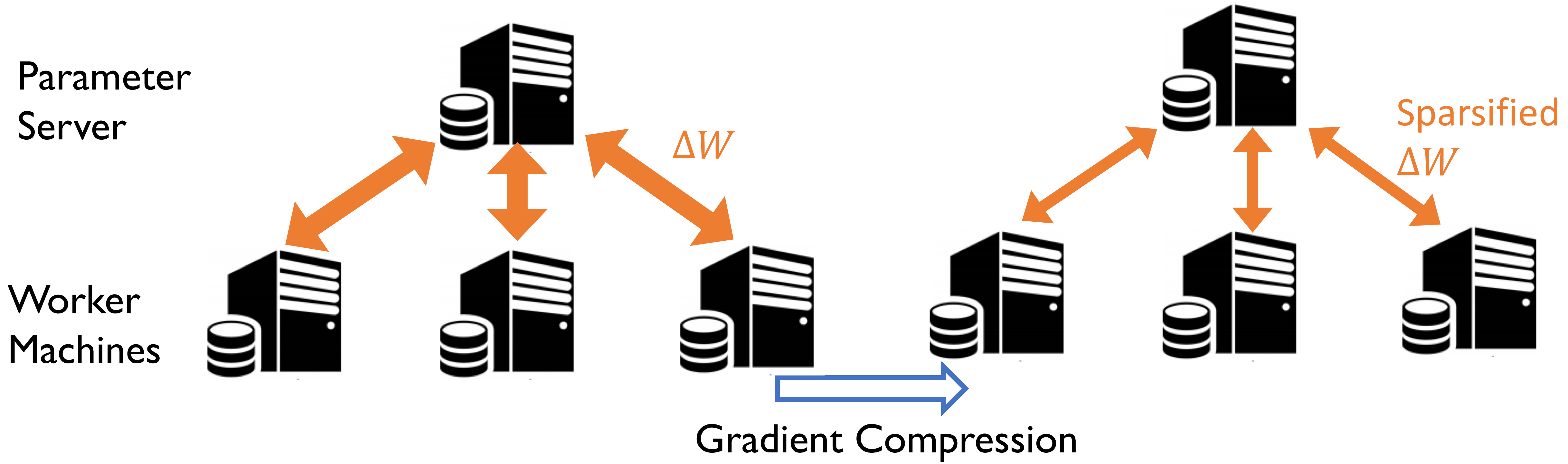
Communication-efficient Distributed Learning for Large Batch Optimization

Rui Liu, Barzan Mozafari
University of Michigan, Ann Arbor

ICML 2022

Gradient Compression in Distributed Learning



Background: Large Batch Optimization

Use largest batch size that still fits the GPU memory

- Local batch size is fixed for each GPU (total batch size increases as the number of GPUs increases)
- Fully utilize the compute power of each node
- Same generalization with some mitigation tricks (e.g., layerwise adaptive learning rates as in **Lars**) [1,2]

[1] Goyal, Priya, et al. "Accurate, large minibatch sgd: Training imagenet in 1 hour." *arXiv* (2017)
[2] You, Yang, et al. "Large batch optimization for deep learning: Training bert in 76 minutes." *arXiv* (2019)

Gradient Compression for Large Batch Optimization

Existing Gradient Compression methods

- Originally designed for the case when communication cost is dominant
- Only reduce the communication cost
- Computation of these dropped gradient coordinates is wasted

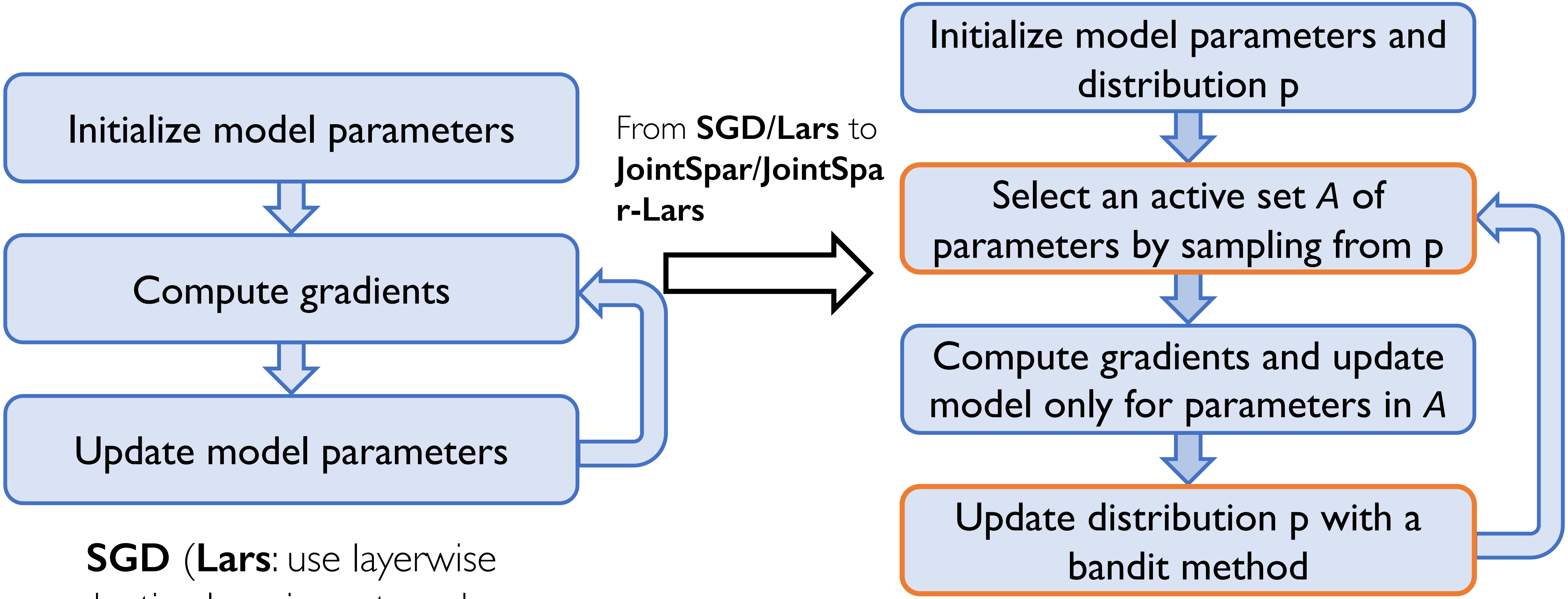
Key Observation

- Communication cost is no longer dominant for large batch optimization especially after applying existing gradient compression methods

Our idea

- Reduce both the computation and communication costs
- Use a bandit method to gradually learn the importance score of each gradient coordinate/block during training
- Skip computing the dropped gradient coordinates/blocks

Our methods: JointSpar and JointSpar-Lars



SGD (**Lars**: use layerwise adaptive learning rates when updating model parameters)

JointSpar (**JointSpar-Lars**: use layerwise adaptive learning rates)

Convergence Rate Analysis

Our methods have the same iteration convergence rates of their respective baselines (assuming nonconvex objective)

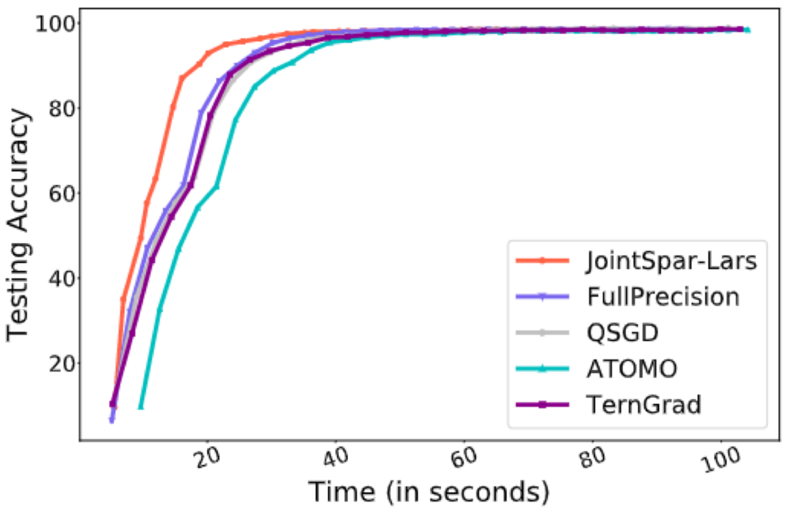
JointSpar: $O\left(\frac{1}{\sqrt{T}}\right)$

same as SGD's convergence rate

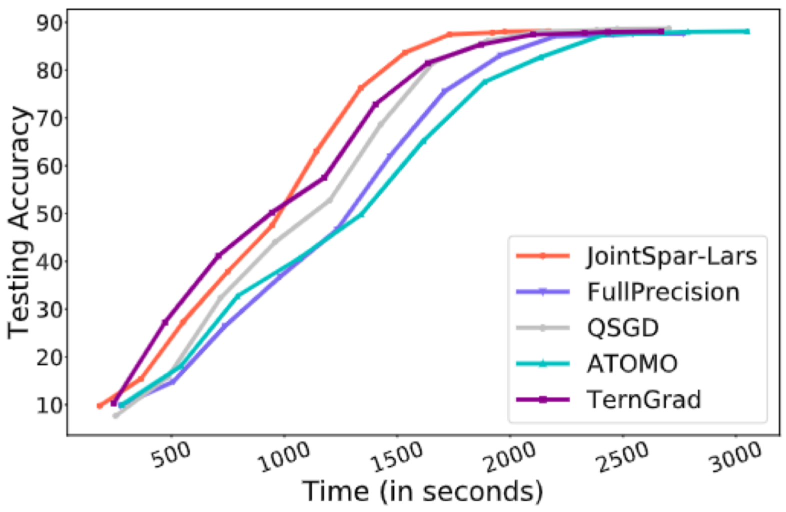
JointSpar-Lars: $O\left(\frac{1}{T}\right)$

same as Lars's convergence rate

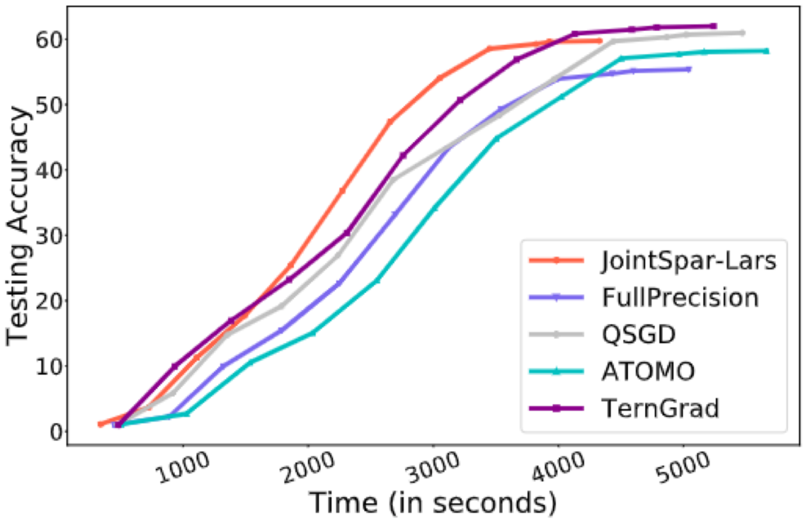
Experiments: Faster Wallclock Time Convergence



(a) LeNet on MNIST



(b) ResNet-18 on CIFAR10



(c) ResNet-18 on CIFAR100

Conclusion

- Propose gradient compression methods for large batch optimization
- Theoretically prove our methods have the same iteration convergence rates as their corresponding baseline methods
- Empirically demonstrate our methods have faster wall-clock time convergence rates

For complete details on this work, please refer to our paper

Rui Liu, Barzan Mozafari. **Communication-efficient Distributed Learning for Large Batch Optimization**, *ICML* 2022