1-bit Adam: Communication Efficient Large-Scale Training with Adam's Convergence Speed

Authors: Hanlin Tang (presenter), Shaoduo Gan, Samyam Rajbhandari, Ammar Ahmad Awan, Conglong Li, Xiangru Lian, Ji Liu, Ce Zhang, Yuxiong He

Adam

Momentum SGD

Fast Convergence

Communication Compression

1-bit Adam

BERT-Large Pretrain (64 GPUs, 40 Gbps Ethernet):

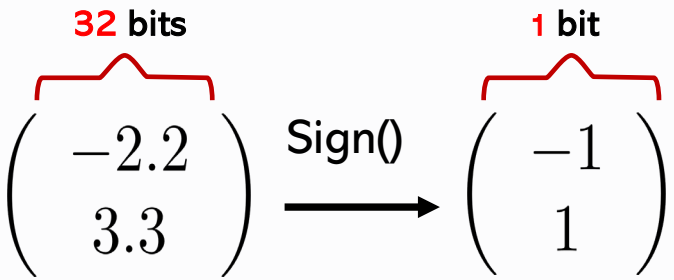
174 hours → (1-bit Adam; Efficient kernels) → 51 hours

3.4x Faster

Algorithm Design

Gradient Compression

Compression:



32x of communication reduction

Slower convergence

Slower training speed

Error Compensation:

$$g_1 = 3.3, \quad g_2 = -1.1, \quad \gamma = 1$$

Step 1: $x = x + Sign(3.3); \quad error = 2.3$

Step 2: $x = x + Sign(-1.1 + 2.3); \quad error = 0.2$

32x of communication reduction

Same convergence

Faster training speed

Adam?

Adam:

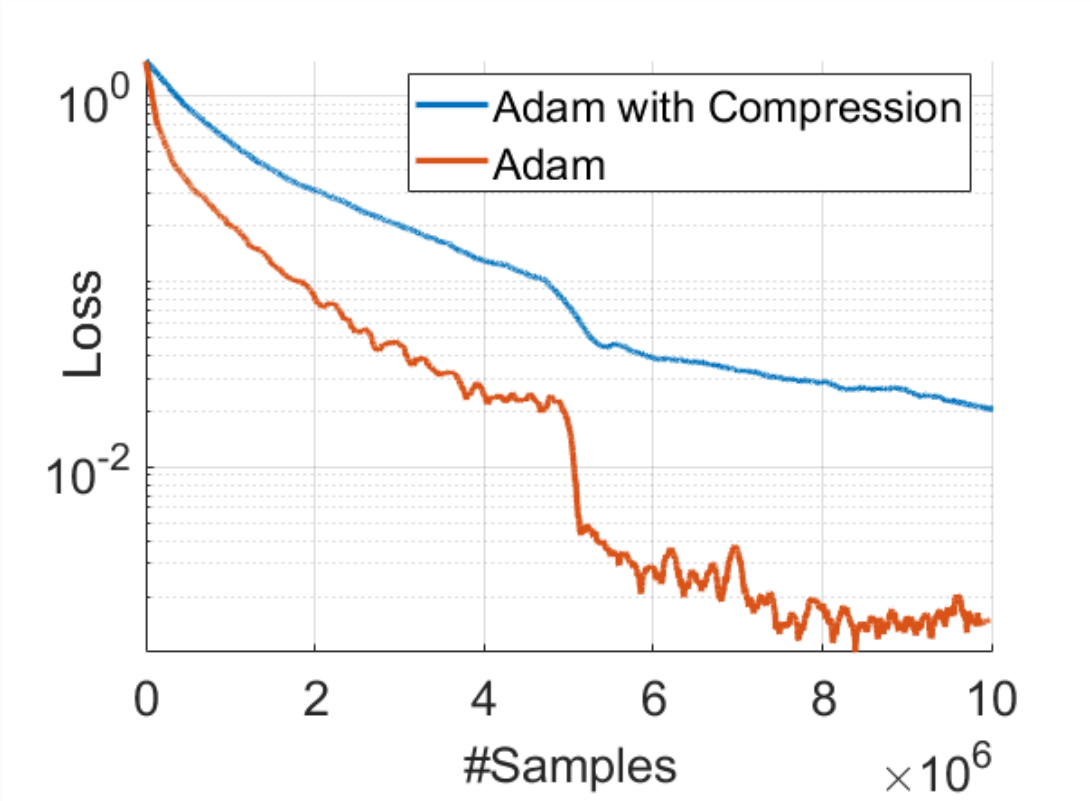
$$\boldsymbol{m}_t = \beta_1 \boldsymbol{m}_{t-1} + (1-\beta_1)\boldsymbol{g}_t$$
$$= \sum_{s=0}^{t} \beta_1^{t-s} \boldsymbol{g}_s$$

$$\boldsymbol{v}_t = \beta_2 \boldsymbol{v}_{t-1} + (1-\beta_2)\boldsymbol{g}_t^2$$
$$= \sum_{s=0}^{t} \beta_2^{t-s} \boldsymbol{g}_s^2$$

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \gamma \frac{\boldsymbol{m}_t}{\sqrt{\boldsymbol{v}_t}}$$

Nonlinear term

Incompatible with error compensation



Adam:

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \gamma \frac{\boldsymbol{m}_t}{\sqrt{\boldsymbol{v}_t}}$$

$$\boldsymbol{m}_t = \beta_1 \boldsymbol{m}_{t-1} + (1-\beta_1)\boldsymbol{g}_t$$

$$\boldsymbol{v}_t = \beta_2 \boldsymbol{v}_{t-1} + (1-\beta_2)\boldsymbol{g}_t^2$$

Constant v

Quasi Adam:

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \gamma \frac{\boldsymbol{m}_t}{\sqrt{\boldsymbol{v}_t}}$$

$$\boldsymbol{m}_t = \beta_1 \boldsymbol{m}_{t-1} + (1-\beta_1)\boldsymbol{g}_t$$

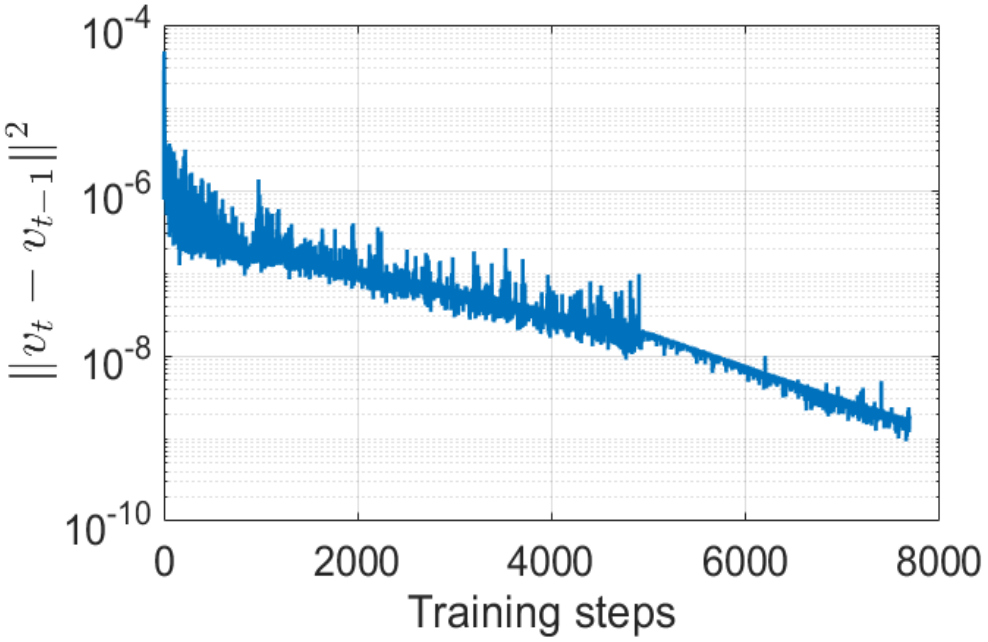
Define: $\gamma_v = \frac{\gamma}{\sqrt{\boldsymbol{v}}}$

Momentum SGD:

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \gamma_v \boldsymbol{m}_t;$$

$$\boldsymbol{m}_t = \beta_1 \boldsymbol{m}_{t-1} + (1-\beta_1)\boldsymbol{g}_t;$$

v_t becomes more stable along with training.



1-bit Adam

Warmup phase:

- Original Adam (No compression)

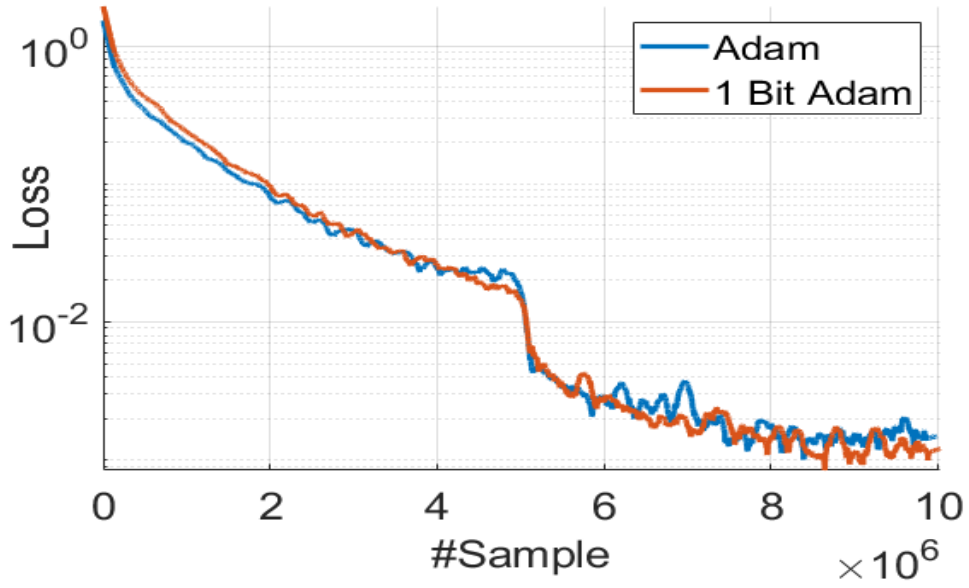
Compression phase:

- Keep v_t unchanged
- Compress m_t only

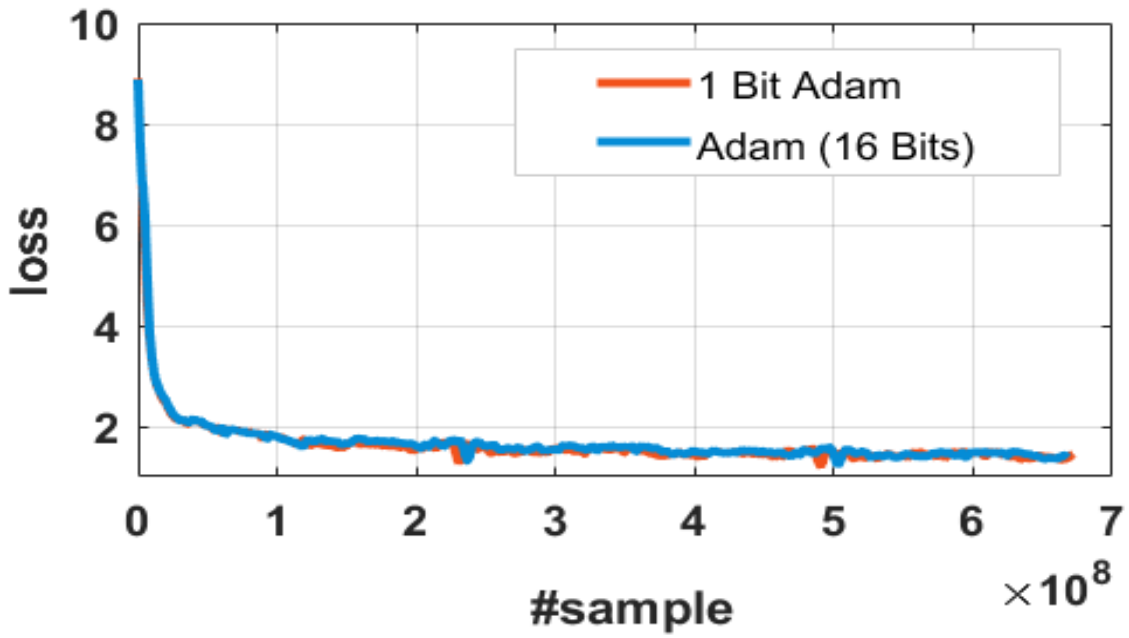
Convergence Results

32x speedup after warmup, **5x** speedup in total (10%~15% warmup needed)

ResNet-18



BERT-large



System Implementation

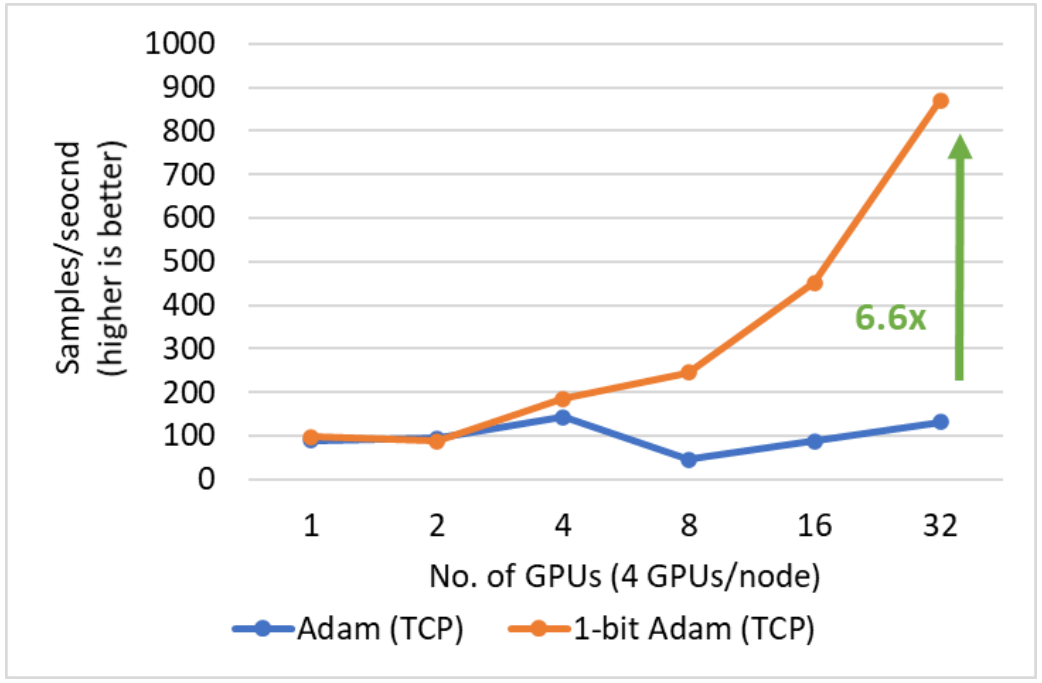
System Implementation

- An efficient implementation of CUDA-Aware Collective Communication backend that works for both InifiniBand and Ethernet network.
- Efficient compression kernel for encoding gradients (32-bits) into 1-bit.
- Compatible with half-precision training (16-bits).
- Released in Micorsoft DeepSpeed.

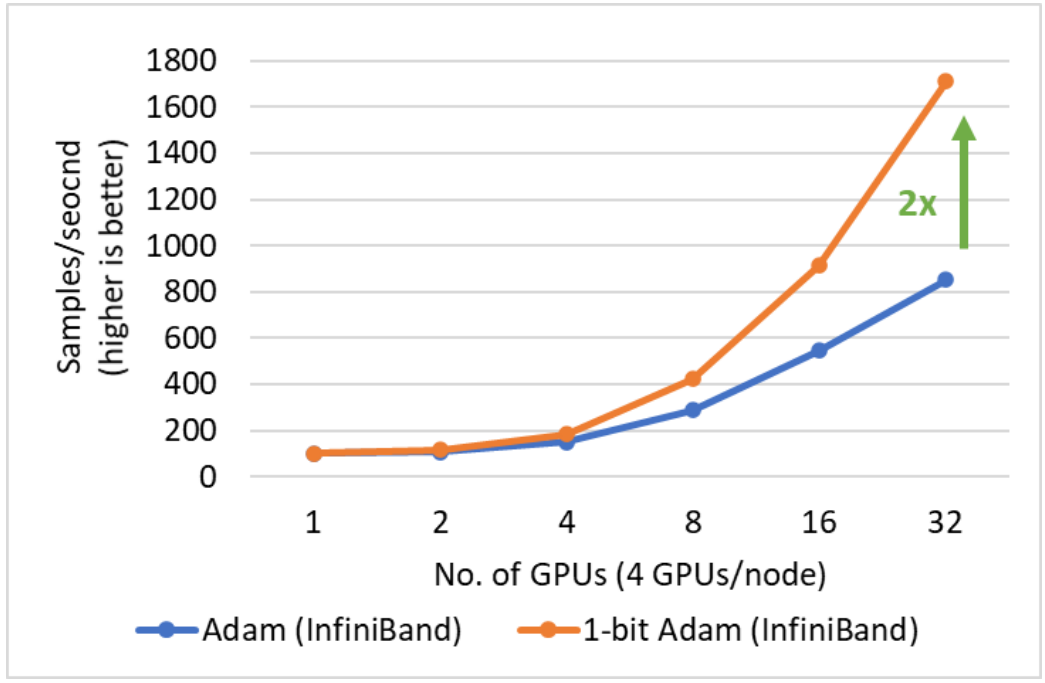
System Implementation

Results: BERT-Large pretraining. End-to-end training speed comparison (including both computation and communication)

4 V100/node, 40 Gbps Ethernet



8 V100/node, 100 Gbps InfiniBand



Thank You