Memory and Bandwidth are All You Need for FSDP

Jiangtao Wang [1], Jan Ebert[1], Oleg Filatov[1], Stefan Kesselheim[1]
[1] Jülich Supercomputing Centre, Forschungszentrum Jülich, Germany

JÜLICH Forschungszentrum

TrustLLM

Introduction

Scaling FSDP and Zero for training large language models demands faster intra-node connections and increased GPU memory. We conducted empirical experiments with various model sizes, GPU setups, and connection speeds to determine the most effective configurations for efficient model training. Our findings identify the specific settings that maximize FSDP efficiency, leading to improved training performance.

Takeaway Messages

- **MAX. Token Capacity**

$$\leq \frac{M_{free}}{2*L*H}$$

- **Max. MFU**

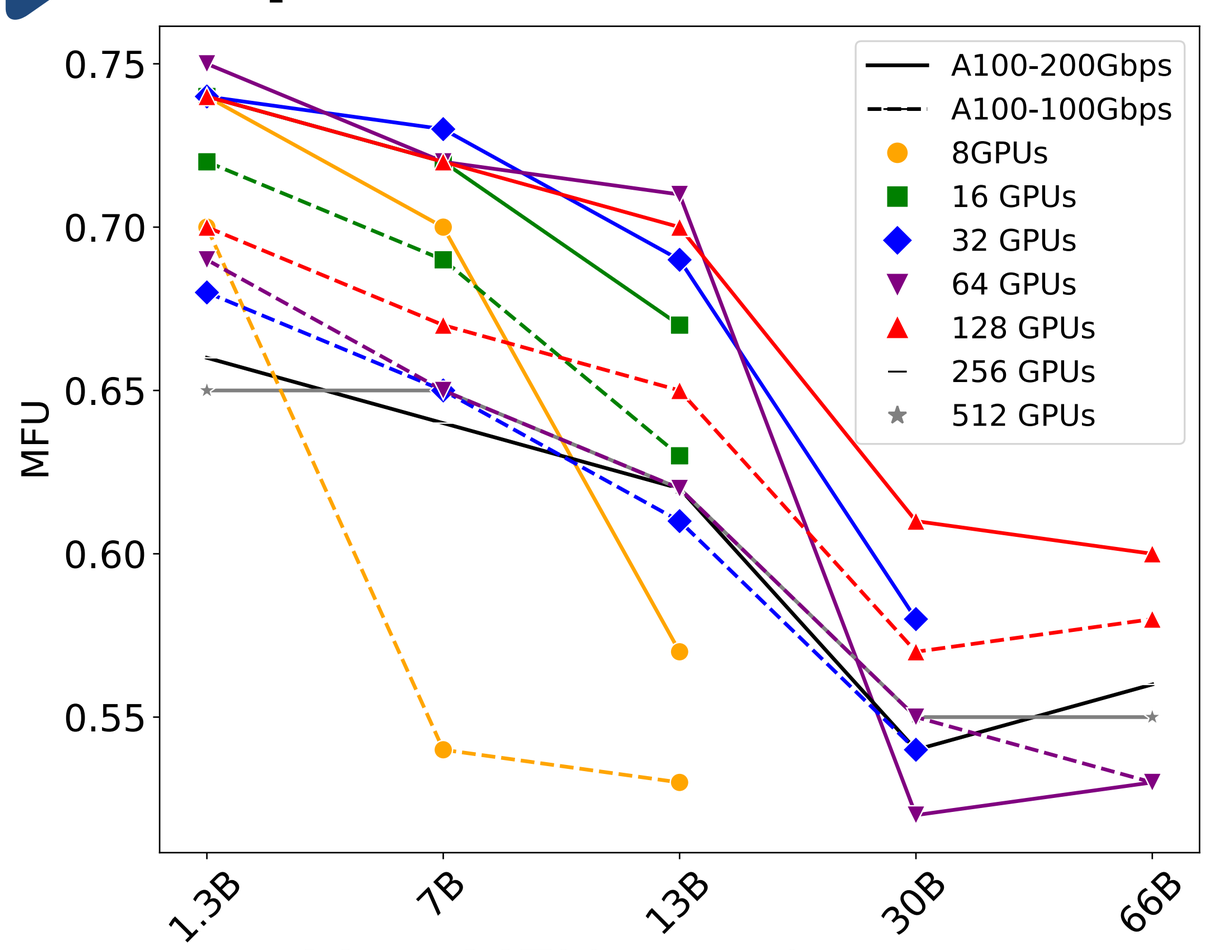
$$\leq \left(2+\frac{L_{seq}}{3H}\right)\frac{3}{4LHQ^2}\frac{S_{volume}M_{free}}{S_{flops}}$$

- **Max. Throughput**

$$\leq \frac{1}{24}\frac{1}{Q^2L^2H^3}M_{free}S_{volume}$$

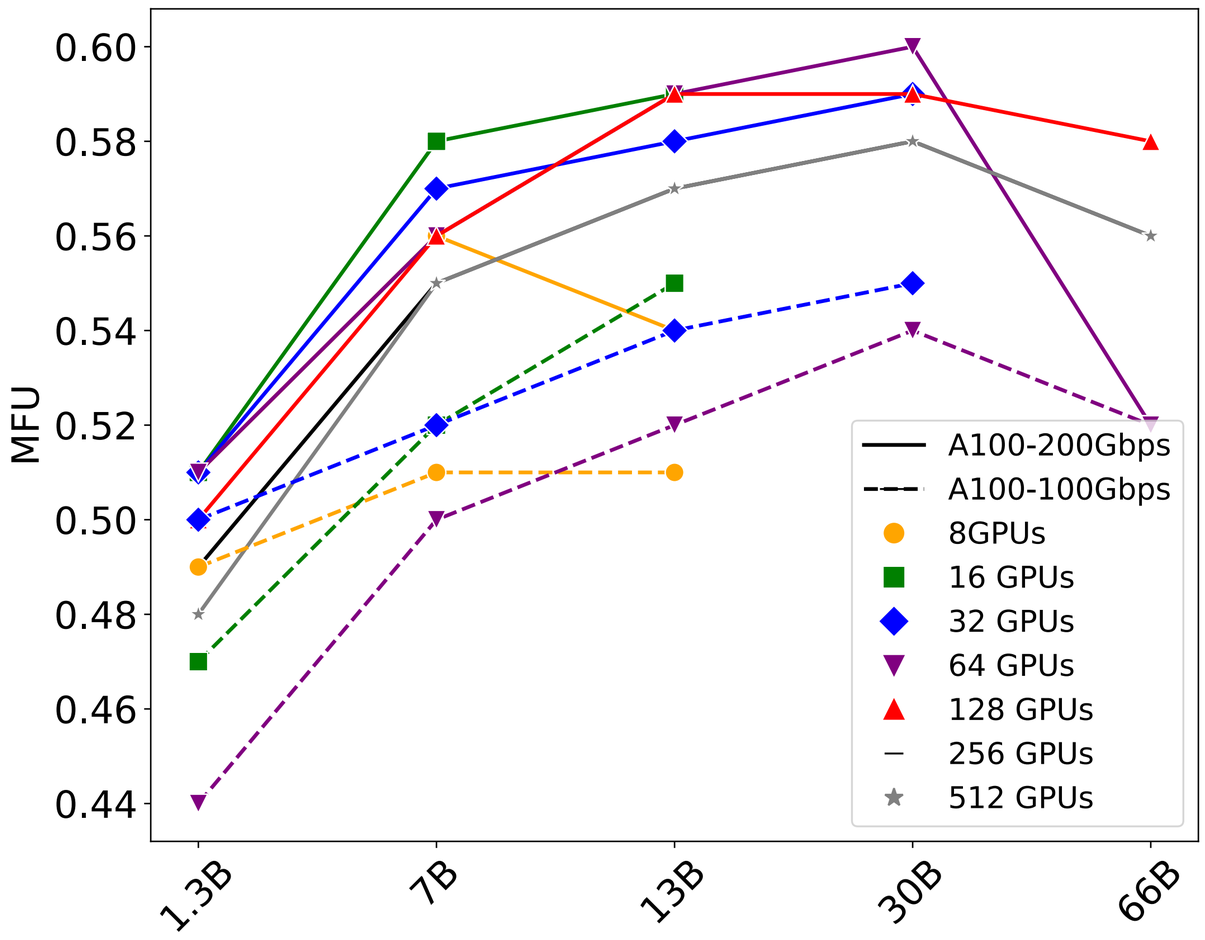
L: Transformer's Layers
H: Hidden Dimension
L_{seq}: Sequence Length
Q: Bytes per parameter, 2 for BF16
M_{free}: Free GPU Memory (byte)
S_{volume}: Node2node connection speed (byte per second)
S_{flops}: GPU Max. Flops

Experiment Results

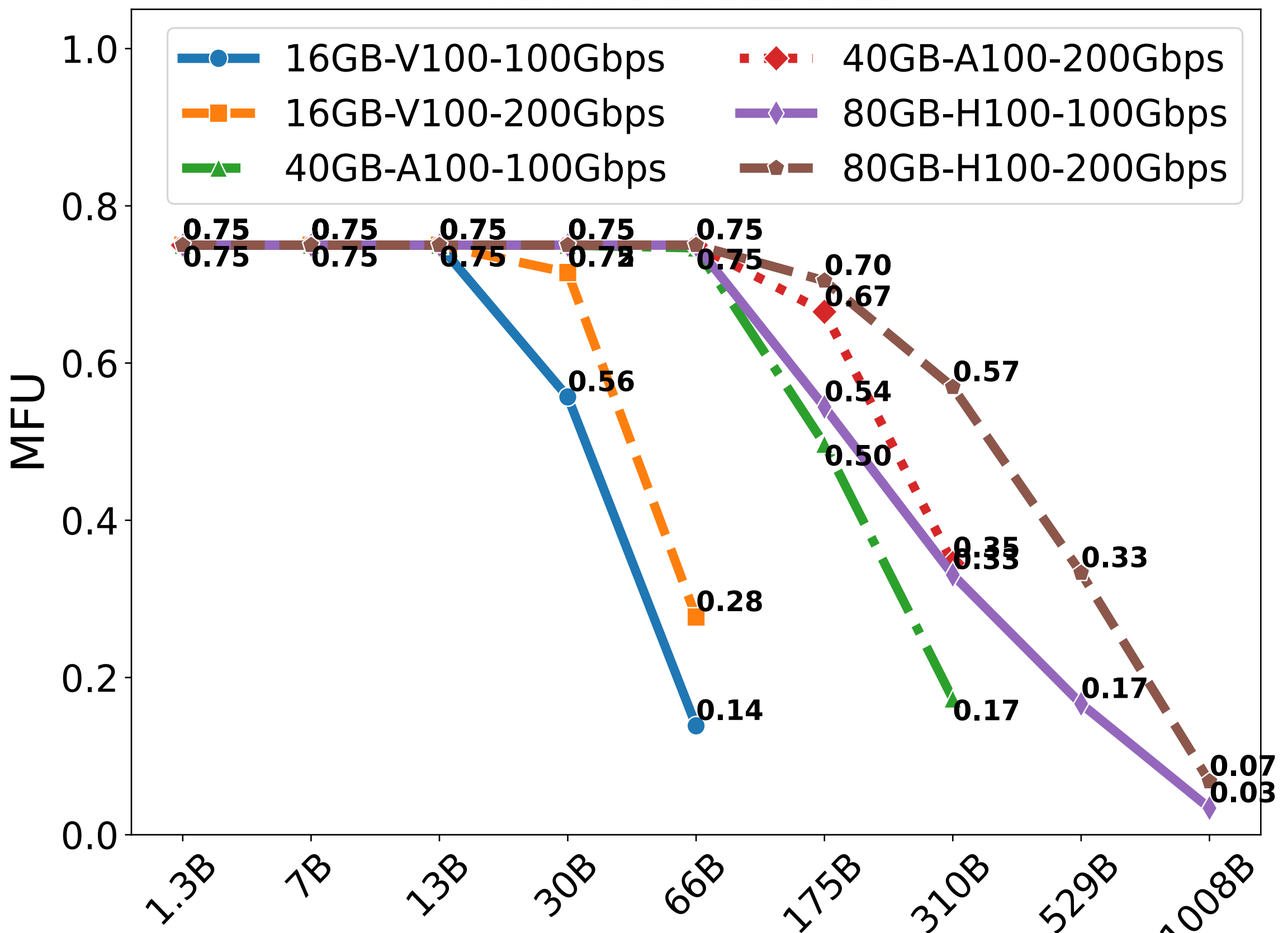


Max. MFU tested on Local Batch Size as 1

Simulated Results



Max. MFU tested on Sequence Length 2048



Max. MFU on 512 simulated GPUs against models