ELECTRONICS RESURGENCE INITIATIVE
ERI SUMMIT
& MTO Symposium
2020 | Seattle, WA | August 18-20

EigenArch: A Low Rank Hardware Machine Learning Accelerator

Gina Adam
George Washington University

Brian Hoskins
NIST Gaithersburg

Martin Lueker-Boden
Western Digital Research

FRANC

Artificial Intelligence

Background

GOAL: Develop **faster** and more **efficient** machine learning accelerators using arrays of **emerging analog devices** called resistive switches (ReRAM) and **digital coprocessors**

CONTEXT: ReRAM technology promise **100-10,000x** lower energy consumption, **30,000x** faster operation than SOA.

PROBLEM: High **write energy**, high **write latency**, limited **endurance**, non-ideal weight updates, limited digital **coprocessor resources**

CURRENT SOLUTIONS: 1) Find better devices → will always be imperfect 2) **Batch training** [1] + duplicate memory + 1-by-1 device programming (**Fig. 1a**, [2]) → **requires expensive coprocessor** or **emerging technology**

OUR VISION: **Low-rank coprocessor** for **batch training** in conventional hardware (systolic arrays [3]) to extract and use only most important information during training (**Fig. 1b**)

+ **high performance, memory efficient**
+ **array level update → speed increase**

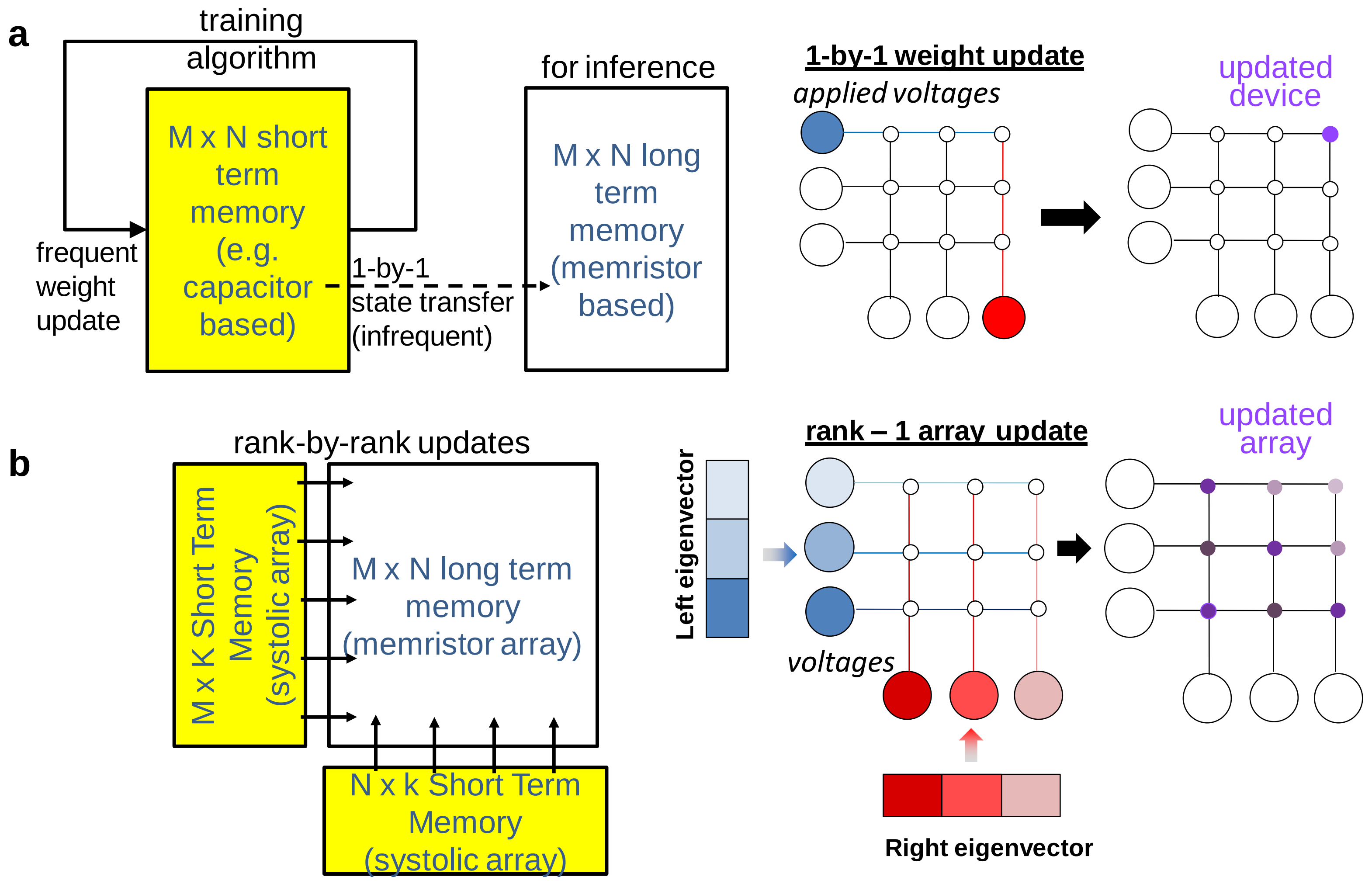

a
training algorithm
M x N short term memory (e.g. capacitor based)
frequent weight update
1-by-1 state transfer (infrequent)
for inference
M x N long term memory (memristor based)
1-by-1 weight update
applied voltages
updated device
b
rank-by-rank updates
M x K Short Term Memory (systolic array)
M x N long term memory (memristor array)
N x k Short Term Memory (systolic array)
Left eigenvector
rank – 1 array update
voltages
updated array
Right eigenvector

Figure 1. Existing vs. proposed approach to training memristor M x N arrays (a) Duplicative short-term memory (M x N capacitor array) overcomes memristor non-ideality and the high programming power of 1-by-1 weight updating [2]; (b) proposed approach with low rank short term memory implemented using systolic arrays that produces rank-by-rank updates at the array level. *"Author's Own"*

Relevance to FRANC: Efficient **digital coprocessors** can improve the **technology readiness level** of emerging technologies: • Exploit new technologies for **inference** to use in **training** • Mitigate device limitations, reduce AI operation energy • bring **edge performance** to the warfighter.

Approach

Our EigenArch is based on three interconnected tasks:

1) **Algorithms to approximate gradient data in machine learning efficiently.** We proposed streaming batch principal component analysis (SBPCA) [4] (**Fig. 2**).
2) **Compact hardware implementations.** We proposed and synthesized quasi-systolic array (QSArray) (**Figs. 3 and 4**).
3) **Experimental validation** in 20,000 ReRAM device platform

ALGORITHMIC IMPLEMENTATION:

SBPCA = approximation of a full gradient matrix from k-rank samples of its contributed parts

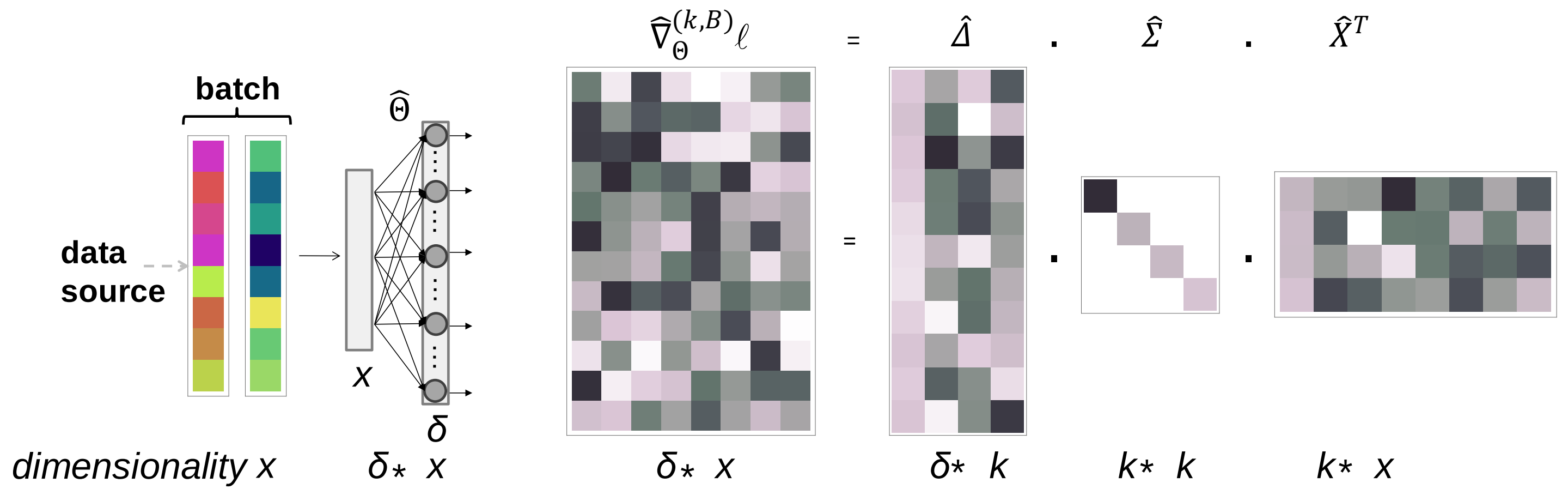

$\widehat{\nabla}_{\Theta}^{(k,B)}\ell = \hat{\Delta} \cdot \hat{\Sigma} \cdot \hat{X}^T$
batch
data source
$\hat{\Theta}$
x
δ
dimensionality x
δ∗ x
δ∗ x
δ∗ k
k∗ k
k∗ x

Figure 2. Streaming Batch Principal Component Analysis (SBPCA). *"Author's Own"*

HARDWARE ARCHITECTURE:

=

ReRAM memory array

+

Quasi-systolic arrays (qsarray) to calculate gradient decomposition

+

Access circuitry (ADCs, DACs)

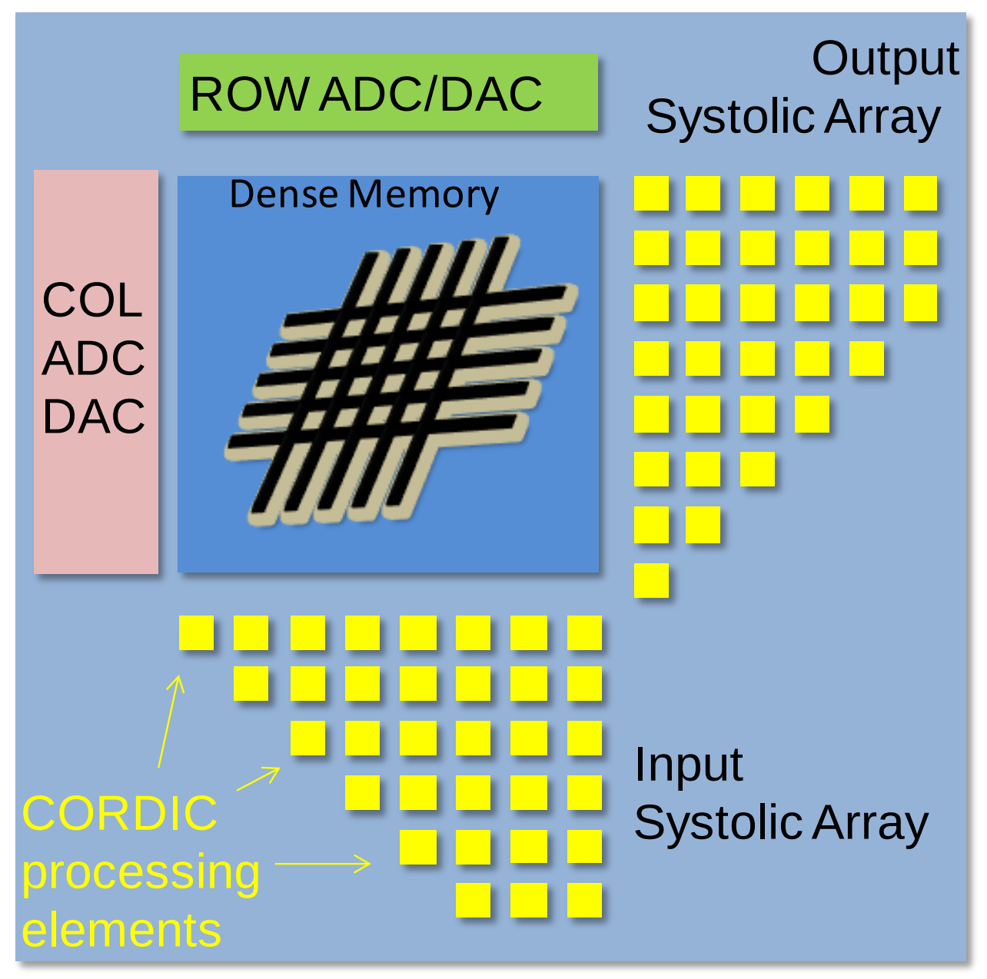
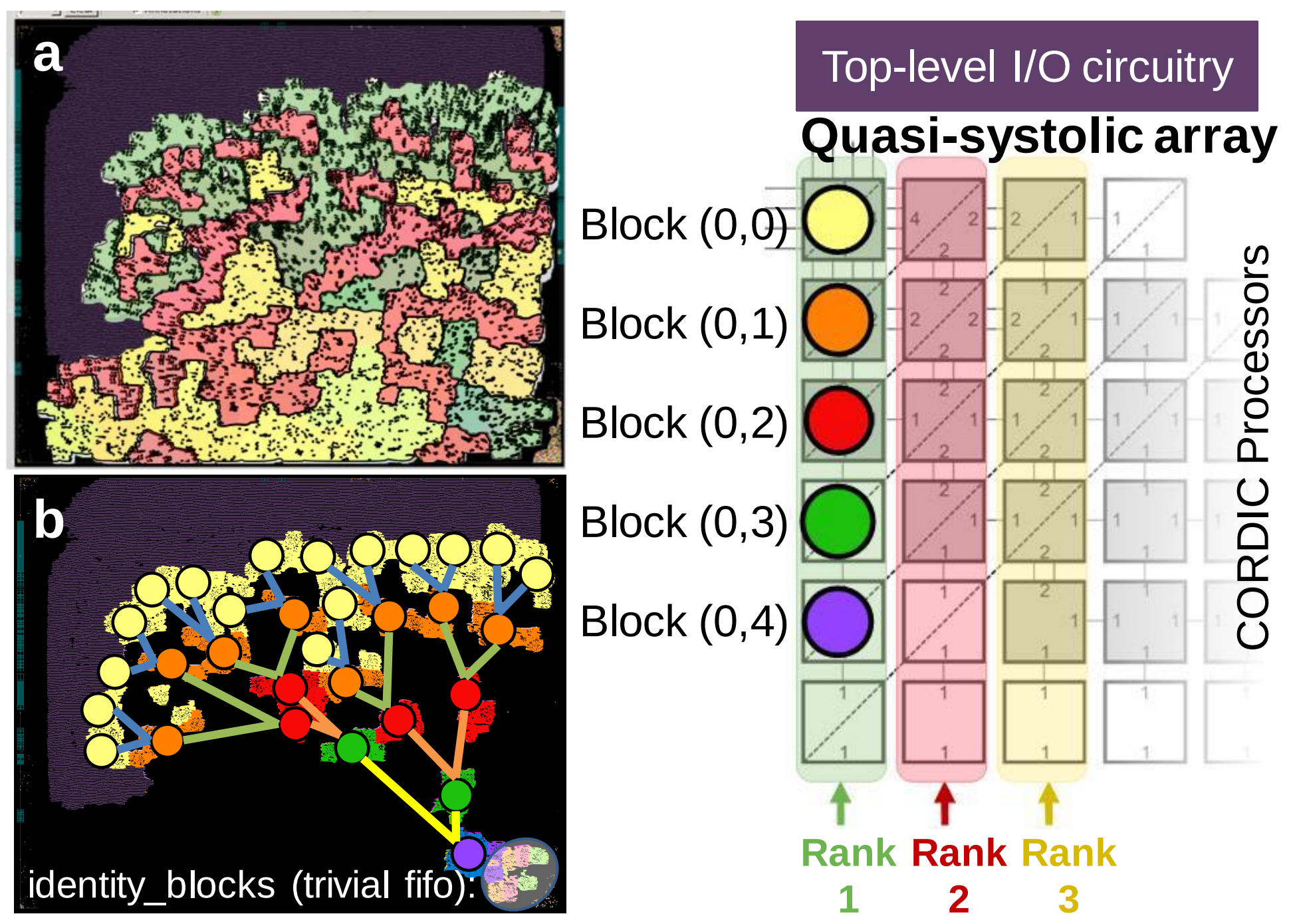


Figure 3. EigenArch implementation. *"Author's Own"*



SYNTHESIS RESULTS:

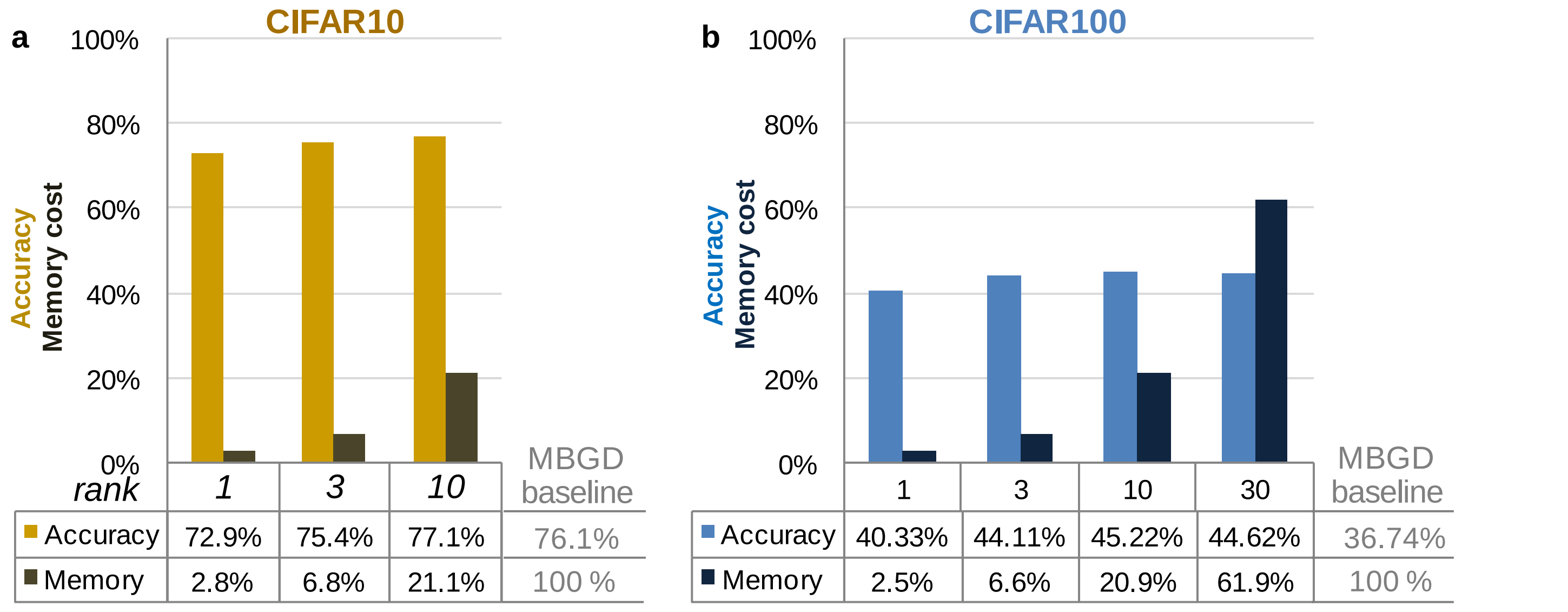
- Each column in the qsarray is a binary tree of CORDIC processing elements (PE)
- Phase angles in each PE store the principal components for the SBPCA algorithm.

Figure 4. Synthesis results (a) rank-level maps for a rank-3 quasi-systolic array with 128-element Input Vectors in a TSMC 16nm process, meets timing of 500 MHz. (b) hypothetical visualization of the binary tree for rank 1. *"Author's Own"*

Results and Impact

The results obtained in the first six months are:

1) **Proposed SBPCA algorithm approximates well gradient data for training on datasets of various complexities e.g. CIFAR-10, CIFAR-100** (**Fig. 5**)
   - **Comparable accuracy** at low rank to traditional mini-batch gradient descent (MBGD) at **lower memory**



rank	1	3	10	MBGD baseline
Accuracy	72.9%	75.4%	77.1%	76.1%
Memory	2.8%	6.8%	21.1%	100 %

	1	3	10	30	MBGD baseline
Accuracy	40.33%	44.11%	45.22%	44.62%	36.74%
Memory	2.5%	6.6%	20.9%	61.9%	100 %

Figure 5. SBPCA accuracy and memory results. For low rank decomposed form has similar accuracy with MBGD for (a) CIFAR10 and (b) CIFAR100. *"Author's Own"*

2) **Systolic hardware implementation has good performance with higher technology readiness** (**Fig. 6**):
   - Area/energy/latency estimates obtained by synthesis of Verilog code in Synopsys Design Compiler

Existing technologies

Google:
TPU
TSMC 28nm Digital, 8 Bit

Property	Metric
Energy VMM Forward (Back)	456 (912) nJ
Array Latency	3000 ns
Pipeline Latency	3 ns
Area (MAC Only)*	317.76 mm^2
Technology readiness	Operational

*original TPU is 256×256 MAC

IBM:
3T1C+PCM*
90 nm Analog, 8 bit

Property	Metric
VMM Forward (Back)	12.51 (25.22) nJ
VMM Latency (Back)*	26.7 (53.4) ns
Area	5.8 mm^2
Transfer Energy	22894 nJ
TE/Example (8000)	2.87 nJ
Technology readiness	Lower

*Energy and area: ~60% capacitors + ~40% PCM
*For 3 layer network, total latency ~240 ns

EigenArch

NIST\WD\GWU:
QSARRAY: 512, 3 ranks
TSMC 16nm Digital, 8 Bit*

Property	Metric
Energy/Example Left (Right)	8.6 (17.2) nJ ✓
Array Latency	240 ns
Pipeline Latency	12 ns
Area Left (Right)	1.05 (2.10) mm^2
Technology readiness	Higher

*projected from model

Figure 6. Performance estimates and comparisons *"Author's Own"*

Impact for DoD and commercial efforts:

- Increases technology readiness level of ReRAM for AI
- Proposed systolic implementation in existing technology → quick prototyping for DoD and commercialization
- Applicable beyond ReRAM to other analog technologies

References

[1] Y. Gao, S. Wu, G.C. Adam, accepted ICONS 2020.
[2] S. Ambrogio et al., Nature, 558, 7708, 2018.
[3] F. Vanpoucke, M. Moonen, E. Deprettere, TCAS II, 44(3), 253-256, 1997.
[4] S. Huang, B. D.Hoskins, M. W. Daniels, M. Stiles, G.C. Adam, AAAI 2020.

This research was developed with funding from the Defense Advanced Research Projects Agency (DARPA).
The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.
Distribution Statement A – Approved for Public Release, Distribution Unlimited