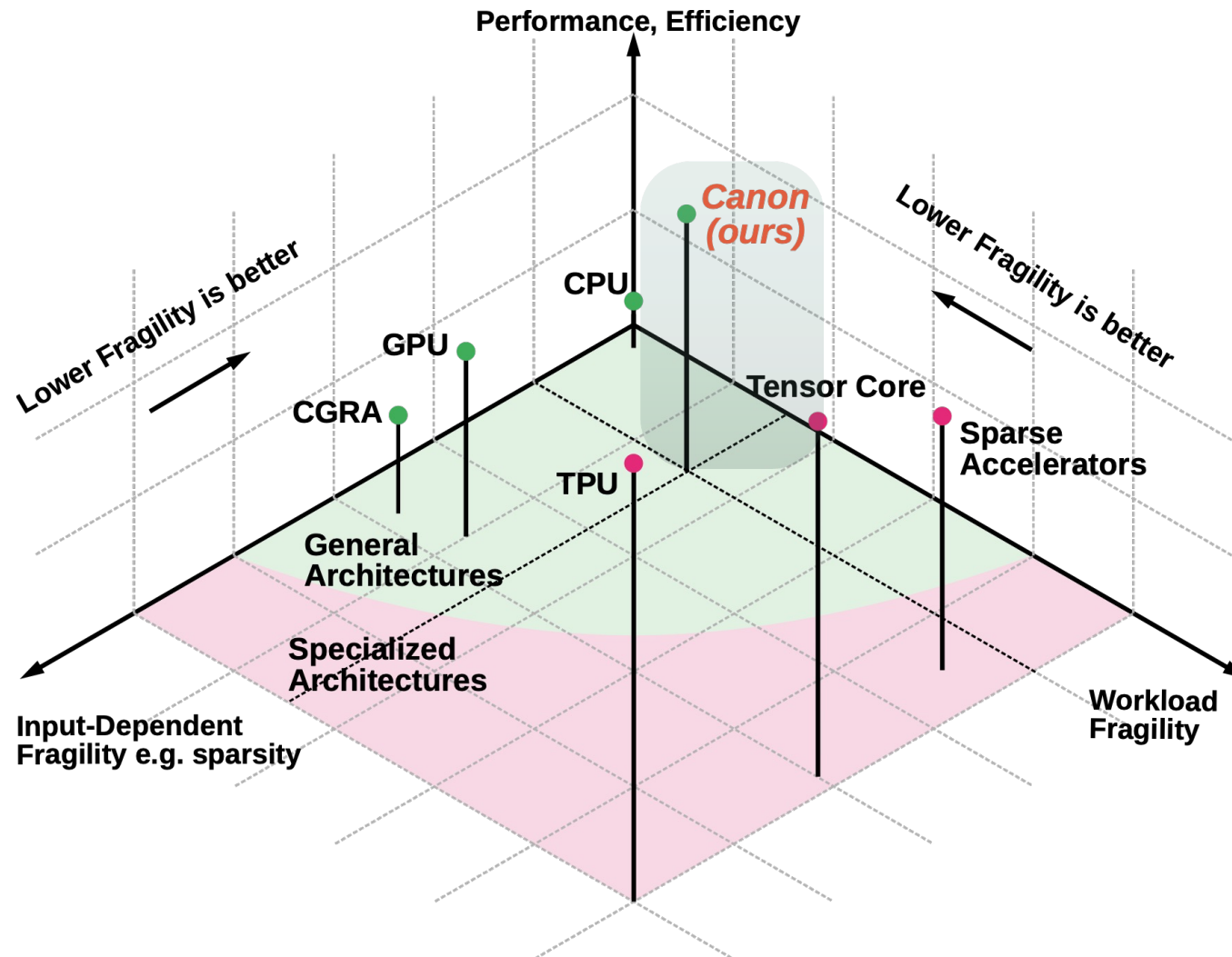
Making Reconfigurability Data-Driven and Cheap

- As data becomes smaller (FP4), instruction cost is very high!
- Domain-specific accelerator has minimal config storage and access cost
- CGRA accelerator introduces config storage and access cost

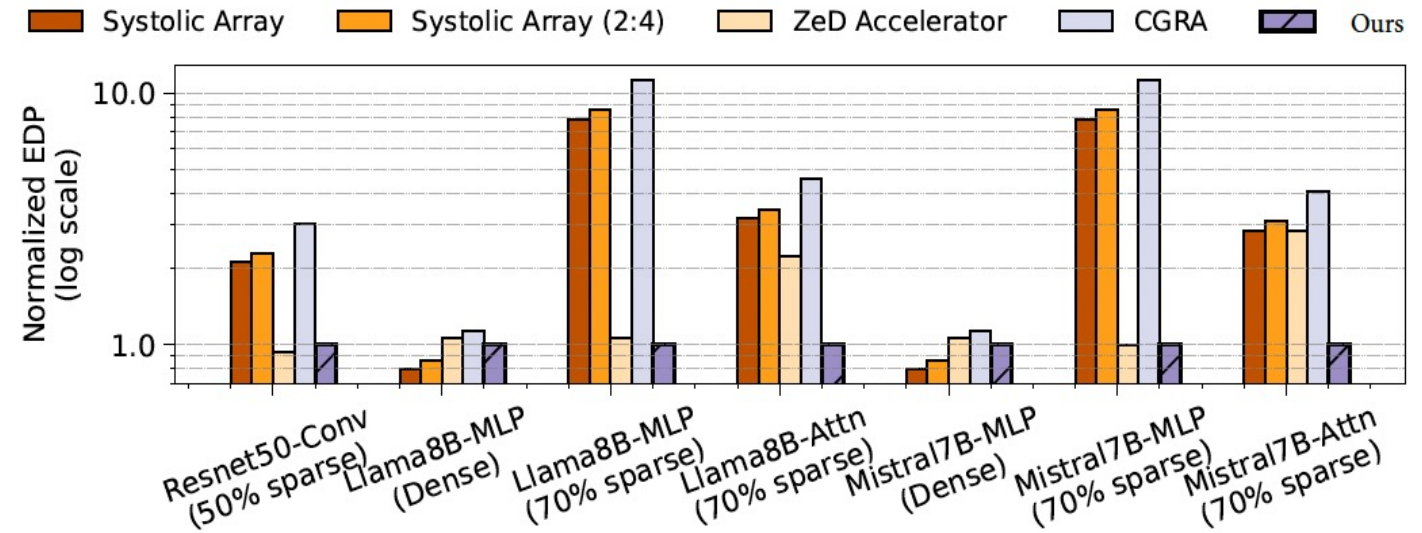
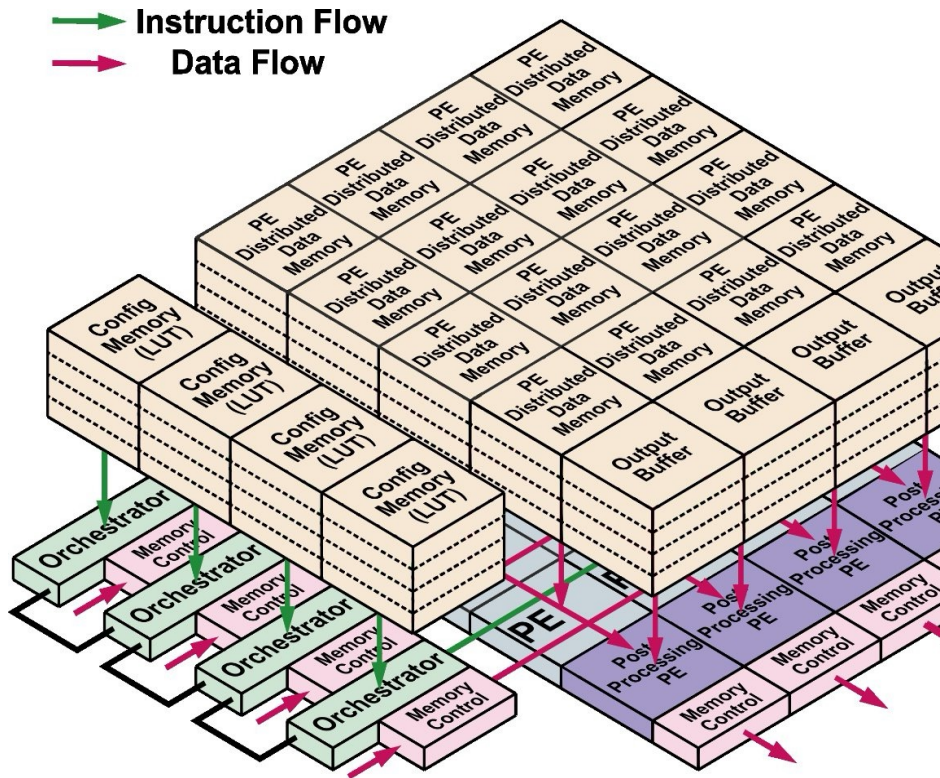


Canon: Data-Adaptive Dynamic Execution

Dataflow follows sparsity and metadata at runtime



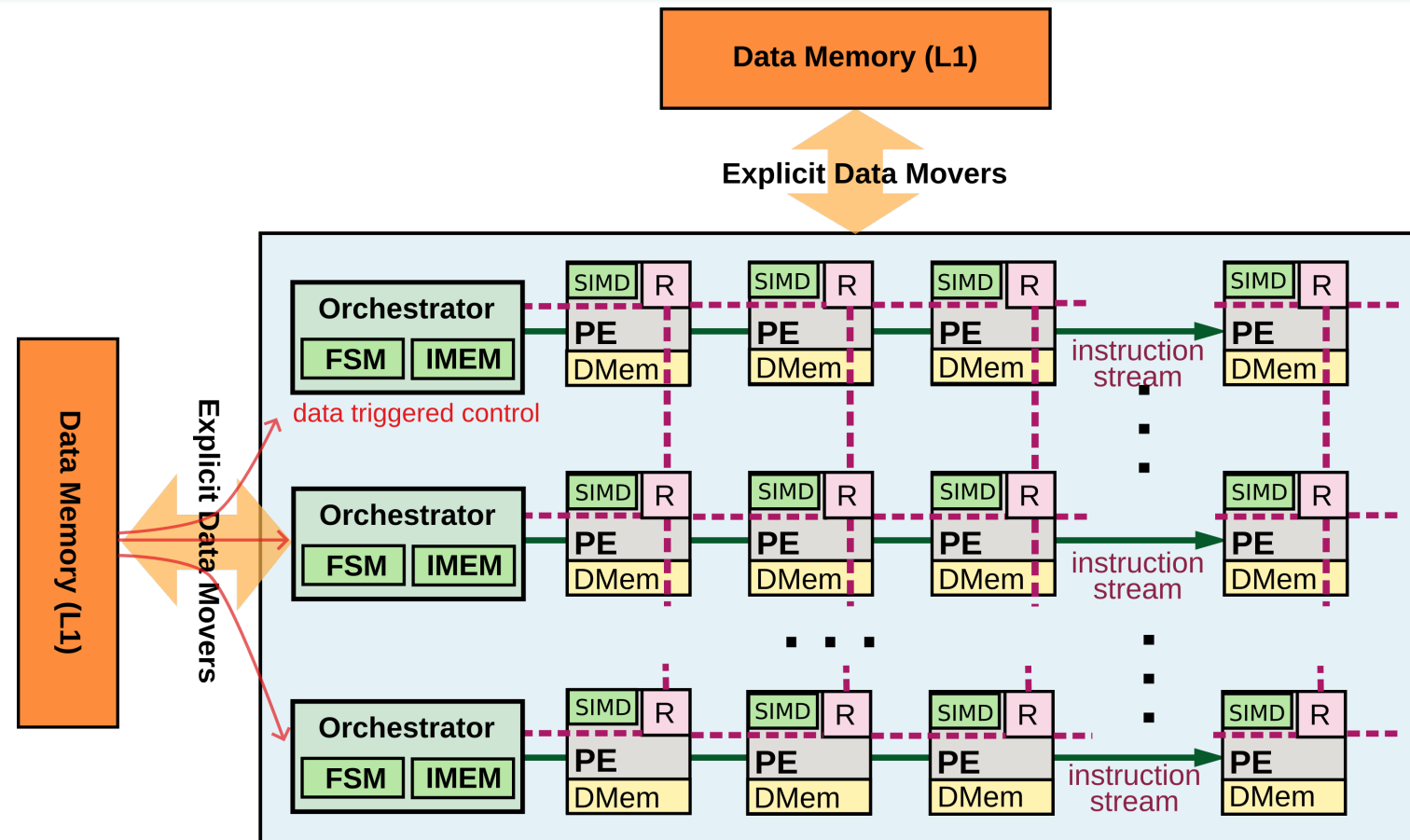
Canon: Data-driven Dynamic Execution Orchestration



- Minimalist reconfigurable processing array that adapts to diverse sparsity with minimal reconfiguration cost
- Smart algorithm that maps arbitrary dataflows on the fabric
- High parallel processing while handling complex dataflow dependencies

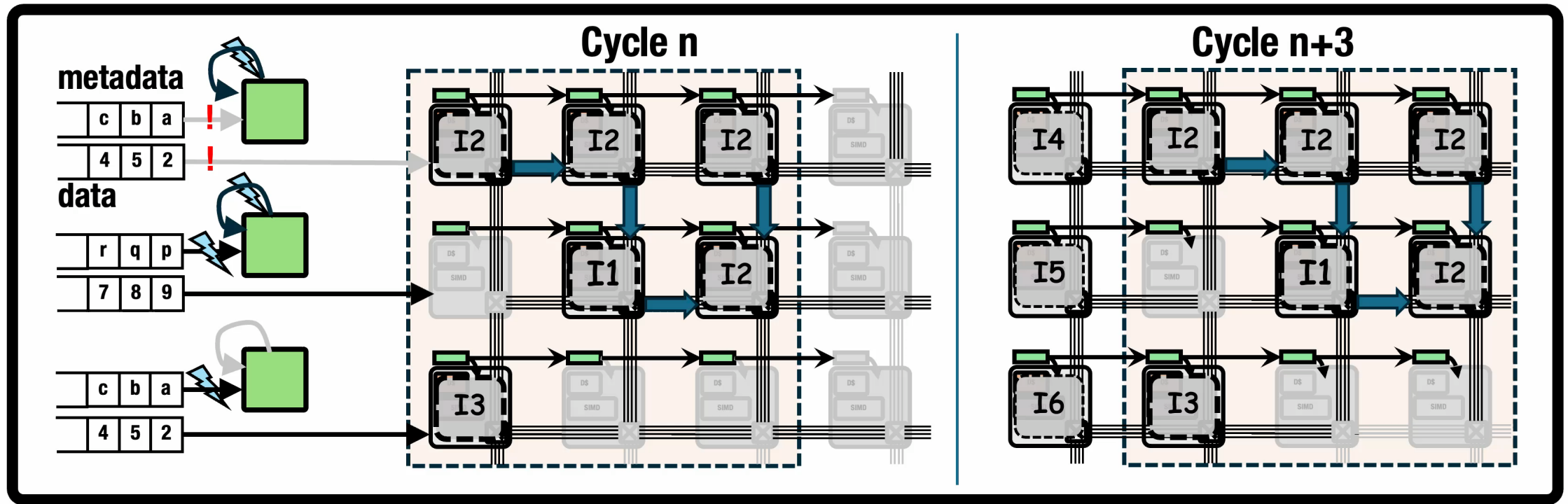
Canon: Data-driven Dynamic Execution Orchestration

- compiler handles regular parts of the application and orchestrates a high level dataflow for sparse workload
- Hardware FSMs trigger instruction issue dynamically in response to kernel state and input data
- Programmability cost amortized over compute row (Time-lapsed SIMD)



Canon Time-Lapsed SIMD

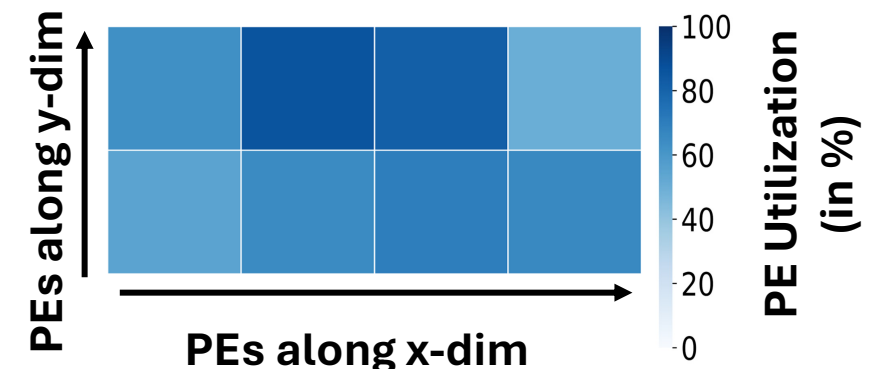
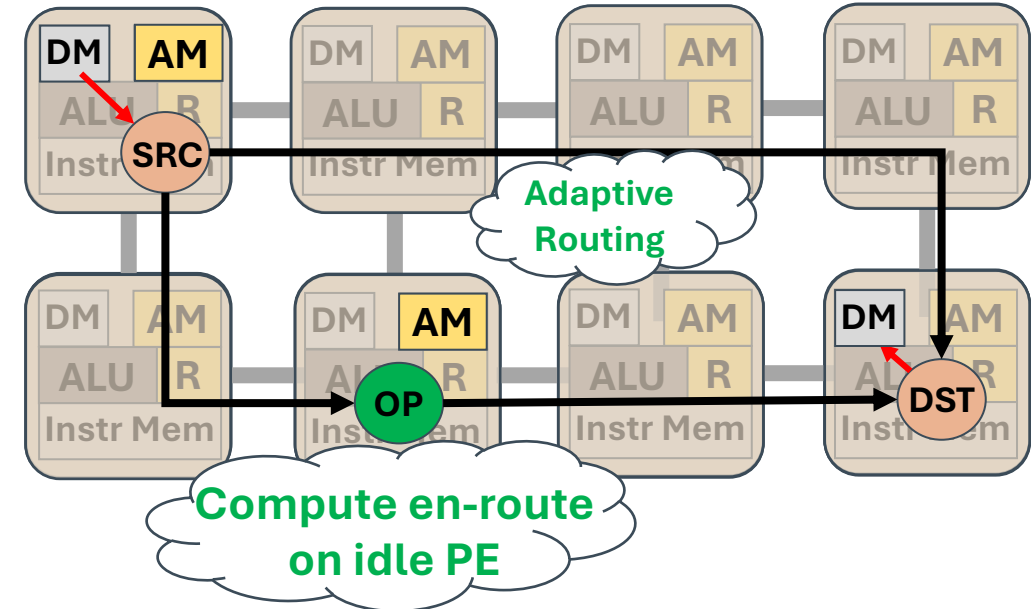
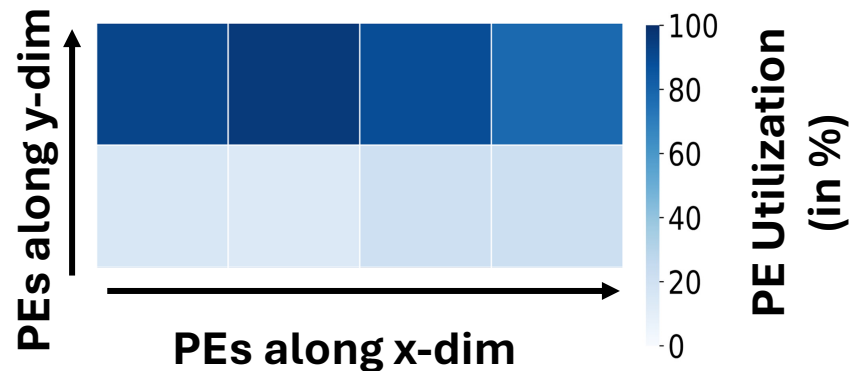
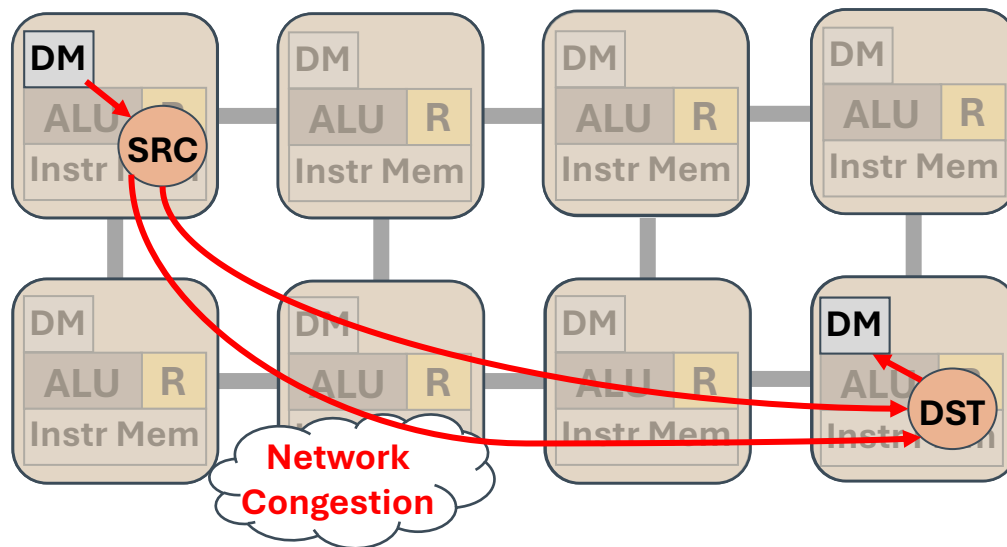
Amortization of Programmability Cost



Nexus Machine: System-adaptive execution

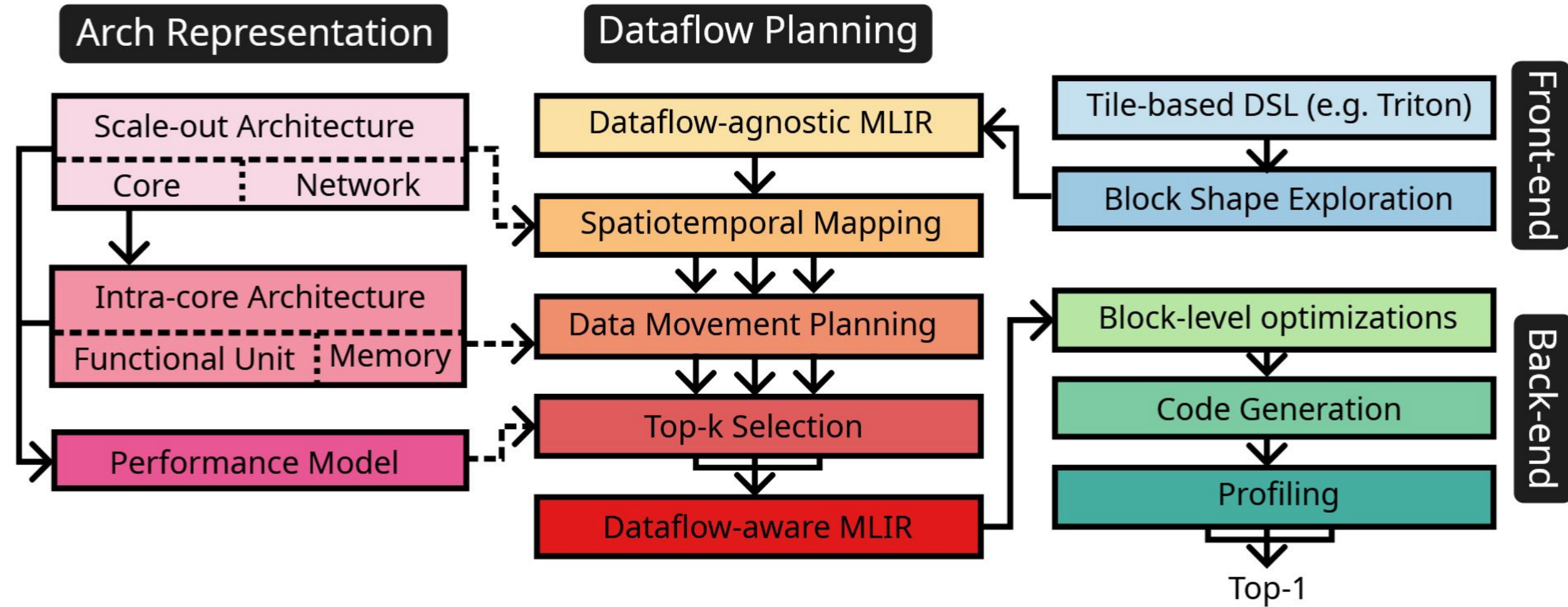
Routing and placement follow load and congestion at runtime

- Active Messages carrying both instruction and data trigger compute en-route on idle PE improving load balancing
- 1.9× performance and 1.7× fabric utilization than generic CGRA for irregular workload

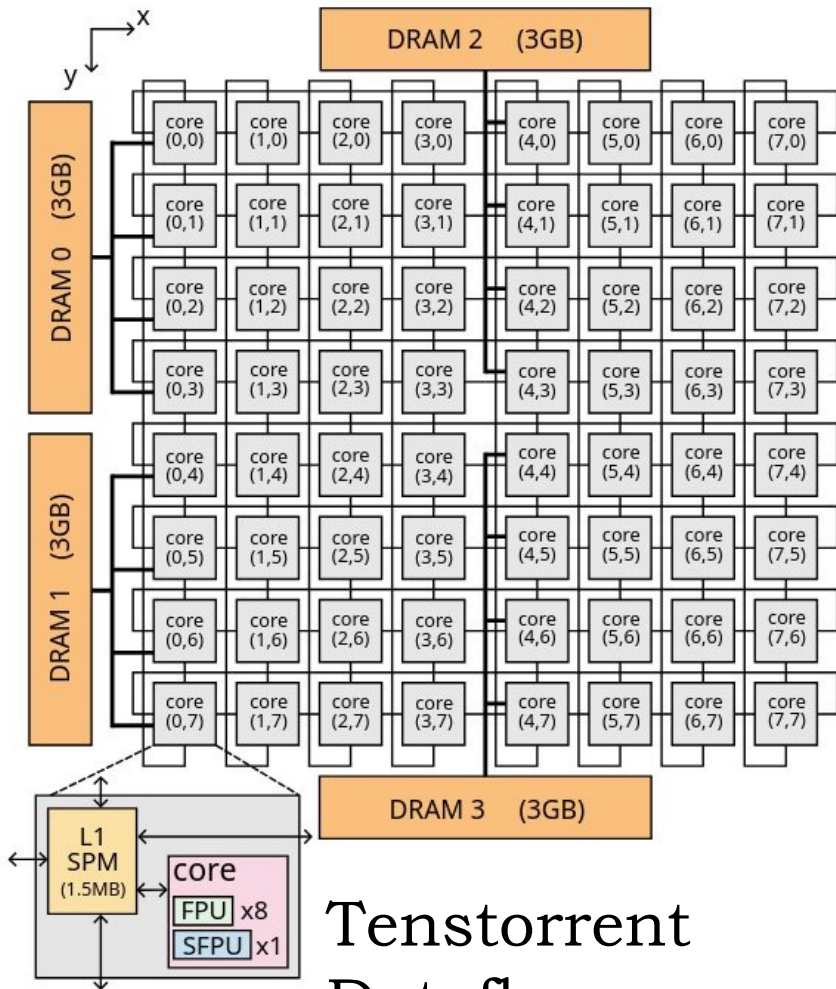


But who programs the dataflow?

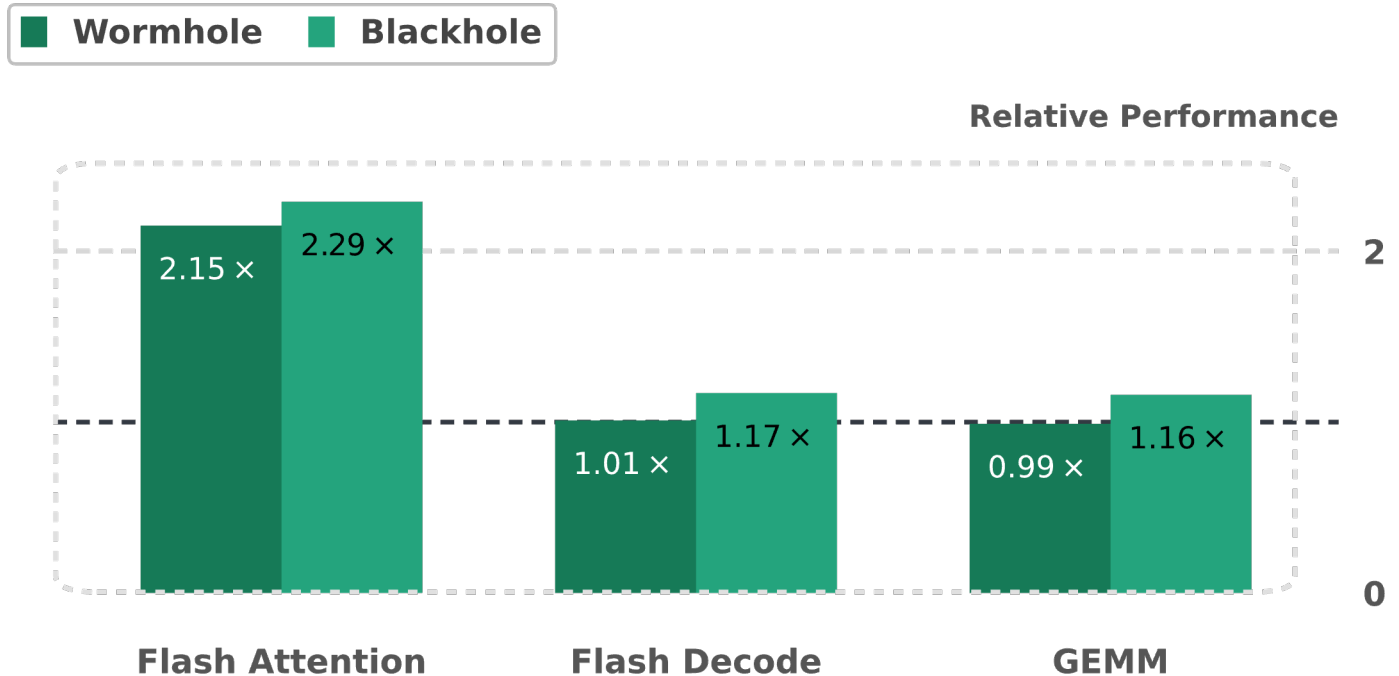
TileLoom: Compiler for Dataflow Planning



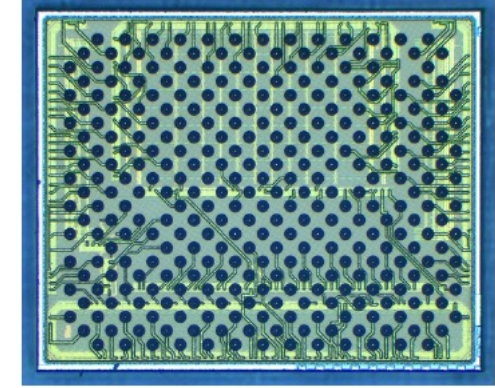
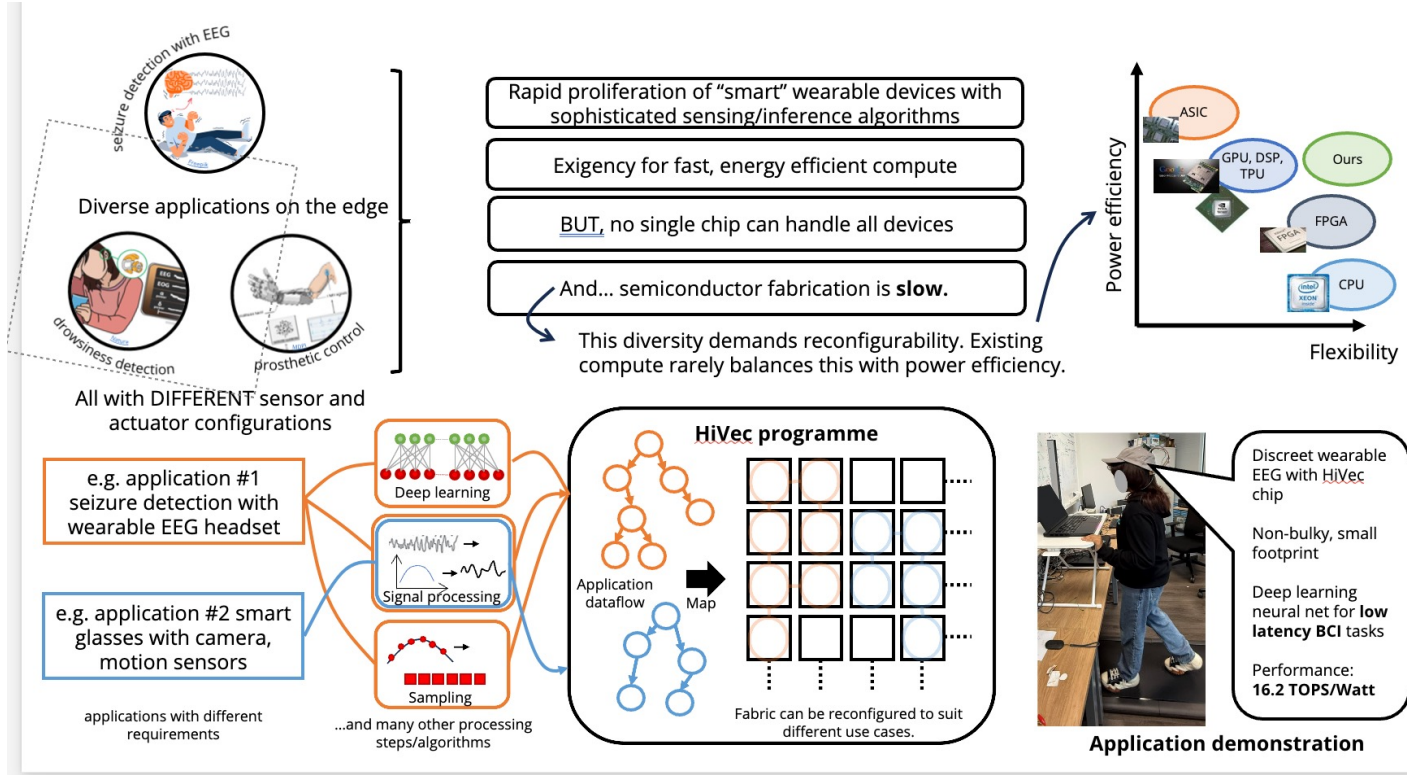
TileLoom $\gtrsim$ Hand-optimized library



Tenstorrent
Dataflow
Accelerator



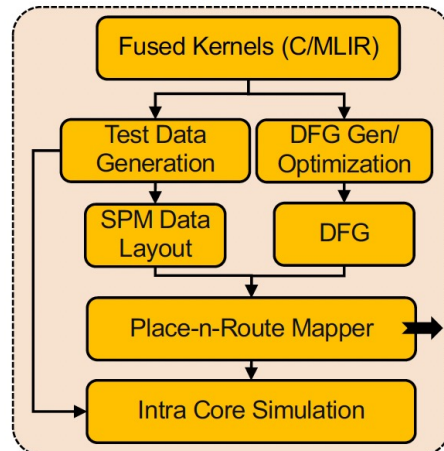
HiVec: AI Accelerator for Multisensory edge analytics



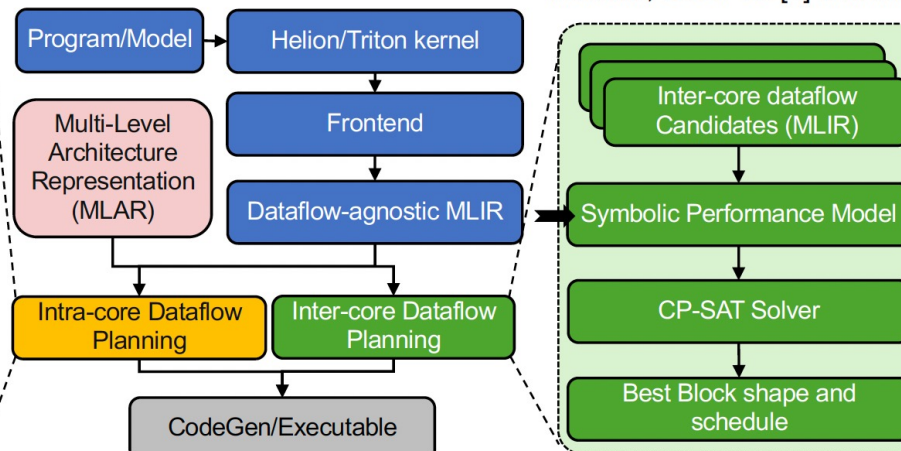
- ✓ 12nm TSMC HiVec CGRA
- ✓ 16.2 TOPS/W @ 0.4V, 267 MHz
- ✓ 7.3X higher normalized energy efficiency

HOT
CHIPS
2026

Morpher, ICCAD' 25 [1]



Tileloom, OSDI' 26 [2] and more



	ISSCC'21	DATE'20	DSD'24	SOCC'25	JSSCC'24	ISSCC'26	HiVec
Tech Node	16	28	45	45	16	3	12
Frequency (Ghz)	1.44	0.05	0.1	0.7	955	0.456	1@0.8V 0.267@0.4V
Peak Performance (GOPS)	388.4	NA	2.845	1.25	367@1.29V	2520	512@0.8V 0.136@0.4V
Peak Efficiency (GOPS/W)	209.5@0.55V	224	519	156	538@0.84V	28500	2416@0.8V 16200@0.4V
Normalised Efficiency (GOPS/W)	372.4	1219.5	1744	2193	956.4	1087	16200
Area (mm ²)	4.4	0.174	0.2	0.013	20.1	0.338	0.391
Normalised Area (mm ²)	2.93	0.174	0.0256	0.0016	2.048	1.86	0.391
Normalised Compute Density (TOPS/mm ²)	0.125	NA	0.1109	0.781	0.179	1.36	0.04 - 1.309

The Hardware Lottery

Sara Hooker

Google Brain

shooker@google.com

Algorithms have been constrained by available hardware.
Hardware has been constrained by programmability.

Hardware, systems and algorithms research communities have historically had different incentive structures and fluctuating motivation to engage with each other explicitly. This historical treatment is odd given that hardware and software have frequently determined which research ideas succeed (and fail). This essay introduces the term hardware lottery to describe when a research idea wins because it is suited to the available software and hardware and *not* because the idea is superior to alternative research directions. Examples from early computer science history illustrate how hardware lotteries can delay research progress by casting successful ideas as failures. These lessons are particularly salient given the advent of domain specialized hardware which makes it increasingly costly to stray off of the beaten path of research ideas. This essay posits that the gains from progress in computing are likely to become even more uneven, with certain research directions moving into the fast-lane while progress on others is further obstructed.



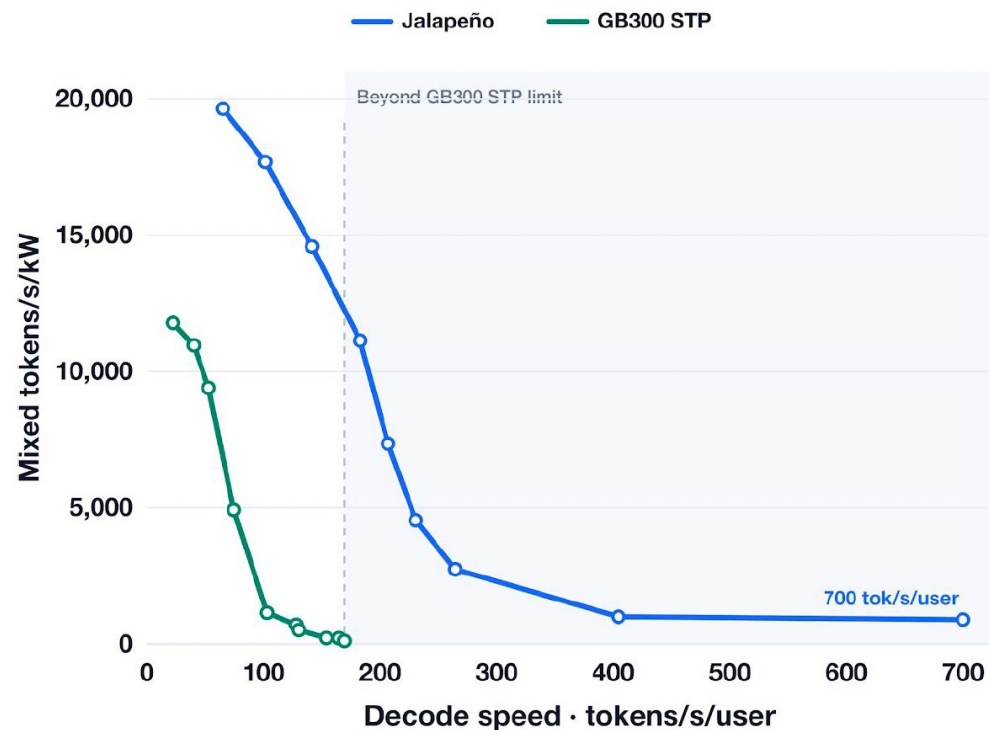
THROUGHPUT-per-kW FRONTIER

Jalapeño is pareto frontier at DeepSeek R1 670B

InferenceX · DeepSeek R1 MXFP4 · nominal 8k/1k · STP · package TDP: Jalapeño 700 W; GB300 1,400 W

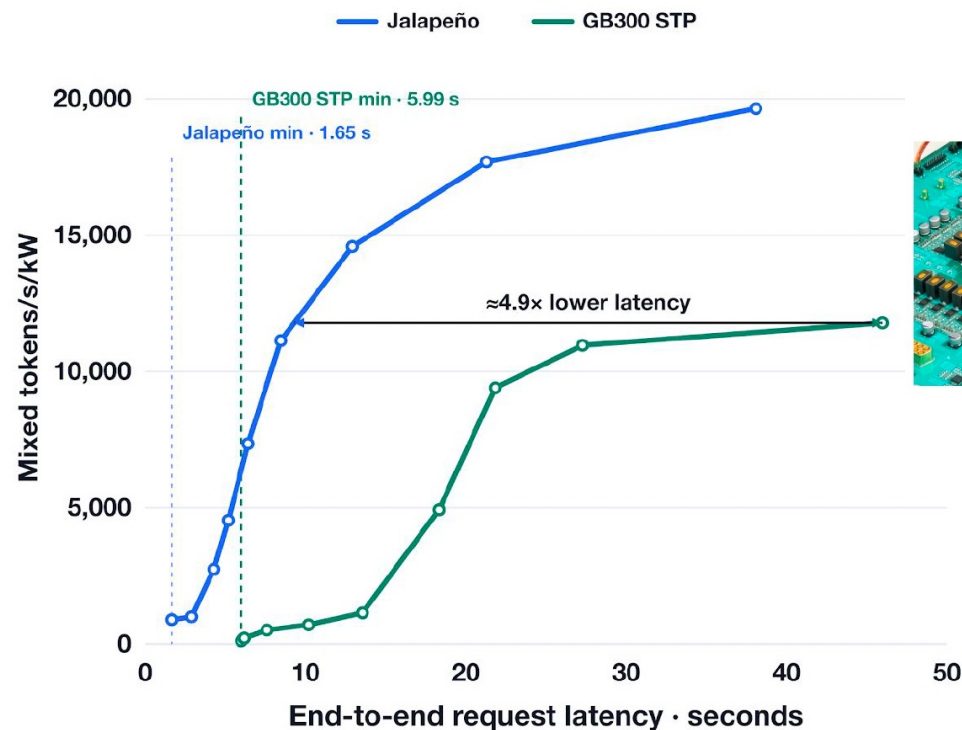
MIXED THROUGHPUT VS. INTERACTIVITY

Efficiency across decoding speeds



MIXED THROUGHPUT VS. END-TO-END LATENCY

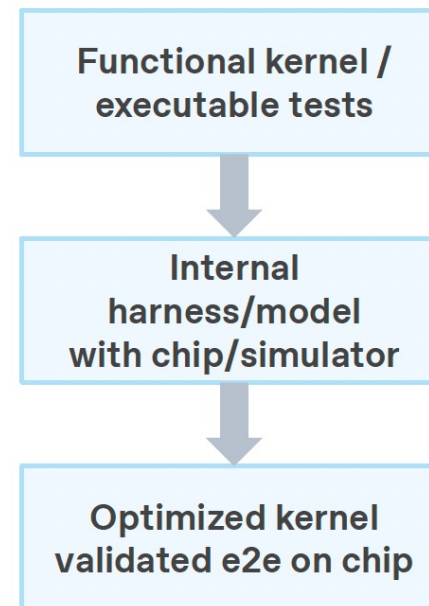
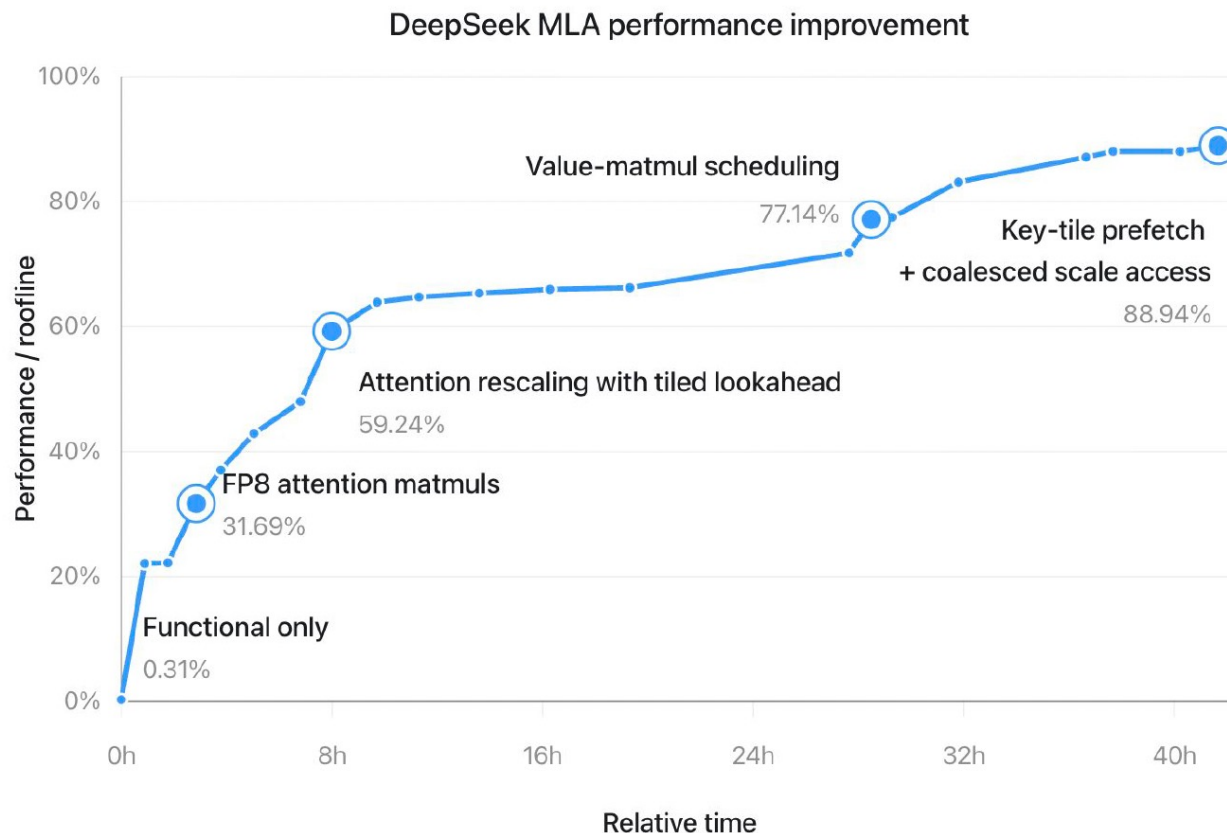
Efficiency across full-request latency



Jalapeño ASIC

What if AI can program the hardware?

AI turns functional kernels into high-performance implementations



DeepSeek kernels

AI drove most of the optimization from a functional baseline.

GPT-OSS blocks

1.5-1.8× faster than the existing expert-written implementation on attention and MoE blocks.

Programmability bottleneck might be shifting

**If AI can program the accelerator,
how ambitious can the architecture be?**

Toward AI-Programmable Adaptive Dataflow

- **More adaptive hardware**

Dataflow changes with workload and runtime data

- **Minimalist runtime control**

Pay only for decisions that depend on runtime data

- **AI-driven mapping and orchestration**

Let AI navigate the large search space

Instead of simplifying architecture for the compiler,
Design architecture that is tractable for AI to program.
Can AI help break the hardware-algorithm lock-in?

Acknowledgments

- **Li-Shiuan Peh**
- **Anh Tuan Do**
- **Pranav Dangi**
- **Rohan Juneja**
- **Zhenyu Bai**
- **Bin Gao**
- **Yujie Zhang**
- **Chen Liang**
- **Tan Cheng**
- **Manupa Karunaratne**
- **Wang Bo**
- **Aditi Kulkarni**
- **Dhananjaya Wijerathne**
- **Zhaoying Li**
- **Thilini Kaushalya Bandara**
- **Dan Wu**
- **Vishnu Paramasivam**
- **Chong Yi Sheng**

**NATIONAL
RESEARCH
FOUNDATION**

PRIME MINISTER'S OFFICE
SINGAPORE



Ministry of Education
SINGAPORE



Institute of
Microelectronics

