Distributed First-Order Optimization with Tamed Communications

Dmitry Grishchenko[1,2] Franck IUTZELER[1,2] Jérôme MALICK[2,3]
[1]Université Grenoble Alpes [2]LJK [3]CNRS

LABORATOIRE JEAN KUNTZMANN

UNIVERSITÉ Grenoble Alpes

Problem

$$\min_{x\in\mathbb{R}^n} \sum_{i=1}^{M} \pi_i f_i(x) + g(x)$$

amount of machines — M

convex, proper, l.s.c. — g

$x^\star$ - unique minimizer

BIG — n

proportion on i^{th} machine — π_i

L - smooth, μ - s. convex — f_i

Motivation

Intuition

proximal method

after — unknown — finite — moment of time

enlarged identification

subspace descent

$x^\star \in \mathcal{R} \subseteq \mathbb{R}^d$, $d \ll n$

random subspace selection

- structure dependent
- iterate dependent
- speed-up after identification

"place" when sparsity happens

Subspace families and projections

The family of linear subspaces $\mathscr{C} = \{\mathscr{C}_i\}_i$ is called covering if $\sum_i \mathscr{C}_i = \mathbb{R}^n$

The random generation of a subspace by selecting randomly some subspaces in some covering family is called admissible selection if it samples whole space.

$$P_{\mathfrak{S}} = \begin{array}{c} \begin{matrix} \overbrace{\hphantom{\tfrac{1}{n_1}\ \cdots\ \tfrac{1}{n_1}}}^{n_1} & \hphantom{0\ \cdots} & \overbrace{\hphantom{\tfrac{1}{n-n_s}\ \cdots\ \tfrac{1}{n-n_s}}}^{n-n_s} \end{matrix} \\ \begin{pmatrix} \frac{1}{n_1} & \cdots & \frac{1}{n_1} & 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \frac{1}{n_1} & \cdots & \frac{1}{n_1} & 0 & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 & \frac{1}{n-n_s} & \cdots & \frac{1}{n-n_s} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 & \frac{1}{n-n_s} & \cdots & \frac{1}{n-n_s} \end{pmatrix} \end{array} \begin{array}{l} \left.\vphantom{\begin{matrix}a\\a\\a\end{matrix}}\right\} n_1 \\ \\ \\ \left.\vphantom{\begin{matrix}a\\a\\a\end{matrix}}\right\} n-n_s \end{array}$$

$P_{\mathscr{F}}$ is an orthogonal projection onto linear subspace $\mathscr{F}$

If $\mathfrak{S}$ is admissible selection, then $\mathsf{P} := \mathbb{E}\left[P_{\mathfrak{S}}\right]$ (average projection) is invertible.

Adaptive Distributed Randomized Proximal Subspace Descent - ADRPSD

```
[M] Generate the first admissible selection 𝔖^0, compute
[M] Initialize z^0, x^1 = prox_{γg}(Q_0^{-1}(z^0)), ℓ = 0, L = {0}.
[W_i] Receive 𝔖^0 from master                                   [SPARSE]
[M, W_i] Compute P_0 = E[P_{i,𝔖^0}], Q_0 = P_0^{-1/2}, and Q_0^{-1}
for k = 1,... in parallel do
[W_i] Receive x^k from master                       [SPARSE for some g]
[W_i] Select independently P_{i,𝔖^ℓ}
[W_i] y_i^k = Q_ℓ (x^k − γ∇f_i(x^k))
[W_i] Send P_{i,𝔖^k}(y_i^k) to master                           [SPARSE]
[M] z^k = Σ_{i=1}^M π_i (P_{i,𝔖^k}(y_i^k) + (I − P_{i,𝔖^k})(z^{k−1}))
[M] x^{k+1} = prox_{γg}(Q_ℓ^{-1}(z^k))
if an adaptation is decided then
[M] L ← L ∪ {k+1}, ℓ ← ℓ+1
[M] Generate a new admissible selection 𝔖^l
[W_i] Receive 𝔖^ℓ from master                                   [SPARSE]
[M, W_i] Compute P_ℓ = E[P_{i,𝔖^ℓ}], Q_ℓ = P_ℓ^{-1/2}, and Q_ℓ^{-1}
[M] Rescale z^k ← Q_ℓ Q_{ℓ−1}^{-1} z^k
end if
end for
```

Adaptive selection

structure — iterate

Update the deterministic part of selection basing on the sparsity pattern of the current iterate with respect to the geometric structure of g

example of

projection for 1d-TV

ADRPSD convergence For any $\gamma \in (0, 2/(\mu+L)]$, let the user choose its adaptation strategy so that:

- the *adaptation cost* is upper bounded by a deterministic sequence: $\|\mathsf{Q}_\ell \mathsf{Q}_{\ell-1}^{-1}\|_2^2 \le \mathbf{a}_\ell$;
- the *inter-adaptation time* is lower bounded by a deterministic sequence: $k_\ell - k_{\ell-1} \ge \mathbf{c}_\ell$;
- the *selection uniformity* is lower bounded by a deterministic sequence: $\lambda_{\min}(\mathsf{P}_\ell) \ge \lambda_\ell$;

then, from the *previous instantaneous rate* $1-\alpha_{\ell-1} := 1 - 2\gamma\mu L\lambda_{\ell-1}/(\mu+L)$, the *corrected rate* for cycle ℓ writes

$$(1-\beta_\ell) := (1-\alpha_{\ell-1})\mathbf{a}_\ell^{1/\mathbf{c}_\ell}.$$

Then, we have for any $k \in [k_\ell, k_{\ell+1})$

$$\mathbb{E}\left[\|x^{k+1} - x^\star\|_2^2\right] \le (1-\alpha_\ell)^{k-k_\ell} \prod_{m=1}^{\ell} (1-\beta_m)^{\mathbf{c}_m} \|z^0 - \mathsf{Q}_0\left(x^\star - \gamma\nabla f(x^\star)\right)\|_2^2.$$

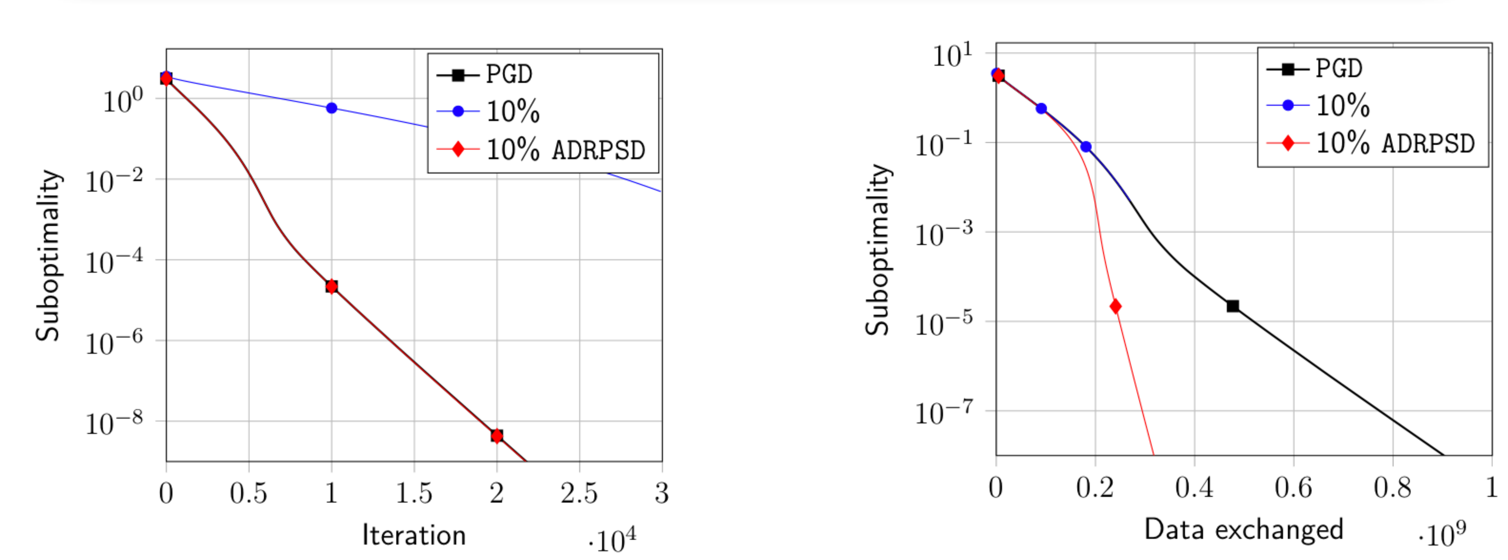

Figure 1: ℓ_1 regularized logistic regression on rcv_1 dataset

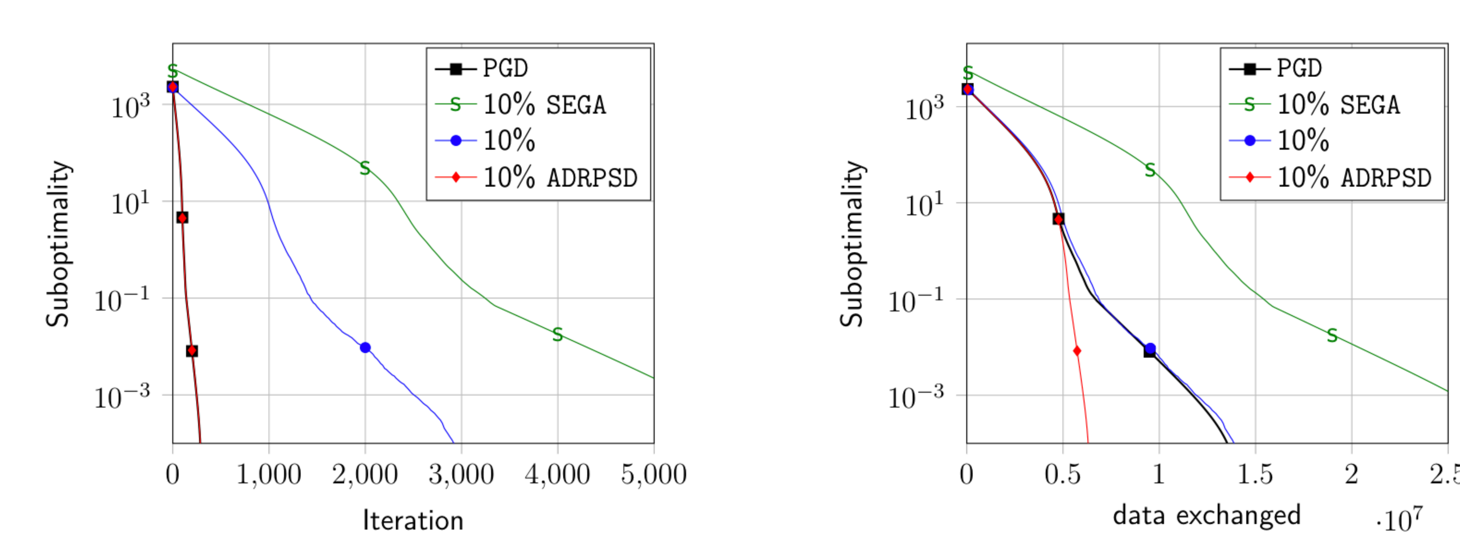

Figure 2: $\ell_{1,2}$ regularized logistic regression on rcv_1 dataset

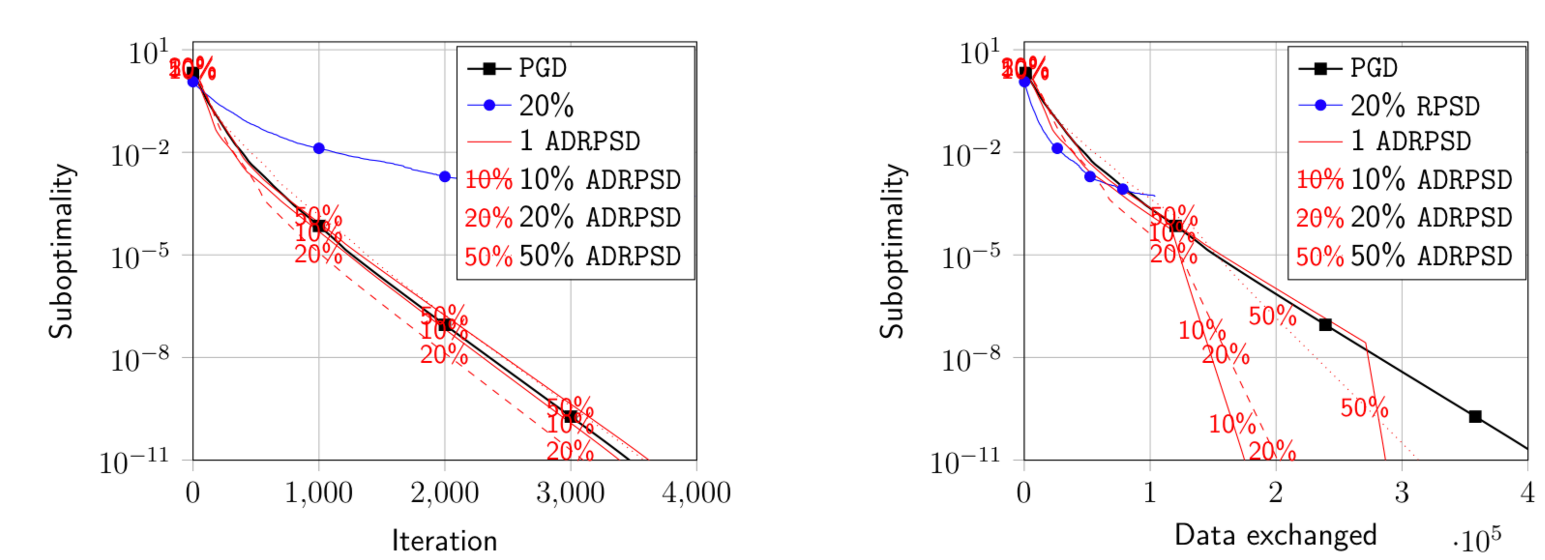

Figure 3: TV regularized logistic regression on a1a datase

[1] Bach, F., Jenatton, R., Mairal, J., Obozinski, G.: Optimization with sparsity-inducing penalties. Foundations and Trends® in Machine Learning (2012)
[2] Fadili, J., Malick, J., Peyré, G.: Sensitivity analysis for mirror-stratifiable convex functions. SIAM Journal on Optimization (2018)
[3] Grishchenko, D., et al.: Asynchronous distributed learning with sparse communications and identification. arXiv preprint arXiv:1812.03871 (2018)
[4] Hanzely, F., Mishchenko, K., Richtárik, P.: Sega: Variance reduction via gradient sketching. In: Advances in NeurIPS (2018)