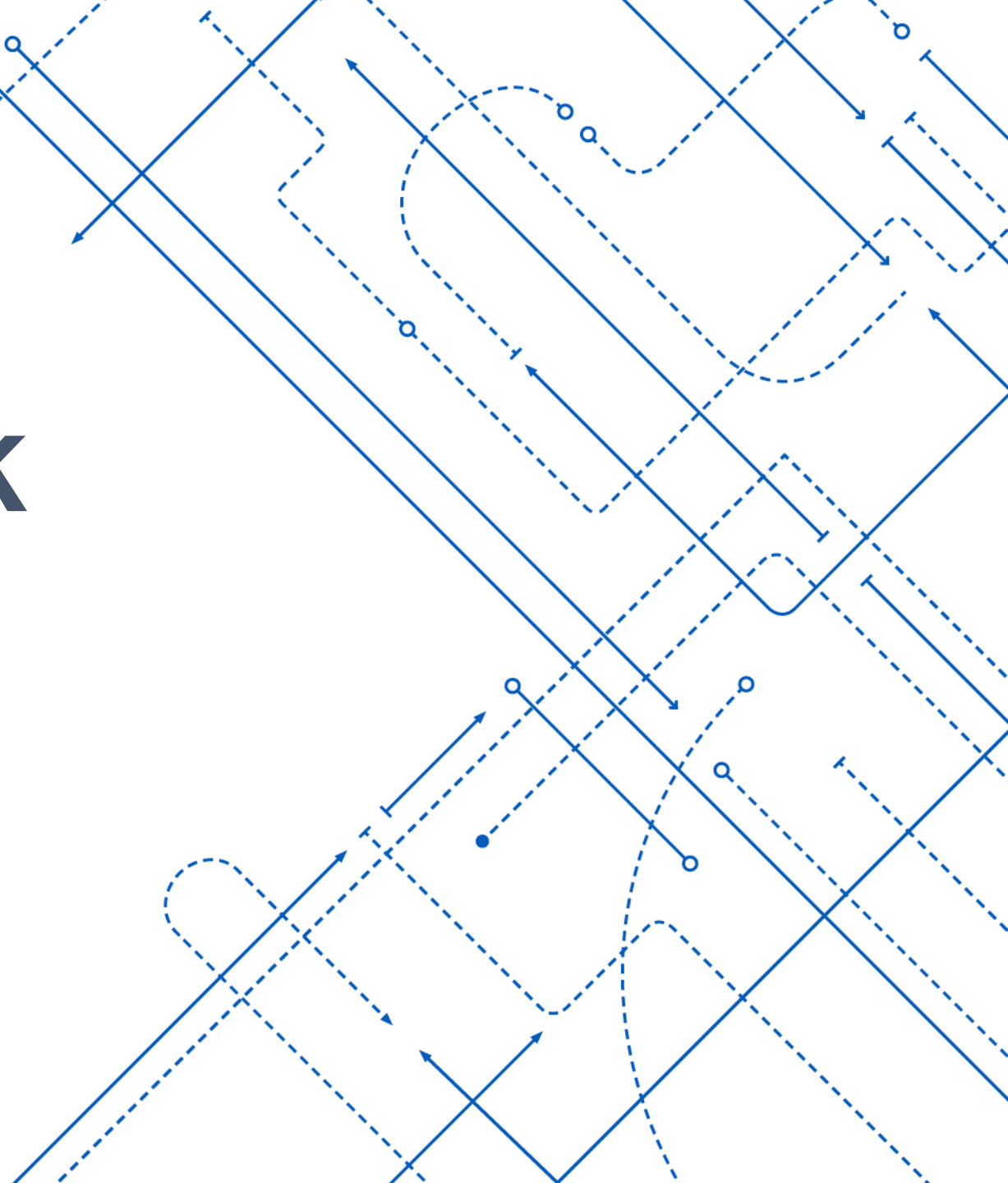
CONVOLUTION NEURAL NETWORK IN CUDA

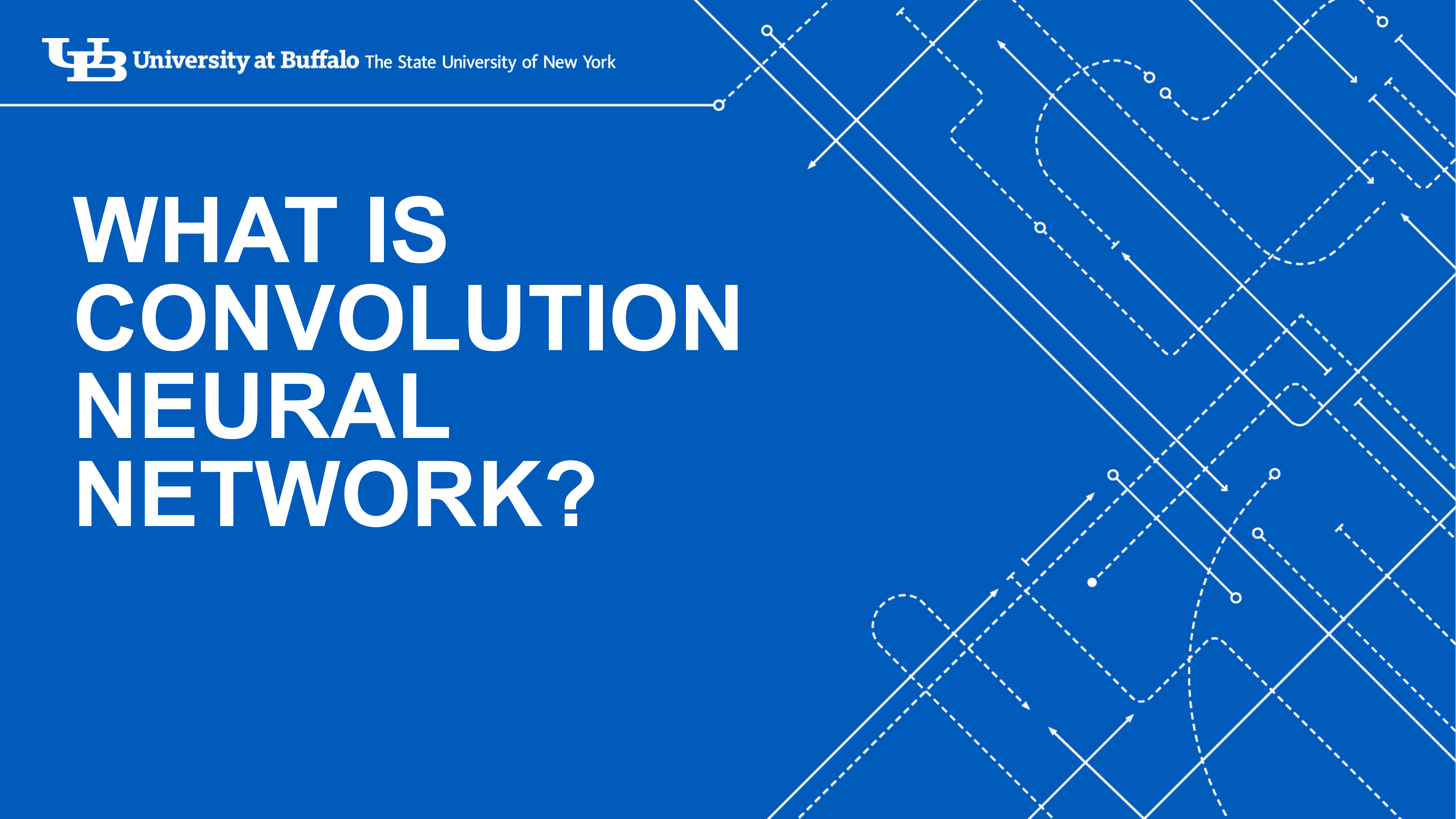
Jean Vigroux



University at Buffalo The State University of New York

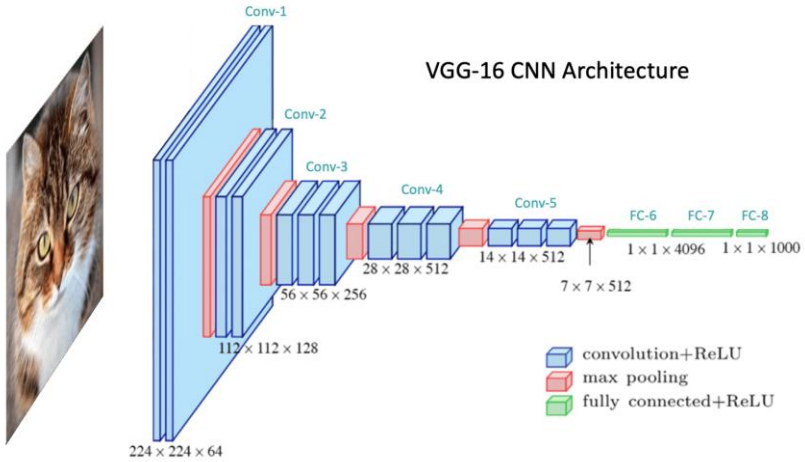
University at Buffalo The State University of New York

WHAT IS CONVOLUTION NEURAL NETWORK?



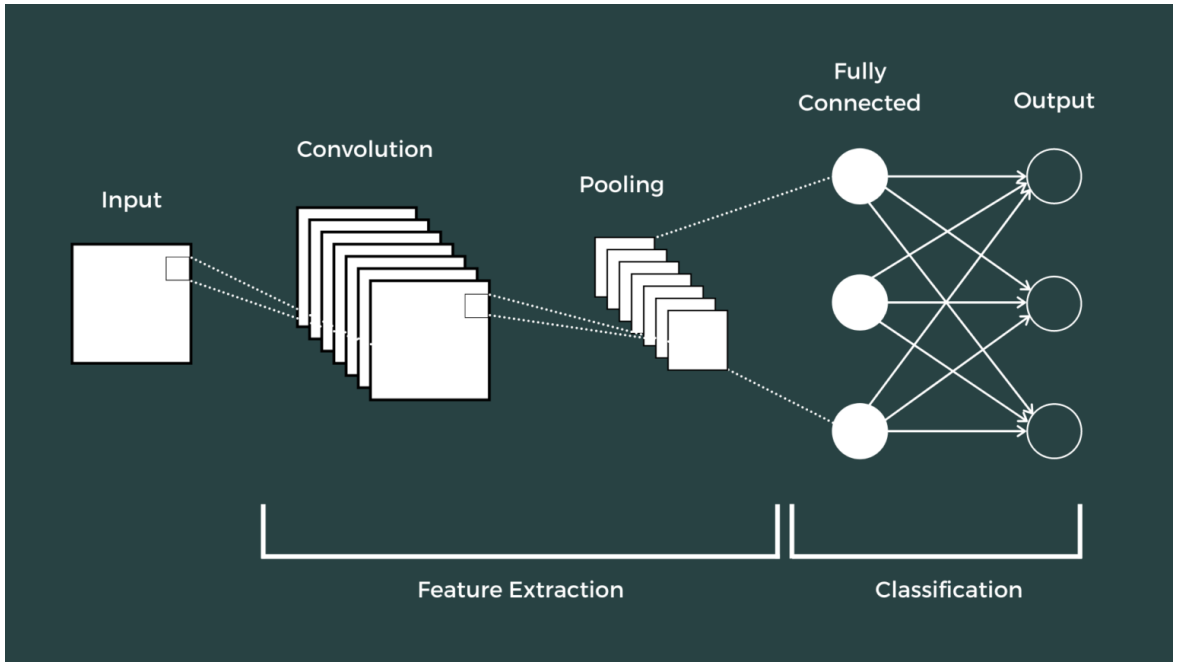
Convolution Neural Network

- Type of deep learning neural network
- Typically used for classification using images
- Computational demanding
  - Training
  - Large Labeled data



CNN Training

- **Inference / Forward Pass**
  - Model makes a prediction
- **Loss Computation**
  - How wrong you are
- **Backward Pass**
  - How much each weight contributed to the error
- **Optimizer**
  - Update weights based on backward pass
- **Repeat**



CUDA for Parallelizing

- Convolutional Neural Network are ideal for parallel implementation
  - Matrix Multiplication
  - Large Data

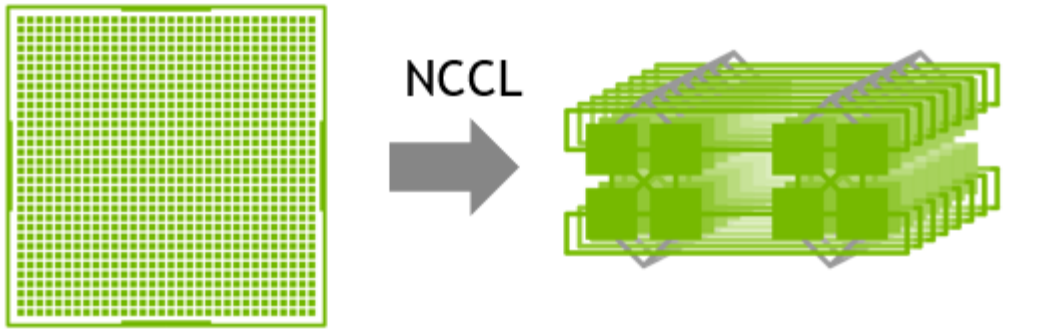
- Nvidia CUDA parallel computing platform
  - Allows utilization of Nvidia GPUs
  - GPU has the ability to run more threads
  - Shared memory within thread blocks

University at Buffalo The State University of New York

WHAT'S NEW?

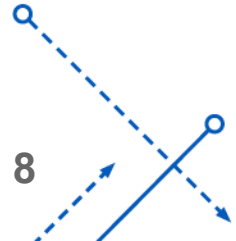
NCCL – NVIDIA Collective Communications Library

- GPU-to-GPU Communication
- Uses NVLink, but works with PCIe and InfiniBand
- Best Performance for multiple GPUs on the same node connected by NVLink



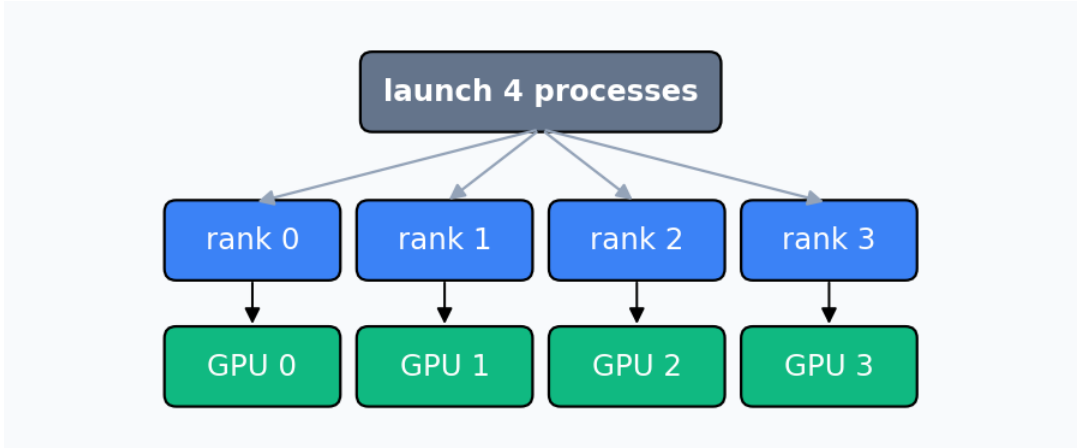
Multi-GPU Data Parallel Training

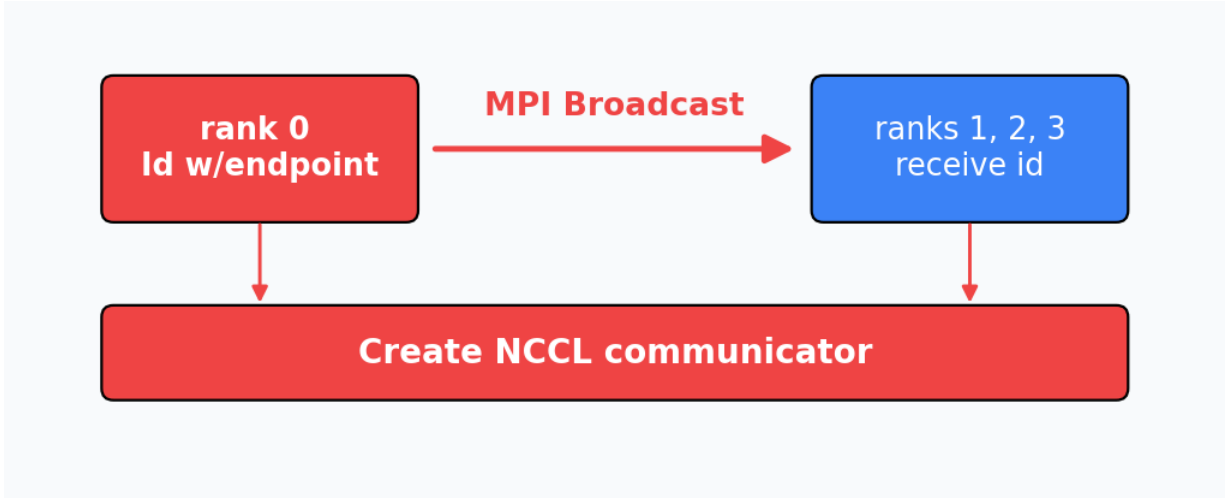
- Using MPI & NCCL, we distribute a version of the CNN model onto each GPU
- Each GPU handles their own slice of the data
- Training weights are averaged then synced each iteration



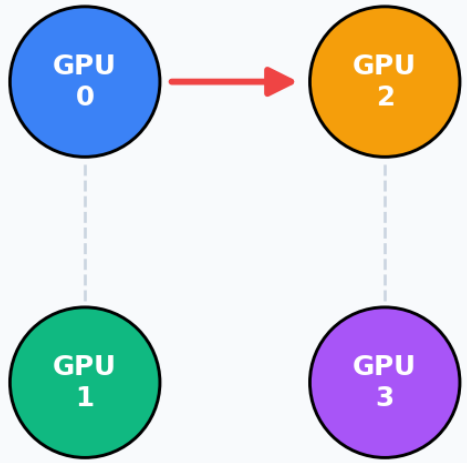
Bootstrap

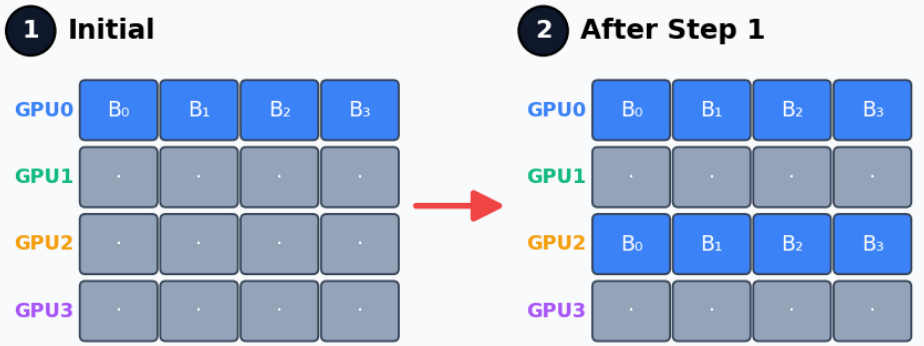
- MPI launches N processes based on the number of available GPUS
- Rank 0 creates a TCP endpoint for communication channel
- That endpoint is then broadcasted to other ranks

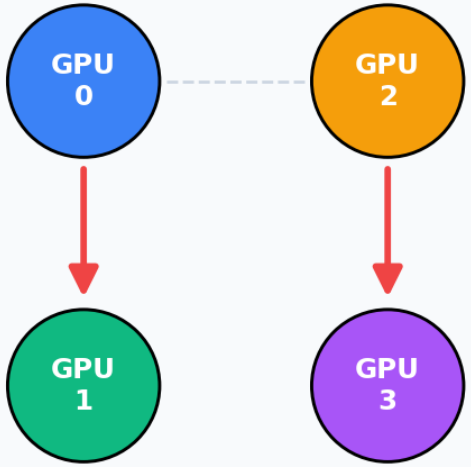


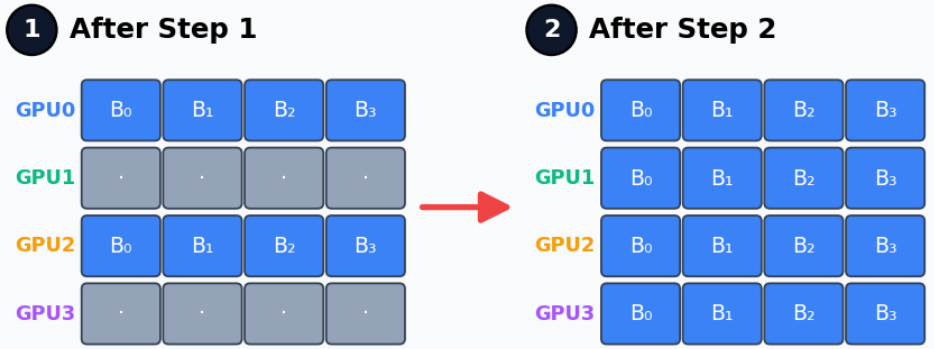


Broadcast



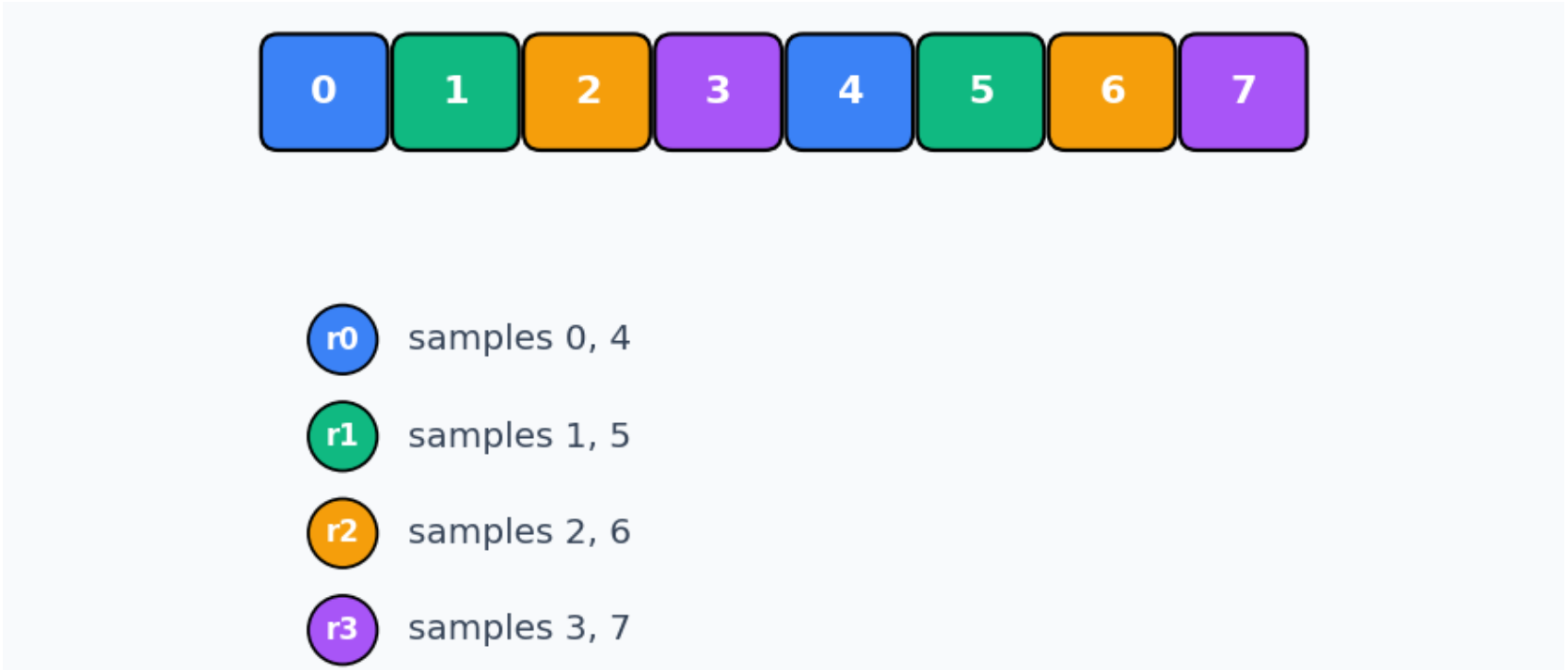






every rank now holds the buffer

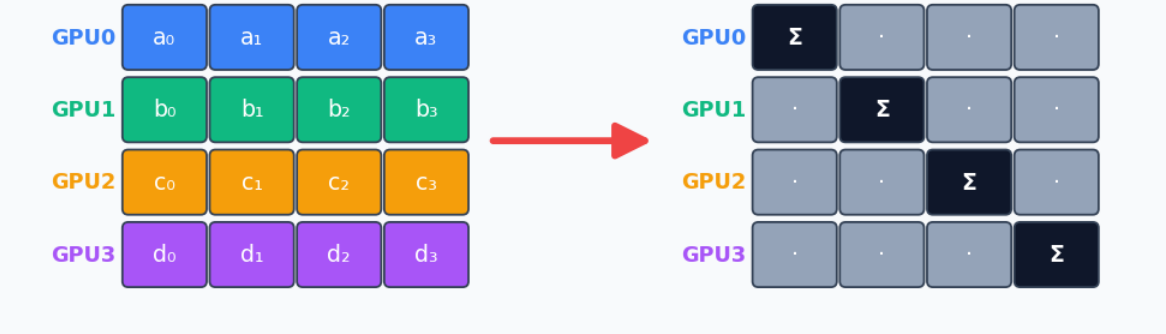
Data Distribution: Stride Sharding



Sync weights across GPUs | All Reduce

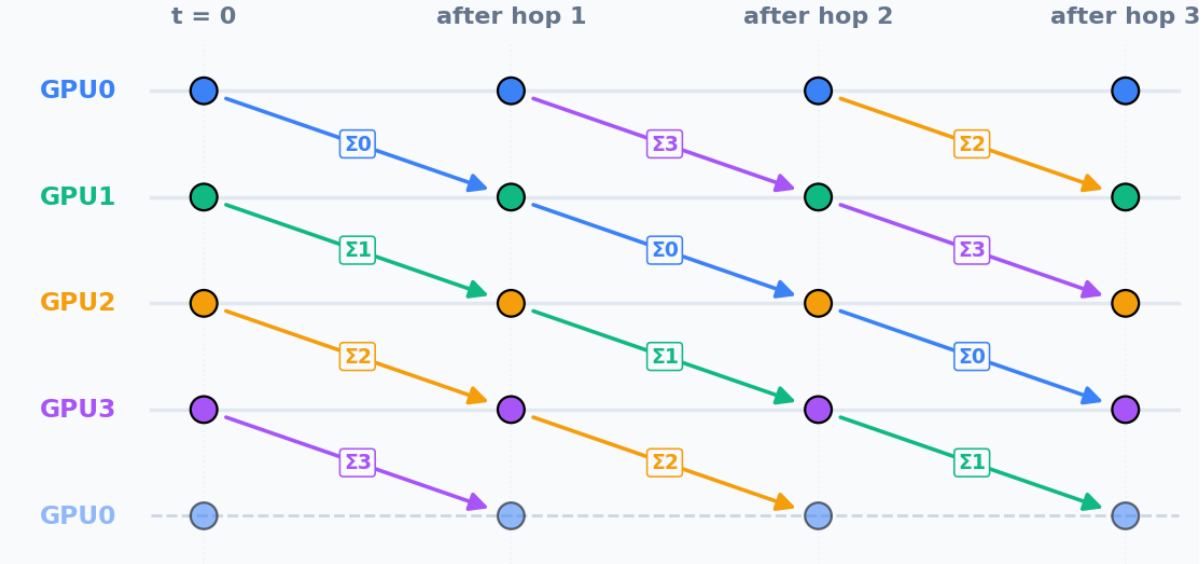
- Each GPU sums up their gradients

  Reduce Scatter



- Then each GPU shares their gradient sum

  All Gather



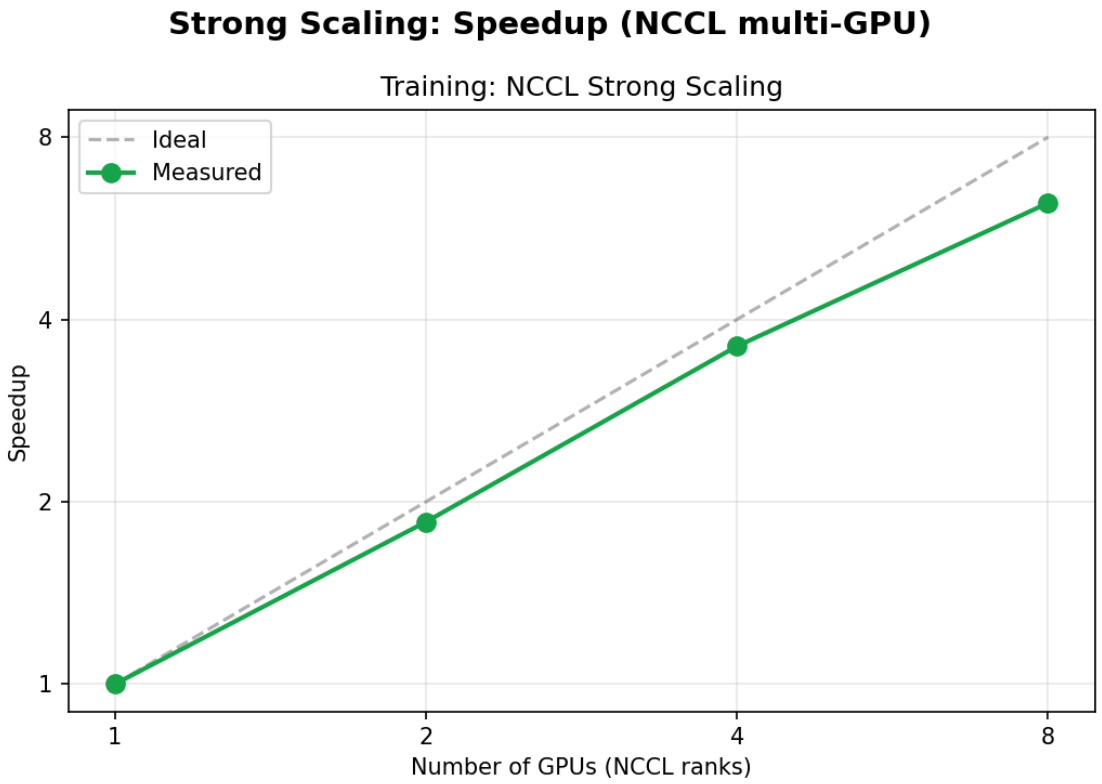
University at Buffalo The State University of New York

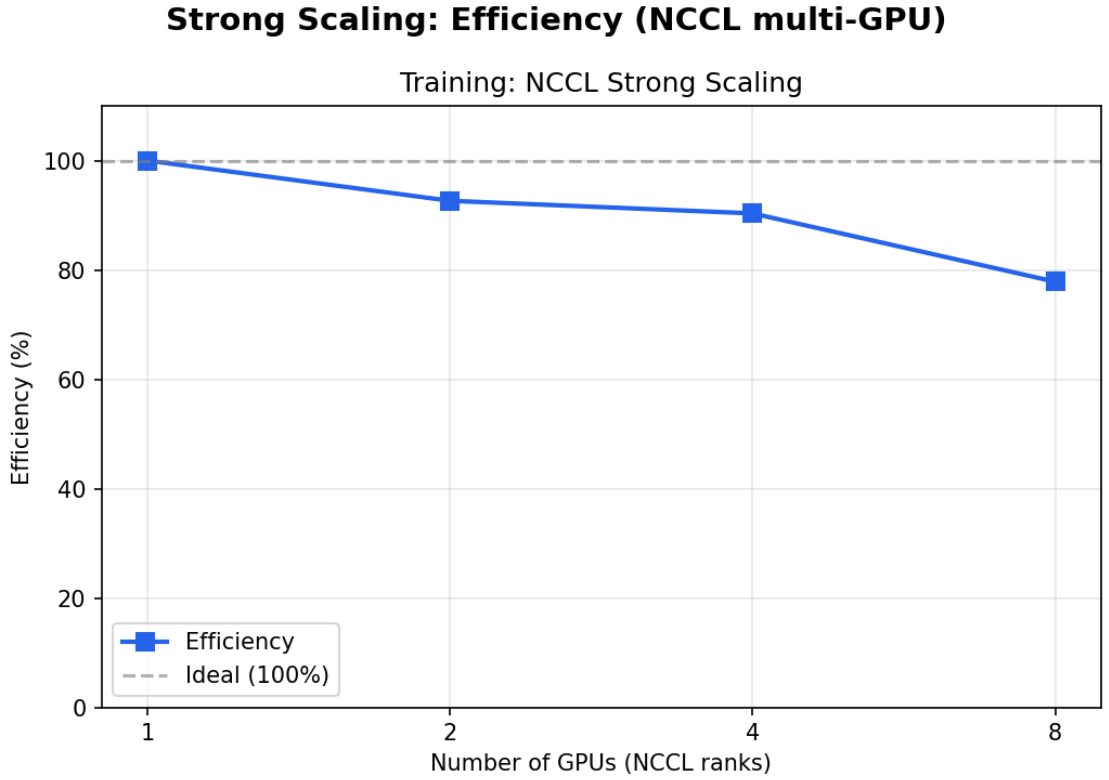
RESULTS

Testing Environment | NCCL

- 8× NVIDIA A100-SXM4-40GB
- Single Node
- NVLink-equipped
- Ran on 1 million Images 48 x 48

Strong Scaling NCCL

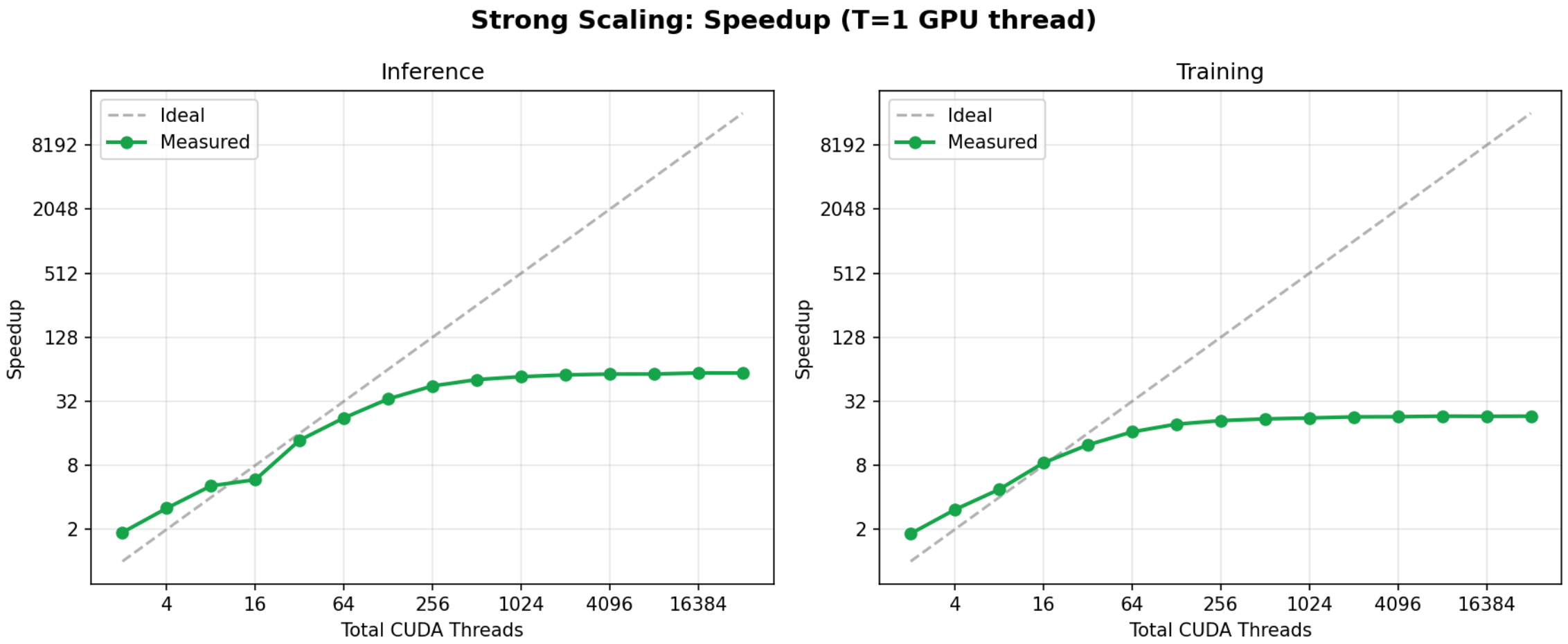




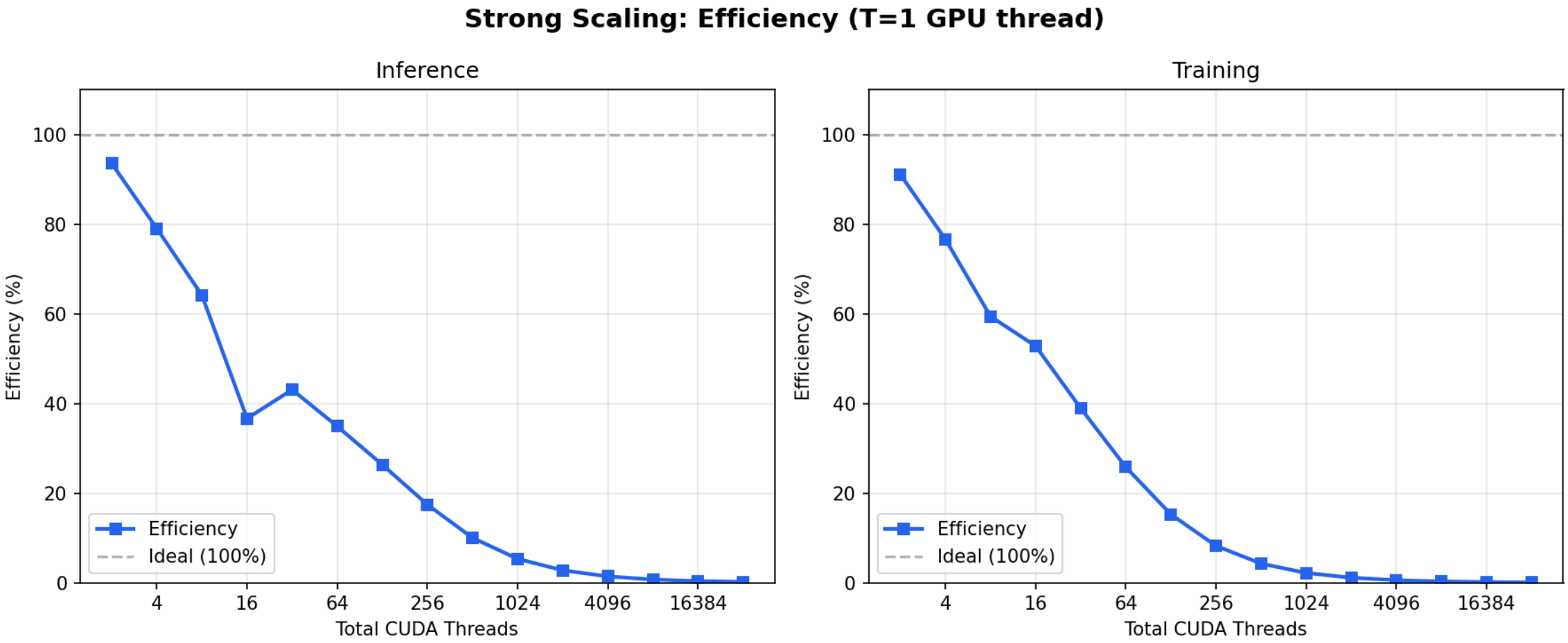
Strong Scaling NCCL

GPUs	Time (s)	Speedup	Efficiency (%)
1	698.55	1.00	100.00
2	377.04	1.85	92.64
4	193.34	3.61	90.33
8	112.19	6.23	77.83

Speedup – Strong Scaling



Efficiency – Strong Scaling



Forward Pass/Inference – Strong Scaling

Threads	Time (ms)	Speedup	Efficiency (%)
1	1893.11	1.00	100.00
2	1012.52	1.87	93.48
4	599.13	3.16	78.99
8	369.20	5.13	64.10
16	322.98	5.86	36.63
32	137.44	13.77	43.04
64	84.82	22.32	34.87
128	56.22	33.67	26.31
256	42.47	44.58	17.41
512	36.99	51.18	10.00
1024	34.72	54.52	5.32
2048	33.43	56.63	2.76
4096	32.82	57.68	1.41
8192	32.78	57.75	0.71
16384	32.03	59.11	0.36
32768	32.07	59.03	0.18

Training – Strong Scaling

Threads	Time (s)	Val Acc	Loss	Speedup	Efficiency (%)
1	4079.19	65.33	1.3129	1.00	100.00
2	2239.79	65.33	1.3129	1.82	91.06
4	1331.66	65.33	1.3129	3.06	76.58
8	858.53	65.33	1.3129	4.75	59.39
16	483.03	65.33	1.3129	8.45	52.78
32	327.63	65.33	1.3129	12.45	38.91
64	247.26	65.33	1.3129	16.50	25.78
128	209.27	65.33	1.3129	19.49	15.23
256	193.77	65.33	1.3129	21.05	8.22
512	186.81	65.33	1.3129	21.84	4.26
1024	182.92	65.33	1.3129	22.30	2.18
2048	178.64	65.33	1.3129	22.83	1.11
4096	178.24	65.33	1.3129	22.89	0.56
8192	176.02	65.33	1.3129	23.17	0.28
16384	176.49	65.33	1.3129	23.11	0.14
32768	176.09	65.33	1.3129	23.17	0.07

Baseline – Strong Scaling

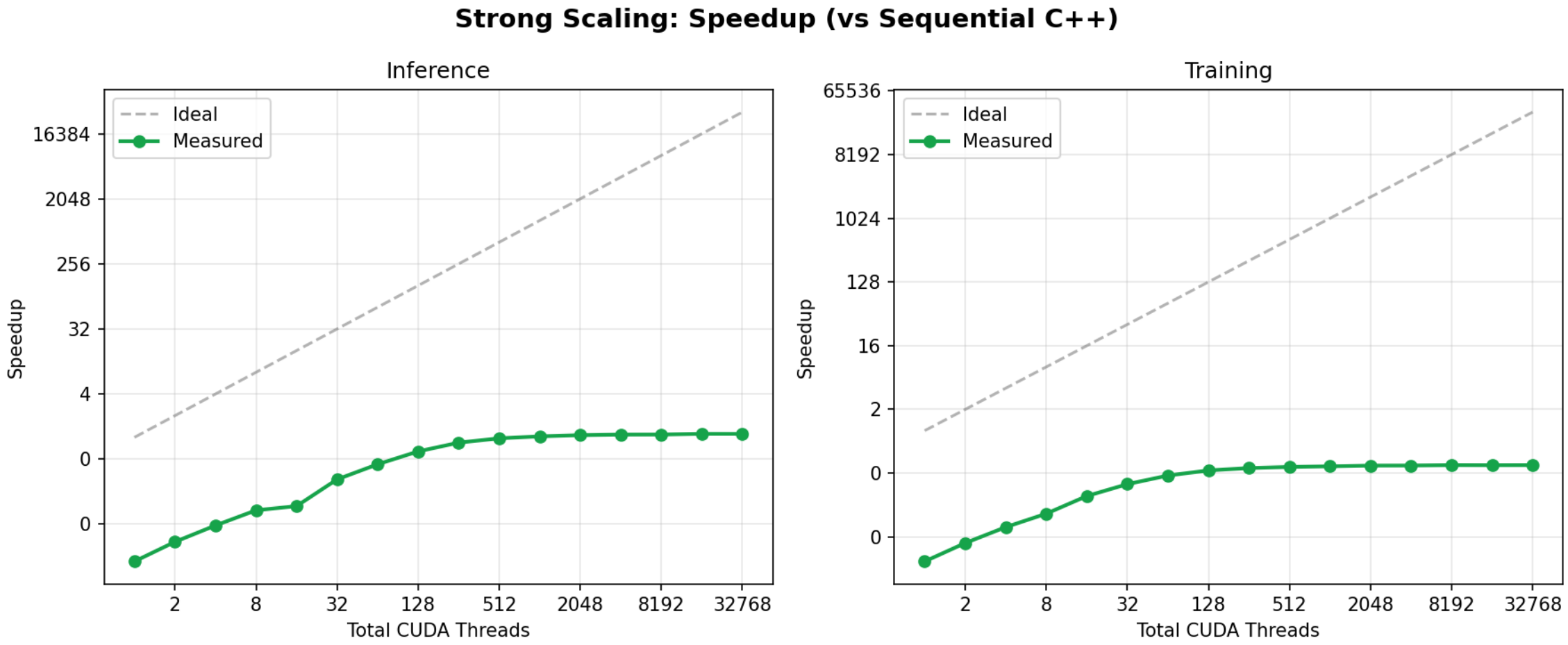
Inference Reference Baselines

Implementation	Time (ms)
Sequential (C++)	35.85
PyTorch	1.04
TensorFlow	1.36

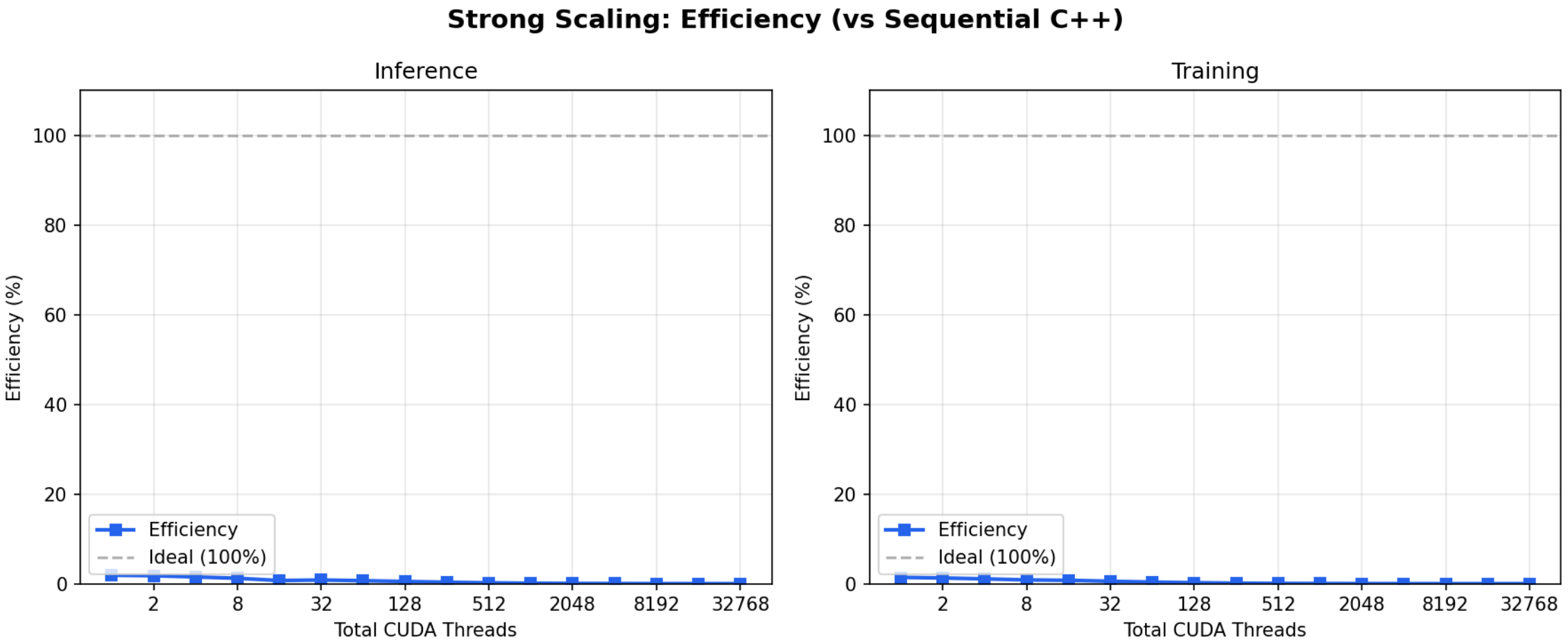
Training Reference Baselines

Implementation	Time (s)	Val Acc	Loss
Sequential (C++)	57.16	65.33	1.3129
PyTorch	2.40	63.06	1.3265
TensorFlow	9.37	57.52	1.4580

Speedup – Strong Scaling



Efficiency – Strong Scaling



University at Buffalo The State University of New York

THANK YOU!!!

Any questions?