

# Resource-Efficient Training via Task Collocation on GPUs

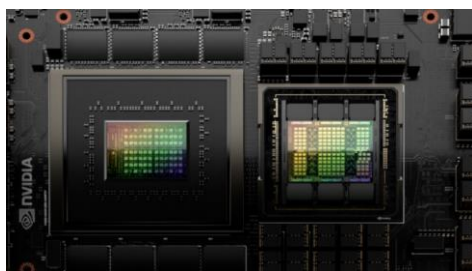
Ehsan Yousefzadeh-Asl-Miandoab

# GPU underutilization for machine learning tasks

- 100,000 real-world jobs show ~52% GPU utilization on average\*
  - **Energy-inefficient** & **waste of hardware resources**

## Hardware problem

1. Compute/memory requirements of models don't match with the modern GPUs
  - e.g., transfer learning, small models
2. No fine-grained sharing mechanism and virtual memory on GPU



NVIDIA H200

141GB GPU memory

50MB L2 cache

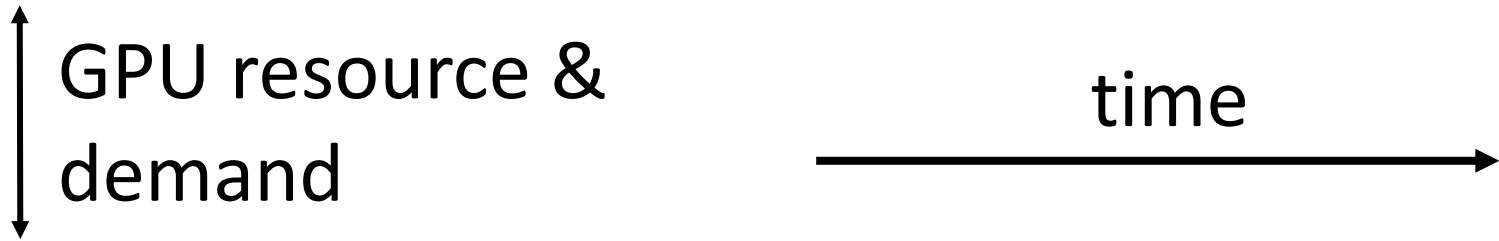
4.8TB/s MemoBand

## Software problem

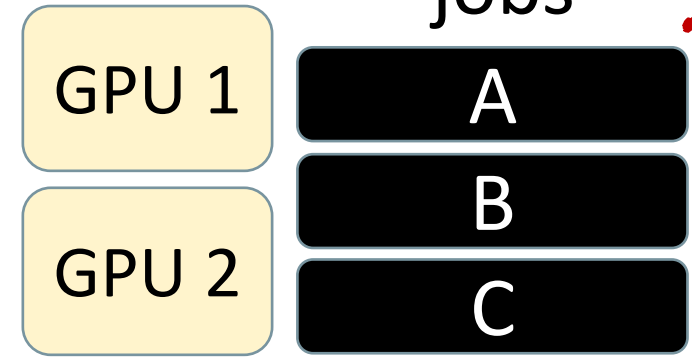
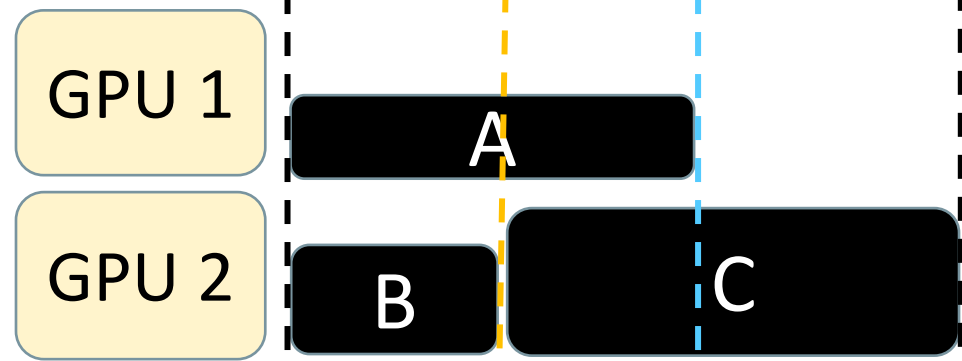
1. Exclusive GPU assignments
  - no finer-grained collocation
2. Mostly black-box view of tasks



# Collocation of Model Training

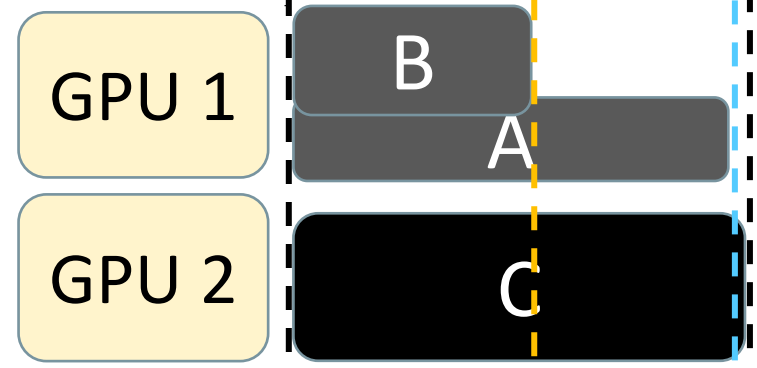


Conventional



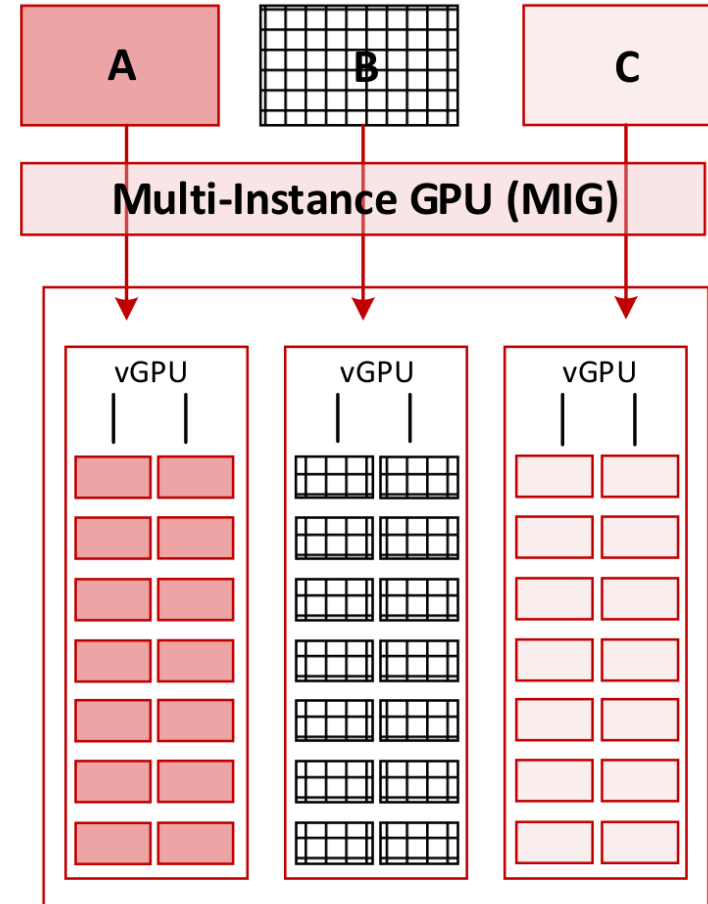
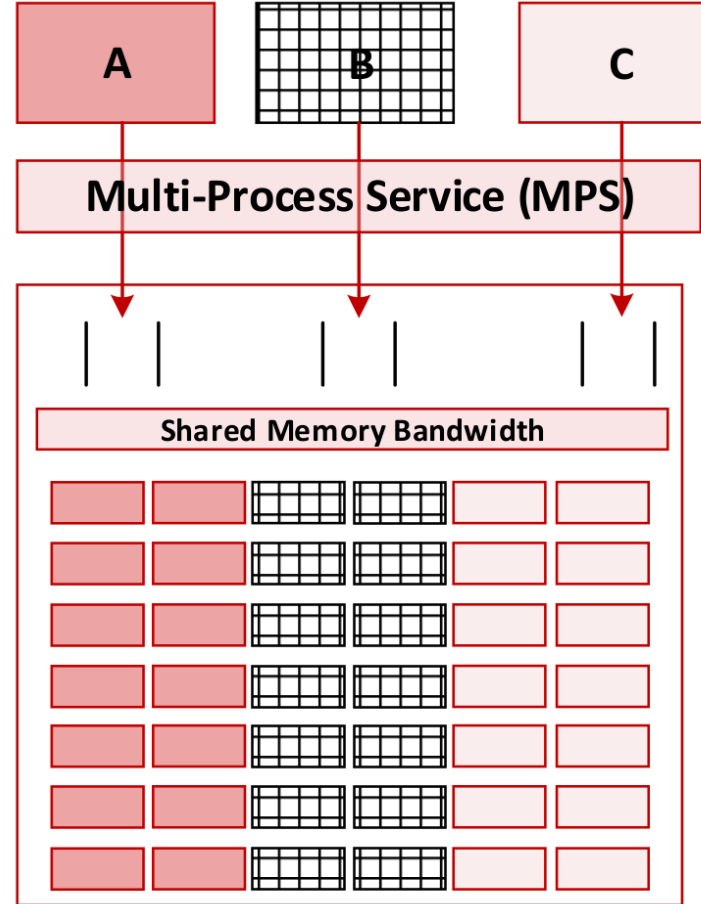
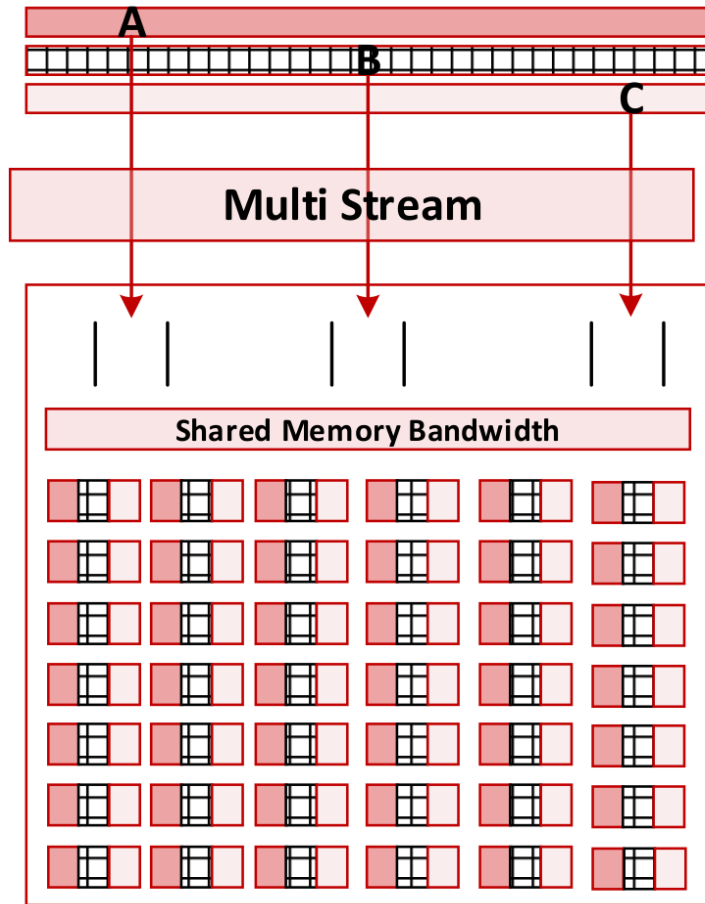
- × C experiences waiting time
- × Waste of energy and resources

Collocation aware

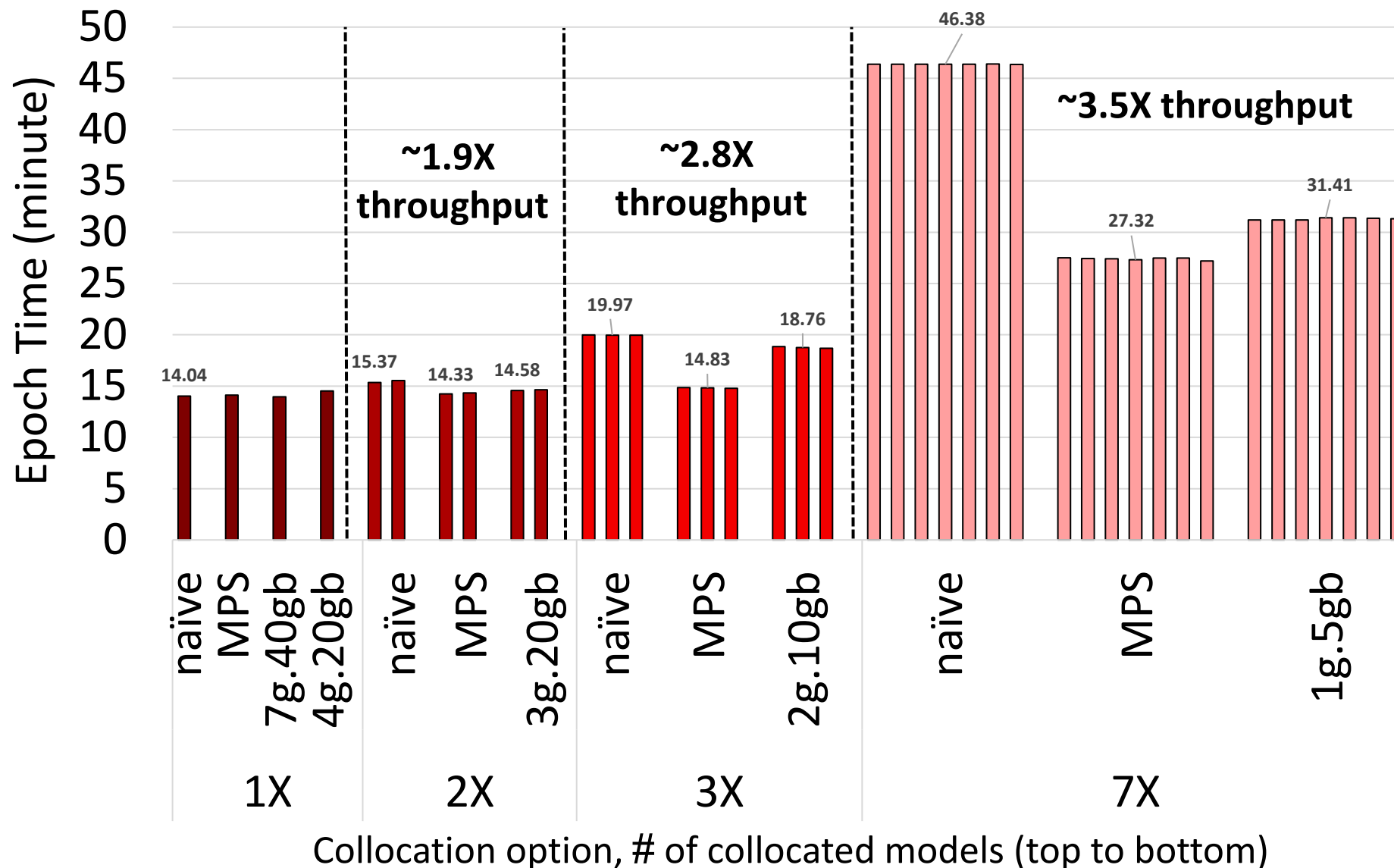


- × Slowdown due to resource interference
- ✓ Higher GPU utilization
- ✓ Higher throughput

# Collocation options on NVIDIA GPUs



# Collocation benefit



Training ResNet50 on ImageNet64 on PyTorch 2

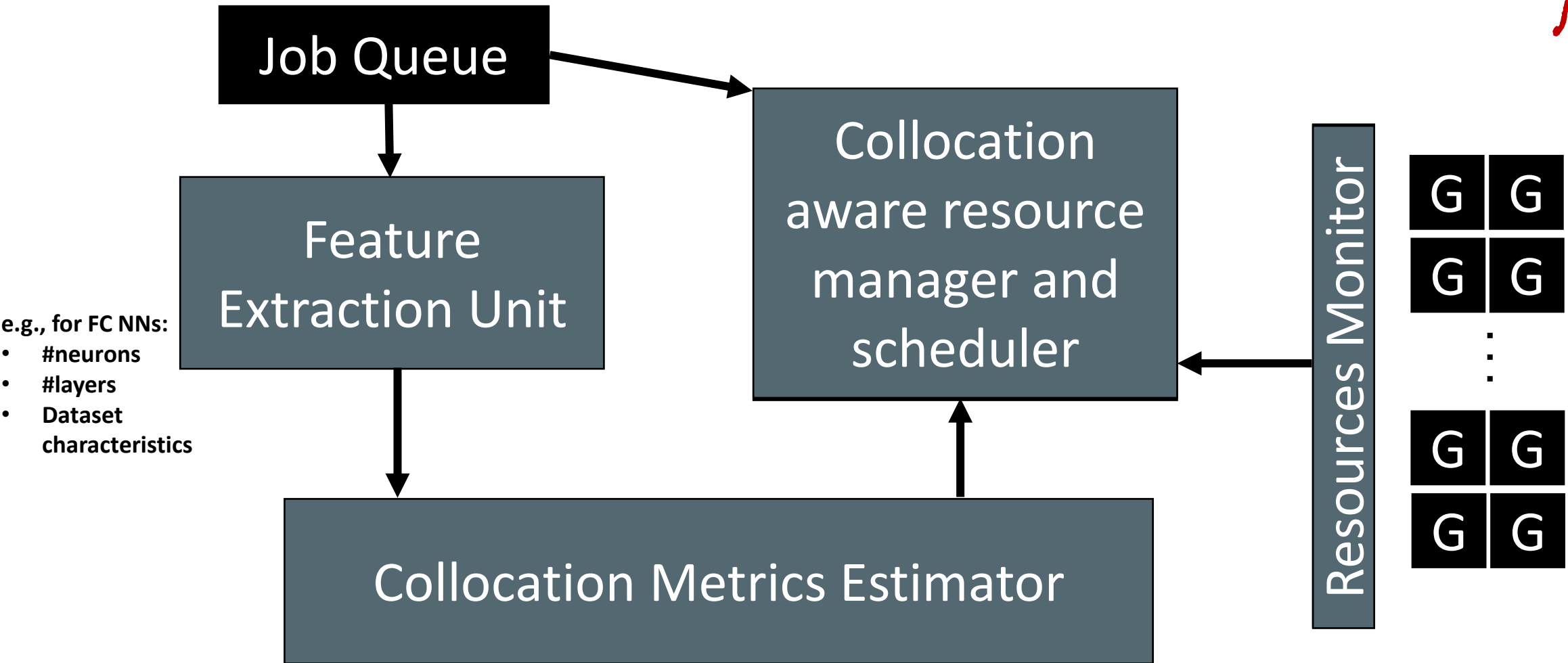
Example scenario:  
**hyper-parameter tuning**

NVIDIA DGX A100 Station

- GPU: A100 40 GB
- CPU: AMD EPYC 7742 64 cores, 512 GB
- 1g = 14 streaming multiprocessors

**Up to 3.5X throughput gain despite increased epoch time**

# Collocation-Aware Resource Manager



- e.g., for FC NNs:
- #neurons
  - #layers
  - Dataset characteristics

e.g., GPU utilization  
GPU memory usage

Thanks! 😊