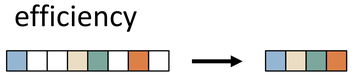
ISOSceles: Accelerating Sparse CNNs through Inter-Layer Pipelining

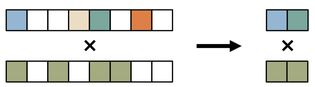
Yifan Yang, **Joel S. Emer**, **Daniel Sanchez**
{yifany, emer, sanchez}@csail.mit.edu

1. Background

- Leveraging sparsity significantly improves CNN inference efficiency

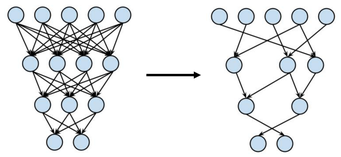


Reduces data movement

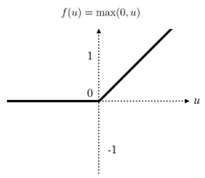


Avoids ineffectual computation

- Sparsity arises from two sources



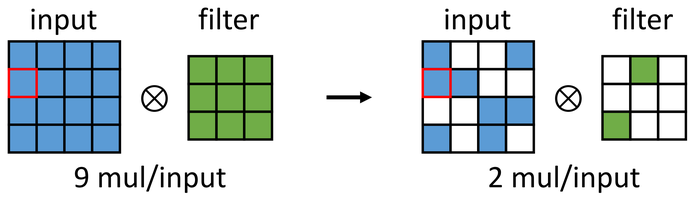
Weight sparsity: pruning



Activation sparsity: ReLU, etc.

2. Motivation

- Low data reuse in sparse CNN inference makes it bottlenecked by memory traffic



9 mul/input 2 mul/input

- Activation dominates memory traffic

Off-chip traffic of prior accelerator SparTen[MICRO 19]

3. Key Observation

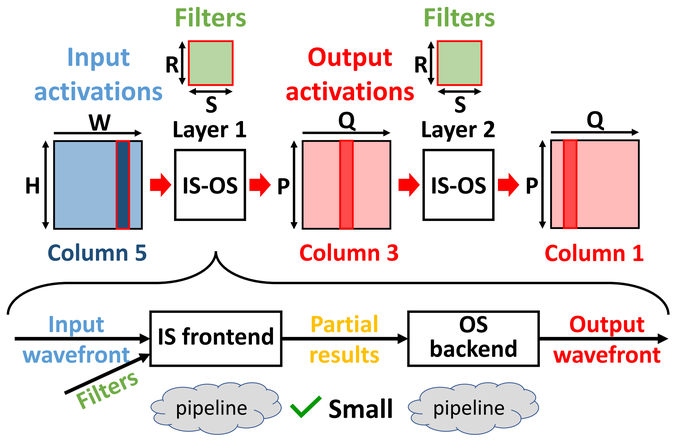
- Most accelerators process CNNs layer by layer
  - Intermediate activations between layers are large and spilled off-chip
- **Insight**: Pipelining the execution of multiple layers effectively reduces activation traffic
  - Intermediate activations are consumed immediately without spilling them off-chip

4. Our Contributions

- A **dataflow**, Input-Stationary Output-Stationary (IS-OS), that allows efficient pipelining of sparse layers
- A **hardware** accelerator, ISOSceles, that implements the dataflow

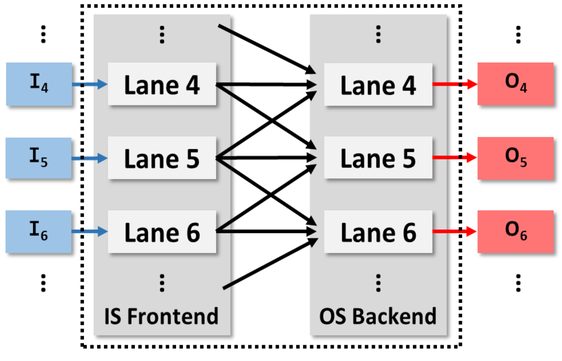
5. IS-OS Dataflow

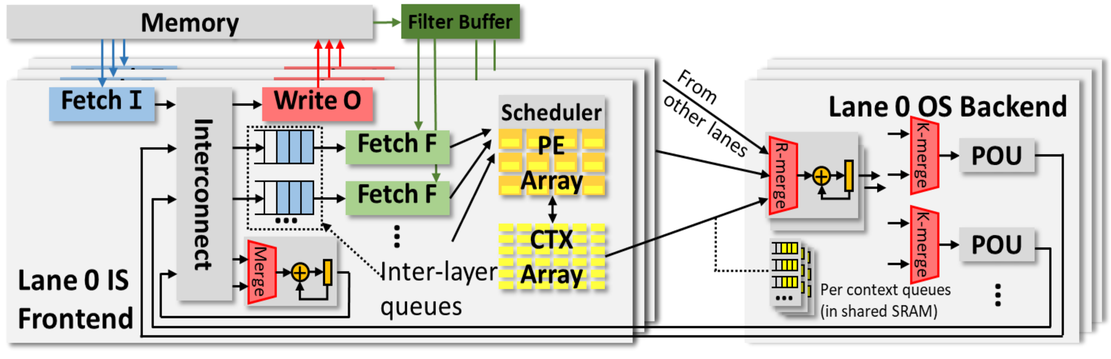
- Each layer consumes inputs and produces outputs in thin **wavefronts**
- Input-stationary (IS): each input wavefront is fully used with all relevant filters before moving onto next wavefront
- Output-stationary (OS): each output wavefront is fully accumulated before moving onto next wavefront



- Pipeline IS frontend and OS backend so that partial result storage is small

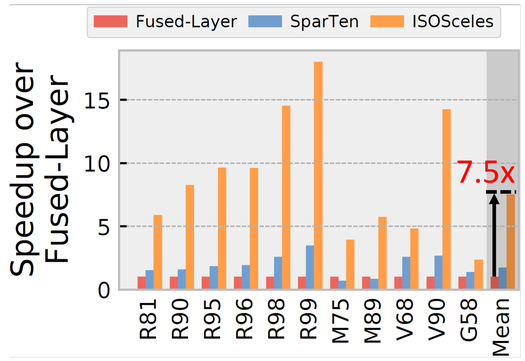
6. ISOSceles Accelerator

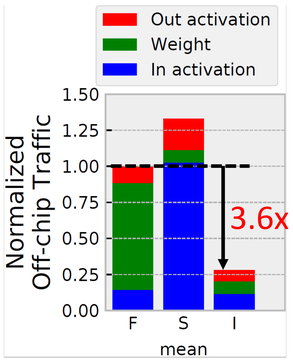




7. Evaluation

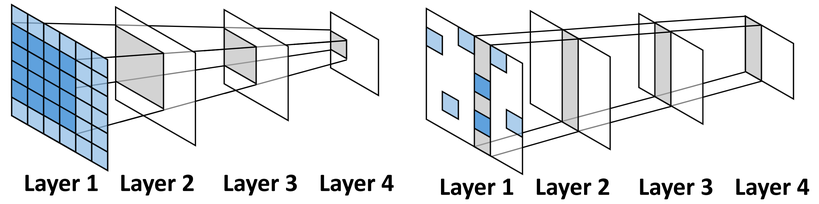
ISOSceles improves performance and reduces traffic





- Workloads: sparse ResNet-50, MobileNetV1, VGG-16, GoogLeNet on ImageNet
- Baselines: Fused-Layer[MICRO 16], SparTen[MICRO 19]

Comparison with 2D tiled dataflow



Fused-Layer[MICRO 16] OS dataflow

- ✗ Poor input reuse
- ✗ Large intermediate storage
- ✗ Sparsity unfriendly

IS-OS dataflow

- ✓ Good input & output reuse
- ✓ Small intermediate storage
- ✓ Sparsity friendly