## 1. LINEARLY CONSTRAINED STO-OPT

- Linearly constrained stochastic convex programs:

  \[
  \min_{x_i \in X_i, i = 1}^N f_i(x_i) \text{ s.t. } \sum_{i=1}^N a_i x_i = b
  \]

- \(N\) arbitrary; Each \(f_i(x_i) \triangleq \mathbb{E}_\xi[f_i(x_i; \xi)]\) 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

**Algorithm 1**: 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 \(x_i^{(0)} \in X_i\), \(\forall i\), \(x^{(0)} = \sum_{i=1}^N a_i x_i^{(0)} - b\), \(\lambda^{(0)} = 0\), and \(t = 1\).
3. While \(t \leq T\) do
4. Each node \(i\) defines \(\phi_i^{(t)}(x_i) \triangleq f_i(x_i) + \rho^{(t)}(x_i - x_i^{(t-1)}) + \frac{1}{\rho^{(t)}} \lambda^{(t-1)} - b\)

and in parallel updates \(x_i^{(t)}, y_i^{(t)}\) using local sub-procedure Alg 2 via

\((x_i^{(t)}, y_i^{(t)}) = \text{STO-LOCAL}(\phi_i^{(t)}(\cdot), x_i^{(t-1)}, y_i^{(t-1)}, K^{(t)})\)

5. Each node \(i\) passes \(x_i^{(t)}\) and \(y_i^{(t)}\) between nodes or to a parameter server. Update \(\lambda^{(t)}\) and \(r^{(t)}\)

\(\lambda^{(t)} = \lambda^{(t-1)} + \rho^{(t)} \left( \sum_{i=1}^N a_i x_i^{(t)} - b \right)\)

\[r^{(t)} = \sum_{i=1}^N a_i y_i^{(t)} - b.\]

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

- 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 \(O(1/\epsilon^2)\) SGD update rounds and \(O(1/\epsilon)\) inter-node communication rounds.
  - **Strongly Convex**: Alg 1 uses \(\tilde{O}(1/\epsilon)\) SGD update rounds and \(O(1/\sqrt{\epsilon})\) 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 shared data jointly train a common model
  - RPDBUS ADMM [Gao et.al. 2016]
  - DCS [Lan et.al. 2017]

Compare “loss/consensus” vs. “# comp/comm”