On the Linear Speedup Analysis of Communication Efficient Momentum SGD for Distributed Non-Convex Optimization

{ Hao Yu, Rong Jin, Sen Yang }
Machine Intelligence Technology Lab, Alibaba Group (US) Inc, Bellevue, WA

1. Distributed Non-Convex Opt

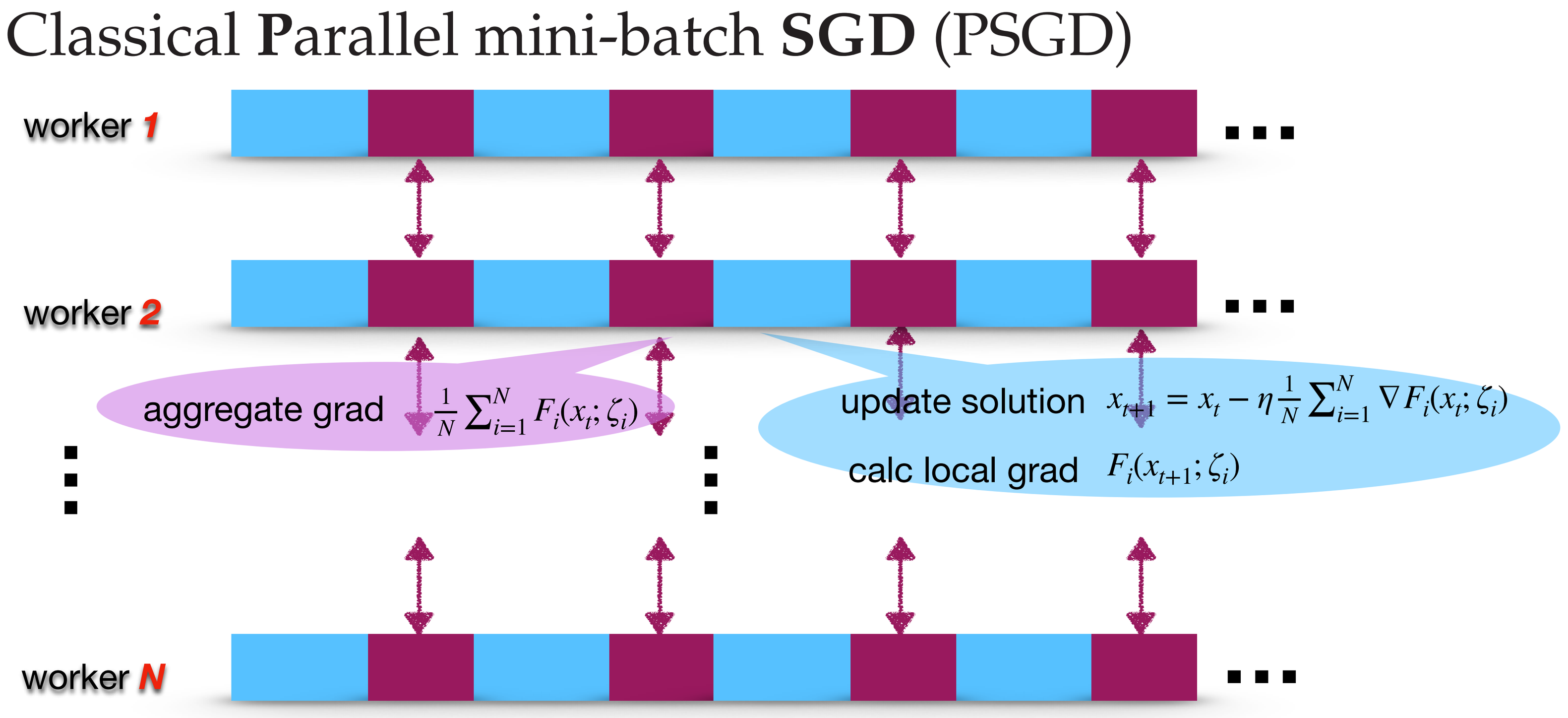
- Consensus non-convex stochastic optimization

$$\min_{\mathbf{x}\in\mathbb{R}^m} \quad \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\zeta_i}[F_i(\mathbf{x};\zeta_i)]$$

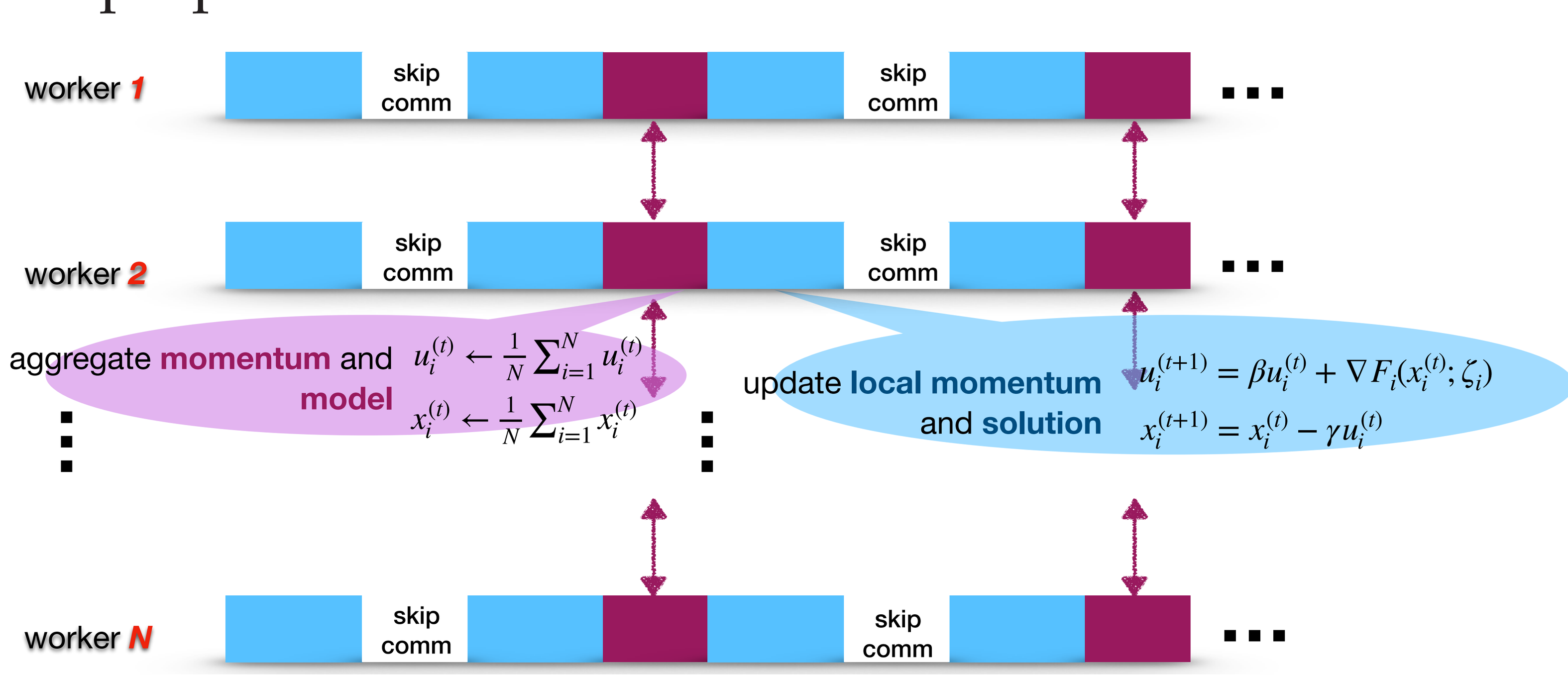
  - N parallel nodes with possibly different non-convex obj
  - Find a consensus solution in a distributed environment
- Applications:
  - Parallel training of deep neural networks
  - **Federated Learning**: users with non-identical private date learn a common ML model with intermittent comm.

2. Algorithms

- Classical **P**arallel mini-batch **SGD** (PSGD)



- PSGD has $O(1/\sqrt{NT})$ convergence, i.e., linear speedup w.r.t. number of workers, with **drawbacks**:
  - much communication: **every** iteration requires to aggregate gradients from all workers!
  - lose privacy when passing gradients/data.
  - unclear if momentum SGD (more widely used than SGD for DL) has linear speedup
- We propose **P**arallel **R**estarted **SGD** with momentum



ALG1: Parallel Restarted SGD with Momentum

1: **Parameters:** γ, $\beta\in[0,1)$, N, I, T
2: **for** $t = 1, 2\ldots, T-1$ **do**
3: Each worker obtains $\mathbf{g}_i^{(t-1)} = \nabla F_i(\mathbf{x}_i^{(t-1)};\xi_i^{(t-1)})$
4: Each worker **in parallel** updates via

$$\text{Option I: } \begin{cases} \mathbf{u}_i^{(t)} = \beta\mathbf{u}_i^{(t-1)} + \mathbf{g}_i^{(t-1)} \\ \mathbf{x}_i^{(t)} = \mathbf{x}_i^{(t-1)} - \gamma\mathbf{u}_i^{(t)} \end{cases} \quad \forall i.$$

$$\text{Option II: } \begin{cases} \mathbf{u}_i^{(t)} = \beta\mathbf{u}_i^{(t-1)} + \mathbf{g}_i^{(t-1)} \\ \mathbf{v}_i^{(t)} = \beta\mathbf{u}_i^{(t)} + \mathbf{g}_i^{(t-1)} \\ \mathbf{x}_i^{(t)} = \mathbf{x}_i^{(t-1)} - \gamma\mathbf{v}_i^{(t)} \end{cases} \quad \forall i.$$

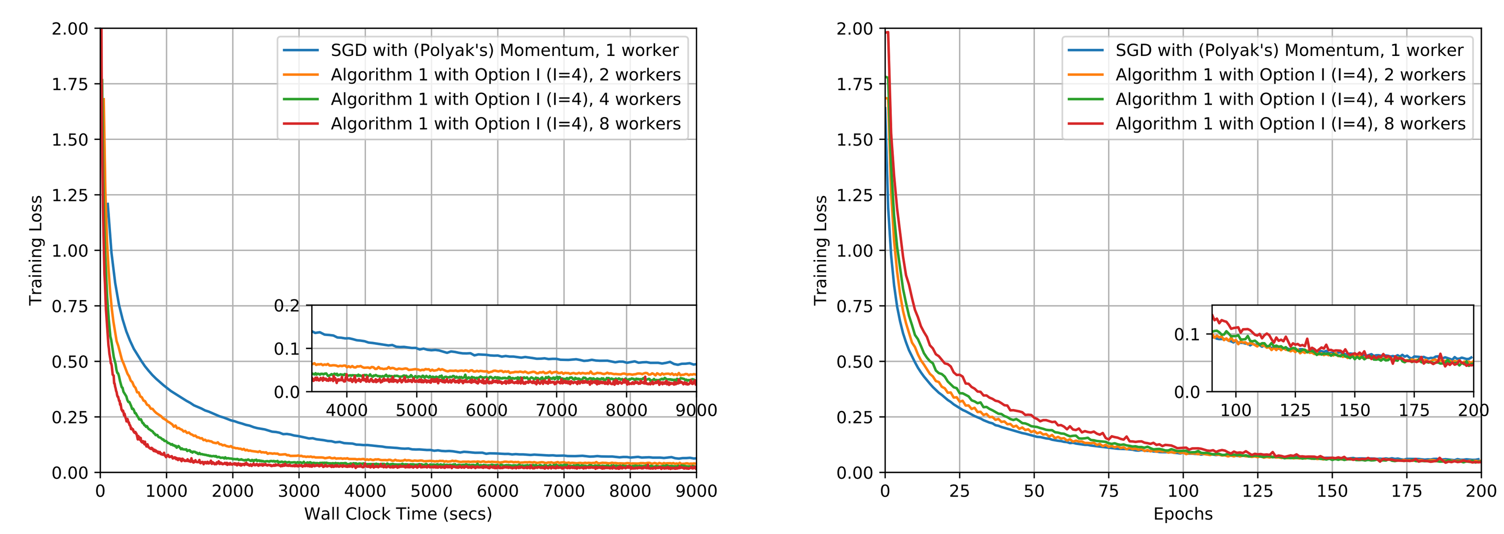
5: **if** t mod $I = 0$, **then**
6: Each worker **resets** its momentum and sol

$$\begin{cases} \mathbf{u}_i^{(t)} = \hat{\mathbf{u}} \triangleq \frac{1}{N}\sum_{j=1}^{N}\mathbf{u}_j^{(t)} \\ \mathbf{x}_i^{(t)} = \hat{\mathbf{x}} \triangleq \frac{1}{N}\sum_{j=1}^{N}\mathbf{x}_j^{(t)} \end{cases} \quad \forall i$$

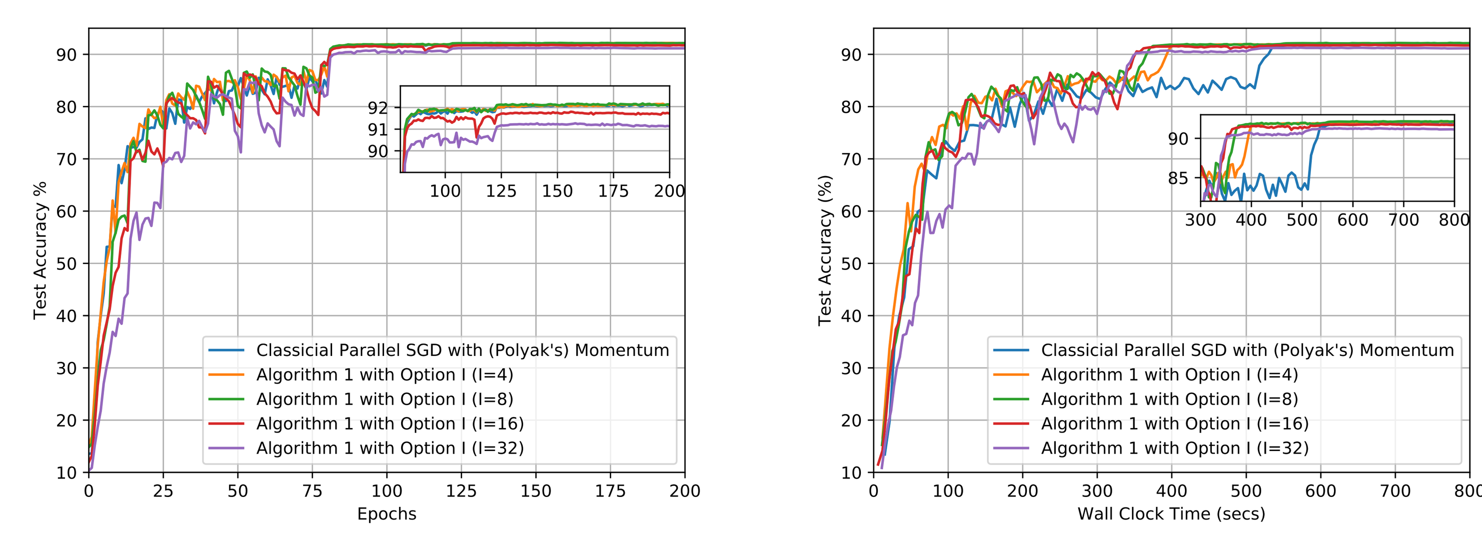
7: **end if**
8: **end for**

3. Main Results and Experiments

- # of comm rounds in PR-SGD-Momentum is I times fewer than in PSGD.
  - PR-SGD-Momentum has $O(1/\sqrt{NT})$ convergence with $I = O(T^{1/2}/N^{3/2})$ when workers access i.i.d. data sets.
  - PR-SGD-Momentum has $O(1/\sqrt{NT})$ convergence with $I = O(T^{1/4}/N^{3/4})$ when workers access distinct data sets.
- Experiments: Train ResNet56 for CIFAR10
  - Constant LR to verify speedup with $N \in \{2, 4, 8\}$.



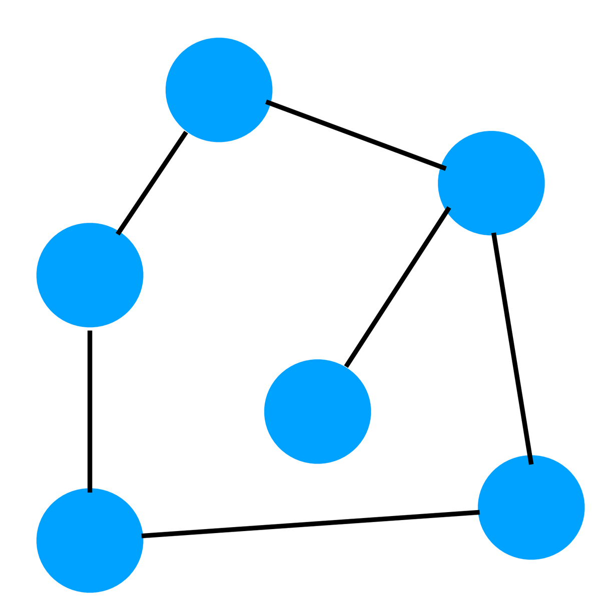
  - Decaying LR to attain SoA test accuracy with speedup.



  - # epochs in figures means **jointly** accessed by **all** workers.

4. Extension: Decentralized Comm

- Mmomentum/model aggregations in PR-SGD-Momentum uses global comm.
- **What if only decentralized comm between neighbors are used?**
- Decentralized SGD **w/o momentum** is analyzed in [Lian et al.'17]
- Mixing matrix $\mathbf{W}$ encodes comm faithful to net topology.



ALG2: Momentum SGD w/ Decentralized Comm

1: **Parameters:** $\mathbf{W}$,γ, $\beta\in[0,1)$, N, T
2: **for** $t = 1, 2\ldots, T-1$ **do**
3: Each worker obtains $\mathbf{g}_i^{(t-1)} = \nabla F_i(\mathbf{x}_i^{(t-1)};\xi_i^{(t-1)})$.
4: Each worker **in parallel** updates via "Option I" or "Option II" in Alg1.
5: Each worker i updates its momentum and sol

$$\begin{cases} \mathbf{u}_i^{(t)} = \sum_{j=1}^{N}\tilde{\mathbf{u}}_j^{(t)}W_{ji} \\ \mathbf{x}_i^{(t)} = \sum_{j=1}^{N}\tilde{\mathbf{x}}_j^{(t)}W_{ji} \end{cases} \quad \forall i$$

6: **end for**

- This paper proves that Alg2 has $O(1/\sqrt{NT})$ convergence, i.e., linear speedup w.r.t. number of workers.