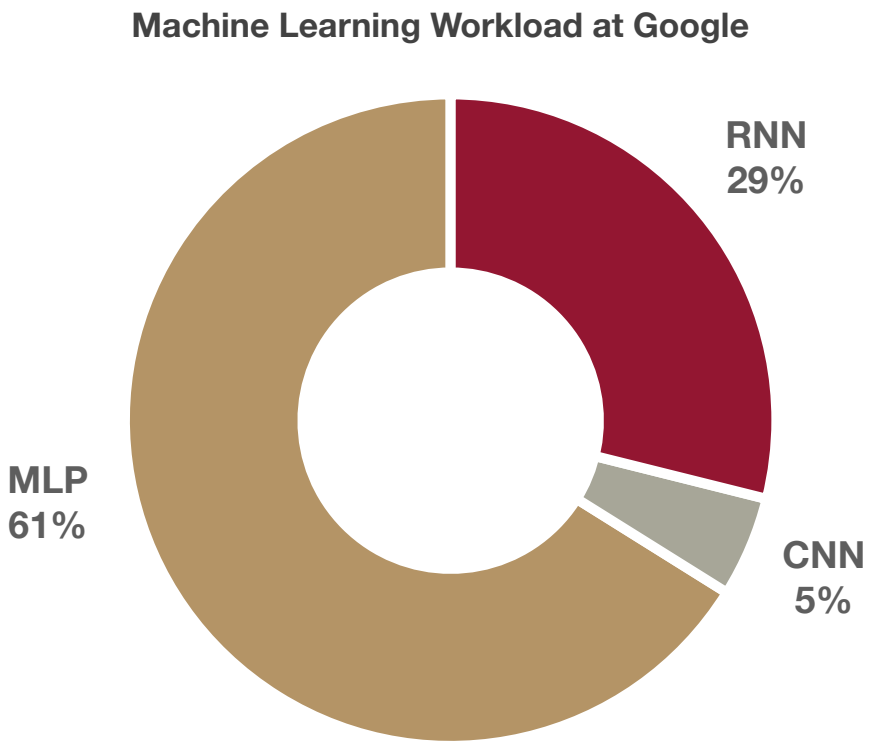
Serving Recurrent Neural Networks Efficiently with a Spatial Accelerator

Tian Zhao, Yaqi Zhang, Kunle Olukotun

Stanford University

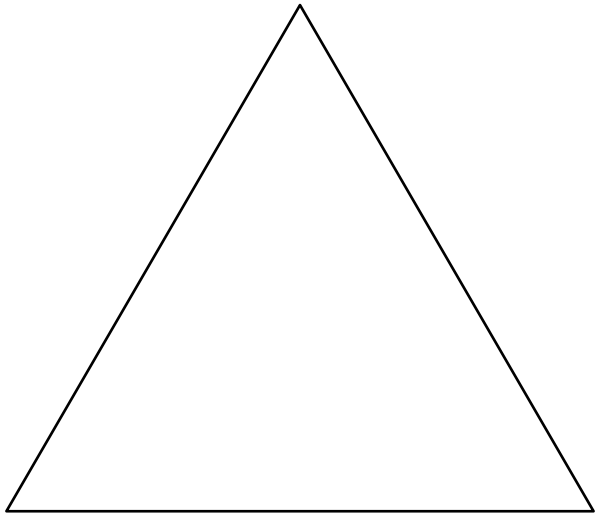
RNNs are Popular Data Center Workloads



Source: In-Datacenter Performance Analysis of a Tensor Processing Unit (2017)

How to design an efficient hardware accelerator for all the RNN kernels?

Performance



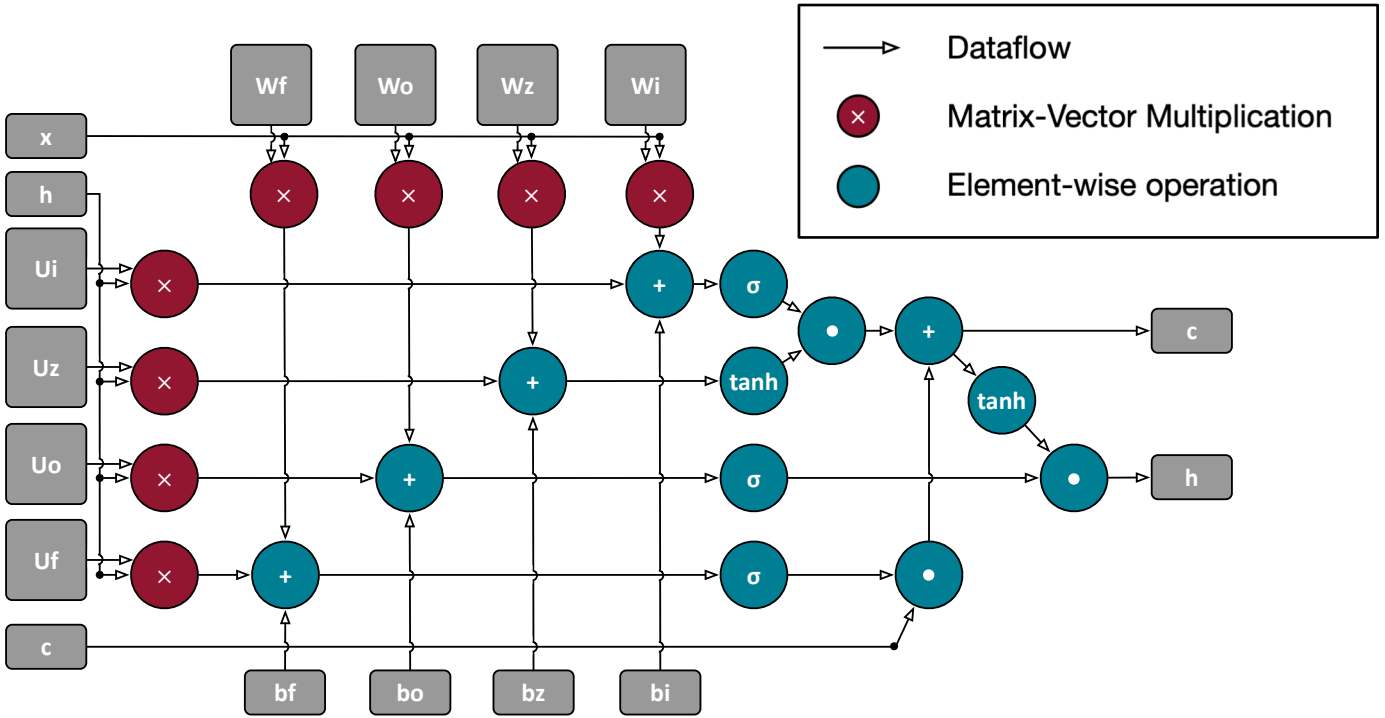
Performance / Area

Performance / Watt

RNN is Hard to Serve Efficiently

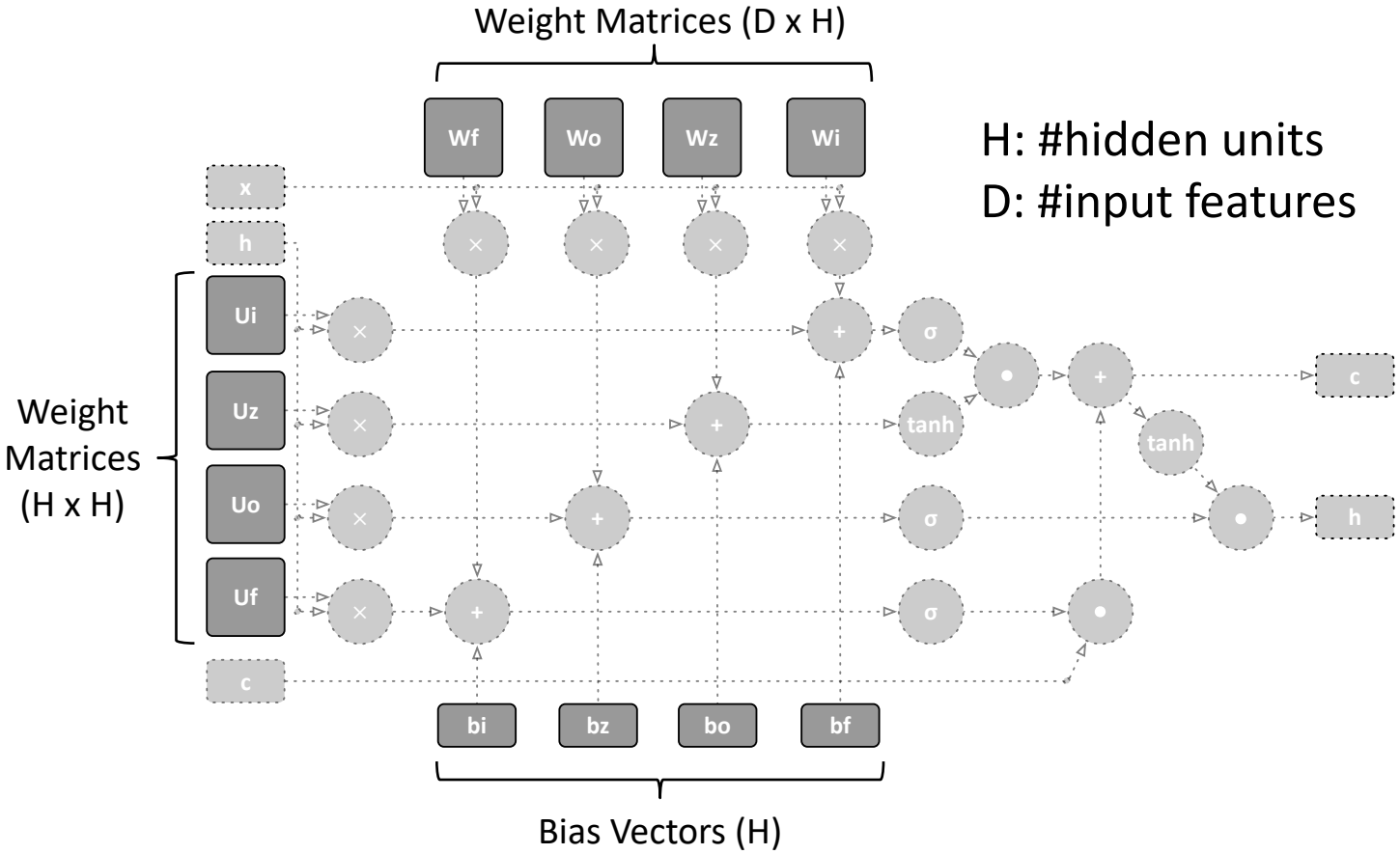
- RNN kernels contain complex dataflow.
- RNN sizes can vary a lot over different problems.

RNN Kernels Contain Complex Dataflow

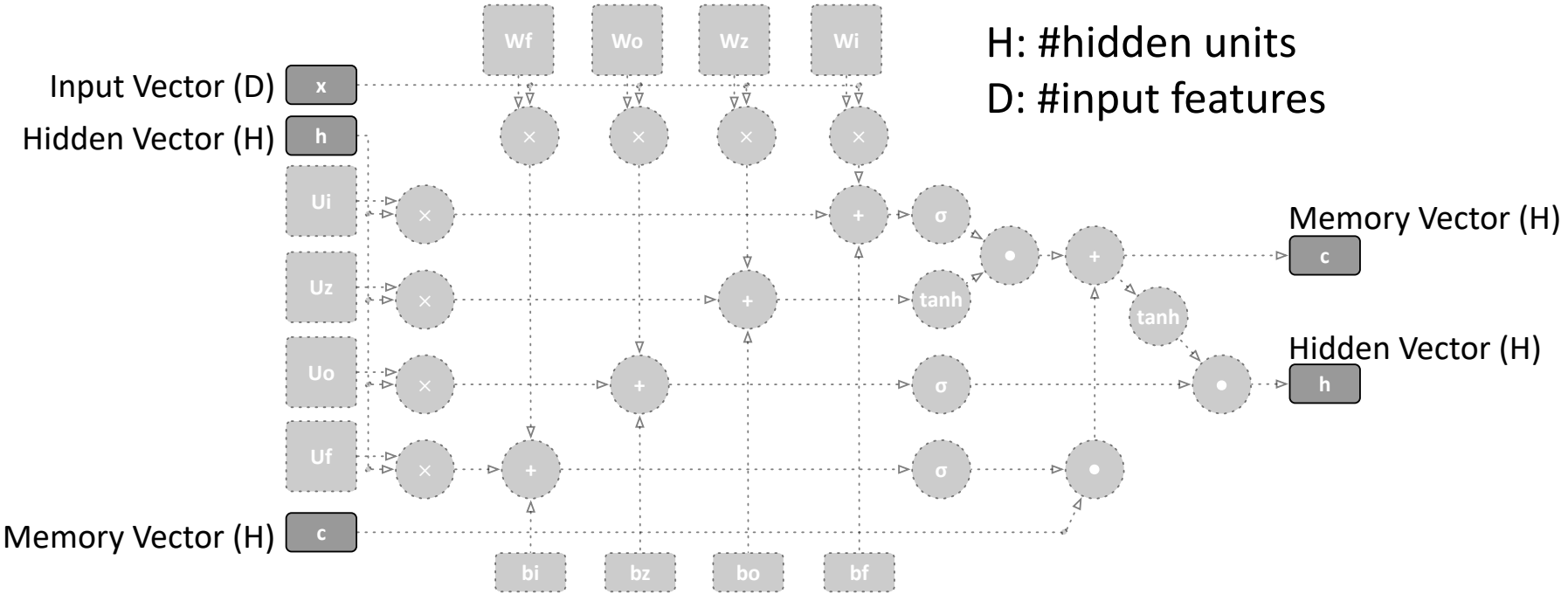


LSTM Example

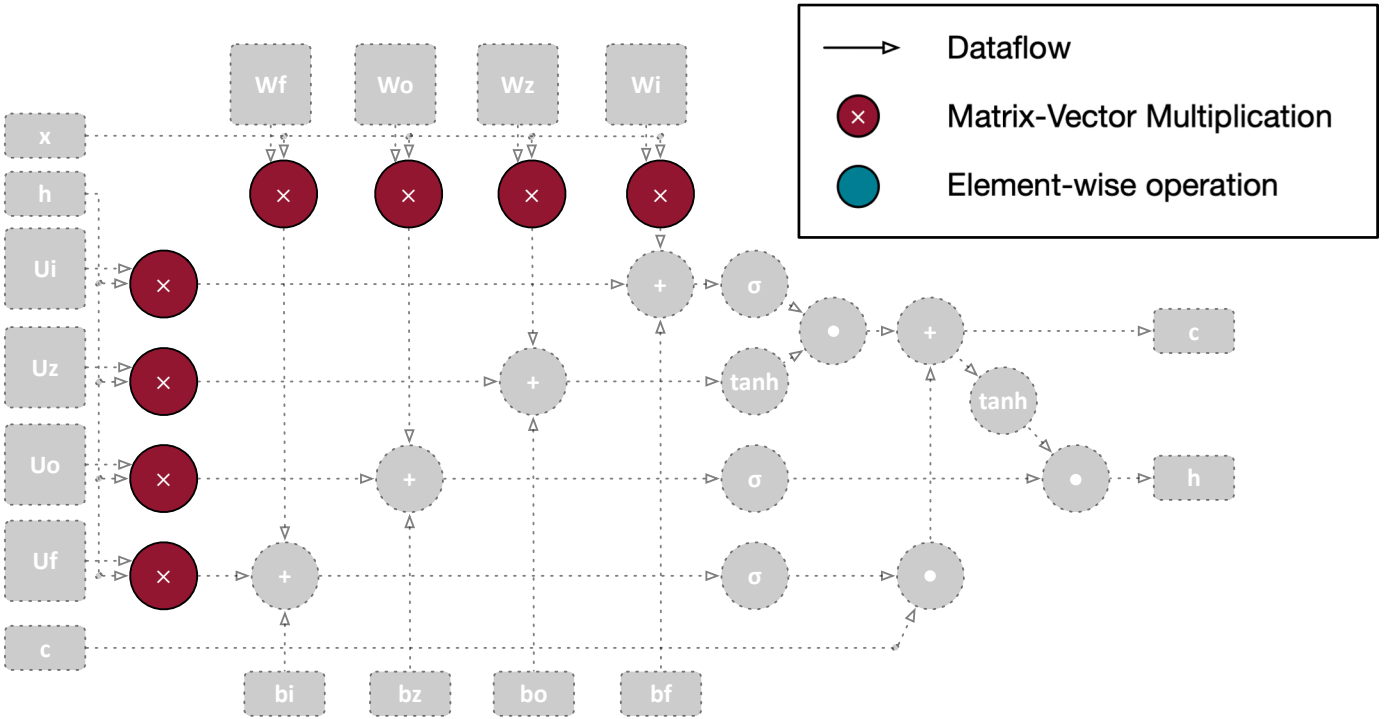
RNN Kernels Contain Complex Dataflow



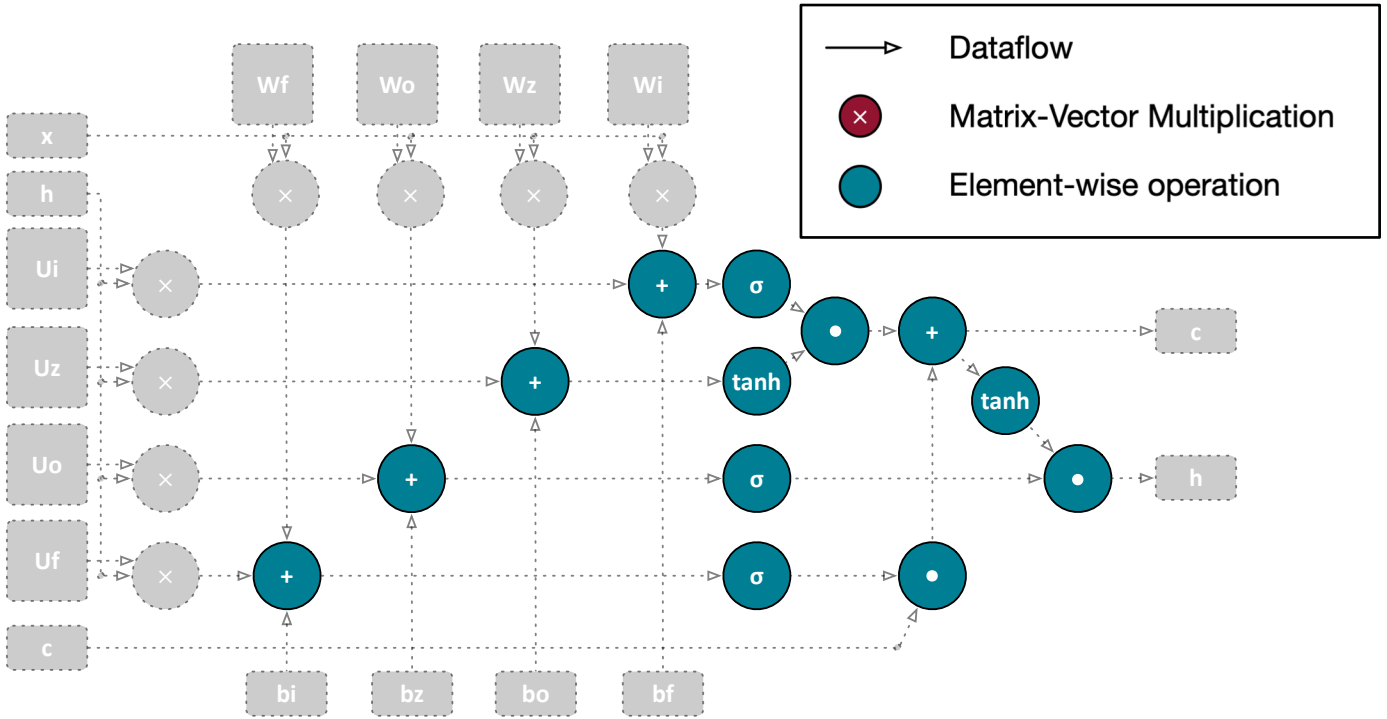
RNN Kernels Contain Complex Dataflow



RNN Kernels Contain Complex Dataflow



RNN Kernels Contain Complex Dataflow



RNN Sizes Can Vary over Different Problems

Tasks	RNN Type	RNN Size
Sequence Classification	Long Short-term Memory (LSTM)	128
Speech Recognition	Gated Recurrent Unit (GRU)	2816

RNN is Hard to Serve Efficiently

- RNN kernels contain complex dataflow.
- RNN sizes can vary a lot over different problems.

Accelerators with BLAS Abstraction

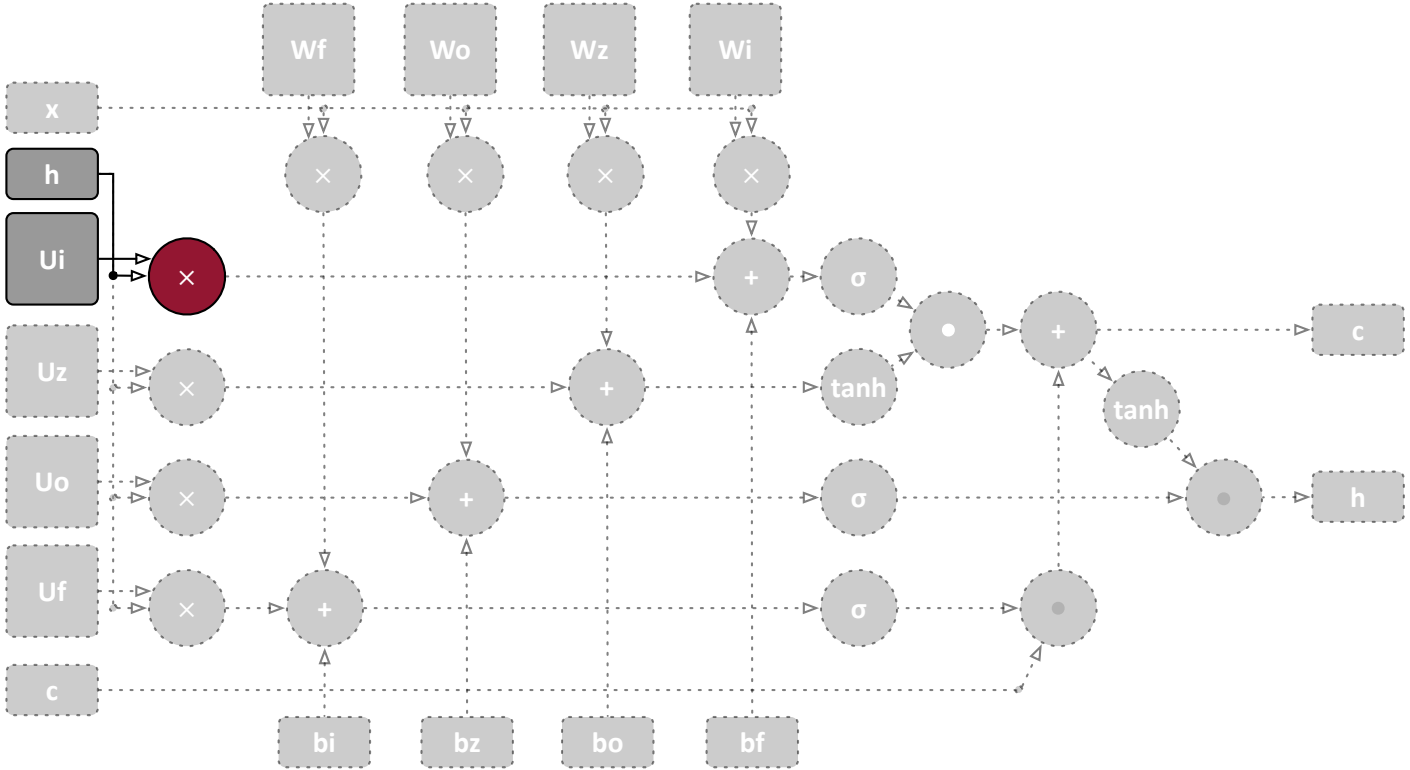
BLAS Level	Example Operation	Accelerator Example
2	Matrix Vector Multiplication (MVM)	Brainwave Neural Processing Unit (BW NPU)
3	Matrix Matrix Multiplication (MMM)	Tensor Processing Unit (TPU)

Is BLAS the right ISA for accelerators?

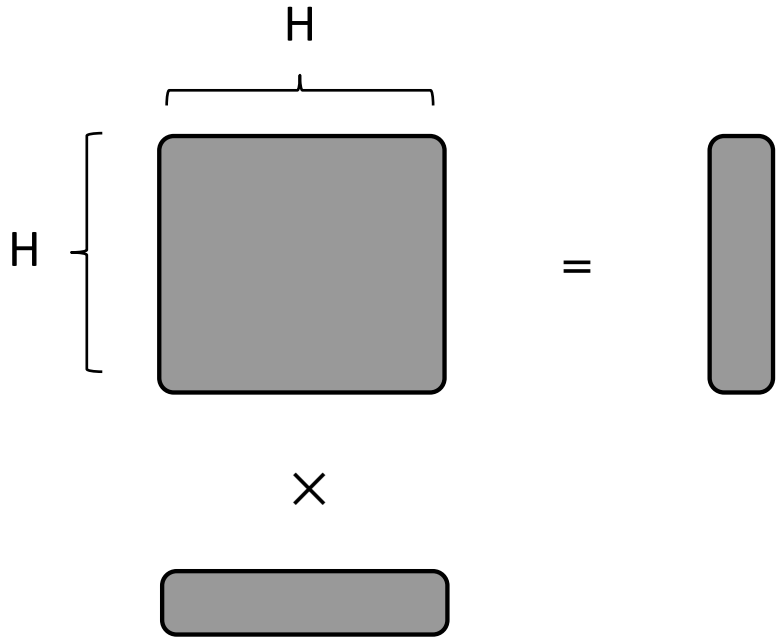
Is BLAS the right ISA for accelerators?

- Programmability (+)
- Efficiency on
  - individual kernel (+)
  - end-to-end task (-)

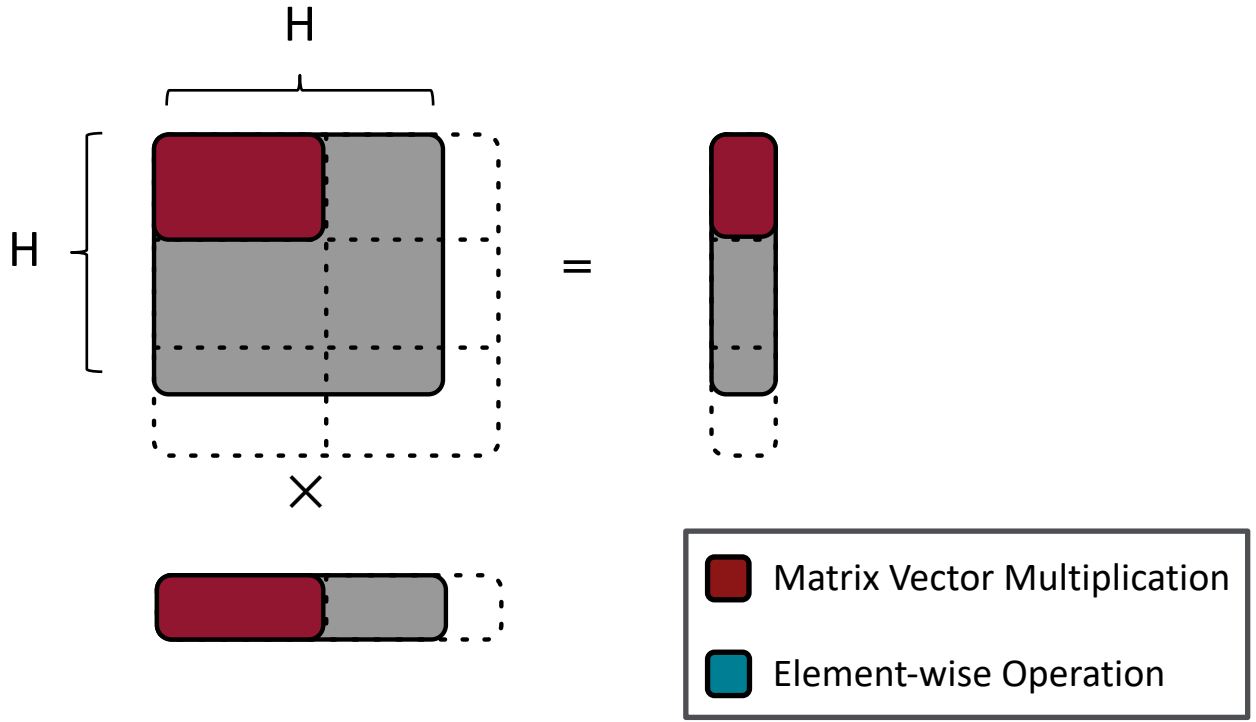
Using BLAS ISA Leads to HW Underutilization



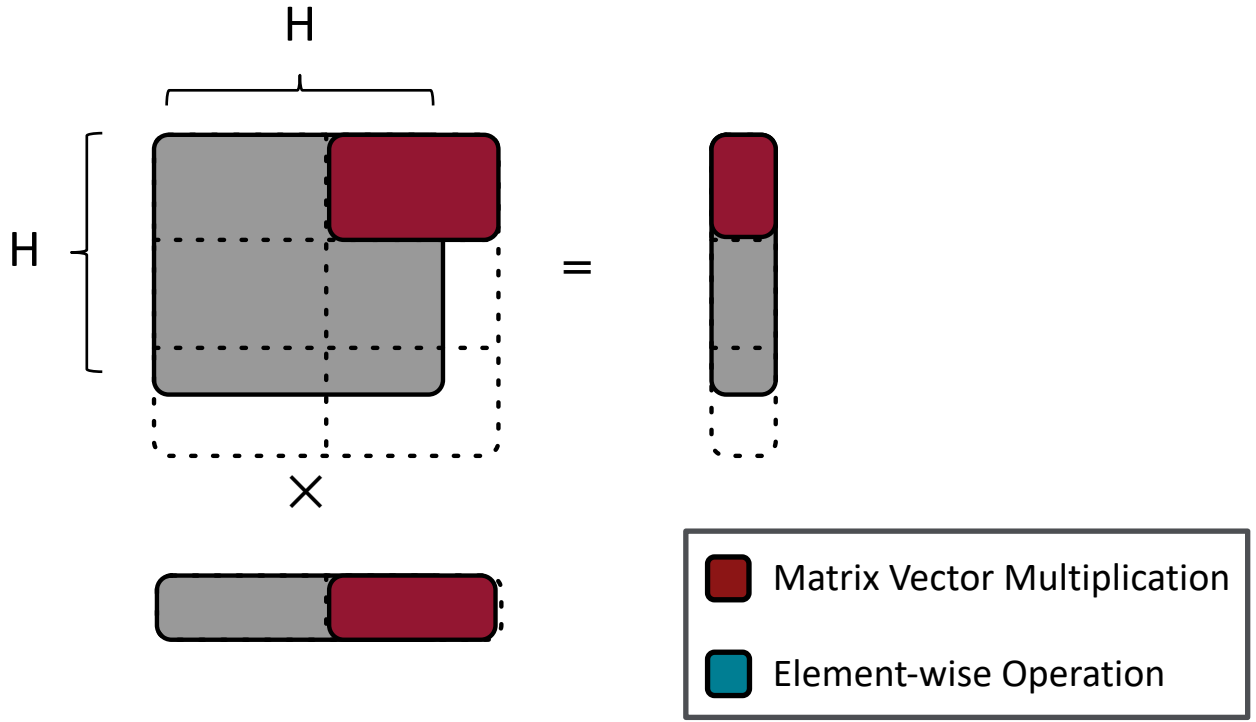
Using BLAS ISA Leads to HW Underutilization



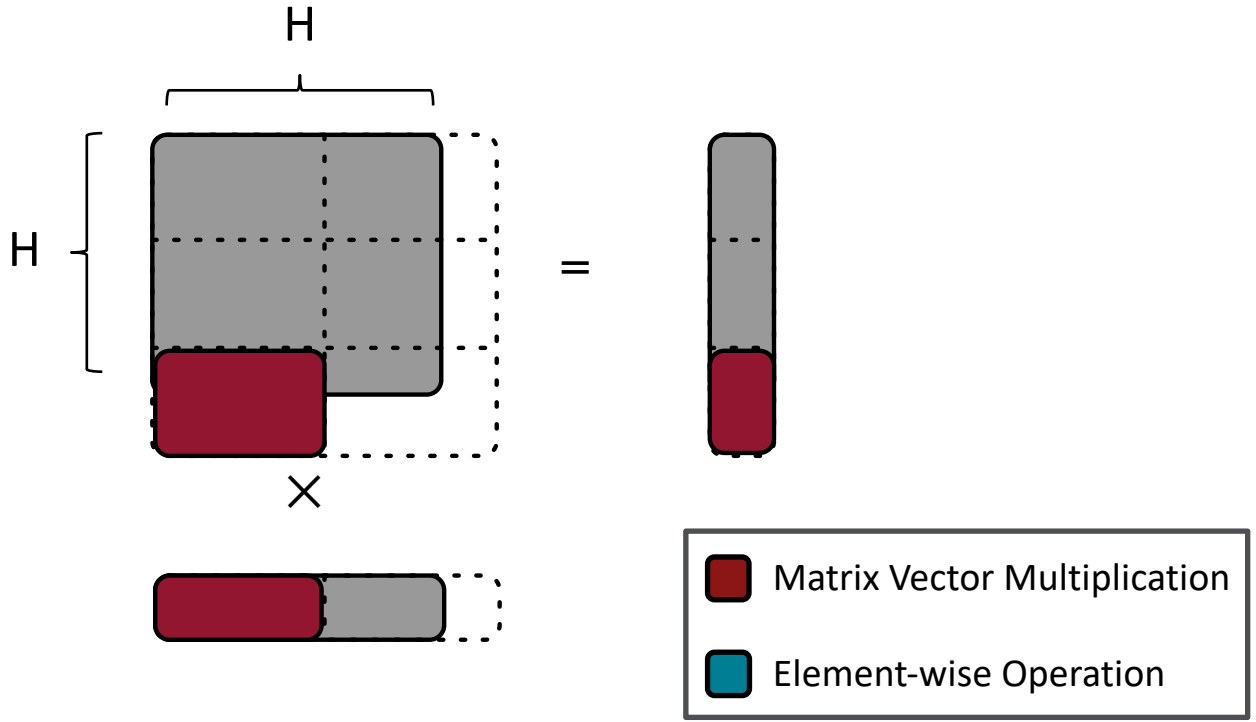
Using BLAS ISA Leads to HW Underutilization



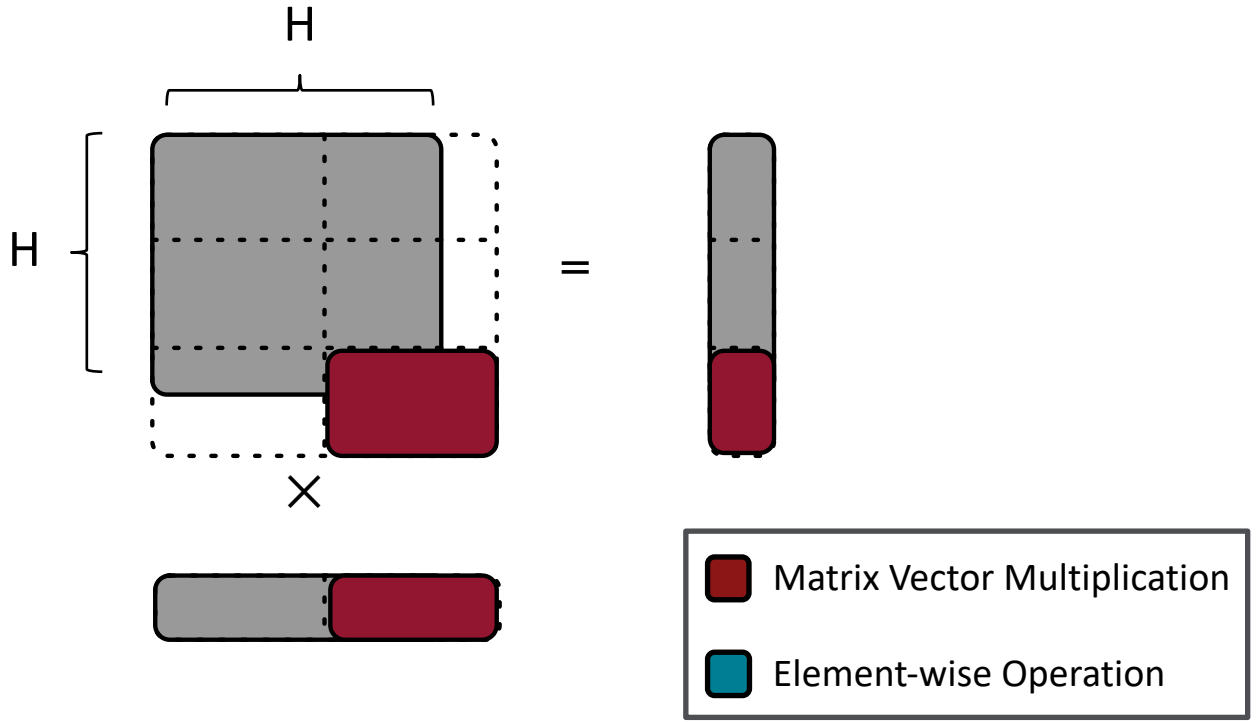
Using BLAS ISA Leads to HW Underutilization



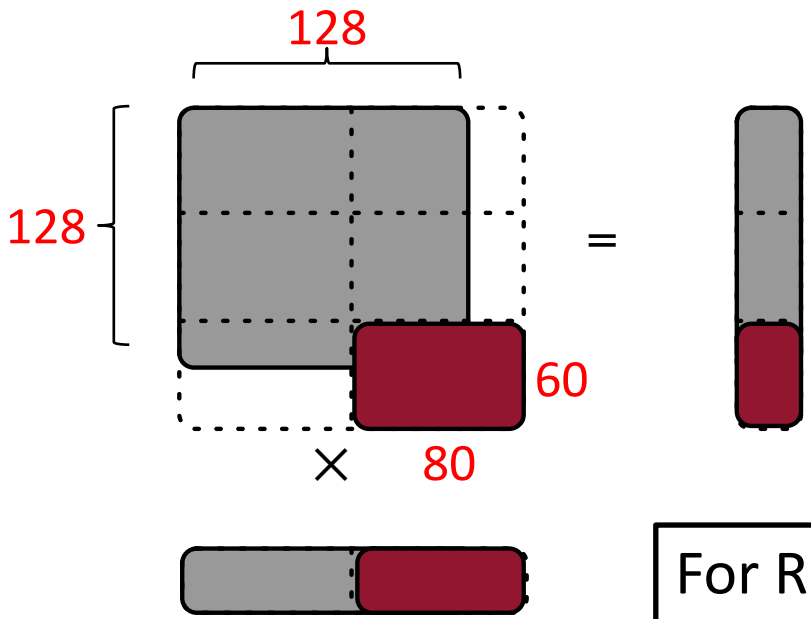
Using BLAS ISA Leads to HW Underutilization



Using BLAS ISA Leads to HW Underutilization

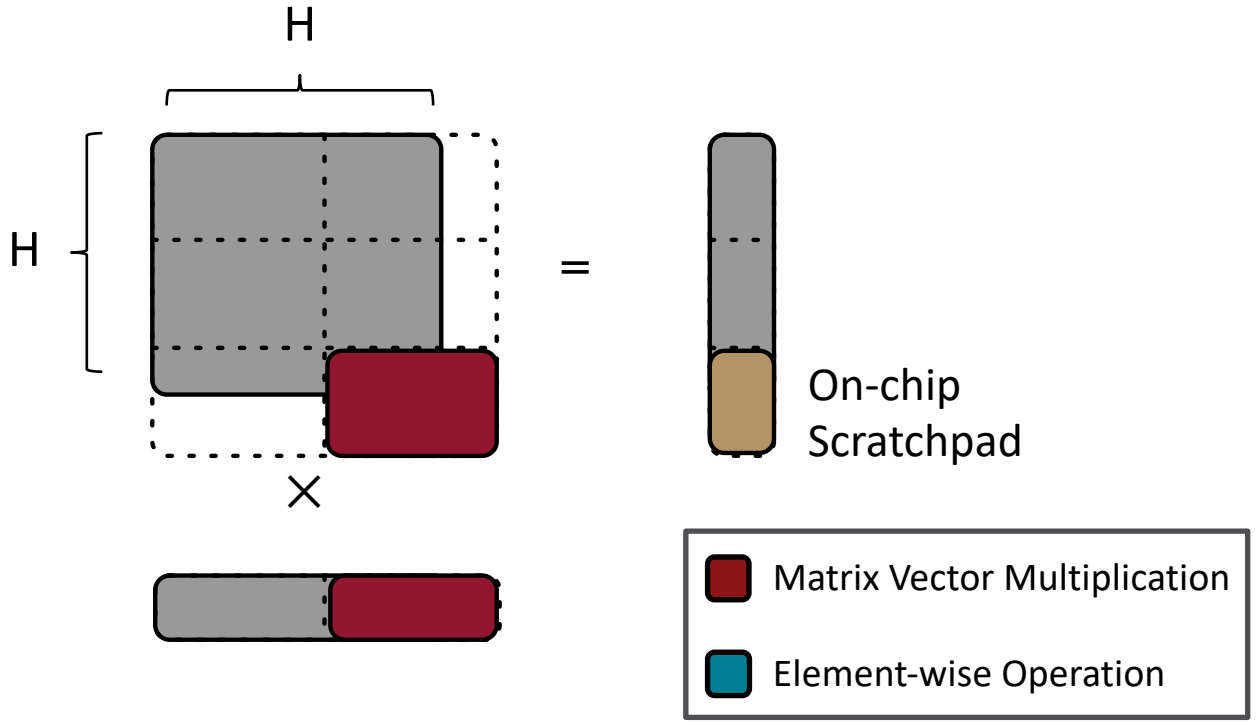


Using BLAS ISA Leads to HW Underutilization

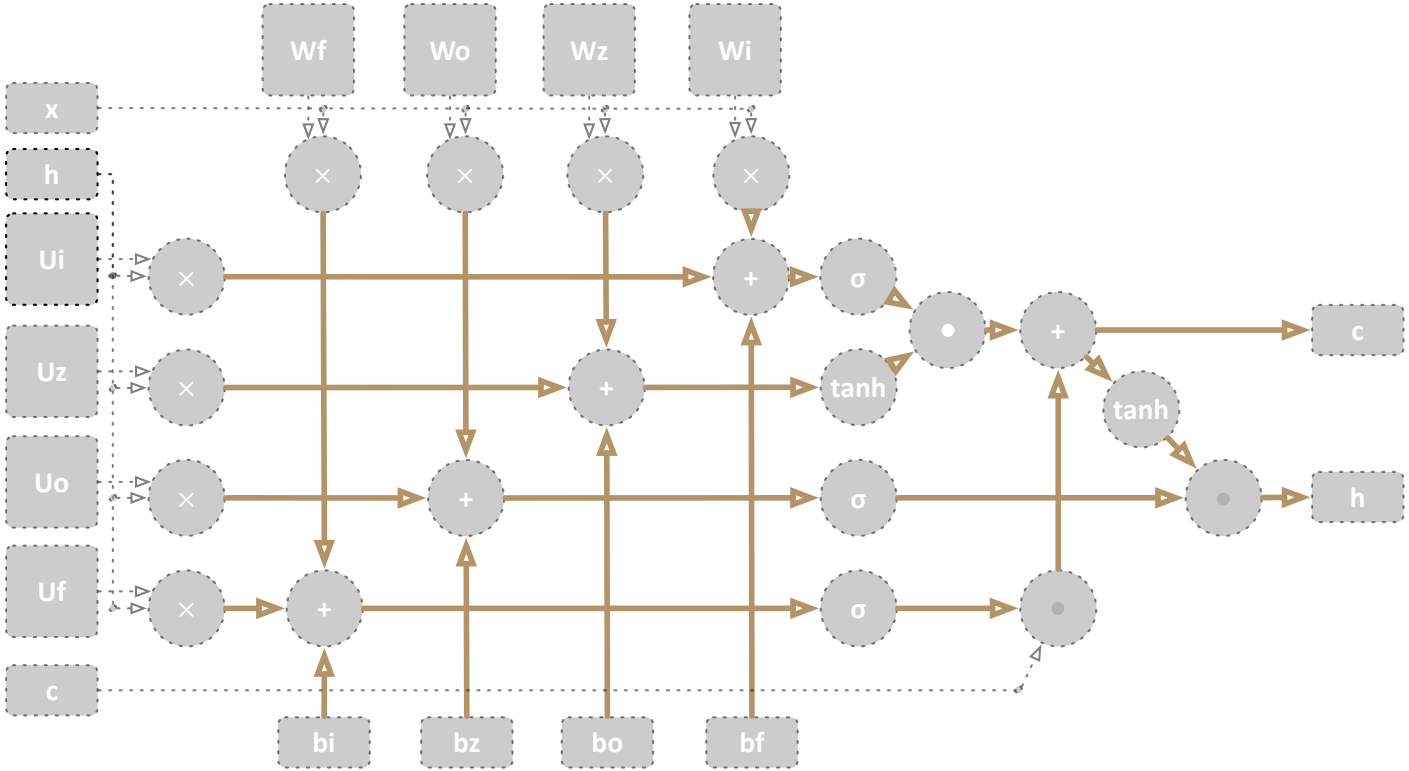


For RNN-128, utilization is 56%!

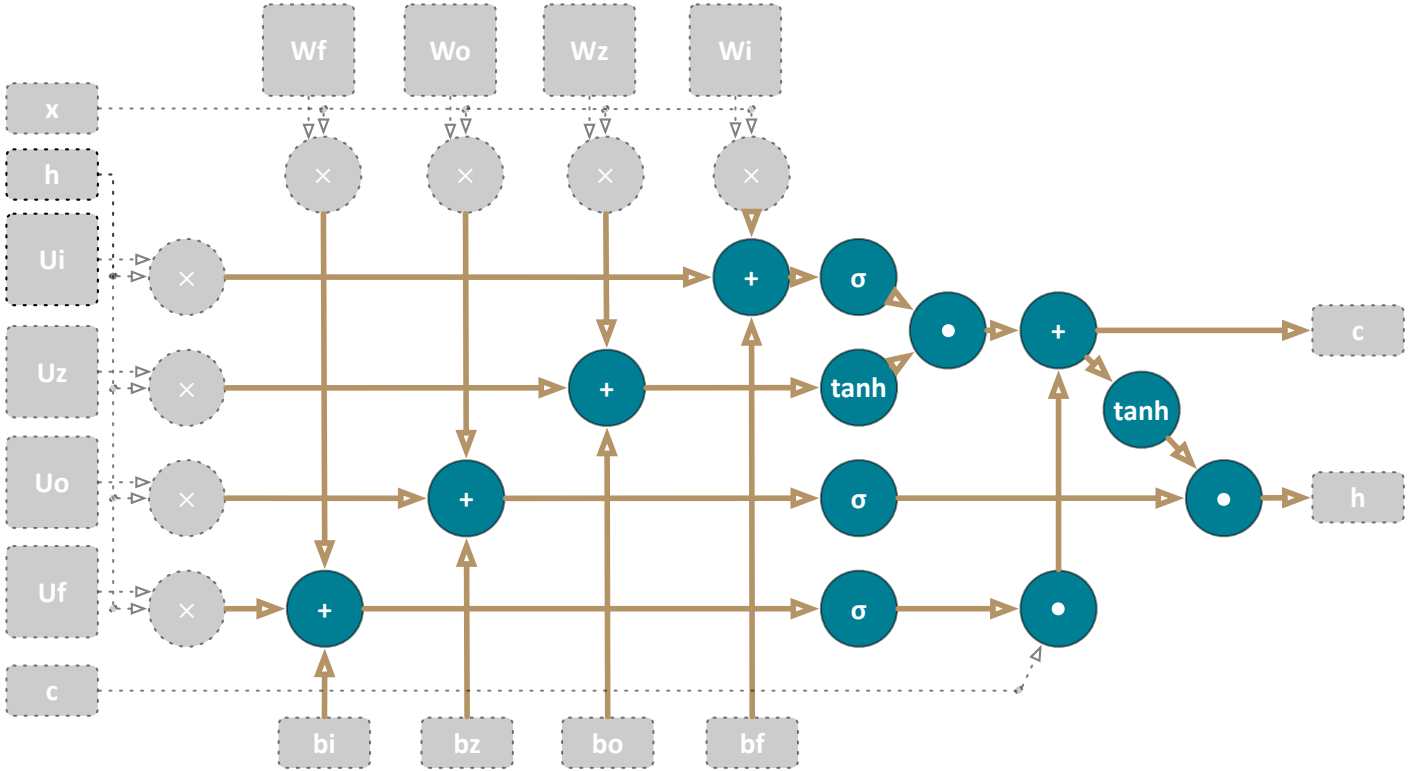
Intermediate Results Buffered in On-chip Scratchpad



Intermediate Results Buffered in On-chip Scratchpad



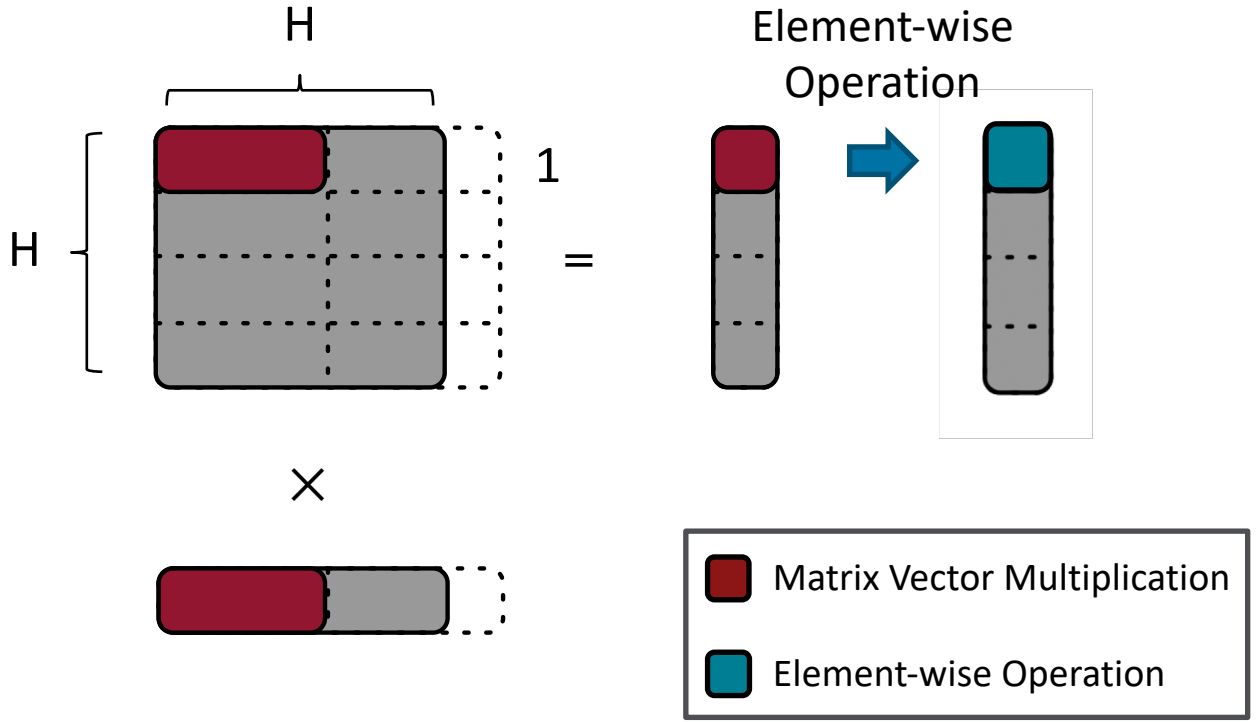
Frequent Access to the On-chip Scratchpad



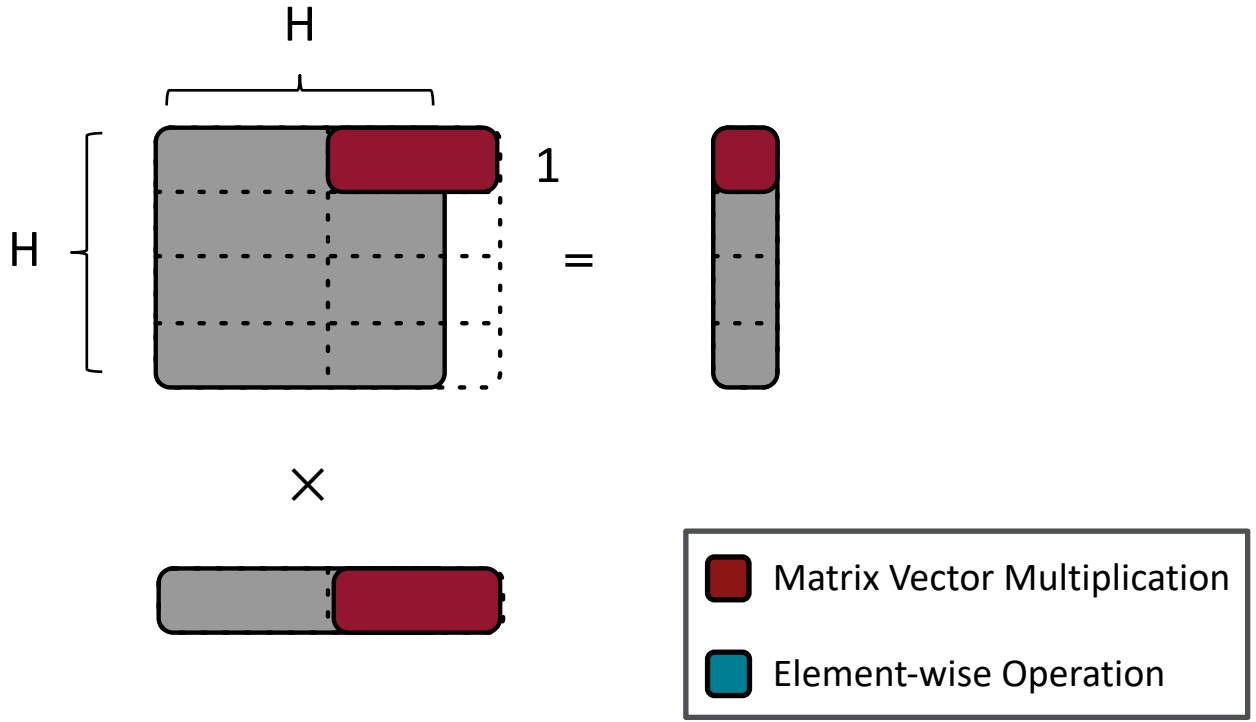
BLAS abstraction leads to hardware underutilization caused by misalignment.

Alternative: Loop-level abstraction

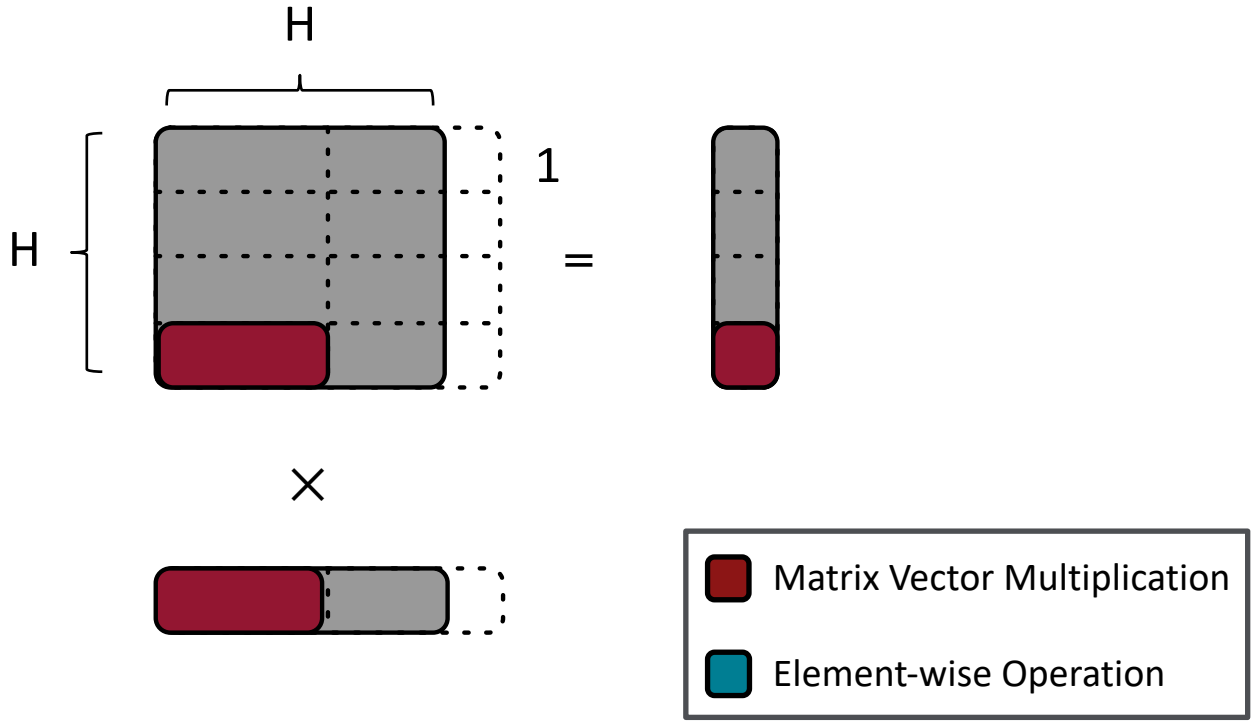
Fine Grain Tiling Leads to Better HW Utilization



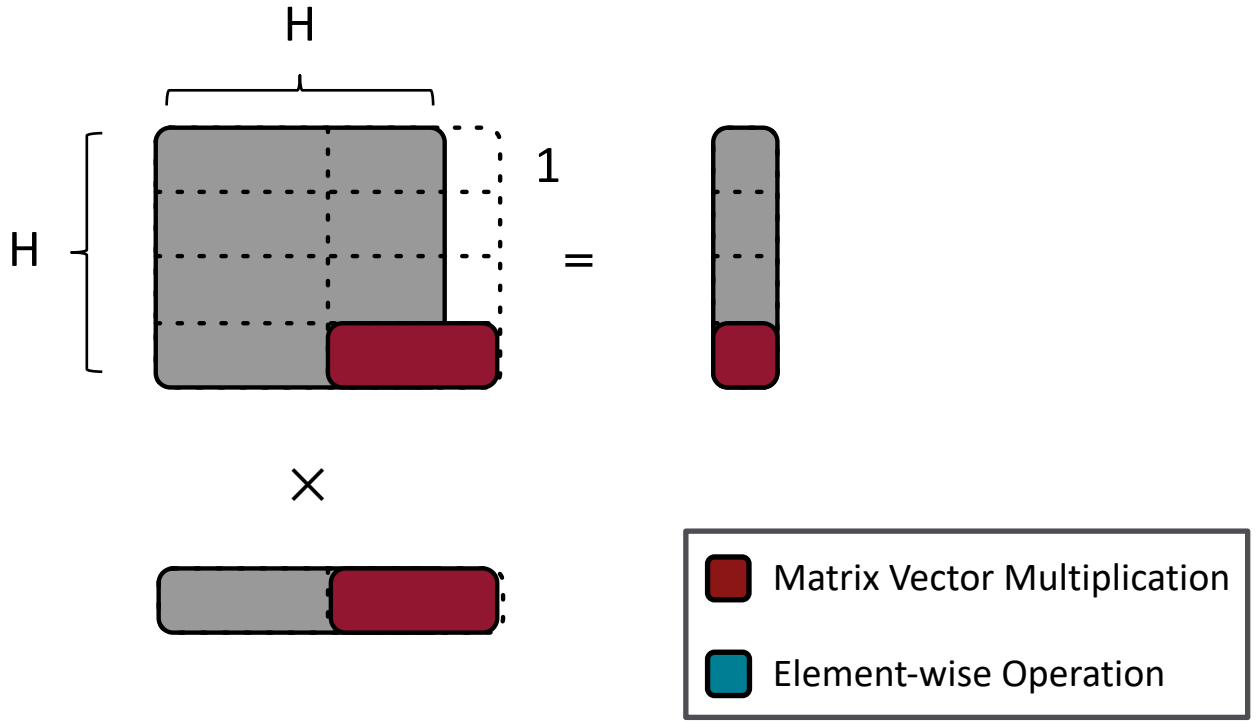
Fine Grain Tiling Leads to Better HW Utilization



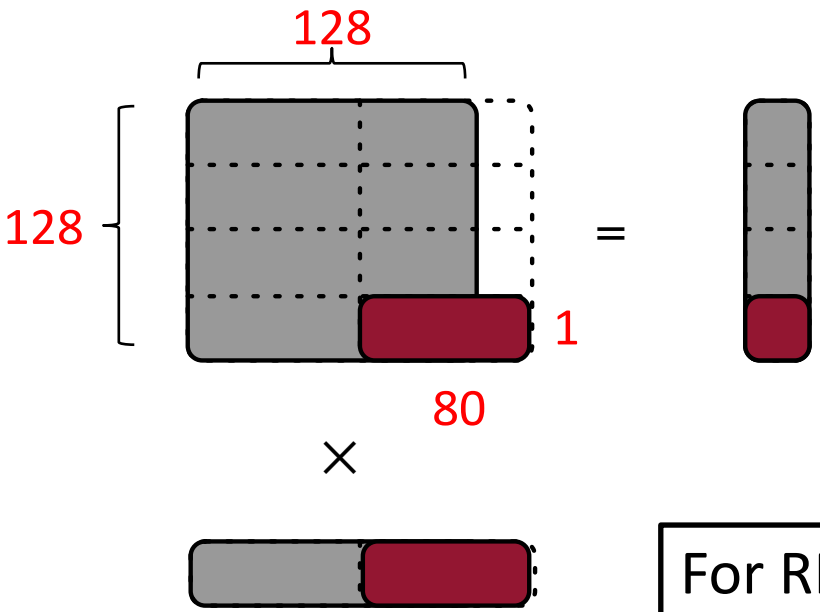
Fine Grain Tiling Leads to Better HW Utilization



Fine Grain Tiling Leads to Better HW Utilization

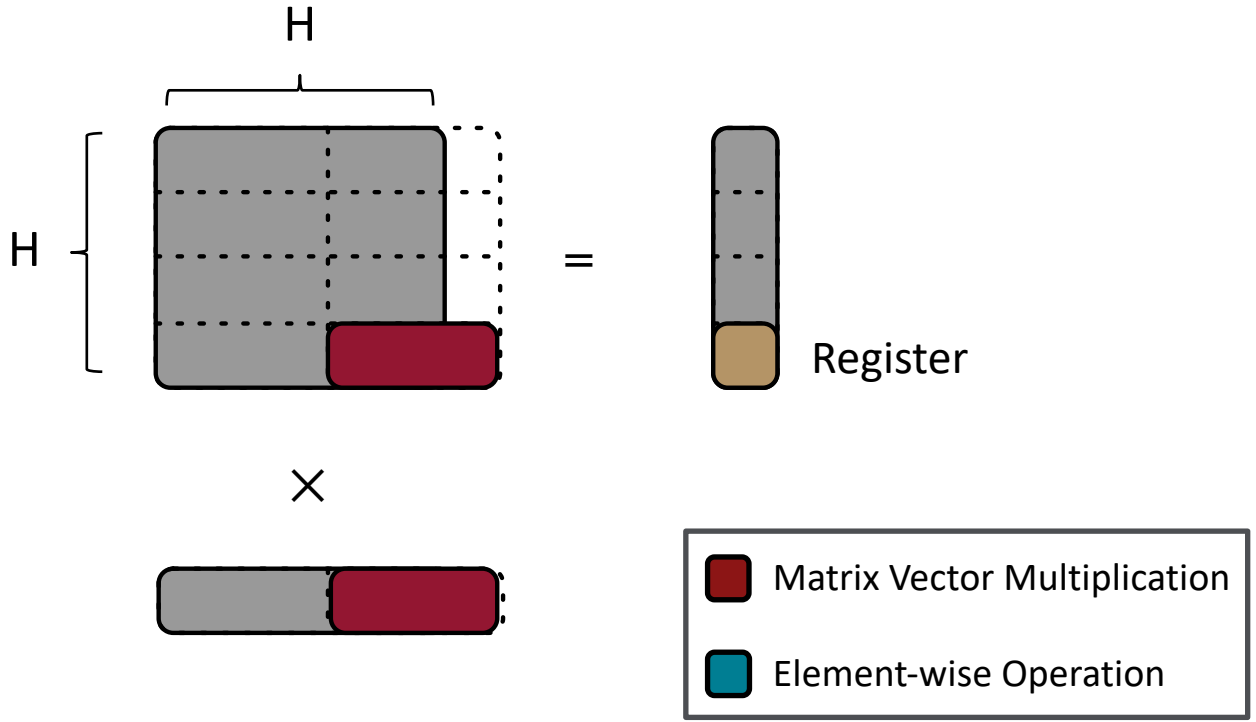


Fine Grain Tiling Leads to Better HW Utilization

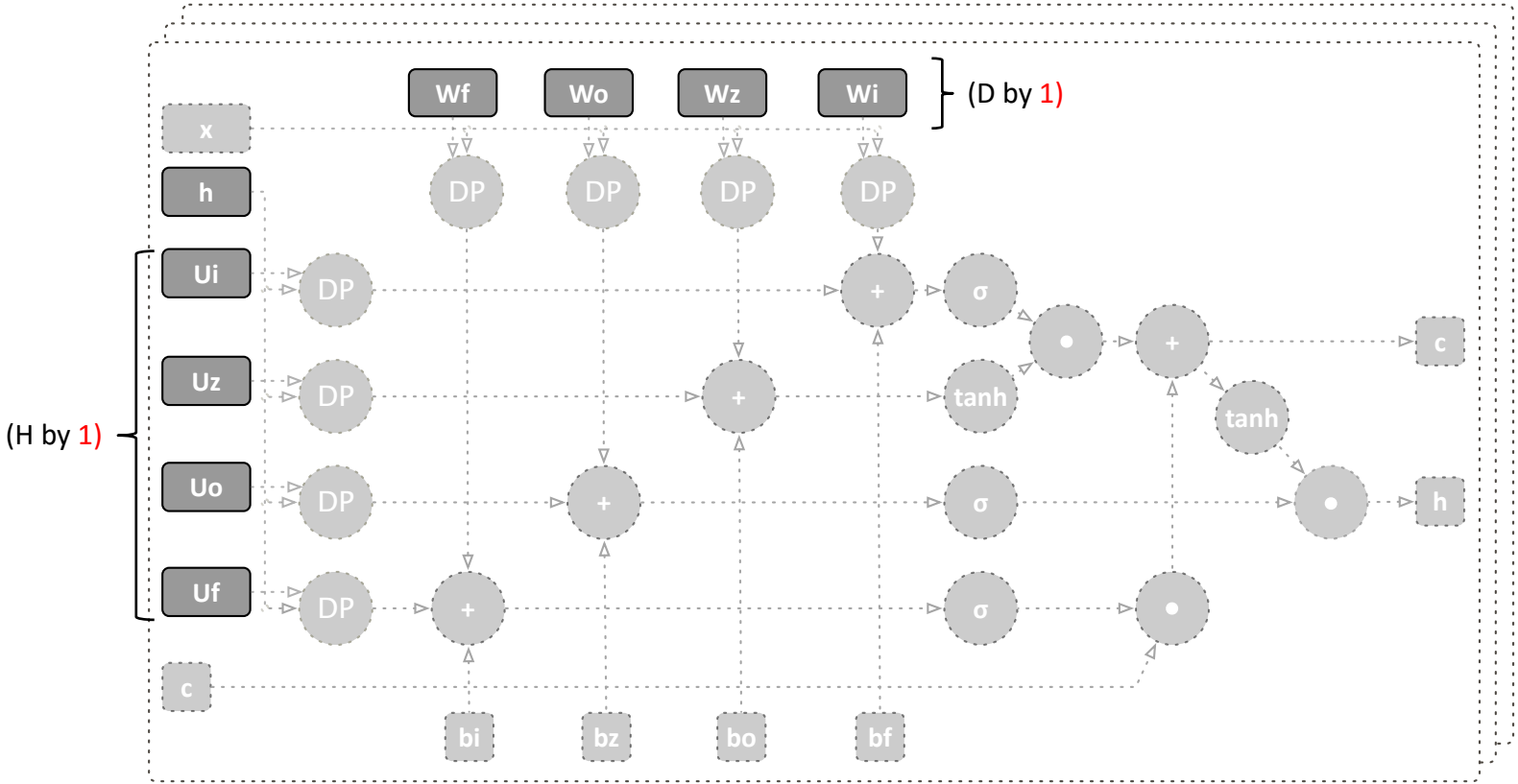


For RNN-128, utilization is 80%!

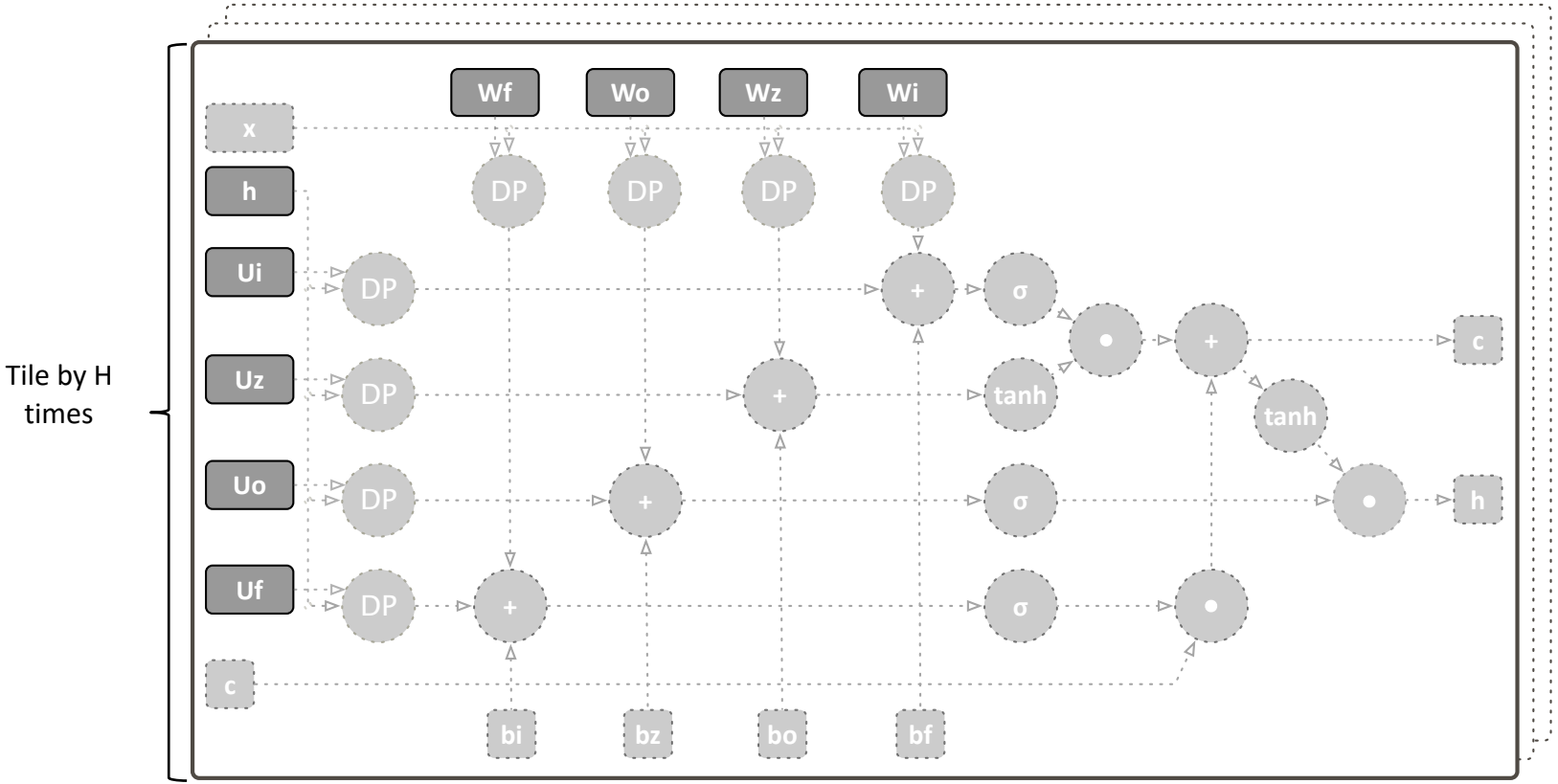
Intermediate Results Buffered in Register



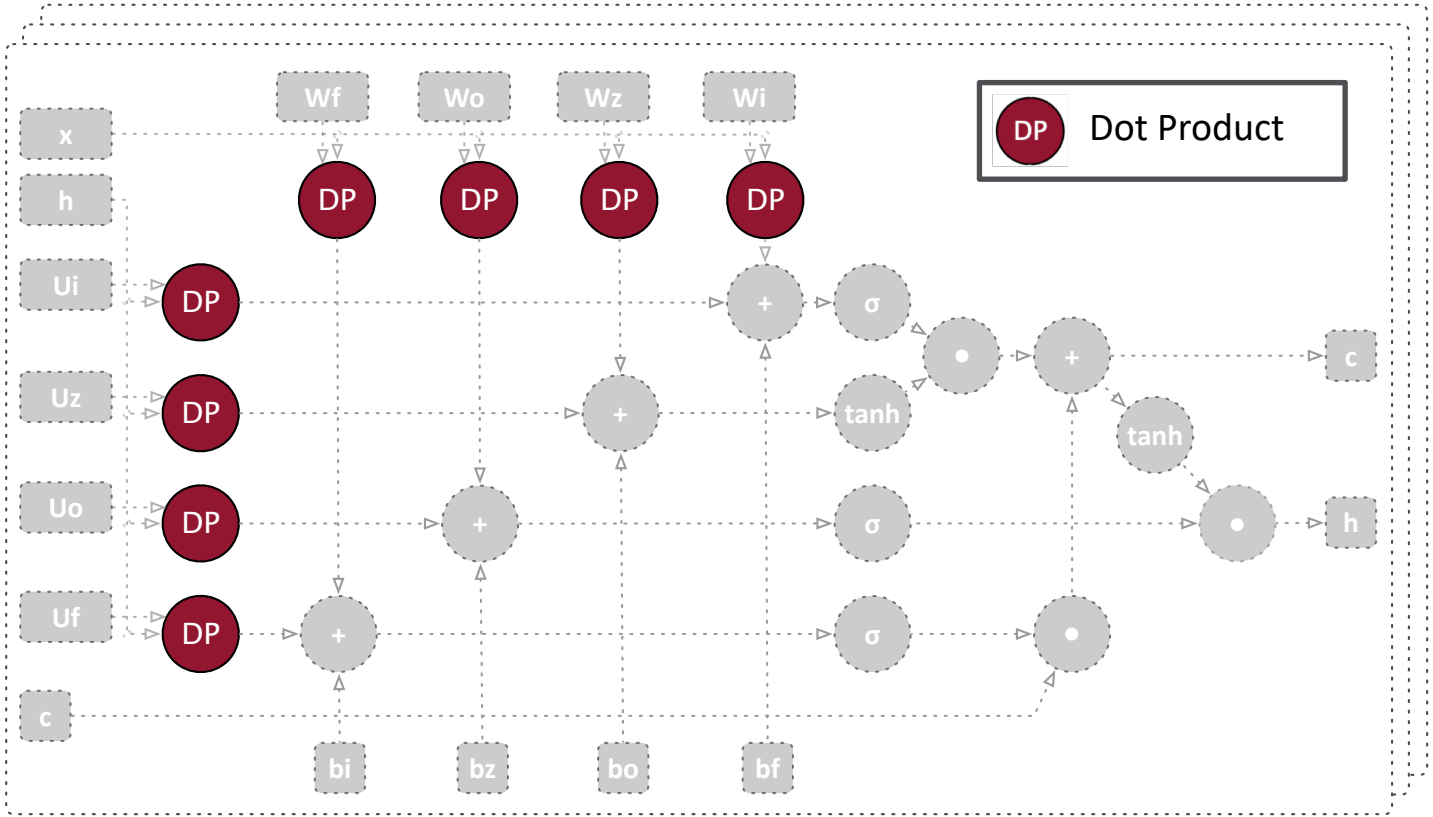
Fine Grain Tiling along the Hidden Unit Dimension



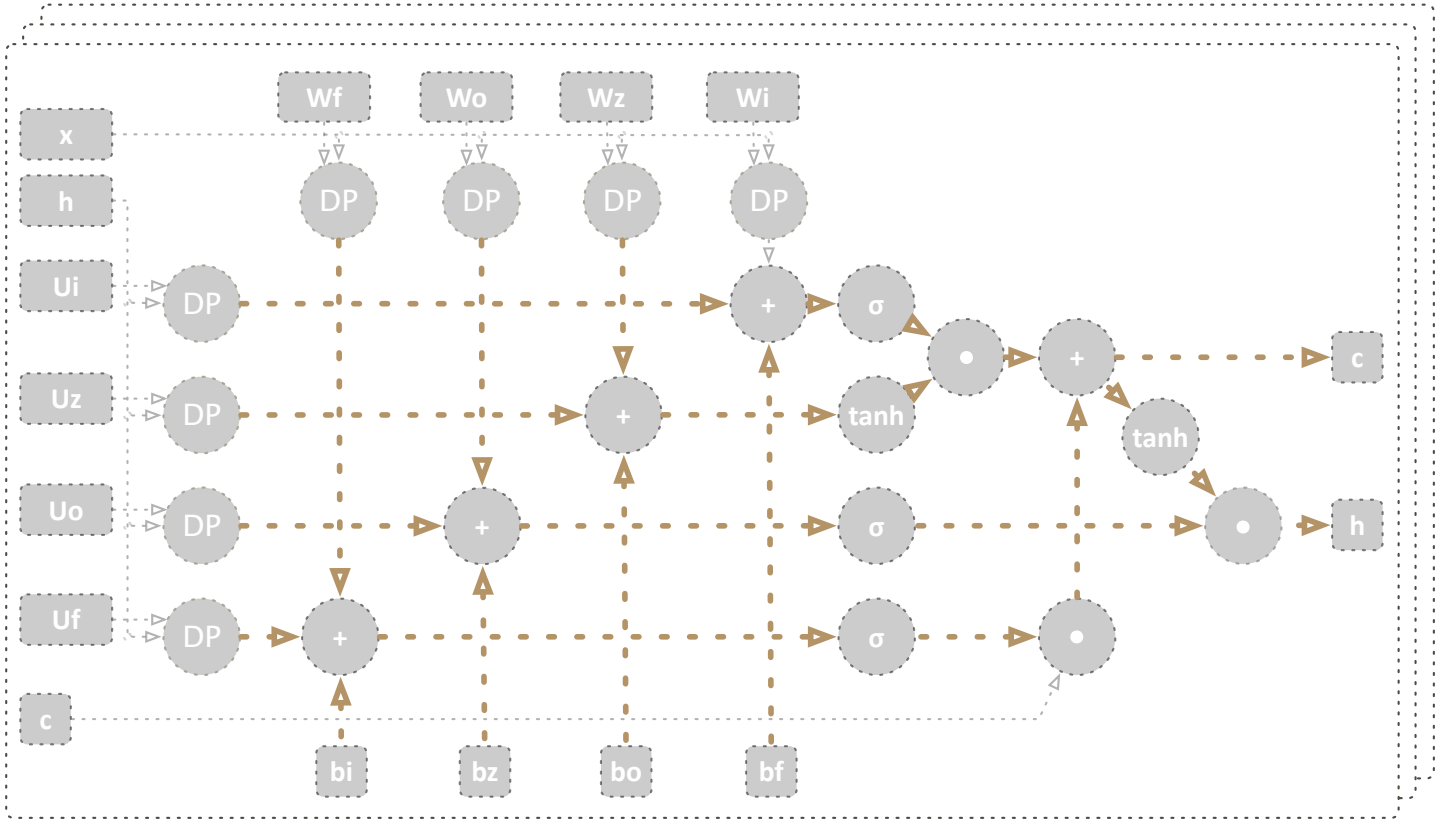
Fine Grain Tiling along the Hidden Unit Dimension



Fine Grain Tiling Converts MVM to DP



Fine Grain Tiling Uses Cheaper Memory Elements

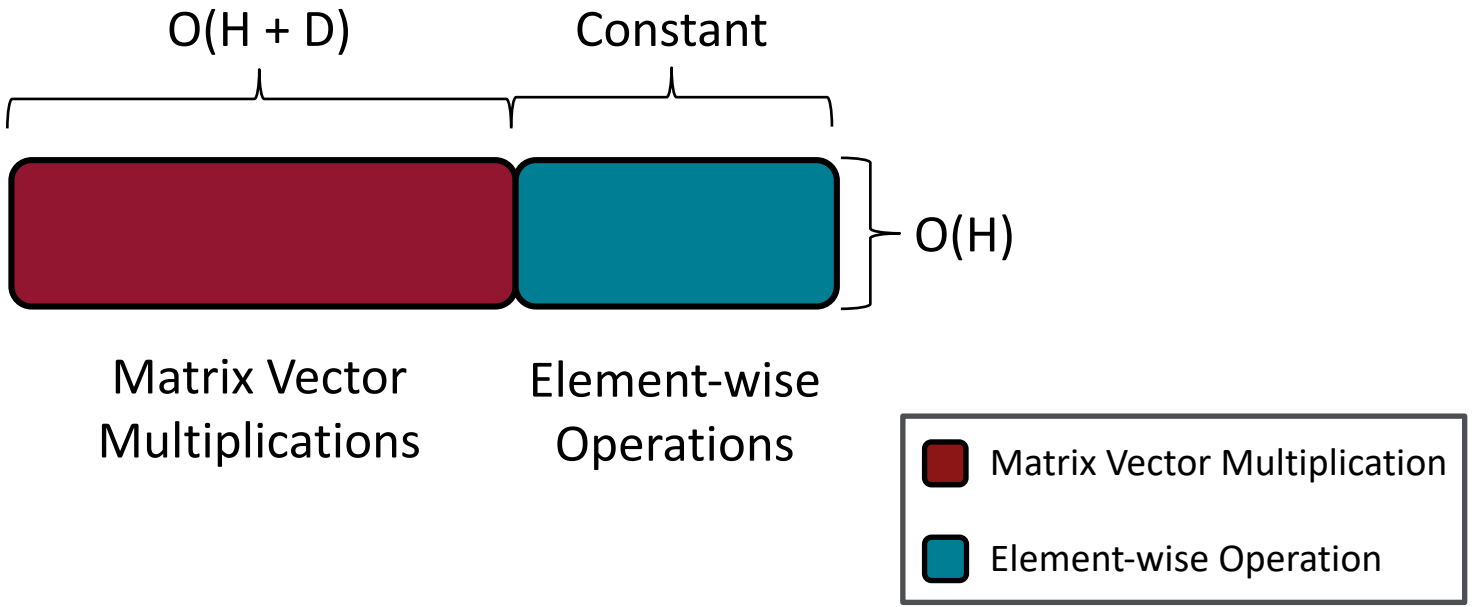


Loop Abstraction Enables Fine Grain Tiling to:

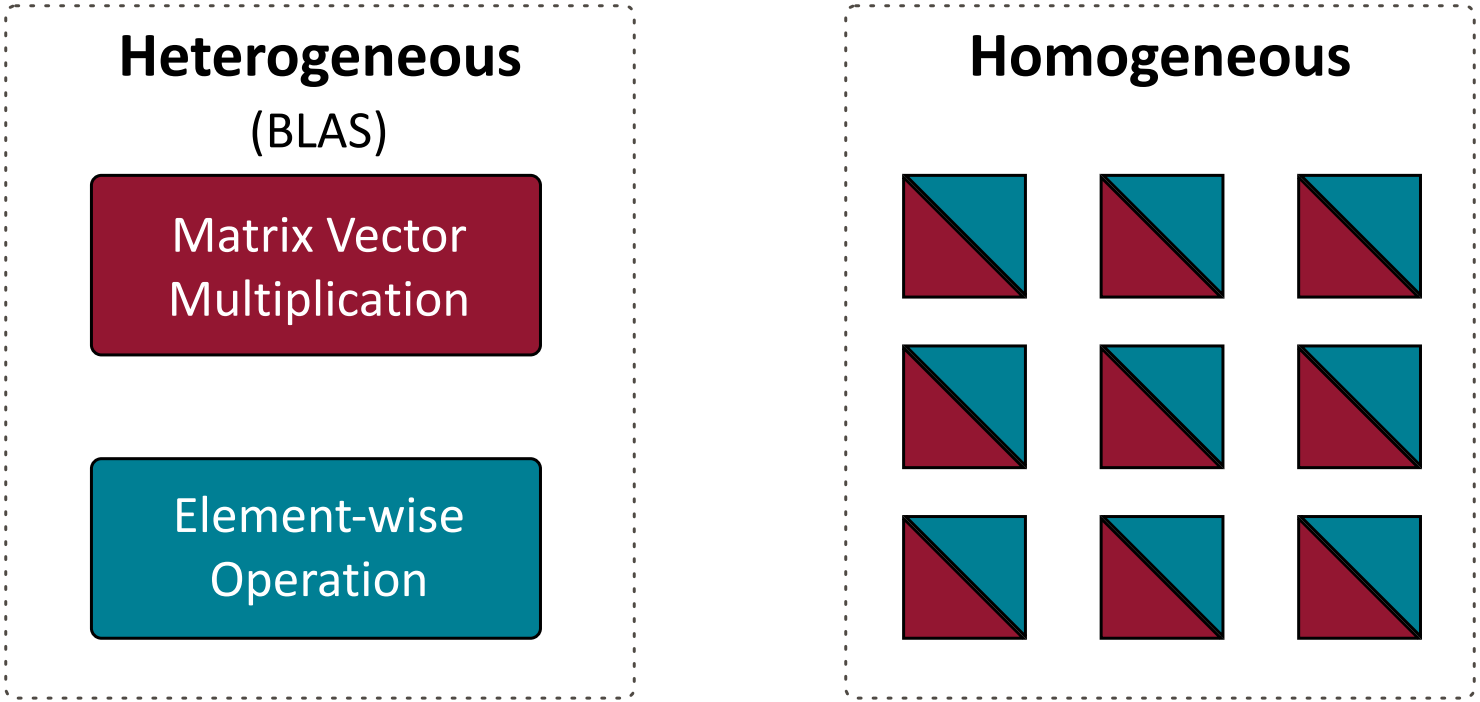
- reduce hardware underutilization due to unalignment.
- reduce the size of the intermediate buffers.

BLAS abstraction leads to a heterogenous accelerator design that contains unbalanced pipeline.

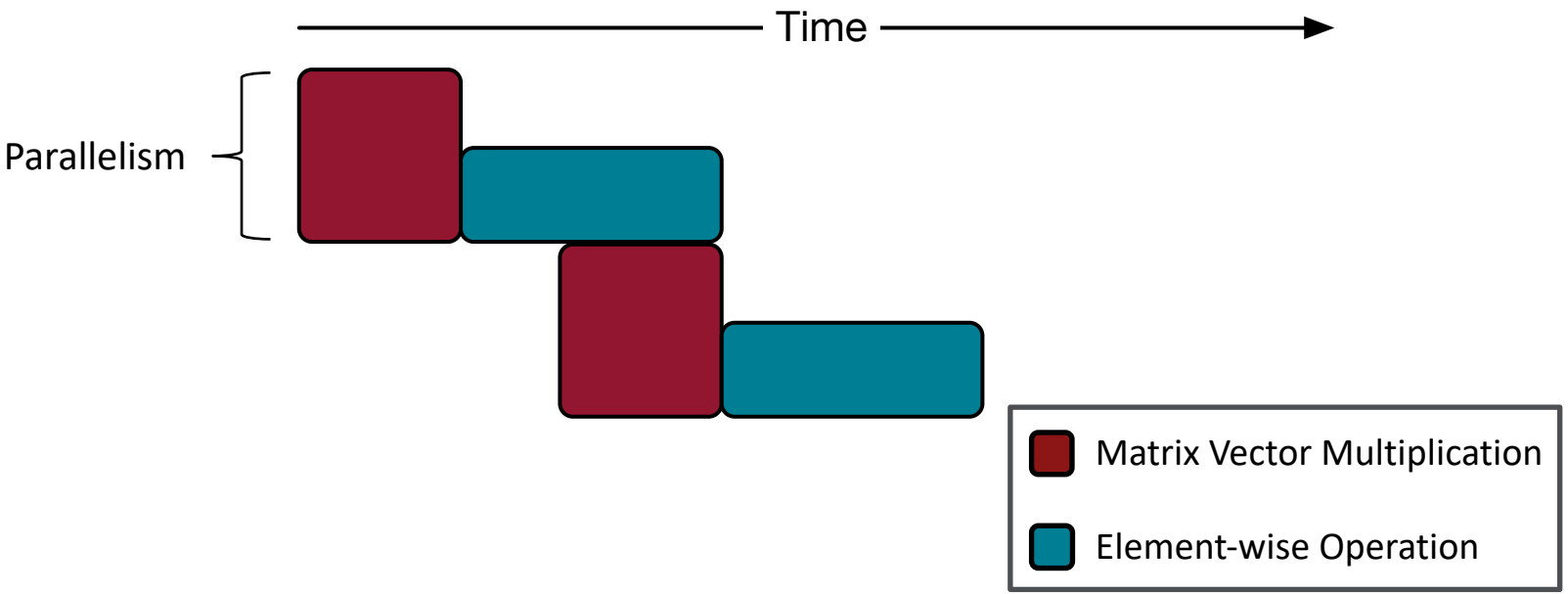
Pipelining the RNN Serving Task



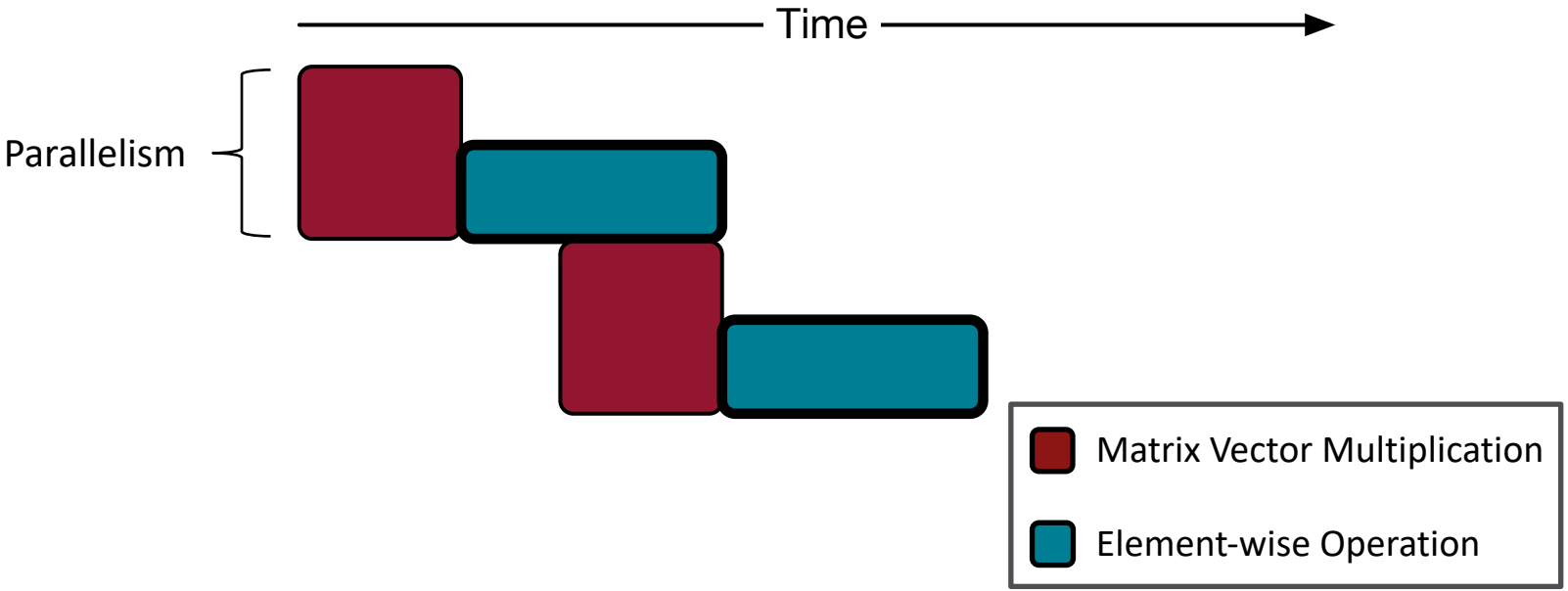
Heterogeneous vs. Homogeneous Accelerators



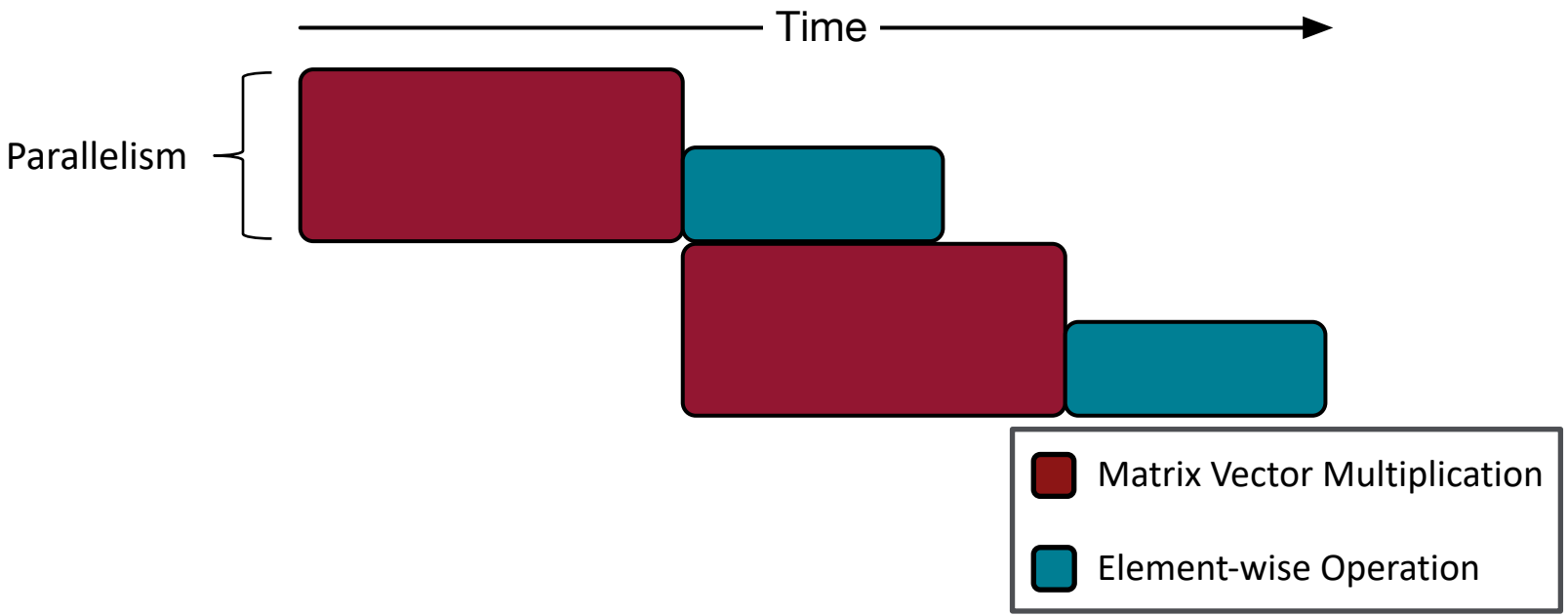
Pipeline within a Heterogeneous Accelerator



Pipeline within a Heterogeneous Accelerator

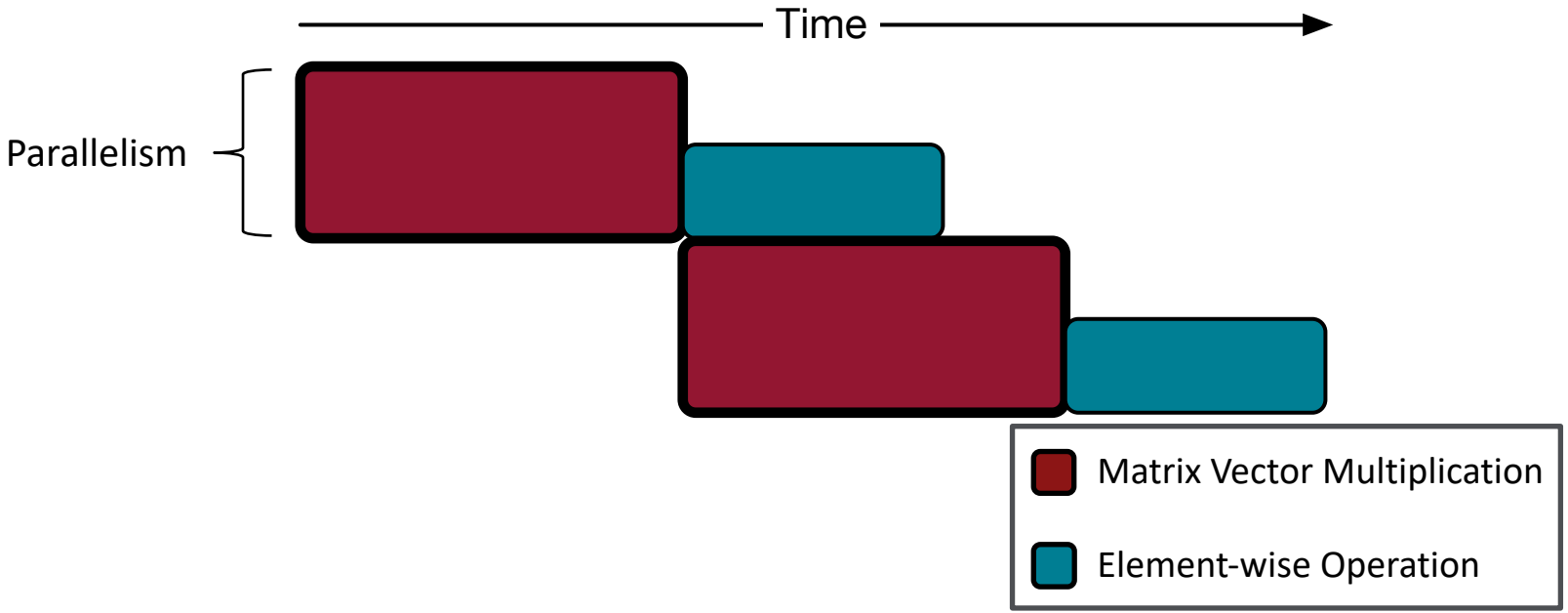


Pipeline within a Heterogeneous Accelerator

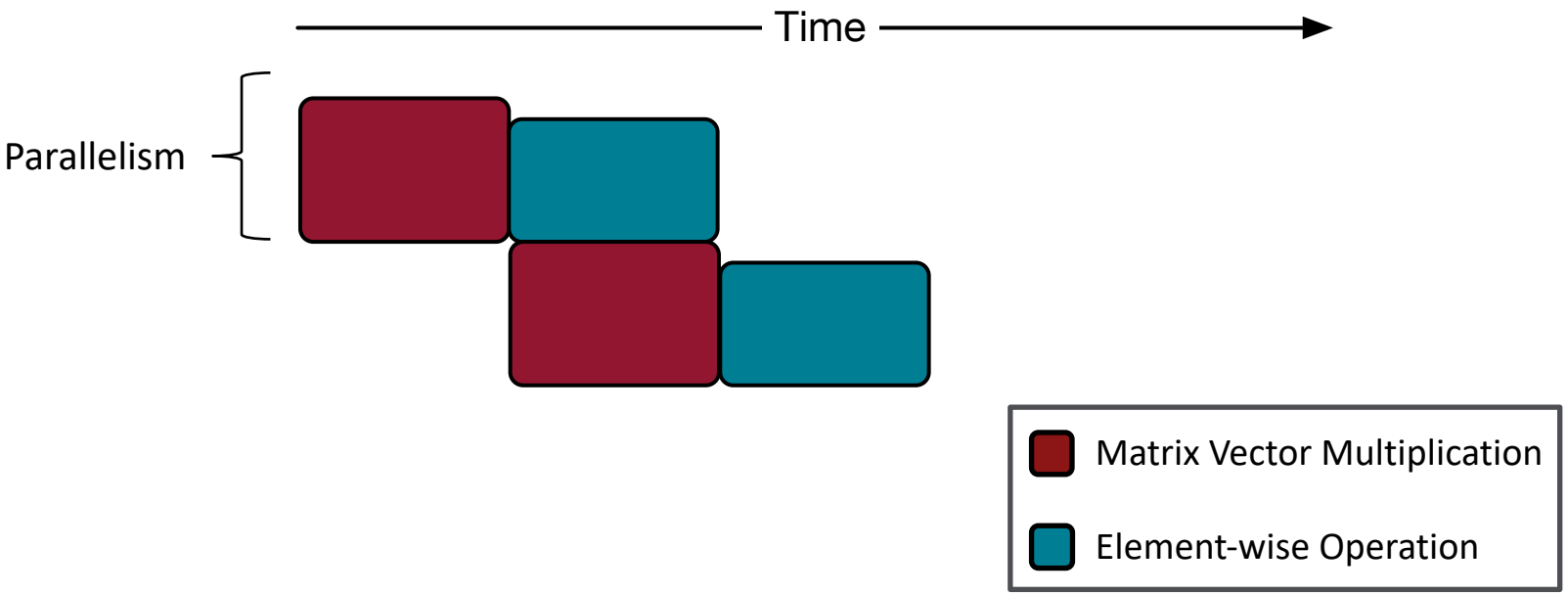


A **heterogenous** accelerator will have an unbalanced pipeline with respect to different problems.

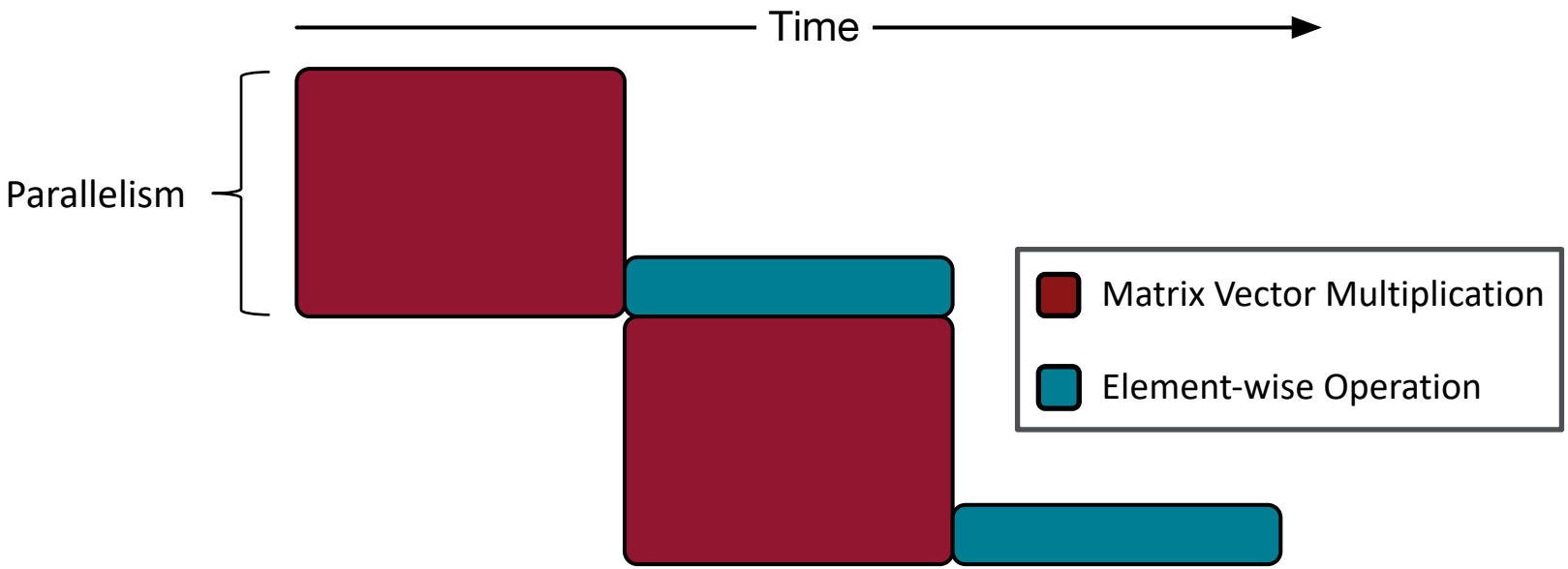
Pipeline within a Heterogeneous Accelerator



Pipeline within a Homogeneous Accelerator



Pipeline within a Homogeneous Accelerator



A **homogenous** accelerator can achieve a balanced pipeline regardless of the problem sizes.

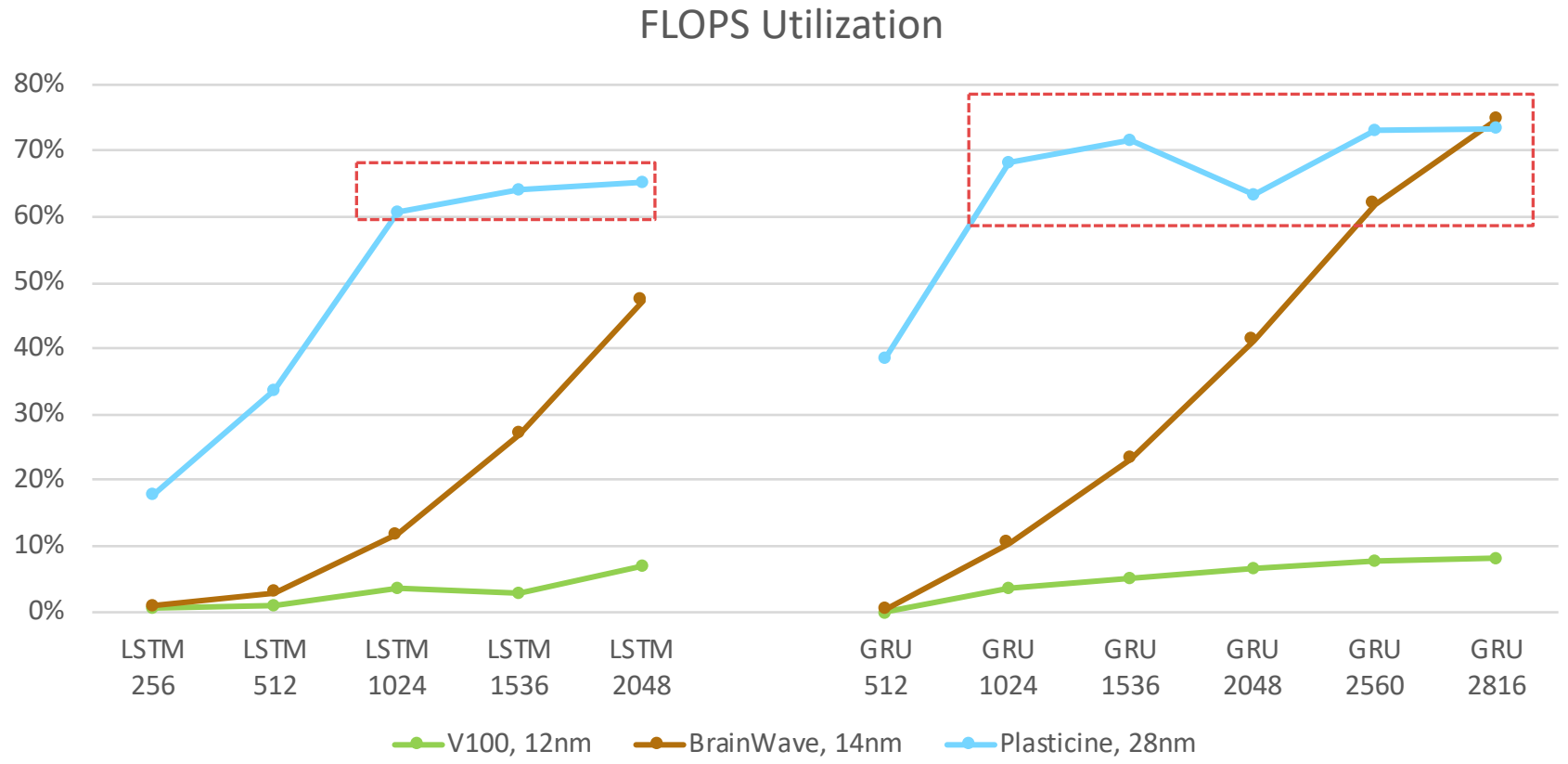
Evaluation Configurations

Specification	Tesla V100 GPU	Stratix 10 FPGA	Plasticine CGRA
Programming Language	TensorFlow + cuDNN	Brainwave ISA	Spatial Lang.
Accelerator Type	Temporal	Spatial	Spatial
ISA Type	MMM	MVM	Loop
Implementation Type	Heterogeneous	Heterogeneous	Homogeneous

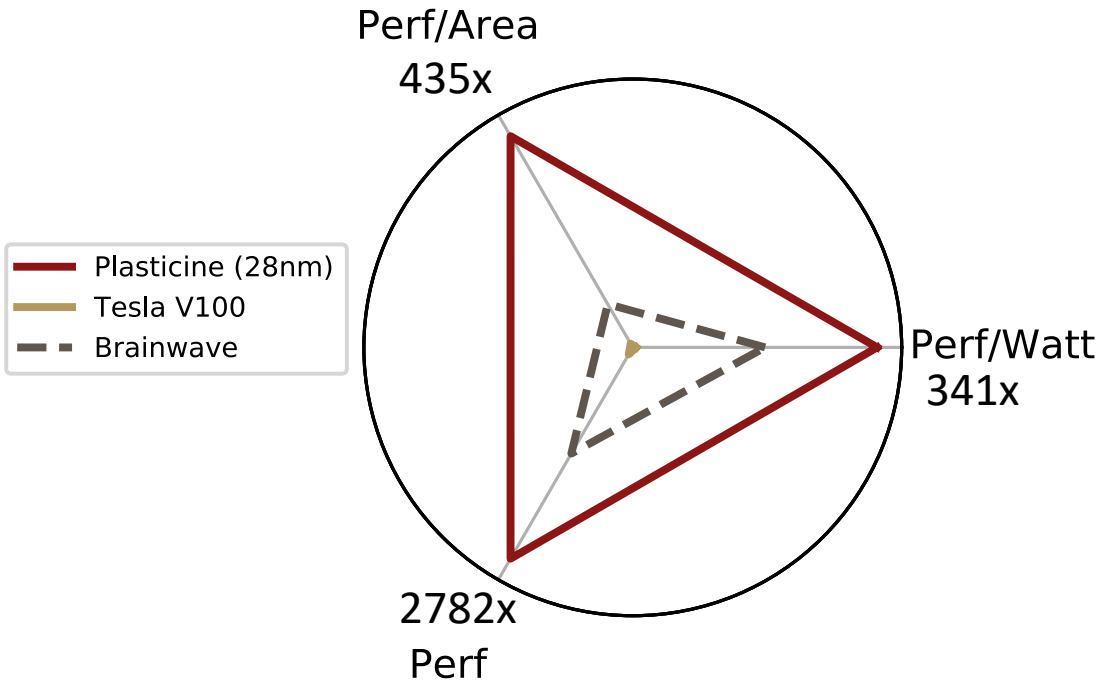
Evaluation Configurations

Specification	Tesla V100 GPU	Stratix 10 FPGA	Plasticine CGRA
Peak 32-bit TFLOPS	15.7	10	12.5
Technology (nm)	12	14	28
Die Area (mm^2)	815	1200	494
TDP (W)	300	148	160

Evaluation on DeepBench



Improvement over CPU Baseline



Homogeneous accelerators with **loop-level abstraction** achieves better HW utilization