FreeLunch: Compression-based GPU Memory Management for Convolutional Neural Networks

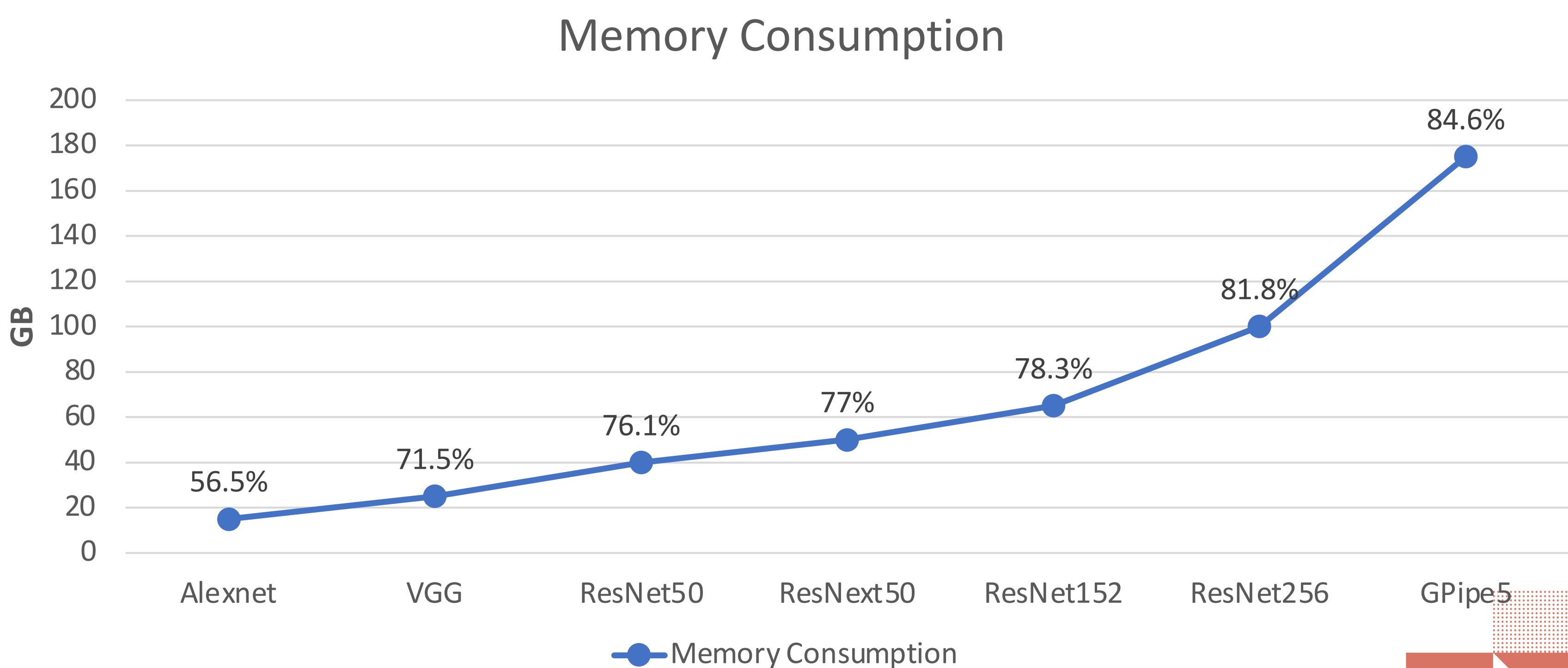
Shaurya Patel
University of British Columbia
University of Massachusetts, Amherst

Tongping Liu
University of Massachusetts, Amherst

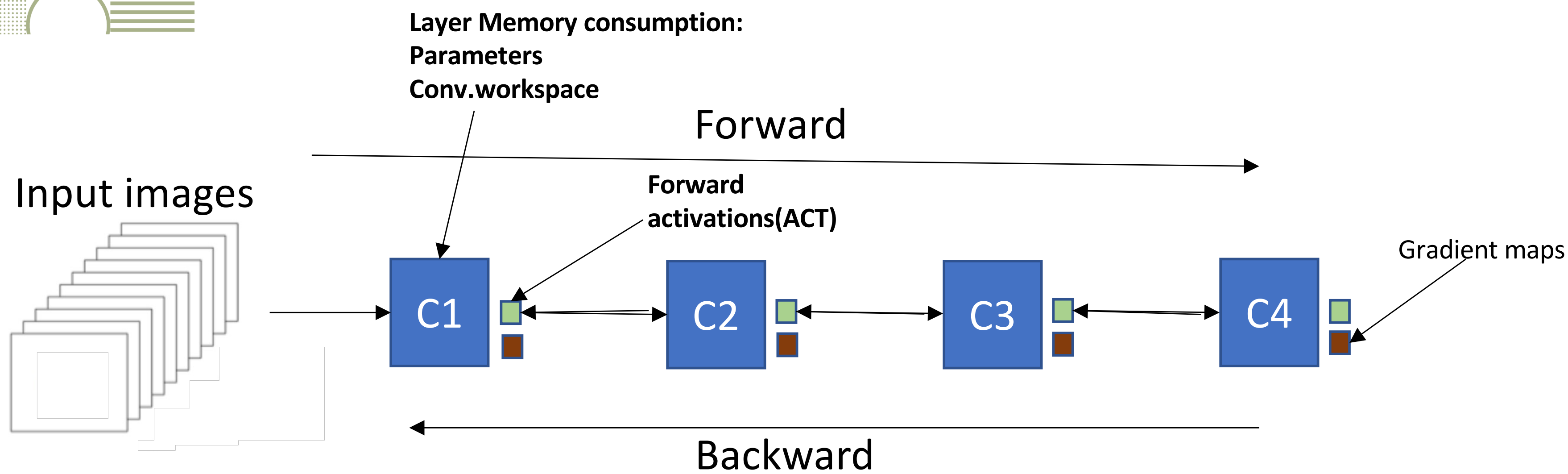
Hui Guan
University of Massachusetts, Amherst

SC21
St. Louis, MO | science & beyond.

CNN memory consumption trend



CNN training



Forward Activations(much larger in size than model params) need to persist in memory until the gradient updates in backward phase!

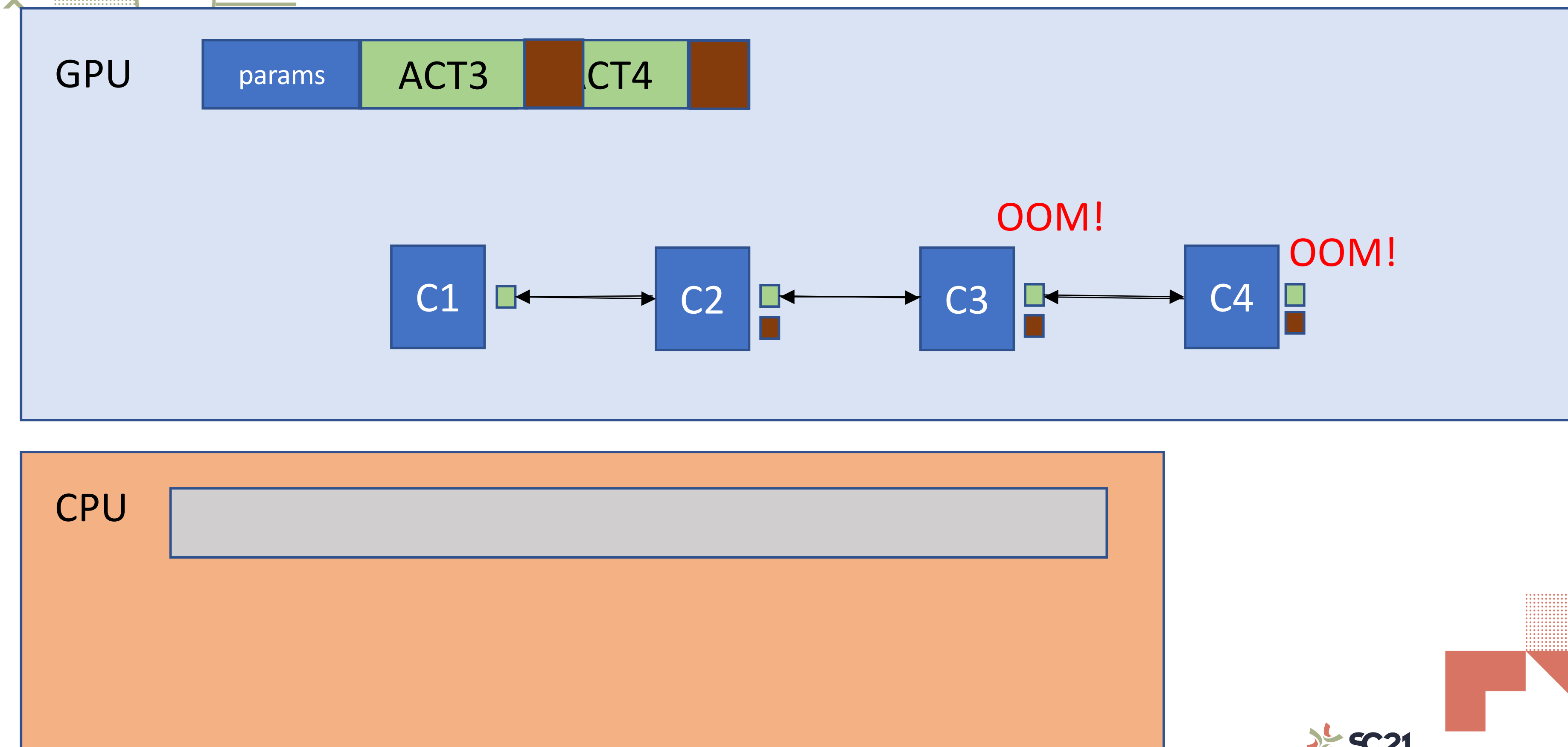
Policies for memory management

- Swapping
  - Capuchin [X. Peng et al., 2020]
  - SwapAdvisor [C-C. Huang et al., 2020]
  - Superneurons [L. Wang et al., 2018]
  - …

- Recomputation
  - Capuchin [X. Peng et al., 2020]
  - Superneurons [L. Wang et al., 2018]
  - …

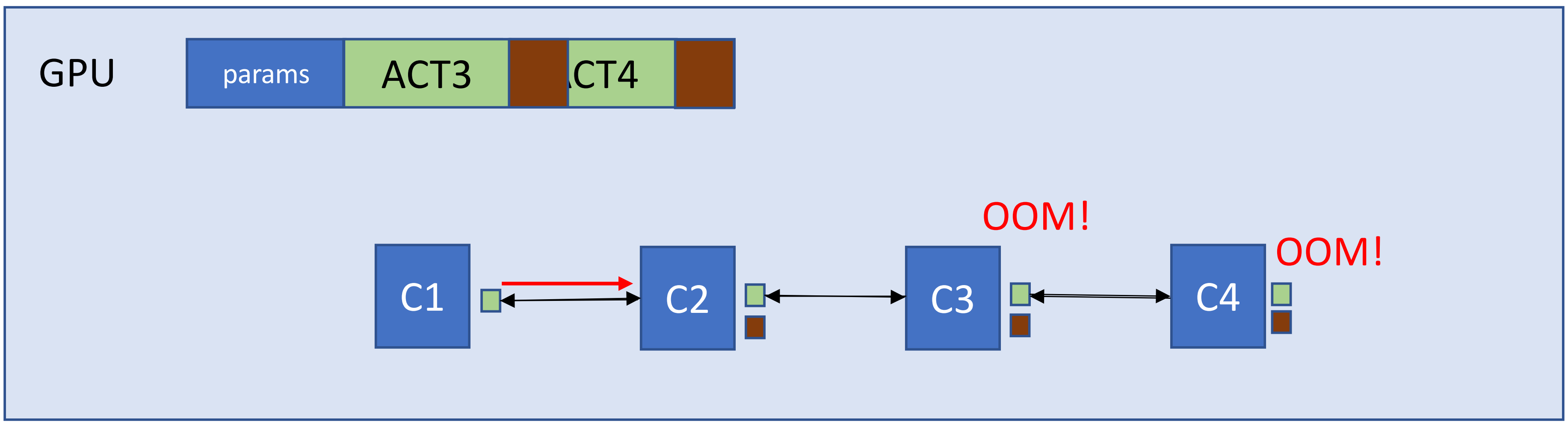
Swapping

CPU-GPU bandwidth is a bottleneck!



Recomputation

Recomputation is complex and has lineage dependencies!



FreeLunch

- A compression based policy for CNN training.
  - basic Idea: **compress and keep the tensors on GPU memory**.
  - avoids the bandwidth issue introduced by swapping.
  - avoids the computation complexity of recomputation.

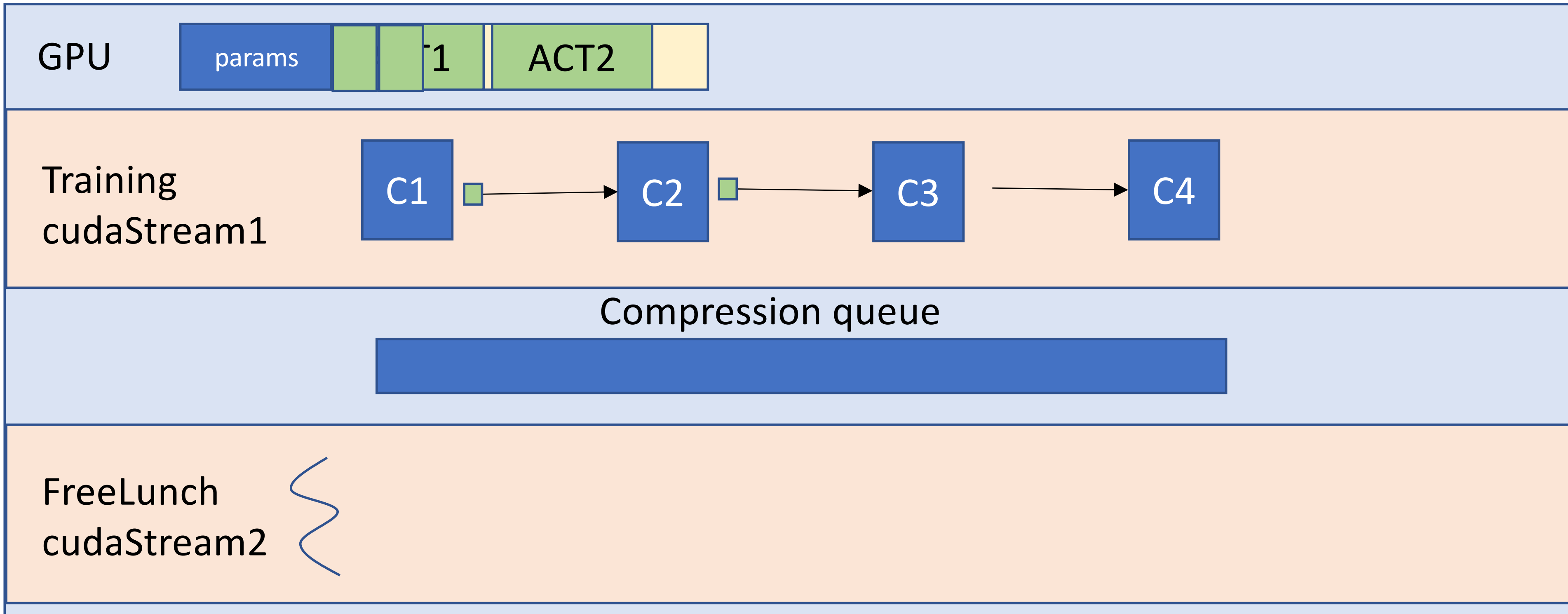
- Challenges:
  - How to reduce the compression overhead?

Parallel workflow

Optimizations:

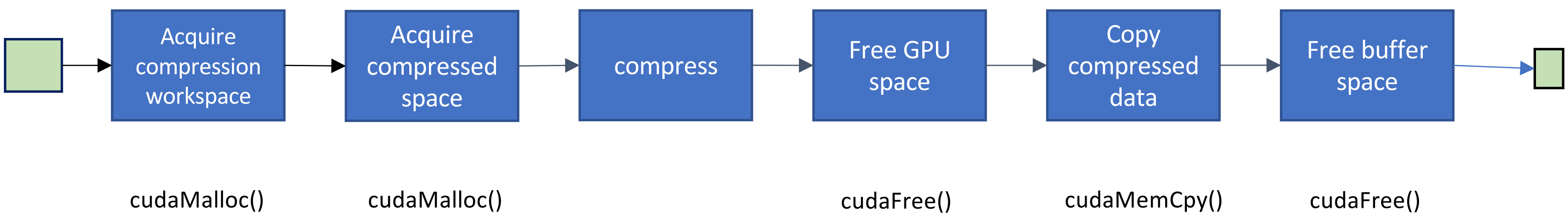
- Sliding Compression Workspace
- Persistent Tensor Buffers

Parallel workflow



Typical Compression workflow

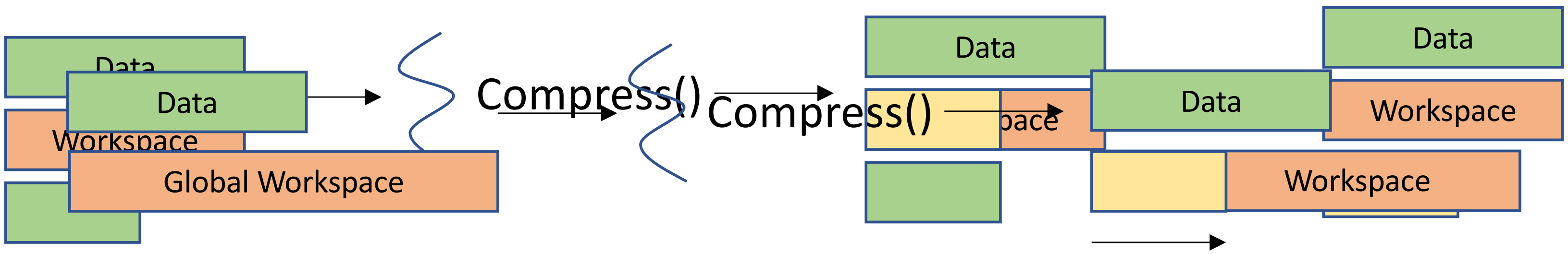
Memory operations synchronize all cuda streams!



Sliding Compression Workspace

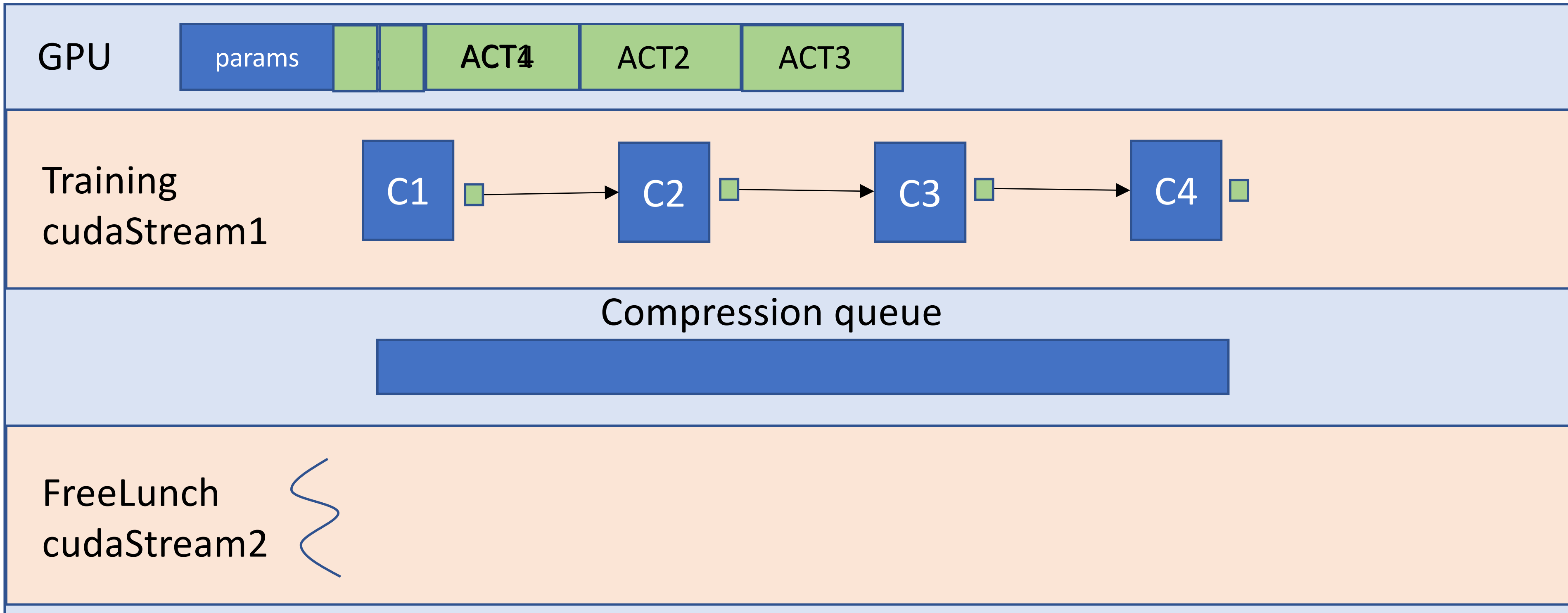
Typical workflow implementation

Sliding compression workspace workflow



This workflow introduces multiple blocking operations!

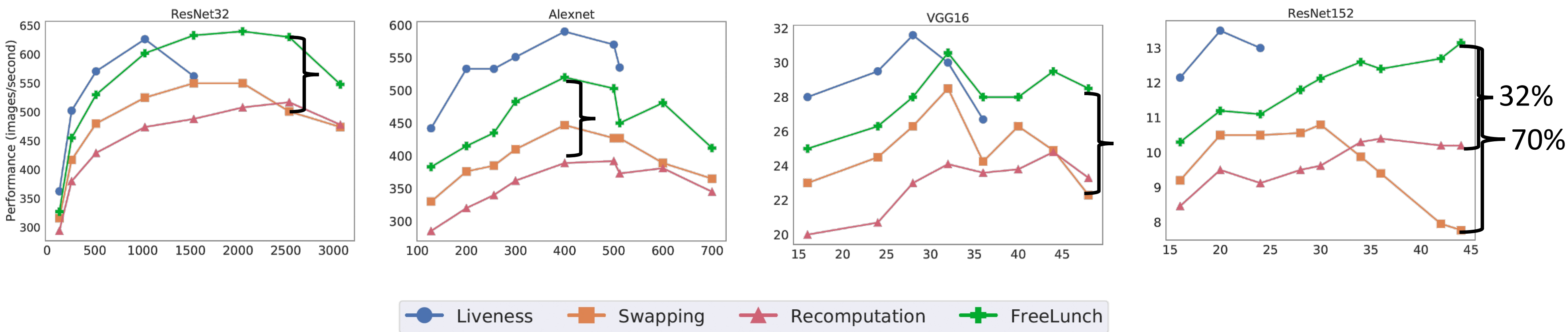
Persistent tensor buffers



Experiment setup

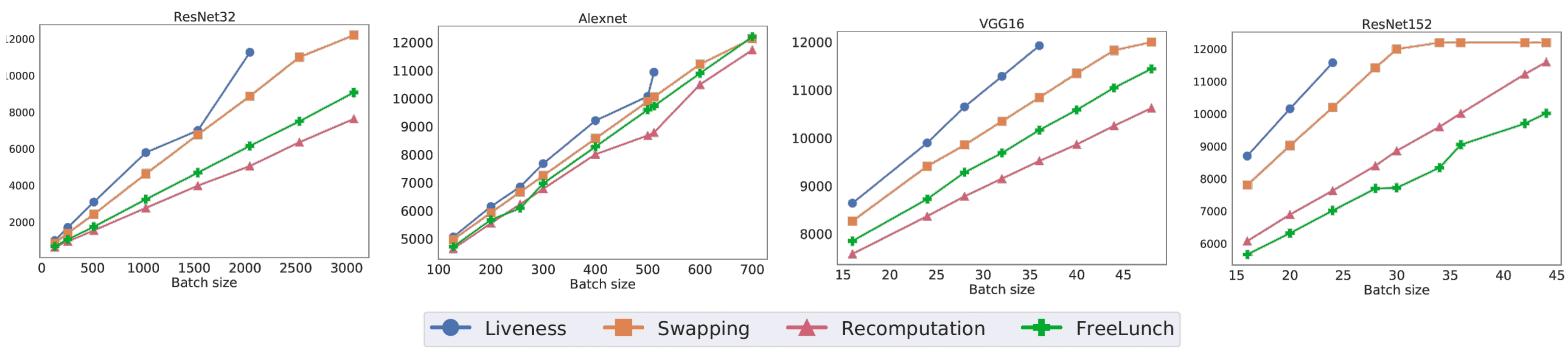
- Can FreeLunch improve training throughput while reducing memory consumption of CNN training?
- How effective are the optimizations in FreeLunch compared with other compression-based baselines?

Throughput as compared to other policies

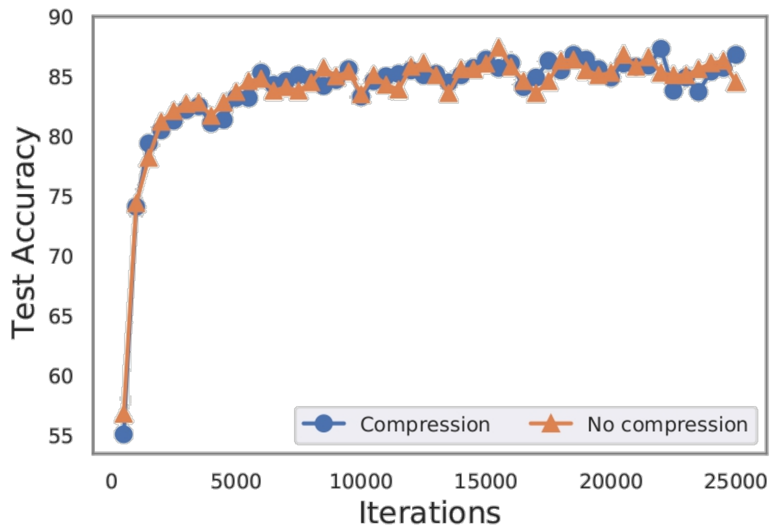


SC21

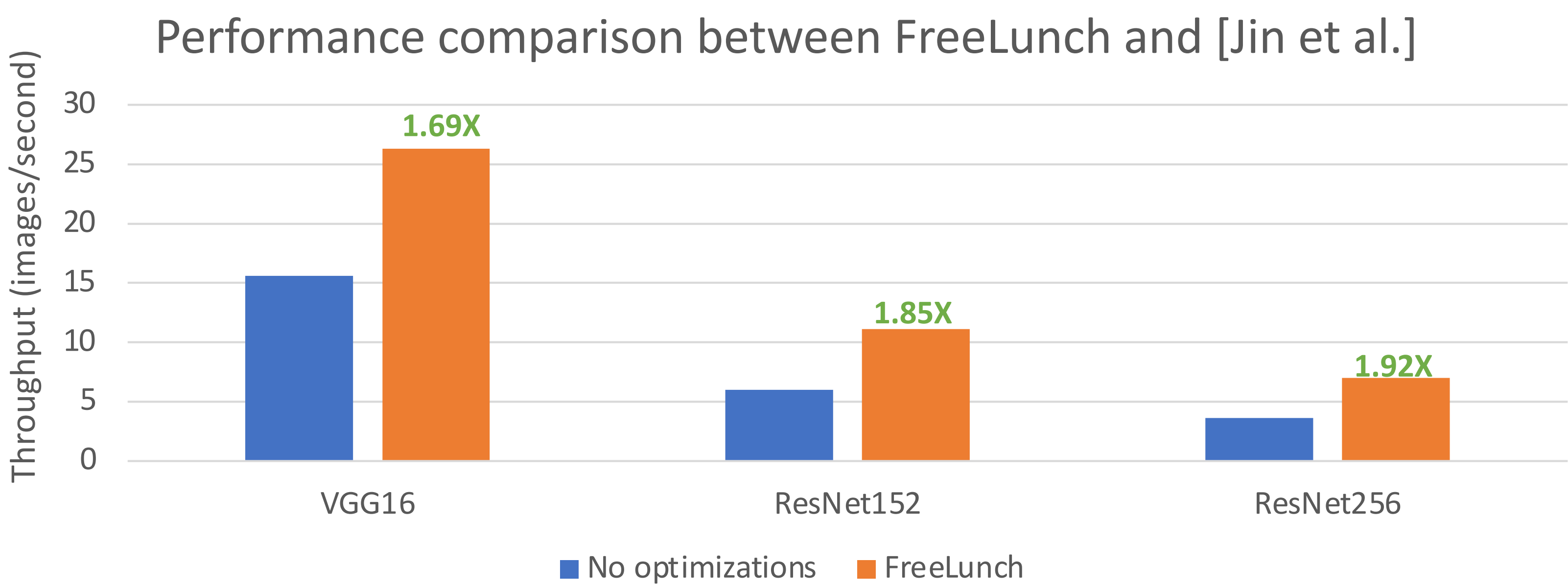
Memory consumed as compared to other policies



No observed impact on accuracy of model

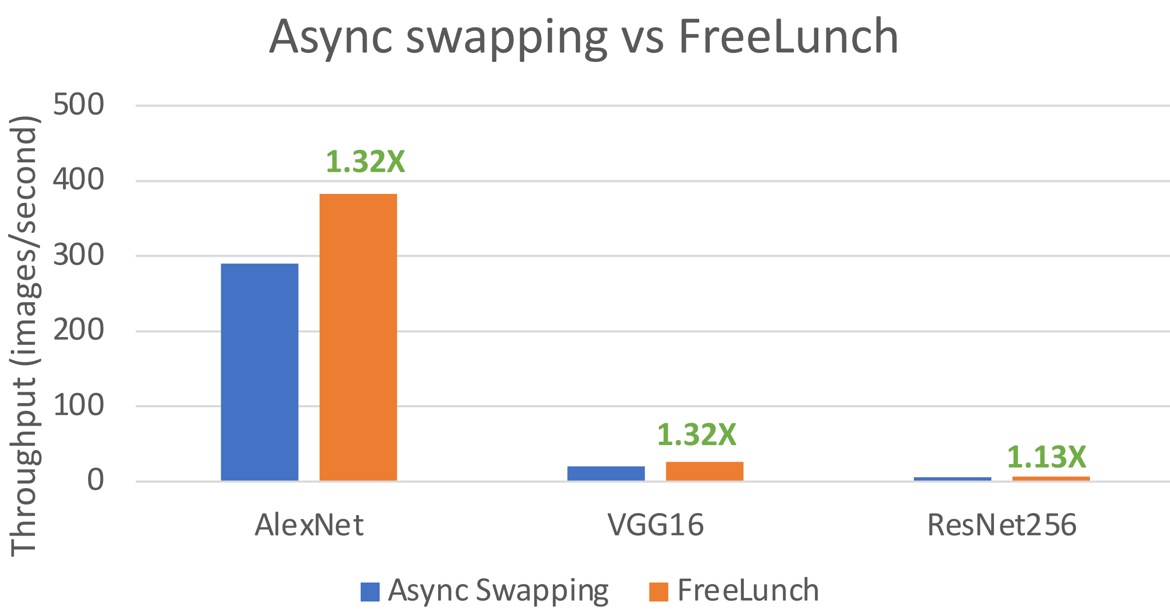


Impact of optimizations



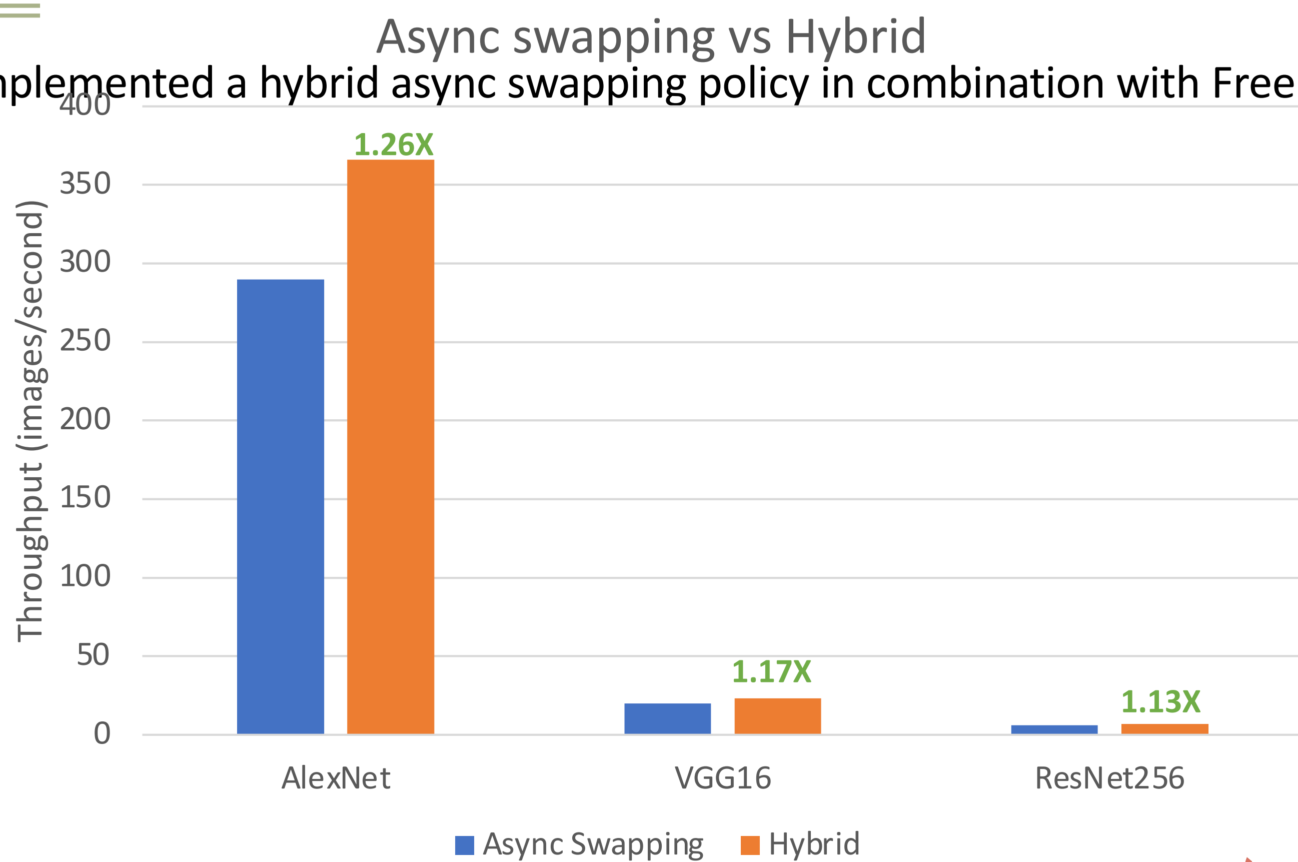
Throughput comparison with async swapping

- Capuchin and SwapAdvisor use swapping in an async manner.
- We implement async swapping and compare it to FreeLunch.



Hybrid policy

- We implemented a hybrid async swapping policy in combination with FreeLunch



Summary

- We introduce FreeLunch that effectively avoids the bandwidth and concurrent execution that swapping and recomputation face.
- We incorporate two optimizations as part of FreeLunch to make compression parallelizable and improve performance.
- We show that FreeLunch achieves up to 70% better throughput and up to 32% better memory consumption.