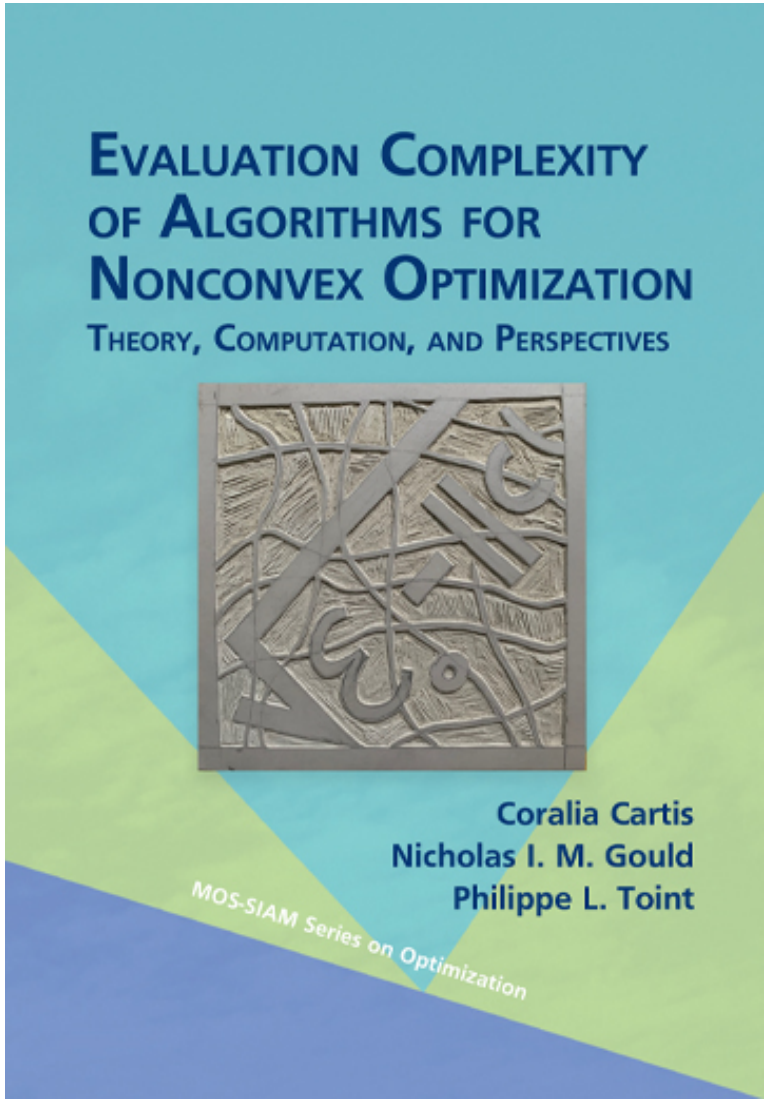
Objective-function-free optimization

Serge Gratton, Sadok Jerad,
Alena Kopaničáková and Philippe Toint

INP - ANITI / UNamur

Louvain-La-Neuve, August 2024

First: a brief publicity break :-)



The problem

Once more, the standard unconstrained nonconvex optimization problem

$$\min_{x \in \mathbb{R}^n} f(x)$$

where the objective function f is

- "sufficiently" smooth
- bounded below

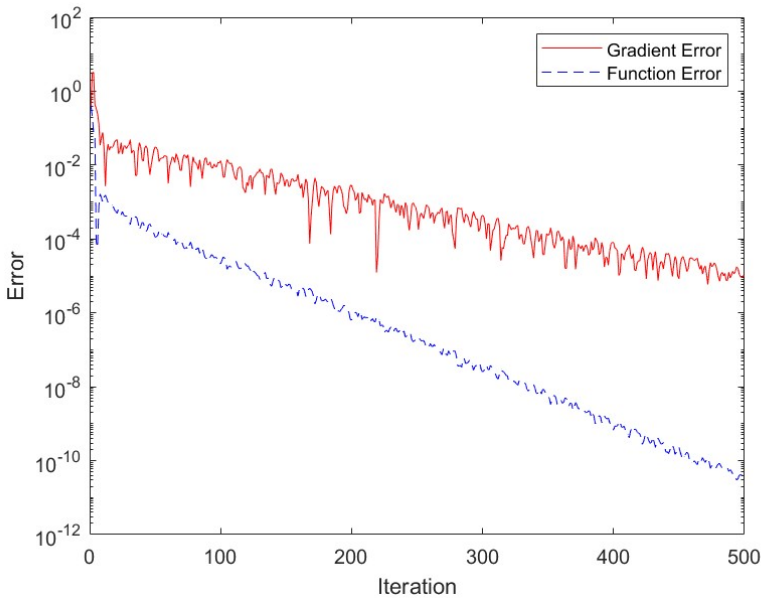
Remarkable one can still say (hopefully) interesting things on this subject!

In this talk: focus on the deterministic case, but ...

Why OFFO?

Our target: robust algorithms for noisy functions/inexact arithmetic

For convergence, standard methods (TR, AR) requires an error on function values which is the square (!) of that on the gradient (e.g. Bellavia et al, 22)



$\Rightarrow$ Design algorithms that

do not evaluate the function

Adaptive gradient methods:

- Adagrad (Duchi et al, 2011)
- WNGrad (Wu, Ward, Bottou, 2018)
- Adam (Kingma, Ba, 2014)

A trust-region method:

- Adatr (Grapiglia, 2022)

$\Rightarrow$ Objective Function Free Optimization = OFFO

ASTR1 an adaptive trust-region algorithm

Step 0: Initialization. x_0 is given. Set $k = 0$.

Step 1: Define the TR. Compute $g_k = g(x_k)$ and define

$$\Delta_{i,k} = \frac{|g_{i,k}|}{w_{i,k}}$$

where $w_{i,k} \geq \varsigma_i > 0$ are weights.

Step 2: Hessian approximation. Select a symmetric B_k.

Step 3: GCP. Define

$$s^L_{i,k} = -\mathrm{sgn}(g_{i,k})\Delta_{i,k} \quad \text{and} \quad s^Q_k = \gamma_k s^L_k$$

with

$$\gamma_k = \begin{cases} \min\left[1, \dfrac{|g_k^T s^L_k|}{(s^L_k)^T B_k s^L_k}\right] & \text{if} \quad (s^L_k)^T B_k s^L_k > 0, \\ 1 & \text{otherwise.} \end{cases}$$

Step 3: Step. Compute a step s_k such that $|s_{i,k}| \leq \Delta_{i,k}$ $(\forall i)$ and

$$g_k^T s_k + \tfrac{1}{2} s_k^T B_k s_k \leq g_k^T s^Q_k + \tfrac{1}{2}(s^Q_k)^T B_k s^Q_k$$

Step 5: New iterate. Set $x_{k+1} = x_k + s_k$, increment k, and go to Step 1.

ASTR1: comments

- the objective function is not evaluated $\Rightarrow$ OFFO . . . and thus the TR radius cannot depend on ared/prered.
- large weights $\Rightarrow$ short steps
- γ_k minimize the quadratic model between 0 and s_k^L

Suppose that $f \in C^1$, has Lipschitz gradient with constant L and that $\|B_k\| \leq \kappa_B$. Then

$$f(x_{k+1}) \leq f(x_k) - \sum_{i=1}^{n} \frac{\varsigma_{\min} g_{i,j}^2}{2\kappa_B w_{i,j}} + \tfrac{1}{2}(\kappa_B + L) \sum_{i=1}^{n} \frac{g_{i,j}^2}{w_{i,j}^2}$$

$\Rightarrow$ descent for large enough weights $w_{i,k}$

ASTR1 with ADAGRAD-like weights (1)

For given $\varsigma \in (0,1]$, $\vartheta \in (0,1]$ and $\mu \in (0,1)$, define

$$w_{i,k} \in \left[\vartheta \left(\varsigma + \sum_{\ell=0}^{k} g_{i,\ell}^2 \right)^{\mu}, \left(\varsigma + \sum_{\ell=0}^{k} g_{i,\ell}^2 \right)^{\mu} \right]$$

For $\vartheta = 1$ and $\mu = \frac{1}{2}$, $w_{i,k} = \sqrt{\varsigma + \sum_{\ell=0}^{k} g_{i,\ell}^2}$ and

ASTR1 with $\vartheta = 1$, $\mu = \frac{1}{2}$ and $B_k = 0$ is ADAGRAD

Suppose that $f \in C^1$, has Lipschitz gradient with constant L and is bounded below. Then ASTR1 with ADAGRAD-like weights, $\mu \in (0,1]$ and $\|B_k\|$ uniformly bounded requires at most

$$\mathcal{O}\left(\epsilon^{-1}\right)$$

iterations to produce an iterate k such that $\text{average}_{0,\ldots,k} \|g_\ell\|^2 \leq \epsilon$.

More on ASTR1

- Extends known results (e.g., Wu, Ward, Bottou, 2018)
- Allows the use of curvature information in an ADAGRAD-like method (Barzilai-Borwein, LBFGS, quasi-Newton, . . . true Hessian)
- The above bound is essentially sharp.

Also possible with the "divergent" weights

$$w_{i,k} \in [v_{i,k}(k+1)^{\nu}, v_{i,k}(k+1)^{\mu}]$$

for $0 < \nu \leq \mu < 1$ and

$$v_{i,k} = \max_{0,\ldots,k} |g_{i,\ell}| \quad \text{or} \quad v_{i,k} = \operatorname*{average}_{0,\ldots,k} |g_{i,\ell}|$$

Slightly weaker (sharp) complexity result

Some results on the small noiseless OPM problems

Method	$\pi_{\texttt{algo}}$	$\rho_{\texttt{algo}}$
`adagbfgs3`	0.75	69.75
`sdba` (using f)	0.73	68.91
`adagH`	0.72	69.75
`adagrad`	0.69	73.11
`maxg`	0.66	66.39
`adagbb`	0.63	64.71
`adam`	0.54	30.25

Performance and reliability statistics for deterministic OFFO and steepest descent algorithms on small OPM problems ($\epsilon = 10^{-6}$)

The impact of noise

algo	$\rho_{\texttt{algo}}$/relative noise level				
	0%	5%	15%	25%	50%
adagH	83.19	84.96	84.20	84.71	82.18
adagbfgs3	78.15	80.50	80.50	80.84	80.18
adagrad	77.31	80.50	80.25	80.17	80.17
adagbb	75.69	80.08	80.17	79.58	79.41
maxg	74.79	74.37	75.55	78.15	78.07
adam	40.34	35.55	36.30	44.03	45.80
sdba	81.51	30.92	31.85	34.87	29.58

Reliability of OFFO algorithms and steepest descent as a function of the level of relative Gaussian noise ($\epsilon = 10^{-3}$)

OFFO and multilevel optimization: context

Statistical machine learning for solving PDE's

Key ingredients:

- an approximation set, typically a nonlinear Neural Network (NN) architecture
- a sampling technique of the space and time domains
- "training" = minimization of a loss encoding the PDE (or the underlying physics) $\Rightarrow$ Physically Informed Neural Networks (PINNs) See [Raissi, Perdikaris, Karniadakis, 2017]

Single level convergence theory exists (e.g. [Shin, Darbon, Karniadakis, 2020]) and involves

- the universality property of NN,
- statistical sampling,
- ability of numerical optimizers (ADAM, SGD,...) to reach an approximate global optimum of nonconvex function

OFFO and multilevel optimization: idea(s) (1)

New algorithm:

- assume a hierarchy of (smooth) models $h_\ell(x)$ (from fine to coarse)
- ensure first-order model coherence between levels by adding a linear term if necessary (cfr FAS))
- an OFFO trust-region based algorithm at each level ℓ: (approx) minimize a quadratic model

$$g_{\ell,k}^T s + \tfrac{1}{2} s^T H_{\ell,k} s$$

in $\mathcal{B}_{\ell,k} = \{s \text{ at level } \ell \ \ \|s\| \leq \Delta_{\ell,k}\}$

- (no evaluation of $f(x) \Rightarrow$ accept all iterates)

OFFO and multilevel optimization: idea(s) (2)

- at each level ℓ compute $s_{\ell,k}$ by
  - either using a Taylor approximation of the model at level ℓ (standard OFFO step), or
  - use the OFFO algorithm to minimize the lower level model $h_{\ell-1}$ with the TR at level ℓ (recursivity)
- only use lower level if lower-level gradient is not too small wrt upper-level one
- when using lower level, ensure the trust-region radius has a suitable size (a bit technical)

OFFO and multilevel optimization: algorithm and results

Note:

- Detailed algorithm does not fit on slide, but...
- Subsumes most existing (first-order) OFFO methods in the single level case
- allows use of second-order information (LBFGS,... or even true Hessian)

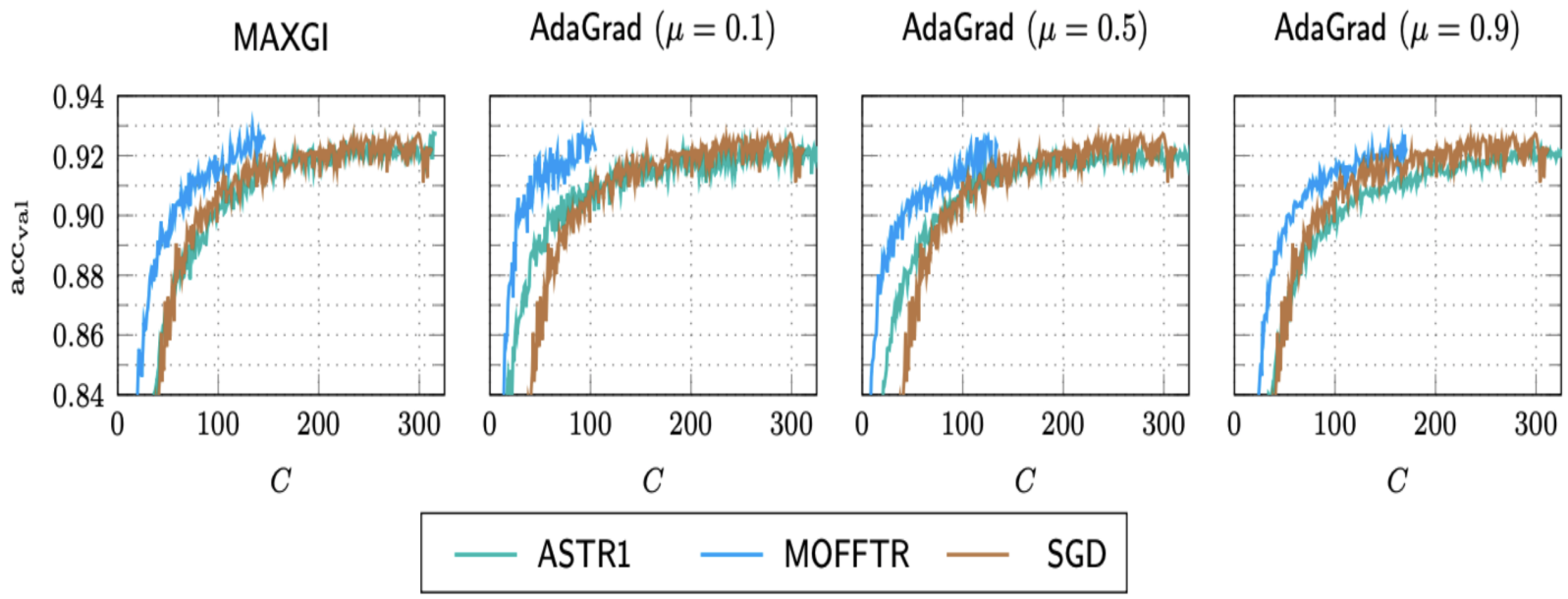
Result:

- complexity is $\mathcal{O}(\epsilon^{-2})$ (optimal for single level case)
- first-order version works well in experiments (stochastic context, momentum, resnets, ... see (Gratton, Kopaničáková,T., 2023))

More can be said about PINN's (Gratton, Mercier, Riccetti, T., 2024)

An example

Image classification using SVNH dataset[Netzer et al. '11]



Towards second-order criticality

Use a trust-region mechanism for second-order criticality?

At x_k, let

$$T_{f,2}(x_k, d) = f(x_k) + g(x)_k^T d + \tfrac{1}{2} d^T H(x_k) d.$$

and the second-order criticality measure

$$\phi_{f,2}^{\delta}(x_k) = \max_{\|d\| \leq \delta} -\left(g(x_k)^T d + \tfrac{1}{2} d^T H(x_k) d \right) = \max_{\|d\| \leq \delta} \Delta q_k(d)$$

Define:

x_k is ϵ-second-order critical if $\phi_{f,2}^{\delta}(x_k) \leq \epsilon$

Idea: Use $\phi_{f,2}^{\delta}(x_k)$ to define weights for the trust-region

ASTR2: a TR OFFO method for 2nd-order criticality

Step 0: Initialization. Given: x_0 and algo constants. Set $k = 0$.

Step 1: Compute derivatives. Compute g_k and H_k, as well as ϕ_k and $\widehat{\phi}_k \stackrel{\text{def}}{=} \min[\phi^{\delta}_{f,2}(x_k), \kappa]$.

Step 2: Define the TR radii. For weights w_k^L and w_k^Q, set

$$\Delta_k^L = \frac{\|g_k\|}{w_k^L} \quad \text{and} \quad \Delta_k^Q = \frac{\widehat{\phi}_k}{w_k^Q}.$$

Step 3: Step computation. If $\|g_k\|^2 \geq \widehat{\phi}_k^3$, set $s_k = -g_k / w_k^L$. Otherwise, set s_k such that

$$\|s_k\| \leq \Delta_k^Q \quad \text{and} \quad \Delta q_k(s_k) \geq \max\left[\Delta q_k^C, \Delta q_k^E\right]$$

where $\Delta q_k^C = \max_{\alpha \geq 0, \alpha\|g_k\| \leq \Delta_k^Q} \Delta q_k(-\alpha g_k)$ and

$$\Delta q_k^E = \begin{cases} \max_{\alpha \geq 0, \alpha \leq \Delta_k^Q} \Delta q_k(\alpha u_k) & \text{if } \lambda_{\min}[H_k] < 0 \\ 0 & \text{if } \lambda_{\min}[H_k] \geq 0 \end{cases}$$

with

$$u_k^T H_k u_k \leq \kappa \lambda_{\min}[H_k], \quad u_k^T g_k \leq 0 \quad \text{and} \quad \|u_k\| = 1,$$

Step 4: New iterate. Define $x_{k+1} = x_k + s_k$, increment k and return to Step 1.

Function decrease for ASTR2

Suppose that $f \in C^2$ and has Lipschitz continuous gradient and Hessian. Then, if $\|g_k\|^2 \geq \widehat{\phi}_k^3$,

$$f_{k+1} \leq f_k - \frac{\|g_k\|^2}{w_k^L} + \frac{L_1}{2} \frac{\|g_k\|^2}{(w_k^L)^2}$$

while, if $\|g_k\|^2 < \widehat{\phi}_k^3$,

$$f_{k+1} \leq f_k - \kappa \min \left[\frac{1}{2(1+L_1)}, \frac{1}{w_k^Q}, \frac{1}{(w_k^Q)^2} \right] \widehat{\phi}_k^3 + \frac{L_2}{6} \frac{\widehat{\phi}_k^3}{(w_k^Q)^3}.$$

$\Rightarrow$ roles of w_k^L and w_k^Q complementary

Complexity of ASTR2 for ADAGRAD-like weights

When using

$$w_k^L \in \left[\vartheta \left(\varsigma + \sum_{\ell=0, \ell \in \mathcal{K}^L}^{k} \|g_\ell\|^2 \right)^\mu, \left(\varsigma + \sum_{\ell=0, \ell \in \mathcal{K}^L}^{k} \|g_\ell\|^2 \right)^\mu \right]$$

$$w_k^Q \in \left[\vartheta \left(\varsigma + \sum_{\ell=0, \ell \in \mathcal{K}^Q}^{k} \widehat{\phi}_k^3 \right)^\mu, \left(\varsigma + \sum_{\ell=0, \ell \in \mathcal{K}^Q}^{k} \widehat{\phi}_k^3 \right)^\mu \right]$$

Suppose that $f \in C^2$ with Lipschitz gradient and Hessian and is bounded below. Then ASTR2 with the above weights and $\mu \in (0, 1]$ requires at most $\mathcal{O}\left(\epsilon^{-1}\right)$ iterations to produce an iterate k such that $\text{average}_{0,\ldots,k} \|g_\ell\|^2 \leq \epsilon$ and $\text{average}_{0,\ldots,k} \widehat{\phi}_\ell^3 \leq \epsilon$. [Essentially sharp!]

. . . and now for an OFFO regularization algorithm!

Consider now the more general

$$T_{f,p}(x,s) = f(x) + \sum_{i=1}^{p} \frac{1}{i!} \nabla_x^i f(x)[s]^i.$$

and the derived regularized model

$$m_k(s) = T_{f,p}(x_k,s) + \frac{\sigma_k}{(p+1)!}\|s\|^{p+1}$$

We assume that $\nabla_x^p f$ is globally Lipschitz.

The OFFAR algorithm

(again using generic κ)

Step 0: Initialization: x_0, $\nu_0 > 0$, ϵ and constants. Set $k = 0$.

Step 1: Check for termination: Evaluate $g_k = \nabla_x^1 f(x_k)$ and terminate if $\|g_k\| \leq \epsilon$. Else, evaluate $\{\nabla_x^i f(x_k)\}_{i=2}^p$.

Step 2: Step calculation: If $k = 0$, set $\sigma_0 = \mu_0 = \nu_0$. Else set

$$\mu_k = \frac{p!\|g_k\|}{\|s_{k-1}\|^p} - \kappa\sigma_{k-1} \quad \text{and} \quad \sigma_k \in [\kappa\nu_k, \max(\nu_k, \mu_k)].$$

Then compute a step s_k such that

$$m_k(s_k) < m_k(0) \quad \text{and} \quad \|\nabla_s^1 T_{f,p}(x_k, s_k)\| \leq \kappa\frac{\sigma_k}{p!}\|s_k\|^p.$$

Step 3: Updates. Set $x_{k+1} = x_k + s_k$ and $\nu_{k+1} = \nu_k + \nu_k\|s_k\|^{p+1}$. Increment k by one and go to Step 1.

Complexity of OFFAR

- No objective function evaluation $\Rightarrow$ OFFO
- The use of μ_k is optional: one could simply set $\mu_k = 0$ without altering the theory. But it is important for performance.
- The definition of μ_k promotes fast growth of the regularization parameter up the problem's Lispchitz constant
- The definition of σ_k helps to limit this growth once the value of the Lipschitz constant has been reached.
- If $p = 1$, $\nu_{k+1} = \nu_k + \nu_k \|s_k\|^2$, recovering WNGrad (Wu, Ward, Bottou, 2018)

Suppose that $f \in C^p$ with $\nabla_x^p f$ Lipschitz gradient, is bounded below and is such that $\min_{\|d\| \leq 1} \nabla_x^i [d]^i \geq \kappa$ for $i = 2, \ldots, p$. Then OFFAR (with suitable constants) requires at most $\mathcal{O}\left(\epsilon^{-\frac{p+1}{p}}\right)$ iterations to produce an iterate k such that $\|g_k\| \leq \epsilon$.

More on OFFAR

- Same rate as ARp using function values (Birgin et al, 2016)
- For $p = 2$, same rate as ARC/AR2 (Cartis, Gould, T. 2011). Optimal rate for second order methods
- Optimal rates for exact pth order methods (Carmon et al. 2019).

MOFFAR: If one requires that the step also satisfies

$$\max\left(0, -\lambda_{\min}[\nabla_s^2 T_{f,p}(x_k, s_k)]\right) \leq \frac{\kappa\sigma_k}{(p-1)!}\|s_k\|^{p-1}$$

Suppose that $f \in C^p$ with $\nabla_x^p f$ Lipschitz gradient, is bounded below and is such that $\min_{\|d\|\leq 1} \nabla_x^i [d]^i \geq \kappa$ for $i = 2, \ldots, p$. Then MOFFAR (with suitable constants) requires at most $\mathcal{O}\left(\epsilon^{-\frac{p+1}{p-1}}\right)$ iterations to produce an iterate k such that $\|g_k\| \leq \epsilon$ and $\widehat{\phi}_k \leq \epsilon$.

Numerical illustration

For AR2 and two variants of OFFAR with $p = 2$, differing on how aggressively μ_k forces growth in σ_k (b more aggressive than a)

	AR2	OFFAR2a	OFFAR2b
$\pi_{\texttt{algo}}$	0.99	0.78	0.83
$\rho_{\texttt{algo}}$	97.48	81.51	88.24

Performance and reliability statistics on the small OPM problems without noise

	5%	15%	25%	50%
AR2	40.67	30.84	24.54	6.81
OFFAR2a	80.76	75.38	70.76	56.30
OFFAR2b	85.97	80.67	72.69	47.98

Reliability statistics $\rho_{\texttt{algo}}$ for 5%, 15%, 25% and 50% relative random Gaussian noise (averaged on 10 runs)

Stochastic variants

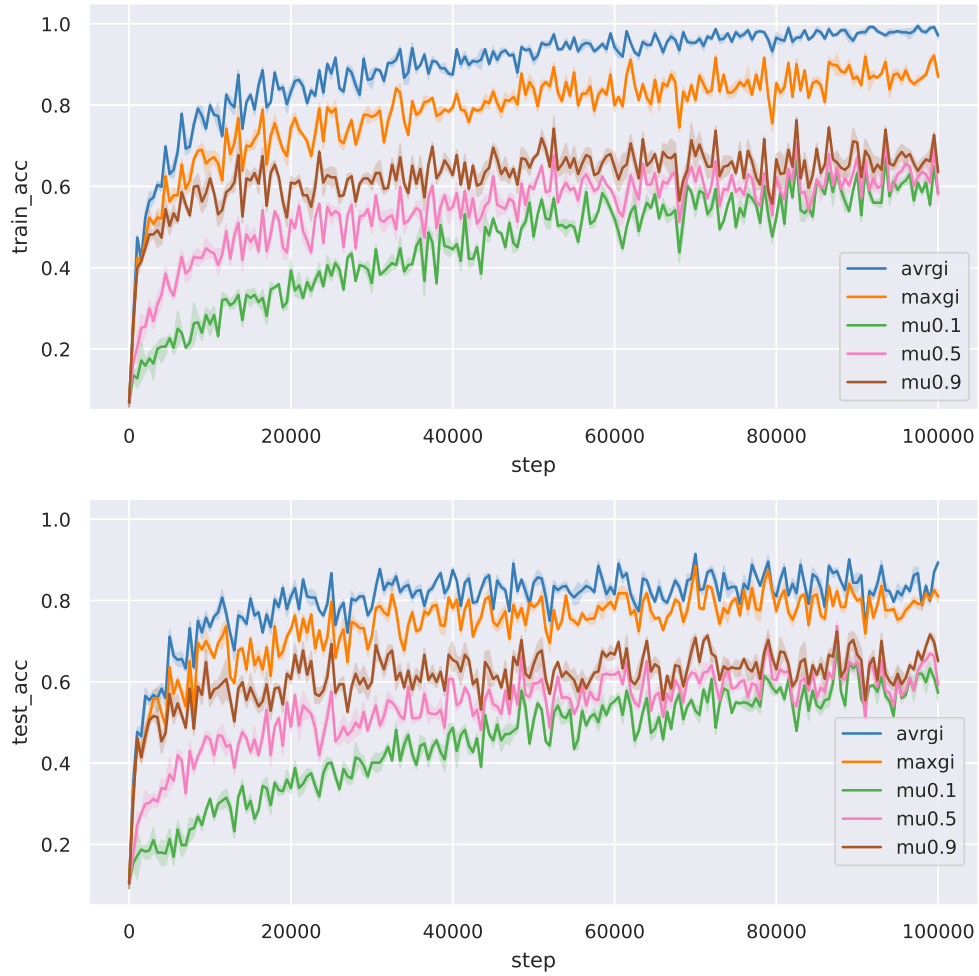
Complexity bounds for first-order criticality (in expectation):

Algorithm	type	Compl.bound
ASTR1	trust-region	$\mathcal{O}\left(\epsilon^{-\frac{1}{2}(1-\mu)}\right)$
ASTR2	trust-region	??
STOFFAR	adat. regularization	$\mathcal{O}(\epsilon^{-3/2})$

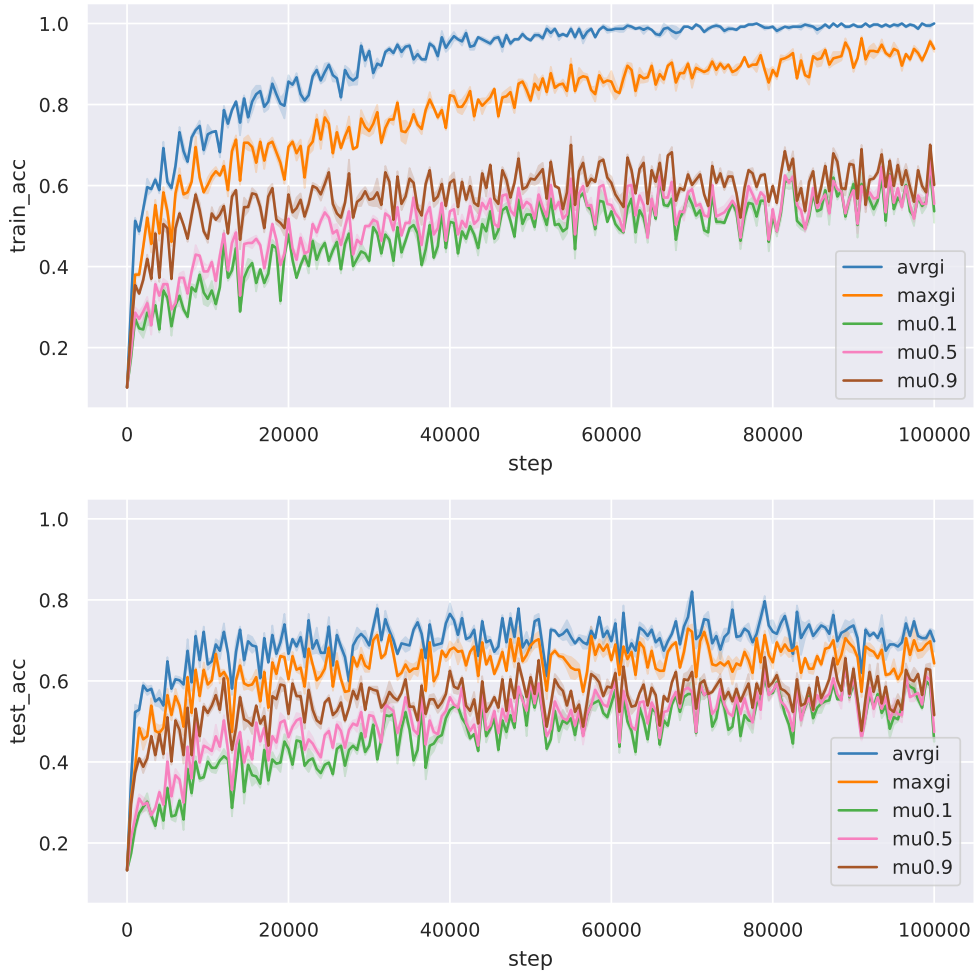
STOFFAR = OFFAR +

$$\mathbb{E}_k\left[\|\nabla_x^i f(X_k) - \overline{\nabla_x^i f}(X_k)\|^{\frac{p+1}{p+1-i}}\right] \leq \kappa_D \sum_{i=1}^{m} \|S_{k-i}\|^{p+1} \quad (i = 1, 2)$$

ASTR1 on CIFAR-10 with cifar10-nv ($\gamma = 10^{-5}$)



ASTR1 on CIFAR-10 with resnet18 ($\gamma = 5.10^{-5}$)



STOFFAR

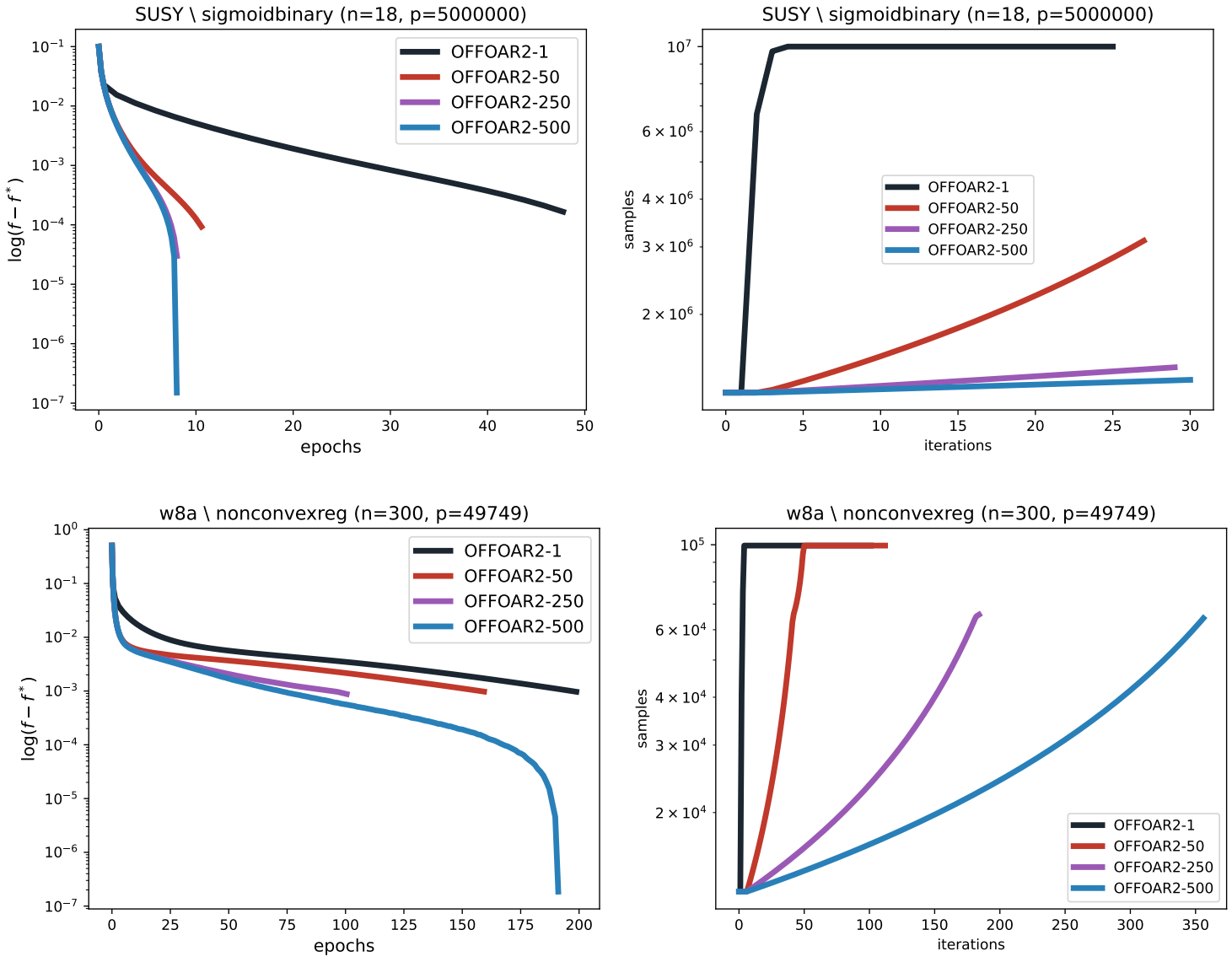


Figure: Loss function and number of samples for SUSY and w8a

Conclusions

Computing the value of f is not necessary for (theoretical) fast convergence

The use of curvature information is possible (and often beneficial) in standard OFFO adaptive methods

OFFO creates some interesting challenges in convergence theory!

Extension of ASTR1 to problems with convex constraints available!

Complexity of stochastic variants (ASTR1, OFFAR) also analyzed

Thank you for your interest and patience . . . and good wind to Yurii!

Details in...

S. Gratton and S. Jerad and Ph. L. Toint, "Parametric Complexity Analysis for a Class of First-Order Adagrad-like Algorithms", to appear in Optimization Methods and Software, 2024.

S. Gratton and Ph. L. Toint, "OFFO minimization algorithms for second-order optimality and their complexity", Computational Optimization and Applications, vol. 84, pp. 573—607, 2022.

S. Gratton and S. Jerad and Ph. L. Toint, "Convergence properties of an Objective-Function-Free Optimization regularization algorithm, including an $\mathcal{O}(\epsilon^{-3/2})$ complexity bound", SIAM Journal on Optimization, vol. 33(3), pp. 1621–1646, 2023.

S. Gratton, A. Kopaničáková and Ph. L. Toint, "Multilevel Objective-Function-Free Optimization with an Application to Neural Networks Training", SIAM Journal on Optimization, vol. 33(4), pp. 2772–2800, 2023.

S. Gratton and S. Jerad and Ph. L. Toint, "Complexity of Adagrad and other first-order methods for nonconvex optimization problems with bounds and convex constraints", arXiv:2406.15793,2024.

S. Gratton and S. Jerad and Ph. L. Toint, "A Stochastic Objective-Function-Free Adaptive Regularization Method with Optimal Complexity", arXiv:2407.08018, 2024.

S. Gratton, V. Mercier, E. Riccietti and Ph. L. Toint, "A Block-Coordinate Approach of Multi-level Optimization with an Application to Physics-Informed Neural Networks", Computational Optimization and Applications, to appear, 2024.