Communication-Efficient Federated Learning with Adaptive Parameter Freezing

ICDCS 2021

DML GROUP MEETING
4.22

OUTLINE

- Introduction
- Design
- Evaluation

Introduction

- 联邦学习中边缘和服务器之间的参数同步成为固有的瓶颈，严重延长训练过程
- 目前已有的工作通常围绕两个方向进行：

  1）quantize the model updates into fewer bits

  2）sparsify local updates by filtering out certain values

- 问题：以上工作假设所有模型参数应该在每一轮通信中不加选择地同步。这是有必要的吗?
- 实验表明：许多参数在模型收敛之前很久就变得稳定; 在这些参数达到其最佳值之后，它们的后续更新只是振荡而没有实质性变化，并且确实可以安全地排除而不会损害模型精度。
- 直观想法：是否可以通过不再同步那些稳定的参数来减少FL通信量?

Theoretical Analysis

Assumption 1. (Strong Convexity.) *The global loss function* $F(\omega)$ *is* $\mu-$*strongly convex, i.e.,*

$$F(y) \geq F(x) + \nabla F(x)^T (y - x) + \frac{\mu}{2}||y - x||^2.$$

An equivalent form of $\mu-$*strong convexity is*

$$(\nabla F(x) - \nabla F(y))^T (x - y) \geq \mu||x - y||^2.$$

Assumption 2. (Bounded Gradient.) *The stochastic gradient calculated from a mini-batch* ξ *is bounded as* $\mathbb{E}||g_\xi(\omega)||^2 \leq \sigma^2$.

Theorem 1. *Suppose* $F(\omega)$ *is* μ*-strongly convex,* σ^2 *is the upper bound of the variance of* $||\nabla F(\omega)||^2$*, then there exist two constants* $A = 1 - 2\mu\eta$ *and* $B = \frac{\eta\sigma^2}{2\mu}$*, such that*

$$\mathbb{E}(||\omega_k - \omega^\star||^2) \leq A^k||\omega_0 - \omega^\star||^2 + B.$$

$$\mathbb{E}(||\omega_k - \omega^\star||^2) \leq (1-2\mu\eta)^k||\omega_0 - \omega^\star||^2 + \sum_{j=0}^{k-1}(1 - 2\mu\eta)^j\eta^2\sigma^2$$

$$\leq \boxed{(1 - 2\mu\eta)}^k||\omega_0 - \omega^\star||^2 + \frac{\eta\sigma^2}{2\mu}.$$

< 1

- 结论：
- 1）在瞬态阶段，ω 在迭代次数上呈指数快速逼近 ω*，而在静止阶段，ω 在 ω* 附近振荡。
- 2）模型参数在开始时发生显着变化，然后逐渐稳定下来

Testbed Measurements on Parameter Variation

- 实验表明：两个参数一开始变化很大，伴随着测试精度的快速上升； 随着精度达到稳定，参数趋于稳定
- 问题：即使参数在稳定之后，仍然会定期更新，这会消耗通信带宽，增加通信成本

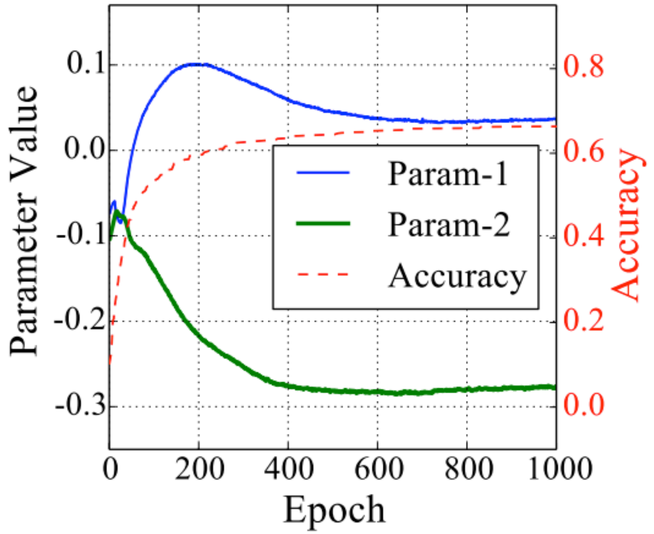


Fig. 1: Evolution of two randomly selected parameters during LeNet-5 training (with the test accuracy for reference.)

Statistical Analysis on Parameter Stability

- 有效扰动的度量：定量描述参数的稳定性（公式2）
- 参数越稳定，其有效扰动就越小。如果所有模型更新的方向相同，P_k^r将为1；如果任何两个连续的模型更新都能很好地相互抵消，那么P_k^r将为0。

$$P_k^r = \frac{||\sum_{i=0}^{r-1} \mathbf{u}_{k-i}||}{\sum_{i=0}^{r-1} ||\mathbf{u}_{k-i}||}. \quad (2)$$

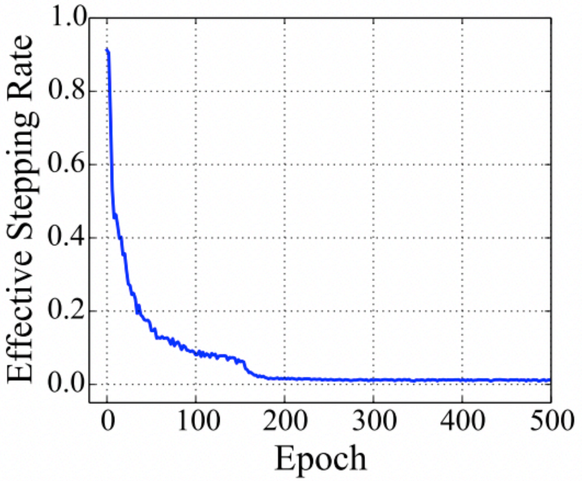


Fig. 2: Variation of the average *effective perturbation* of all the parameters during the LeNet-5 training process.

Granularity of Parameter Manipulation: Tensor or Scalar?

- 问题：同一张量中的所有标量是否共享相同的稳定趋势，以便可以直接处理张量?

- 实验表明：

- 1）不同张量表现出不同的稳定性趋势;

- 2）同时，每个张量的第5和第95误差条之间也存在较大的间隙，这意味着同一层内的参数可能表现出截然不同的稳定性特性。

- 原因：因为输入样本的某些特征可能比其他特征更容易学习，因此在神经网络层中，某些标量参数在其最佳维度上移动得更快，并且收敛得更快。这本质上是一种称为非均匀收敛的性质。

- 结论：参数同步控制的粒度是Scalar而不是Tensor

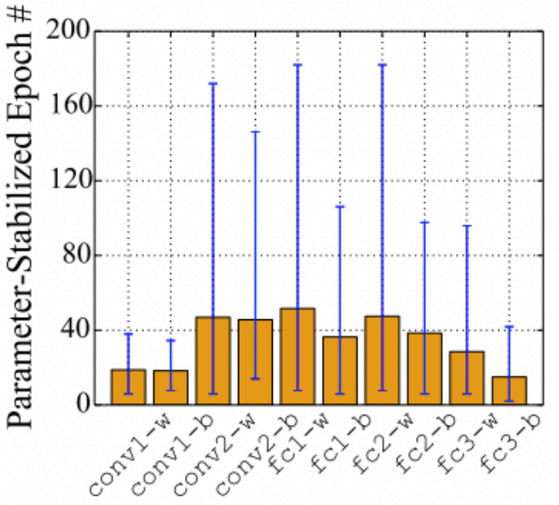


Fig. 3: Average epoch number where parameters in different layers become stable (error bars show the 5^{th}/95^{th} percentiles).

Objective and Challenges

- 目标：在这项工作中的目标是减少 FL 的整体传输量，同时保持精度性能。
- 挑战
- 1）如何有效地检测资源受限边缘设备中的稳定参数？ 在内存中维护观察窗口内的所有历史更新快照是不切实际的。
- 2）当看似稳定的参数不再同步时，如何保证模型收敛?

.

ADAPTIVE PARAMETER FREEZING

- 修改有效扰动的定义，降低边缘设备的资源占用
- 1）将稳定性检查的频率从每轮一次放宽到多轮一次，在两次连续检查之间累积模型更新
- 2）为了节省内存，采用指数移动平均 (EMA) 方法来计算有效扰动，而不是维护先前更新的窗口（占用内存大）
- 特性：如果模型更新的方向相同，则接近 1，如果它们发生振荡，则接近 0
- 如果一个参数在这个新定义下的有效扰动小于给定的稳定性阈值，我们就确定它是稳定的
- 稳定性阈值要求：足够宽松以包括所有稳定参数，同时足够严格以不错误地包括任何不稳定参数
- 引入了一种在运行时自适应调整稳定性阈值的机制：每次当大多数（例如，80%）参数被归类为稳定时，我们将稳定性阈值降低一半。（类似于学习率衰减），可以逐步纠正不正确阈值的负面影响

$$P_K = \frac{|E_K|}{E_K^{\text{abs}}}, \text{ where } E_K = \alpha E_{K-1} + (1-\alpha)\Delta_K, \\ E_K^{\text{abs}} = \alpha E_{K-1}^{\text{abs}} + (1-\alpha)|\Delta_K|. \tag{3}$$

ADAPTIVE PARAMETER FREEZING

- 直观想法：部分同步。将稳定参数排除在同步之外（但仍然在本地更新它们），并且只将模型的其余部分同步到中央服务器
- 缺点：这种部分同步方法可能会导致严重的精度损失
- 原因：边缘客户端上的本地训练数据是在特定设备环境或用户偏好下生成的。本地数据通常不会在不同的客户端上相同且独立地分布（即非 IID）。 仅在本地更新的参数最终会在不同客户端上出现不同的局部最优值
- 原则：稳定 (即不同步) 的参数必须在每个客户端上保持不变

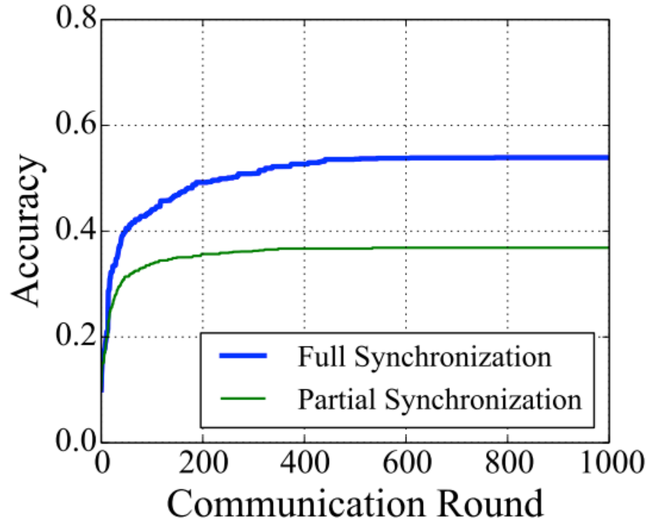


Fig. 5: *Partial synchronization* causes severe accuracy loss on non-IID data.

ADAPTIVE PARAMETER FREEZING

- 另一种直观想法：简单地将稳定的参数固定为它们的当前值。这样的永久冻结方法自然可以防止参数发散。
- 缺点：同样会导致部分精度损失。
- 原因：有些参数只是暂时稳定下来的。实际上，神经网络模型中的参数并不是相互独立的。在训练的不同阶段，它们之间可能有复杂的相互作用，这可能会导致一个看似稳定的参数偏离其当前值 (最终达到最佳)。
- 原则：任何有效的解决方案都必须处理暂时稳定的参数

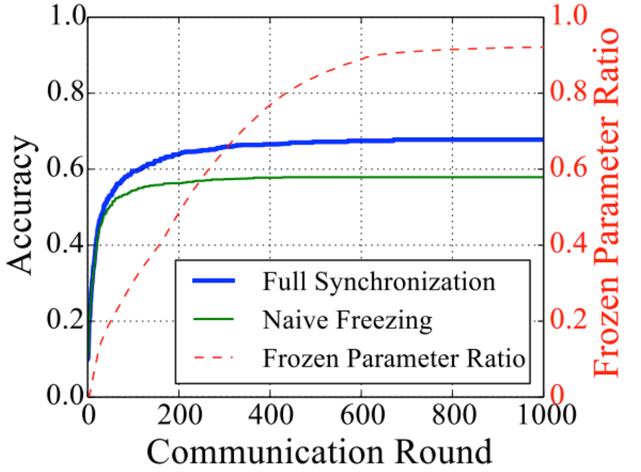


Fig. 6: *Permanent freezing* also causes accuracy loss.

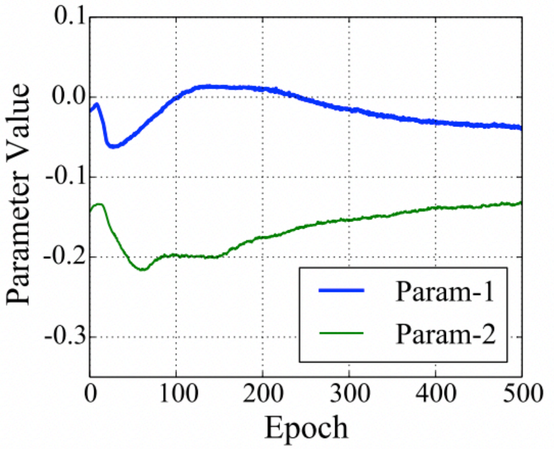


Fig. 7: The two sampled parameters stabilize only *temporarily* between epoch 100 and 200.

Adaptive Parameter Freezing

- 遵循两个原则：
- 1）为了确保模型的一致性，必须冻结非同步的稳定参数
- 2）为了确保冻结的参数可以收敛到真正的最优值，将允许它们在某个时候恢复更新（即解冻）
- 方案：将稳定的参数冻结一定的时间间隔（冻结期）。在冻结期内，该参数固定为之前的值，一旦冻结期到期，该参数应正常更新，直到再次稳定。
- 问题：如何调节冻结期?
- 如果冻结时间设置过大，我们可以显著压缩通信，但可能会影响模型的收敛精度；如果冻结周期设置得太小，参数冻结带来的性能收益将非常有限。
- 自适应参数冻结 (APF)：（受到TCP控制启发）
- 每个稳定参数的相关冻结期从一个小值开始。当冻结期结束时，该参数在下一轮重新加入训练，并更新其有效扰动重新检查其稳定性。如果参数仍然稳定，加法地增加冻结期的持续时间，否则就成倍地减少
- 收敛的参数将在大部分时间保持冻结，同时暂时稳定的参数将会迅速得到解冻

Aggressive APF: Extension for Over-parameterized Models

- DNN 模型大多是非凸的
- 例如 ResNet ，由于平面最小值或鞍点等不规则情况，某些参数可能不会收敛（过度参数化，over-parameterized）。对于这些模型，在 APF 下被冻结的稳定参数的比例可能非常有限
- 解决方案：引入了 APF 的扩展版本：Aggressive APF。
- 即在标准APF之外，我们随机将一些不稳定的参数与非零冻结期相关联，并将其置于冻结状态。（类似Dropout）
- 由于之前的自适应冻结方法可以很好地适应过早冻结的参数，因此对于过度参数化的模型，Aggressive APF 可以在没有额外精度损失的情况下获得更好的通信压缩水平。

Evaluation

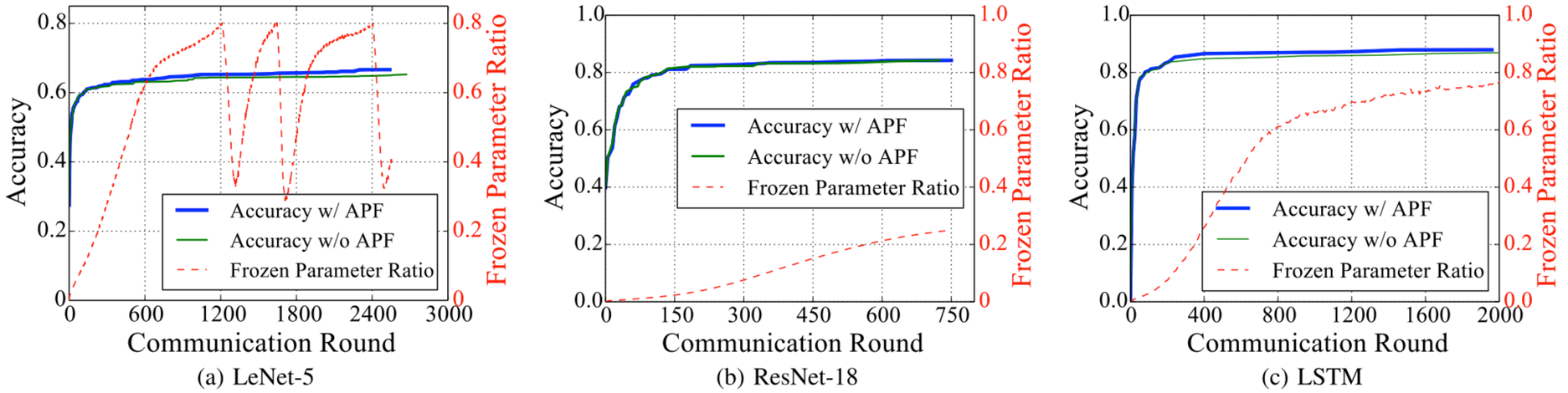


Fig. 9: Test accuracy curves when training different models with and without APF.

Model	LeNet-5	ResNet-18	LSTM
Transmission-Volume w/ APF	239 MB	2.62 (1.44) GB	194 MB
Transmission-Volume w/o APF	651 MB	3.12 GB	428 MB
APF Improvement	63.3%	16.0% (53.8%)	54.7%

TABLE I: Cumulative Transmission Volume (Values in parentheses are from Aggressive APF).

Model	LeNet-5	ResNet-18	LSTM
Per-round Time w/APF	0.74 s	139 (95) s	1.8 s
Per-round Time w/o APF	1.02 s	158 s	2.2 s
Improvement	27.5%	12.1% (39.8%)	18.2%

TABLE II: Average Per-round Time (Values in parentheses are from Aggressive APF).

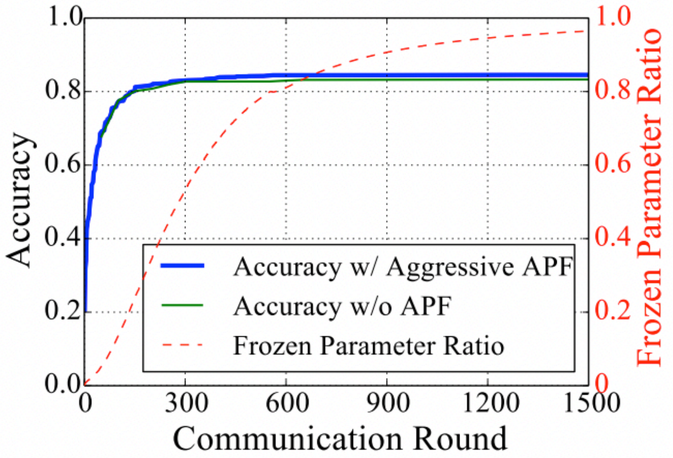
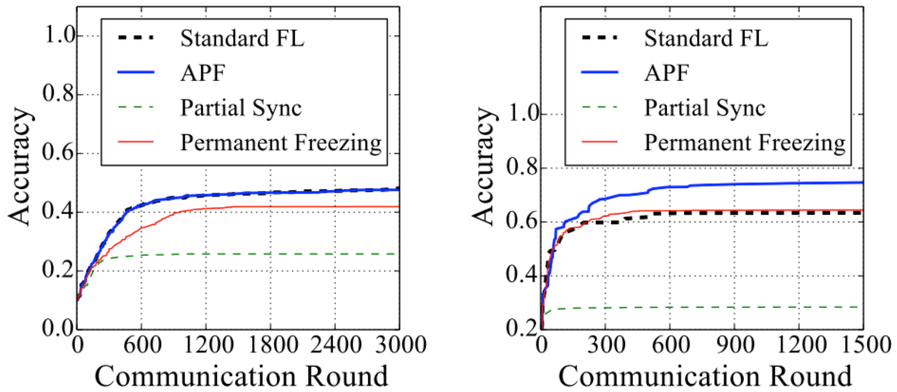


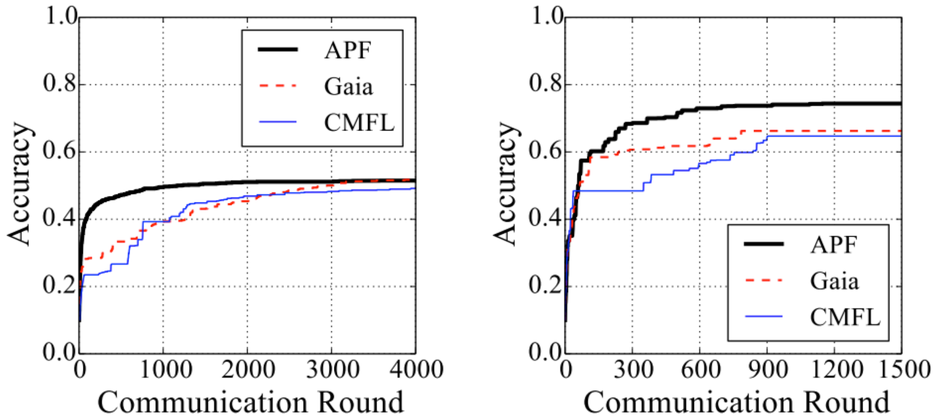
Fig. 10: *Aggressive APF* attains a much larger communication improvement for ResNet-18 without accuracy loss.

Evaluation



(a) LeNet-5 (b) LSTM

Fig. 11: Performance comparison among different schemes when training LeNet-5 and LSTM on extremely non-IID data.



(a) LeNet-5 (b) LSTM

Fig. 13: Performance comparison between APF and two typical sparsification methods—Gaia and CMFL.

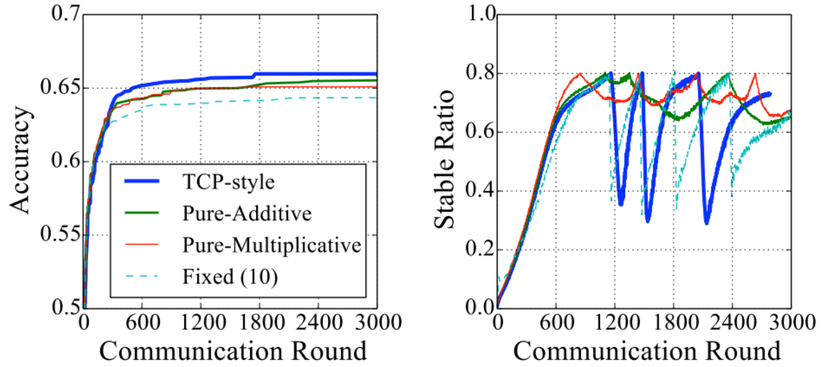


Fig. 12: A comparison of the TCP-style control scheme in APF against other potential design choices. *Pure-Additively* means to additively increase or decrease the freezing period by 1; *Pure-Multiplicatively* means to multiplicatively increase or decrease the freezing period by $2\times$; *Fixed (10)* means to freeze each stabilized parameters for 10 (in number of stability checks, i.e., for $10 \times F_c$ iterations).

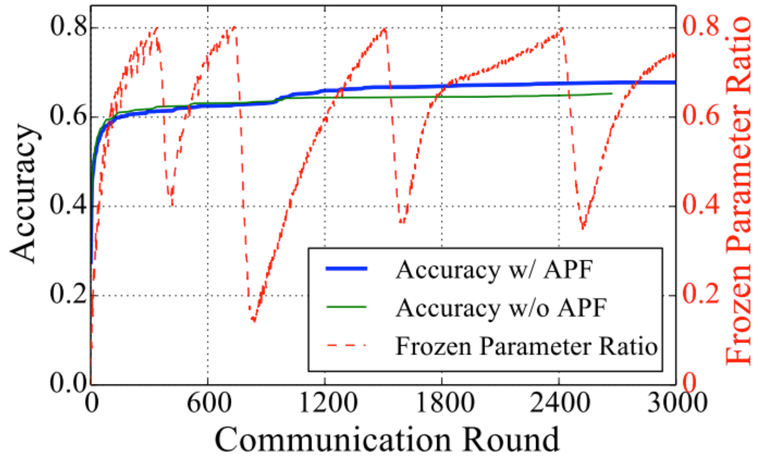


Fig. 14: When the initial stability threshold is set loosely, LeNet-5 can still converge well after several *tighten-up* actions.

Evaluation

Model	LeNet-5	ResNet-18	LSTM
Computation Time Incurred by APF (per-round)	0.009 s	1.278 s	0.011 s
Computation Time Inflation Ratio incurred by APF	1.93%	4.50%	1.42%
Memory Occupied for APF Processing	1.2 MB	142 MB	4.8 MB
Memory Inflation Ratio incurred by APF	0.18%	8.51%	2.35%

TABLE III: Computation (extra time required for each round in average) and memory overheads of APF.

Conclusion

- 疑问：分布式场景下的效果如何?

THANKS!