Identify and Sparsify: Distributed Optimization with Asynchronous Moderate Communications

D. GRISHCHENKO[1,2], F. IUTZELER[1], J. MALICK[1] and M.-R. AMINI[2]

1. DAO team, LJK, Université Grenoble Alpes
2. AMA team, LIG, Université Grenoble Alpes

Correspondence: name.lastname@univ-grenoble-alpes.fr

Overview

Rule Forty-two. All persons more than a mile high to leave the court.

Context

- **Optimization algorithms:** find minimizer of convex functions
- **Distributed setting:** several machines, without shared data
- **Communications between machines:** bottleneck

Results

We present a **distributed** version of **proximal gradient decent** with constant stepsize and **two-way sparse communications** with **linear convergence**.

Model

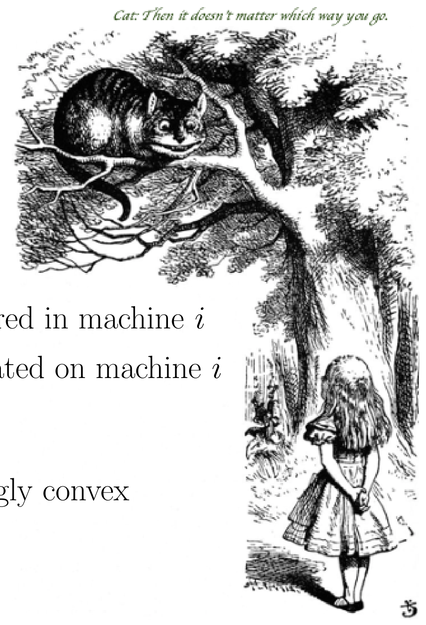
- **Distributed learning:**

  n observations are split over M machines

  machine i has a private examples subset $\mathcal{S}_i$ $\sum \mathcal{S}_i = \mathcal{S}$ – full set of examples

- **Shared prediction without moving data:**
  decoupling the ability to learn from
  the need to store the data in a centralized way.

Cat: Where are you going?
Alice: Which way should I go?
Cat: That depends on where you are going.
Alice: I don't know.
Cat: Then it doesn't matter which way you go.



Problem: $$\min_{x\in\mathbb{R}^d} \quad \sum_{i=1}^{M} \pi_i f_i(x) + r(x)$$

$\pi_i = n_i/n$ the proportion of observations locally stored in machine i

$f_i(x) = \frac{1}{n_i}\sum_{j\in\mathcal{S}_i} \ell_j(x)$ the local empirical risk estimated on machine i

Assumptions

- **On functions:** all f_i are $L-$smooth and $\mu-$strongly convex
- **On regularizer:** r is convex and l.s.c.

  $x^\star$ – unique minimizer

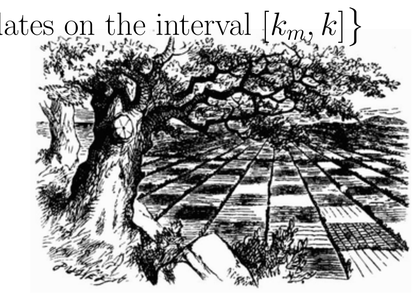
Notations

Alice: How long is forever? White Rabbit: Sometimes, just one second.

- **For the master:**

  k = number of updates the master receives from any of the slaves

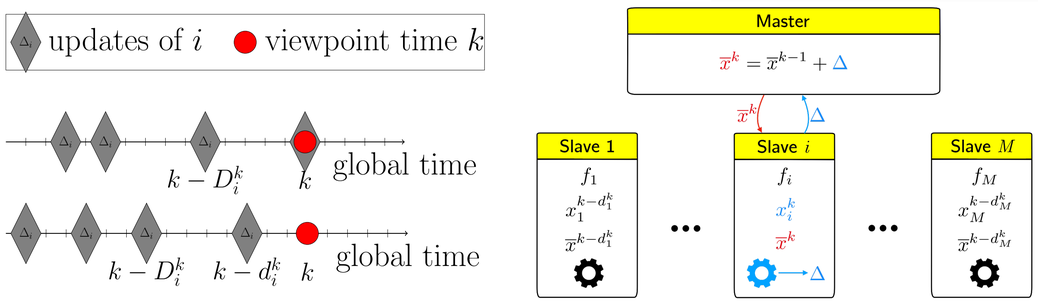
  $k_{m+1} = \min\left\{k : \text{each machine made at least 2 updates on the interval } [k_m, k]\right\}$



- **For slave i:**

  d_i^k = time elapsed from the last update

  D_i^k = time of the penultimate update



Sparsification of local updates

Round the neck of the bottle was a paper label, with the words 'DRINK ME' beautifully printed on it in large letters.

Master machine asynchronously gathers *sparsified* delayed gradient updates from slaves and sends them back the current point. At iteration k, this randomly drawn subset of entries of the gradient to be computed by agent i^k is called **mask** and is denoted by $\mathbf{S}^k$. Using $x_{[j]}$ is the j-th coordinate of $x \in \mathbb{R}^d$

$$x^k_{i[j]} = \begin{cases} \left(x^{k-D_i^k} - \gamma\nabla f_i(x^{k-D_i^k})\right)_{[j]} & \text{if } i = i^k \text{ and } j \in \mathbf{S}^{k-D_i^k} \\ x^{k-1}_{i[j]} & \text{otherwise} \end{cases}$$

$$x^k = \text{prox}_{\gamma r}\underbrace{\left(\sum_{i=1}^{M} \pi_i x_i^k\right)}_{:=\bar{x}^k} = \arg\min_z \left\{r(z) + \frac{1}{2\gamma}\|x - \bar{x}^k\|^2\right\}$$

Assumption on sparsification

The sparsity mask selectors $(\mathbf{S}^k)$ are the only random variables:

$\mathbb{P}[j \in \mathbf{S}^k] = 1$ if $j \in \text{supp}(x^k)$

$\mathbb{P}[j' \in \mathbf{S}^k] = p > 0$ for all $j' \notin \text{supp}(x^k)$

Delays $(D_i^k)_{i=1,..,M}$ are independent of the future mask selectors $\{\mathbf{S}^\ell\}_{\ell\geq k}$.

Master	**Slave i**
Initialize $\bar{x}^0$	Initialize $x_i = x_i^+ = x = \bar{x}^0$
while *not converged* **do**	**while** *not interrupted by master* **do**
Receive $[\Delta^k]_{\mathbf{S}^{k-D_i^k}}$ from agent $i = i^k$	$[x^+]_{\mathbf{S}\setminus\text{supp}(x)} \leftarrow [x - \gamma\nabla f_i(x)]_{\mathbf{S}\setminus\text{supp}(x)}$
$\bar{x}^k \leftarrow \bar{x}^{k-1} + \pi_i[\Delta^k]_{\mathbf{S}^{k-D_i^k}}$	$[x^+]_{\text{supp}(x)} \leftarrow p[x - \gamma\nabla f_i(x))]_{\text{supp}(x)}$
$x^k \leftarrow \text{prox}_{\gamma r}(\bar{x}^k)$	$\quad + (1-p)[x_i]_{\text{supp}(x)}$
Choose sparsity mask $\mathbf{S}^k$	$\Delta \leftarrow x^+ - x$
Send $x^k, \mathbf{S}^k$ to agent $i = i^k$	Send $[\Delta]_{\mathbf{S}}$ to master
	$[x_i]_{\mathbf{S}} \leftarrow [x_i^+]_{\mathbf{S}}$
	Receive x and $\mathbf{S}$ from master
end	**end**

Convergence rate

Alice could not even get her head though the doorway; 'and even if my head would go through,' thought poor Alice, 'it would be of very little use without my shoulders.

Take $\gamma \in (0, 2/(\mu + L)]$. Then, for all $k \in [k_m, k_{m+1})$,

$$\mathbb{E}\|x^k - x^\star\|^2 \leq \left(1 - 2\frac{\gamma p\mu L}{\mu + L}\right)^m \max_{i=1,..,M} \|x_i^0 - x_i^\star\|^2$$

- Linear convergence
- Same step-size as in vanilla proximal gradient
- If $M = 1$ and $p = 1$ – usual convergence rate

Identification

This time there could be NO mistake about it: it was neither more nor less than a pig, and she felt that it would be quite absurd for her to carry it further.

Assumptions

- **On regularizer:**
  $r(x) = \lambda_1\|x\|_1$, then $\text{prox}_{\gamma r}(x)$ is the soft-treshholding operator.
- **On delays:**
  The number of iterations between two full updates cannot grow exponentially, i.e. $k_{m+1} - k_m = o(\exp(m))$. This assumption is rather mild and subsumes the usual bounded delay assumption.

Identification result The algorithm identifies a near-optimal support in finite time with probability one:

$$\exists K : \forall k \geq K, \quad \text{supp}(x^\star) \subseteq \text{supp}(x^k) \subseteq \text{supp}(y_\varepsilon^\star)$$

where $y_\varepsilon^\star = \text{prox}_{\gamma(1-\varepsilon)r}(\bar{x}^\star - x^\star)$ for any $\varepsilon > 0$. Furthermore, if the problem is non-degenerate, i.e. $-\sum_{i=1}^M \pi_i\nabla f_i(x^\star) \in \text{ri}\ \partial r(x^\star)$ then, the algorithm identifies the optimal support with probability one:

$$\exists K : \forall k \geq K, \quad \text{supp}(x^k) = \text{supp}(x^\star)$$

Sparsity This identification result gives us **two-way sparsity** of algorithm in terms of communications.

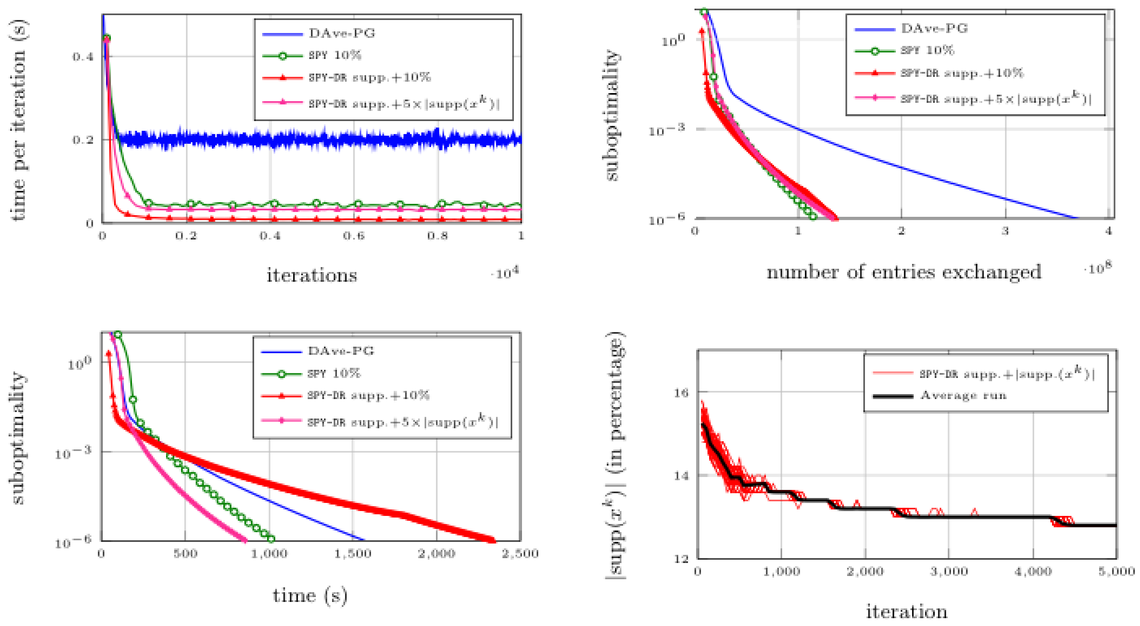


Figure 1: Logistic regression for the rcv1 dataset: evolution of the time per iteration, wallclock time performance, suboptimality vs communication, and robustness of identification.

References

But I don't want to go among mad people, Alice remarked. Oh, you can't help that, said the Cat: we're all mad here. I'm mad. You're mad. How do you know I'm mad?said Alice. You must be, said the Cat, or you wouldn't have come here.

1. Grishchenko, D., Iutzeler, F., Malick, J., & Amini, M. R. *Asynchronous Distributed Learning with Sparse Communications and Identification.* arXiv preprint arXiv:1812.03871.
2. Jalal Fadili, Jérôme Malick, & Gabriel Peyré. *Sensitivity analysis for mirror-stratifiable convex functions.* SIAM Journal on Optimization, 2018.
3. Mishchenko, K., Iutzeler, F., Malick, J., & Amini, M. R. *A delay-tolerant proximal-gradient algorithm for distributed learning.* International Conference on Machine Learning (pp. 3584-3592), 2018.
4. Rémi Leblond, Fabian Pedregosa, & Simon Lacoste-Julien. *Asaga: Asynchronous parallel saga.* AISTATS 2017.