A Communication Efficient Stochastic Multi-Block Alternating Direction Method of Multipliers

{ Hao Yu}
Amazon, Seattle, WA

1. Linearly Constrained Sto-Opt

- Llinearly constrained stochastic convex programs

$$\min_{\mathbf{x}_i \in \mathcal{X}_i, \forall i} \quad f(\mathbf{x}) = \sum_{i=1}^{N} f_i(\mathbf{x}_i) \text{ s.t. } \sum_{i=1}^{N} \mathbf{A}_i \mathbf{x}_i = \mathbf{b}$$

- N arbitrary; Each $f_i(\mathbf{x}_i) \triangleq \mathbb{E}_\xi[f_i(\mathbf{x}_i;\xi)]$ with expensive true gradient but cheap unbiased stochastic gradient.
- **Applications**
  - **Large scale** linearly constrained **optimization**, e.g,. linear programs: Too large to store or solve on a single node.
  - **Distributed machine learning**: N distributed nodes (with possibly non-identical training data) jointly train a common ML model.

2. Communication Efficient ADMM

- **ADMM** is effective and popular for distributed optimization, yet **suffers significant communication overhead** for passing Lagrange multipliers.
- Conventional ADMM involves a communication step following immediately a computation step. (Communication often much more expensive than SGD computation.)
- This paper develops **communication efficient** multi-block stochastic ADMM to **reduce communication rounds w/o. sacrificing convergence**.

3. Our Algorithm

Alg1 : Two-Layer Communication Efficient ADMM

1: **Input:** Algorithm parameters T, $\{\rho^{(t)}\}_{t\geq 1}$, $\{\nu^{(t)}\}_{t\geq 1}$ and $\{K^{(t)}\}_{t\geq 1}$.
2: Initialize arbitrary $\mathbf{y}_i^{(0)} \in \mathcal{X}_i, \forall i$, $\mathbf{r}^{(0)} = \sum_{i=1}^{N} \mathbf{A}_i \mathbf{y}_i^{(0)} - \mathbf{b}$, $\boldsymbol{\lambda}^{(0)} = \mathbf{0}$, and $t = 1$.
3: **while** $t \leq T$ **do**
4: Each node i defines $\phi_i^{(t)}(\mathbf{x}_i) \triangleq$

$$f_i(\mathbf{x}_i) + \rho^{(t)}\langle \mathbf{r}^{(t-1)} + \frac{1}{\rho^{(t)}}\boldsymbol{\lambda}^{(t-1)}, \mathbf{A}_i\mathbf{x}_i - \frac{\mathbf{b}}{N}\rangle + \frac{\nu^{(t)}}{2}\|\mathbf{x}_i - \mathbf{y}_i^{(t-1)}\|^2$$

and **in parallel** updates $\mathbf{x}_i^{(t)}, \mathbf{y}_i^{(t)}$ using local sub-procedure **Alg 2** via

$$(\mathbf{x}_i^{(t)}, \mathbf{y}_i^{(t)}) = \text{STO-LOCAL}(\phi_i^{(t)}(\cdot), \mathcal{X}_i, \mathbf{y}_i^{(t-1)}, K^{(t)})$$

5: Each node i passes $\mathbf{x}_i^{(t)}$ and $\mathbf{y}_i^{(t)}$ between nodes or to a parameter server. Update $\boldsymbol{\lambda}^{(t)}$ and $\mathbf{r}^{(t)}$

$$\boldsymbol{\lambda}^{(t)} = \boldsymbol{\lambda}^{(t-1)} + \rho^{(t)}\Big(\sum_{i=1}^{N}\mathbf{A}_i\mathbf{x}_i^{(t)} - \mathbf{b}\Big)$$

$$\mathbf{r}^{(t)} = \sum_{i=1}^{N}\mathbf{A}_i\mathbf{y}_i^{(t)} - \mathbf{b}.$$

6: Update $t \leftarrow t+1$.
7: **end while**
8: **Output:** $\overline{\mathbf{x}}^{(T)} = \frac{1}{\sum_{t=1}^{T}\rho^{(t)}}\sum_{t=1}^{T}\rho^{(t)}\mathbf{x}^{(t)}$

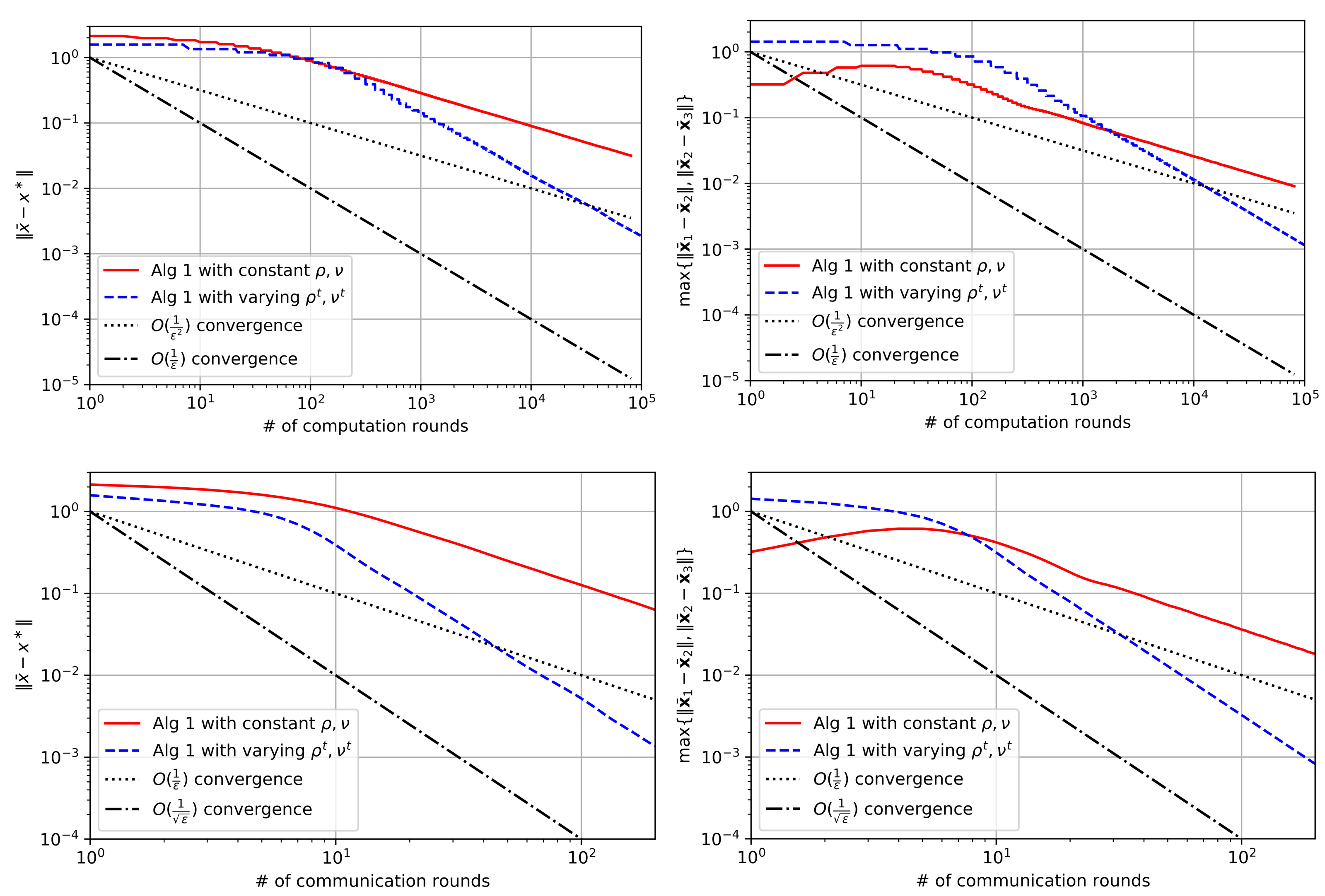
- SGD-LOCAL (Alg 2) is a $K^{(t)}$ step SGD procedure with designed initialization, step size and averaging rules.

4. Performance Analysis

- **Under corresponding algorithm parameter rules**, to achieve an $O(\epsilon)$ accuracy solution
  - **General Convex**: Alg 1 uses $\tilde{O}(1/\epsilon^2)$ SGD update rounds and $\tilde{O}(1/\epsilon)$ inter-node communication rounds.
  - **Strongly Convex**: Alg 1 uses $\tilde{O}(1/\epsilon)$ SGD update rounds and $\tilde{O}(1/\sqrt{\epsilon})$ inter-node communication rounds.
- The # of communication rounds is only the square root of that of computation (SGD update) rounds.
- Lowest computation complexity for stochastic convex opt with lower communication complexity than other stochastic ADMM.

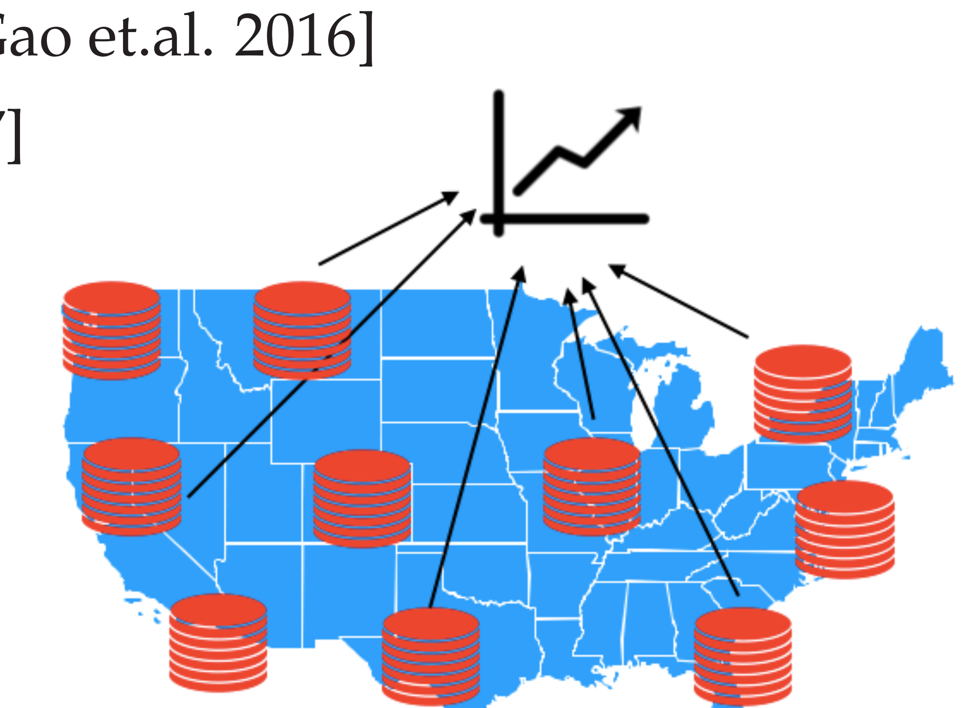
5. Experiments

- Convergence rate verification: smooth strongly convex



Setting algorithm parameters in line with our theory yields the (better) proven convergence rates

- Distributed logistic regression: 10 distributed nodes with sharded data jointly train a common model
  - RPDBUS ADMM [Gao et.al. 2016]
  - DCS [Lan et.al. 2017]



Compare "loss/consensus" v.s. "# comp/comm"