Leader Stochastic Gradient Descent (LSGD) for Distributed Training of Deep Learning Models

Yunfei Teng*, Wenbo Gan*, Francois Chalus, Anna Choromanska, Donald Goldfarb, Adrian Weller

Motivation

We consider distributed optimization under communication constraints for training deep learning models. Our method differs from the state-of-art parameter-averaging scheme (MDP/GSD). In a number of cases:
1. our objective formulation does not change the location of stationary points compared to the original optimization problem;
2. we avoid convergence deteriorations caused by pushing local workers descending to different local minima to each other (i.e., the average of their parameters);
3. our update by design breaks the curse of symmetry (the phenomenon of being trapped in poorly generalizing sub-optimal solutions in symmetric non-convex problems);
4. our approach is more communication efficient since it broadcasts only parameters of the leader rather than all workers.

Multi-Leader Setting

We propose a multi-leader setting well-aligned with the state-of-art parameter-optimization problem; location of stationary points compared to the original

Objective function:

\[ L = \sum_{i=1}^{n} f(x_i) \]

is the global leader (the best worker among local leaders)

\[ \eta \in \mathbb{R}^n \]

is the global best worker among local best workers

\[ \lambda_j \in \mathbb{R}^n \]

are the hyperparameters that denote the strength of the forces pulling the workers to their local and global leader respectively.

Algorithm:

1. **Input:** pulling coefficients \( \lambda \), learning rate \( \eta \), local/global communication periods \( \tau, \tau' \)
2. **Initialization:**
   - Initialize random \( x^0 \)
   - Set iteration counter \( t = 0 \)
   - Set \( m = \arg \min \{ \| \nabla L(x^0) \| \} \)
3. **repeat**
   - for all \( j = 1, \ldots, n \)
     - One random worker \( i \) selects \( x_i \)
     - \( x_i^t = x_i^{t-1} - \eta \nabla L(x_i^t) \)
   - if \( m \) is stochastic
     - Do in parallel for each worker
     - One random worker \( i \) selects \( x_i \)
     - \( x_i^t = x_i^{t-1} - \eta \nabla L(x_i^t) \)
   - Do for each leader:
     - Pull to the local best workers
     - \( x_l^t = \arg \min_{x \in \mathbb{R}^n} \left\{ \| \nabla L(x) \| \right\} \)
     - Pull to the global best worker
     - \( x_g^t = \arg \min_{x \in \mathbb{R}^n} \left\{ \| \nabla L(x) \| \right\} \)
4. **end if**
5. **end for**
6. **end repeat**

Improvements in Step Direction

When the landscape is locally convex, we expect that the new leader term will bring the step direction closer to the global minimizer. This can be shown quantitatively for the quadratic case.

\[ \text{Theorem:} \] (for a convex quadratic) if either \( x \) is small or, in the angle \( \theta \) between the gradient and the Newton step is large, then at least half of the candidate leaders \( x_l \) and \( x_g \) will bring the step direction closer to the Newton direction \( \nabla L(x) \), in the sense that

\[ \arg \min_{x \in \mathbb{R}^n} \left\{ \| \nabla L(x) \| \right\} \leq \arg \min_{x_l \in \mathbb{R}^n} \left\{ \| \nabla L(x) \| \right\} \]

Empirical Results

Figure: VGG and ResNet on CIFAR and ImageNet with 4 workers. Test error for the center variable versus wall-clock time.

References