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

Yunfei Teng*, Wenbo Gao*, 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 EASGD [1] in a number of ways:

1. our objective formulation does not change the location of stationary points compared to the original optimization problem;
2. we avoid convergence decelerations caused by pulling local workers descending to different local minima to each other (i.e. to 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 landscapes);
4. our approach is more communication efficient since it broadcasts only parameters of the leader rather than all workers.

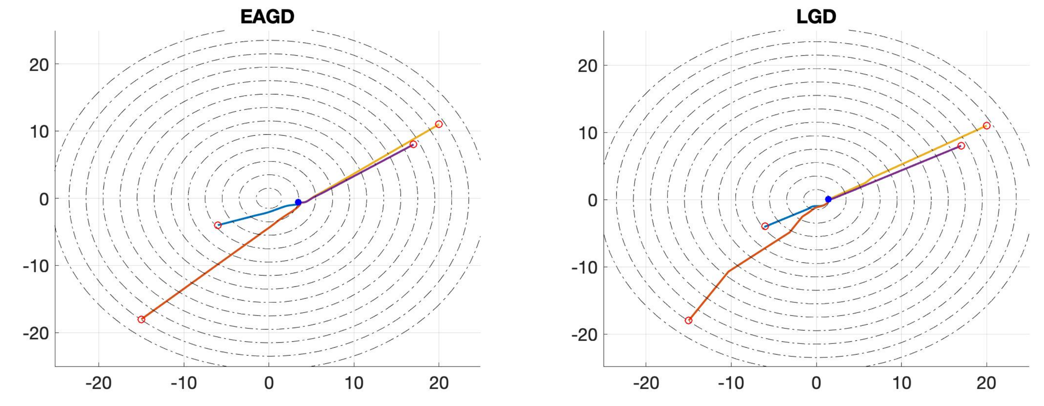


Figure: Trajectories of variables $(\mathbf{x}, \mathbf{y})$ during optimization.

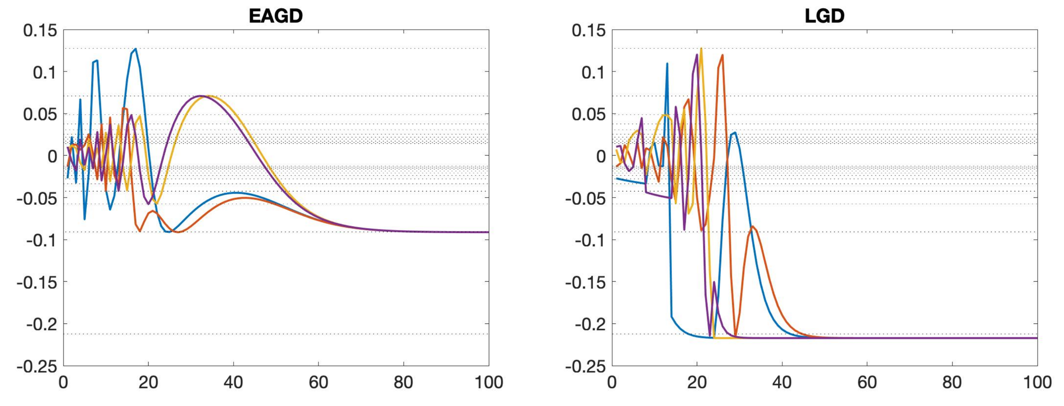
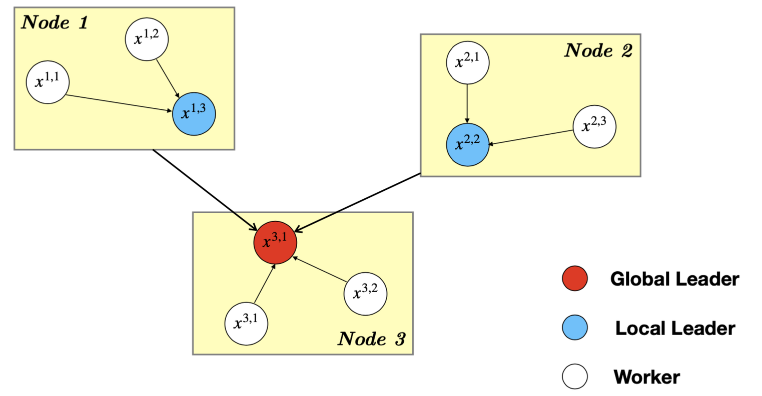


Figure: The value of the objective function $L(x, y)$ for each worker during training on a heavily non-convex problem.

Multi-Leader Setting

We propose a multi-leader setting well-aligned with the hardware architecture:

- single computational node is formed by local workers $\Rightarrow$ local leader
- group of nodes $\Rightarrow$ global leader



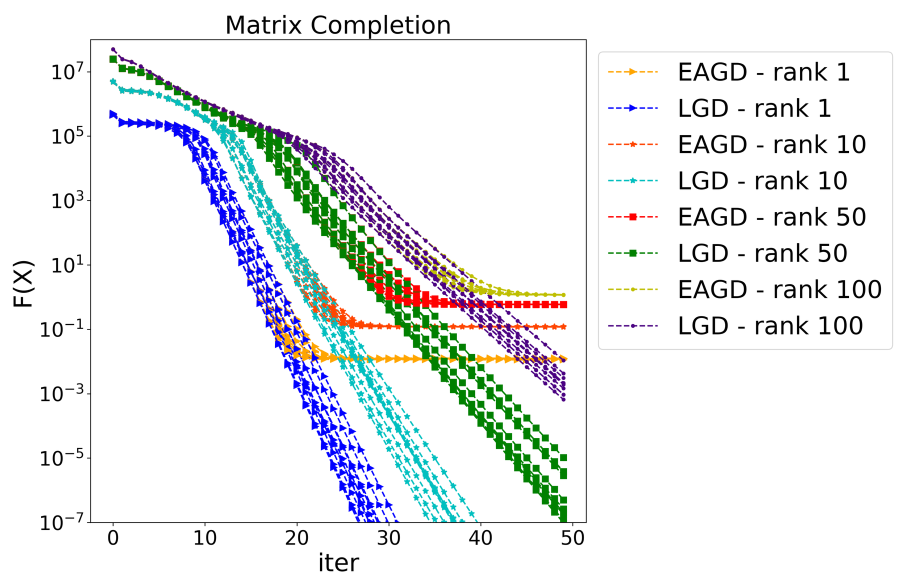
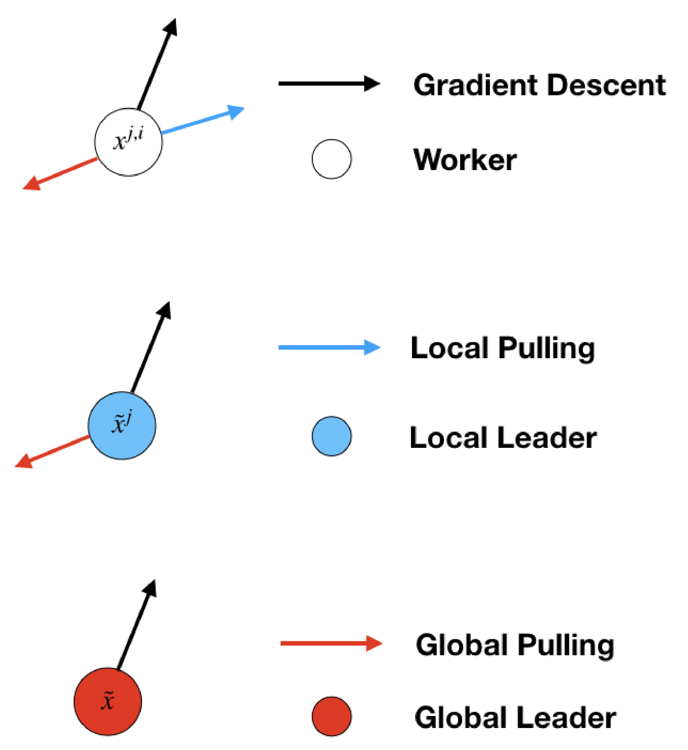


Figure: Low-rank matrix completion problems solved with EAGD and LGD.

LSGD updates



Objective function:

$$\min_{x^{1,1},x^{1,2},\ldots,x^{n,l}} L(x^{1,1}, x^{1,2}, \ldots, x^{n,l})$$

$$:= \min_{x^{1,1},x^{1,2},\ldots,x^{n,l}} \sum_{j=1}^{n}\sum_{i=1}^{l} \mathbb{E}[f(x^{j,i};\xi^{j,i})] + \frac{\lambda}{2}||x^{j,i} - \tilde{x}^{j}||^2 + \frac{\lambda_G}{2}||x^{j,i} - \tilde{x}||^2 \quad (1)$$

Parameter update:

$$x^{j,i}_{t+1} = x^{j,i}_t - \underbrace{\eta g^{j,i}_t(x^{j,i}_t)}_{\text{gradient descent}} - \underbrace{\lambda(x^{j,i}_t - \tilde{x}^{j}_t)}_{\text{local pulling}} - \underbrace{\lambda_G(x^{j,i}_t - \tilde{x}_t)}_{\text{global pulling}} \quad (2)$$

- $\xi^{j,i}$s are the data samples drawn from data distribution $\mathscr{P}$
- n is the number of groups and l is the number of workers in each group
- $x^{j,1}, x^{j,2}, \ldots, x^{j,l}$ are the parameters of the workers in the j^{th} group
- $\tilde{x}^j$ is the local leader and $\tilde{x}$ is the global leader (the best worker among local leaders)
- λ and λ_G are the hyperparameters that denote the strength of the forces pulling the workers to their local and global leader respectively

Algorithm

LSGD Algorithm (Asynchronous)

Input: pulling coefficients λ, λ_G, learning rate η, local/global communication periods τ, τ_G

Initialize:

Randomly initialize $x^{1,1}, x^{1,2}, \ldots, x^{n,l}$

Set iteration counters $t^{j,i} = 0$

Set $\tilde{x}^j_0 = \arg\min_{x^{j,1},\ldots,x^{j,l}} \mathbb{E}[f(x^{j,i};\xi^{j,i}_0)]$, $\tilde{x}_0 = \arg\min_{x^{1,1},\ldots,x^{n,l}} \mathbb{E}[f(x^{j,i};\xi^{j,i}_0)]$;

repeat

for all $j = 1, 2, \ldots, n$, $i = 1, 2, \ldots, l$ **do** ▹ Do in parallel for each worker

Draw random sample $\xi^{j,i}_{t^{j,i}}$

$x^{j,i} \longleftarrow x^{j,i} - \eta g^{j,i}_t(x^{j,i})$

$t^{j,i} = t^{j,i} + 1$;

if $nl\tau$ divides $(\sum_{j=1}^{n}\sum_{i=1}^{l} t^{j,i})$ **then**

$\tilde{x}^j = \arg\min_{x^{j,1},\ldots,x^{j,l}} \mathbb{E}[f(x^{j,i};\xi^{j,i}_{t^{j,i}})]$. ▹ Determine the local best workers

$x^{j,i} \longleftarrow x^{j,i} - \lambda(x^{j,i} - \tilde{x}^j)$ ▹ Pull to the local best workers

end if

if $nl\tau_G$ divides $(\sum_{j=1}^{n}\sum_{i=1}^{l} t^{j,i})$ **then**

$\tilde{x} = \arg\min_{x^{1,1},\ldots,x^{n,l}} \mathbb{E}[f(x^{j,i};\xi^{j,i}_{t^{j,i}})]$. ▹ Determine the global best worker

$x^{j,i} \longleftarrow x^{j,i} - \lambda_G(x^{j,i} - \tilde{x})$ ▹ Pull to the global best worker

end if

end for

until termination

Stationary Points of LSGD

The L(S)GD loss (1) has the property that the leader variable is *always* a stationary point of the underlying objective function. This is not the case for E(A)SGD.

Theorem: Let Ω_i be the points $(x^1, \ldots, x^p)$ where x^i is the unique minimizer among $(x^1, \ldots, x^p)$. If $x^* = (w^1, \ldots, w^p) \in \Omega_i$ is a stationary point of the LSGD objective function, then $\nabla f(w^i) = 0$.

Convergence Rates

Under the assumption of strong convexity of f, L(S)GD recovers the same convergence rate as SGD.

Theorem: For sufficiently small learning rate η, the LSGD step satisfies

$$\mathbb{E}f(x_+) - f(x^*) \leq (1 - m\eta)(f(x) - f(x^*)) - \eta\lambda(f(x) - f(z)) + \frac{\eta^2 M}{2}\sigma^2.$$

Here z denotes the leader point. Since $f(z) \leq f(x^i)$ for each worker x^i, we see that there is an additional reduction $\eta\lambda(f(x) - f(z))$ induced by the leader. We can then obtain the standard convergence rate:

Theorem: If η decreases at the rate $\eta_k = \Theta(\frac{1}{k})$, then $\mathbb{E}f(x_k) - f(x^*) \leq O(\frac{1}{k})$.

This complexity matches other distributed algorithms such as Elastic Averaging SGD and Hogwild.

On large-scale problems, we may not exactly evaluate f to determine the leader, and instead use estimates of f. Suppose that we have unbiased estimates $\widetilde{f}(x^i)$ with uniformly bounded variance σ^2_f, and select the worker with lowest realized estimate to become the leader. This increases the variance of the limit by no more than $O(\lambda\sqrt{n}\sigma_f)$, where n is the number of workers.

Theorem: Suppose the leader is stochastic. Then $\limsup_{k\to\infty} \mathbb{E}f(x_k) - f(x^*) \leq \frac{1}{2}\eta\kappa\sigma^2 + \frac{4}{m}\lambda\sqrt{p}\sigma_f$. If η, λ decrease at the rate $\eta_k = \Theta(\frac{1}{k})$, $\lambda_k = \Theta(\frac{1}{k})$, then $\mathbb{E}f(x_k) - f(x^*) \leq O(\frac{1}{k})$.

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 quadratic problems.

Theorem: Let f be a convex quadratic. If either the λ is small, or the angle at x between the gradient and the Newton step is large, then at least half of the candidate leaders $\{z : f(z) \leq f(x)\}$ will bring the step direction closer to the Newton direction d_N in the sense that $\text{angle}(d_N, -\nabla f(x) + \lambda(z - x)) \leq \text{angle}(d_N, -\nabla f(x))$.

Empirical Results

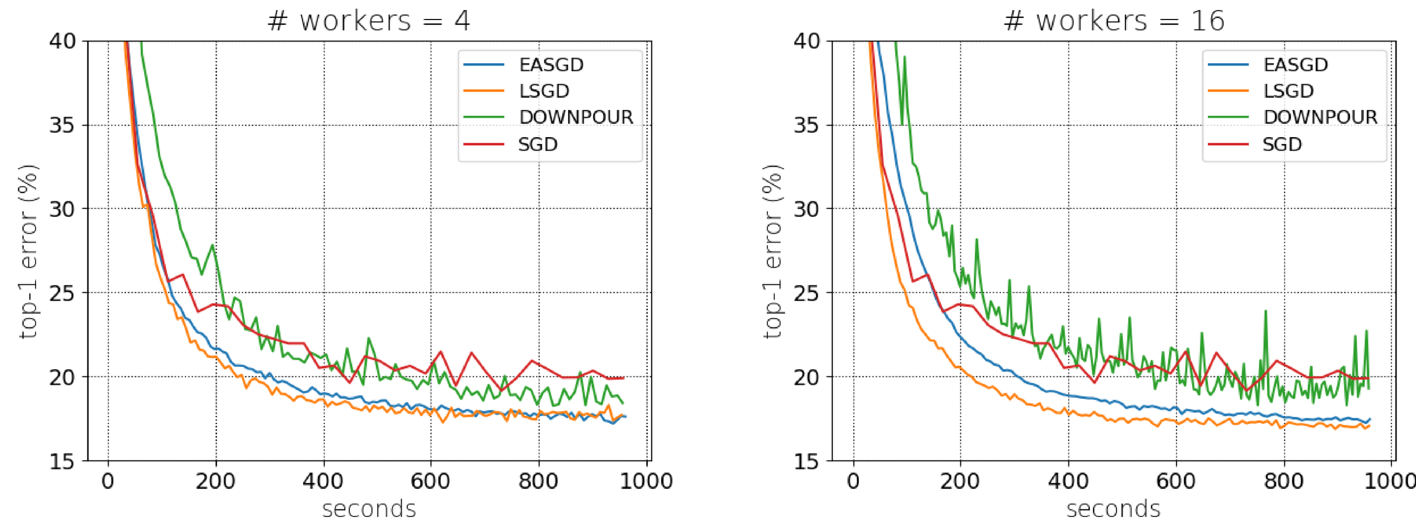


Figure: CNN7 on CIFAR-10 with 4 workers (on the left) and 16 workers (on the right). Test error for the center variable versus wall-clock time.

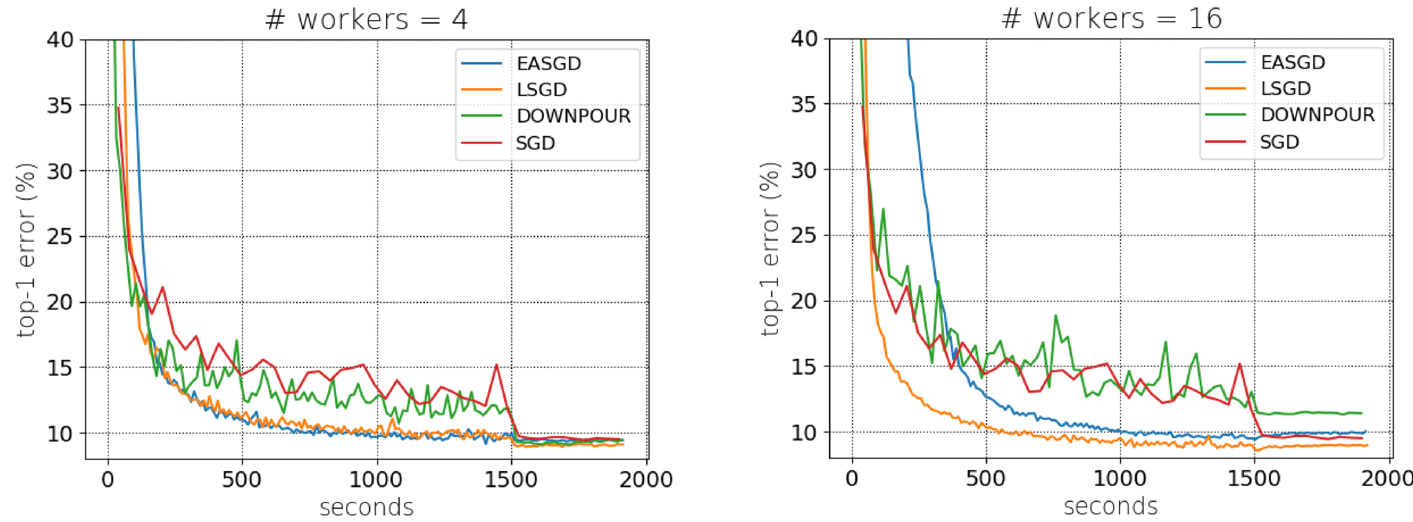


Figure: ResNet20 on CIFAR-10 with 4 workers (on the left) and 16 workers (on the right). Test error for the center variable versus wall-clock time.

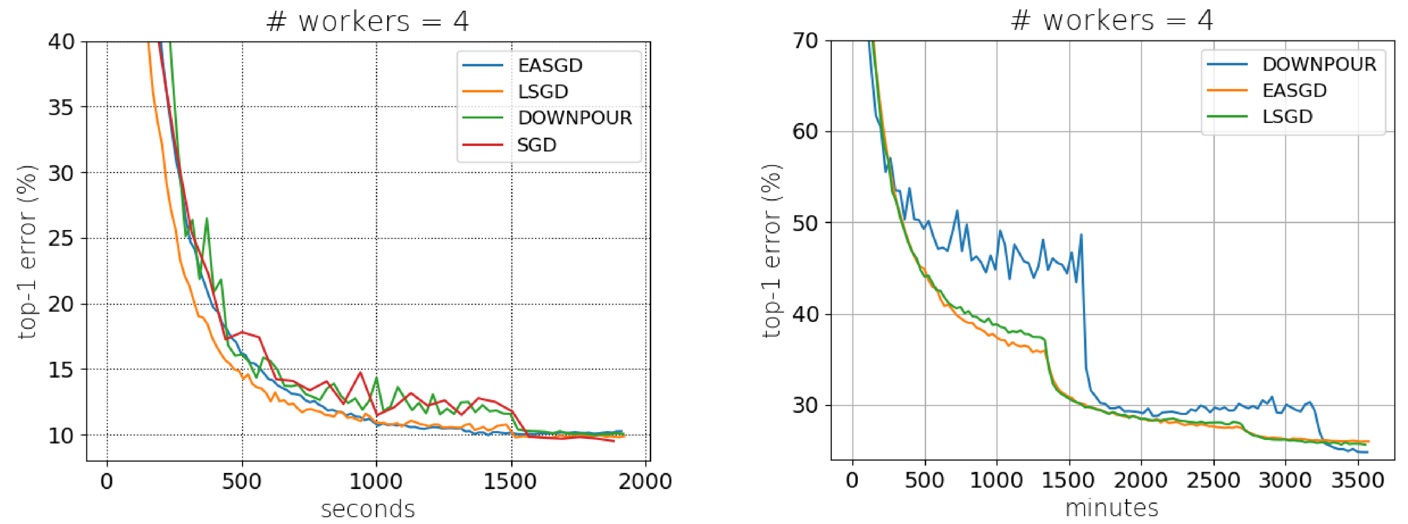


Figure: VGG16 on CIFAR-10 (on the left) and ResNet50 on ImageNet (on the right) with 4 workers. Test error for the center variable versus wall-clock time.

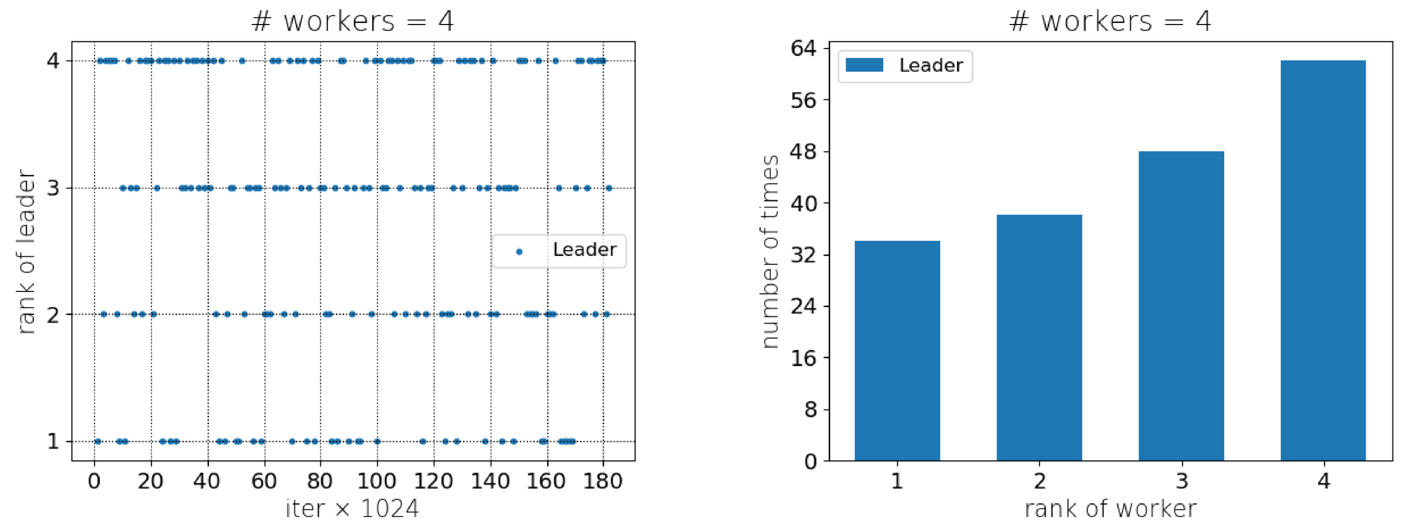


Figure: ResNet20 on CIFAR-10. The identity of the worker that is recognized as the leader (i.e. rank) versus iterations (on the left) and the number of times each worker was the leader (on the right).

References

[1] S. Zhang, A. Choromanska, and Y. LeCun. Deep learning with elastic averaging SGD. In *NIPS*, 2015.