Can Decentralized Algorithms Outperform Centralized Algorithms? 
A Case Study for Decentralized Parallel Stochastic Gradient Descent

**Executive summary**
- In large scale machine learning, instead of using only one machine, we distribute data into multiple machines and let them collaborate on solving the following optimization problem:

$$\min \mathcal{f}(\mathbf{x}) := \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{\mathbf{y} \sim p_{\mathbf{y}}(\mathbf{x}; a)} f_{i}(\mathbf{x}) = f(\mathbf{x})$$

- Based on our case study, the decentralized algorithm may not need more iterations to converge than its centralized counterpart.
- Decentralized algorithms outperform centralized algorithms for networks with low bandwidth and high latency.

**Background:** Centralized Method 1 (Parameter Server)

**Centralized Method 2 (AllReduce)**

**Our Proposal: Decentralized Method**

**Algorithm (Decentralized Parallel SGD)**

$$\mathbf{x}(1) = \mathbf{x}(2) = \ldots = \mathbf{x}(N) \quad \mathbf{W}$$

**Theoretical Results**

- Assumptions
  - Lipschitzian: $f(\mathbf{x})$ is with $L$-Lipschitzian gradient
  - Bounded variance: the variance of each worker's partial gradient is bounded
  - Spectral gap: $\rho = \max(\lambda_{i}(\mathbf{W})|i=1, \ldots, k) < 1$

- The following rate holds:
  $$\mathbb{E}[(f(\mathbf{x}) - f^{*})^{2}] \leq \frac{L}{N} + \frac{1}{2} \rho^{2} + \frac{1}{2} (1-\rho^{2})^{2}$$

**Evaluation:** Proprietary dataset and model (IBM Watson Natural Language Classifier)

**Evaluation:** Public dataset and model (CIFAR10/ResNet)

**Future work**
- Asynchronous parallelism for decentralized algorithms.
- Investigate new topologies to improve communication efficiency.

**Executive summary**
Decentralized algorithms outperform centralized algorithms for networks with low bandwidth and high latency.