# Rest-Katyusha: Exploiting the Solution's Structure via Scheduled Restart Schemes

Junqi Tang[†], Mohammad Golbabaee[#], Francis Bach[⋆], Mike Davies[†]

[†] University of Edinburgh, [#] University of Bath, [⋆] INRIA - ENS

## Towards the Structure-Adaptive Optimization

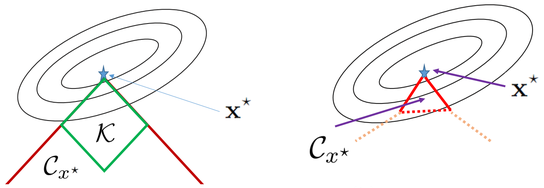
- In supervised machine learning we attempt to infer $x^\dagger \in \mathbb{R}^d$:

$$x^\dagger = \arg\min_x \mathbb{E}_a \bar{f}(a, x), (\text{expected risk}) \quad (1)$$

via *regularized empirical risk minimization* (ERM) :

$$x^\star \in \arg\min_{x\in\mathbb{R}^d} \left\{ F(x) := f(x) + \lambda g(x) \right\}, \quad f(x) := \frac{1}{n}\sum_{i=1}^n f_i(x), \quad (2)$$

- Each $f_i(x) \to$ convex and $L$-smooth, regularizer $g(x) \to$ convex function and possibly non-smooth.
- Strong-convexity assumption is often vacuous for high-dimensions.
- Non-smooth $g(.)$ injects prior information to ERM and often enforce the solution to be *structured*, e.g. sparse, piece-wise smooth, or low rank, etc.




(a) ERM with constraint $g(.) := \iota_{\mathcal{K}}(.)$ (b) Regularized ERM

- **Can we exploit the solution's structure to design even faster optimization algorithms?**

## Solution's Structure and Restricted Strong-Convexity

- Let $x^\star \in \mathcal{M}$ – a low-dimensional subspace in $\mathbb{R}^d$, and define:

$$\Phi(\mathcal{M}) := \sup_{v\in\mathcal{M}\setminus\{0\}} \frac{g(v)}{\|v\|_2}, \quad (3)$$

- We assume the *restricted strong-convexity* [2, 3]:

$$f(x) - f(x^\star) - \langle \nabla f(x^\star), x - x^\star \rangle \geq \frac{\gamma}{2}\|x - x^\star\|_2^2 - \tau g^2(x - x^\star), \quad (4)$$

- Then for statistically-optimal choices of regularization parameter $\lambda \geq 2g^*(\nabla f(x^\dagger))$,

$$F(x) - F^\star \geq \mu_c \|x - x^\star\|_2^2 - \text{residual}, \quad \forall x \in \mathbb{R}^d, \quad (5)$$

where $\mu_c = \frac{\gamma}{2} - 32\tau\Phi^2(\mathcal{M})$, which encodes the intrinsic dimension of $x^\star$.

## References

1. Z. Allen-Zhu, *Katyusha: the First Direct Acceleration of Stochastic Gradient Methods*, JMLR, 2018.
2. A. Agarwal, S. Negahban, and M. Wainwright. *Fast Global Convergence Rates of Gradient Methods for High-Dimensional Statistical Recovery.* The Annals of Statistics, 2012.
3. S. Negahban, B. Yu , M. Wainwright, and P. Ravikumar. *A Unified Framework for High-Dimensional Analysis of M-Estimators with Decomposable Regularizers.* NIPS, 2009.
4. O. Fercoq, and Q. Zheng. *Adaptive Restart of Accelerated Gradient Methods under Local Quadratic Growth Condition.* arXiv, 2017.

## Restarted Katyusha Algorithm

- We focus on a state-of-the-art algorithm Katyusha [1] for solving (2).

**Katyusha($x^0$, $L$, $S$) – *informally written:**

**For (s = 1, 2, .., S)**
**Set the momentum parameter** $\theta = \frac{2}{s+4}$
**Inner loop (i = 1, 2, 3, ... 2n)**
→ Calculate a variance-reduced stochastic gradient $\tilde{\nabla} f(x_i^s)$
→ $z_{i+1}^s = \text{prox}_{\lambda g}^{\frac{1}{3\theta L}}[z_i^s - \frac{1}{3\theta L}\tilde{\nabla} f(x_i^s)]$
→ $y_{i+1}^s = \text{prox}_{\lambda g}^{\frac{1}{3L}}[x_i^s - \frac{1}{3L}\tilde{\nabla} f(x_i^s)]$
→ **Extrapolate** $x_{i+1}^s$ with (stabilized) Nesterov's momentum
Output $\hat{x}^{S+1} = \frac{1}{2n}\sum_{j=1}^{2n} y_j^S$

- The original Katyusha only has sublinear convergence $O(\sqrt{nL/\epsilon})$ without the ordinary strong-convexity assumption.
- Our algorithm – "restart to rescue":

**Rest-Katyusha($x^0$, $L$, $\mu_c$, $S_0$, $T$, $\beta \geq 2$):**
**First stage – warm start:**
→ $x^1 = \text{Katyusha}(x^0, L, S_0)$
**Second stage – periodic restart to exploit the structure:**
**Restart period** $S = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_c}} \right\rceil$
**For (t = 1, 2, 3, ... T)**
→ $x^{t+1} = \text{Katyusha}(x^t, L, S)$
Output $x^{T+1}$

- Accelerated linear convergence $O\left(n + \sqrt{\frac{nL}{\mu_c}}\right)\log\frac{1}{\epsilon}$ towards the statistical accuracy $o\|x^\star - x^\dagger\|_2$.

## Adaptive Rest-Katyusha

**Practical issues**

- $\mu_c \to$ hard to be estimated accurately.
- Inaccurate $\mu_c \to$ compromised convergence.

**Can we estimate $\mu_c$ on the fly? – Yes, by adaptive restart !**

→ With the composite gradient map:

$$\mathcal{T}(x) = \arg\min_q \frac{L}{2}\|x - q\|_2^2 + \langle \nabla f(x), q - x\rangle + \lambda g(q), \quad (6)$$

→ it is known that (see e.g. [4]):

$$F(x) - F^\star = \Theta\left(\|\mathcal{T}(x) - x\|_2^2\right), \quad (7)$$

→ Our Scheme : Tune the $\mu_c$ estimate via tracking the objective-gap by $\mathcal{T}(x)$.

**Adaptive Rest-Katyusha($x^0$, $L$, $\mu_0$ – *initial guess, $S_0$, $T$, $\beta$):**
**First stage – warm start:**
→ $x^1 = \text{Katyusha}(x^0, L, S_0)$
→ Calculate $\mathcal{T}(x^1)$, $S = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_0}} \right\rceil$
**Second stage – adaptive restart via estimating $\mu_c$:**
**For (t = 1, 2, 3, ... T)**
→ $x^{t+1} = \text{Katyusha}(x^t, L, S)$
→ Calculate $\mathcal{T}(x^{t+1})$
→ **if** $\|\mathcal{T}(x^{t+1}) - x^{t+1}\|_2^2 \leq \frac{1}{\beta^2}\|\mathcal{T}(x^t) - x^t\|_2^2$
**then** $\mu_0 \leftarrow 2\mu_0$, **else** $\mu_0 \leftarrow \mu_0/2$.
→ **Tune the restart period** $S = \left\lceil \beta\sqrt{32 + \frac{12L}{n\mu_0}} \right\rceil$
Output $x^{T+1}$

## Numerical Experiments

We experiment on the Lasso regression task:

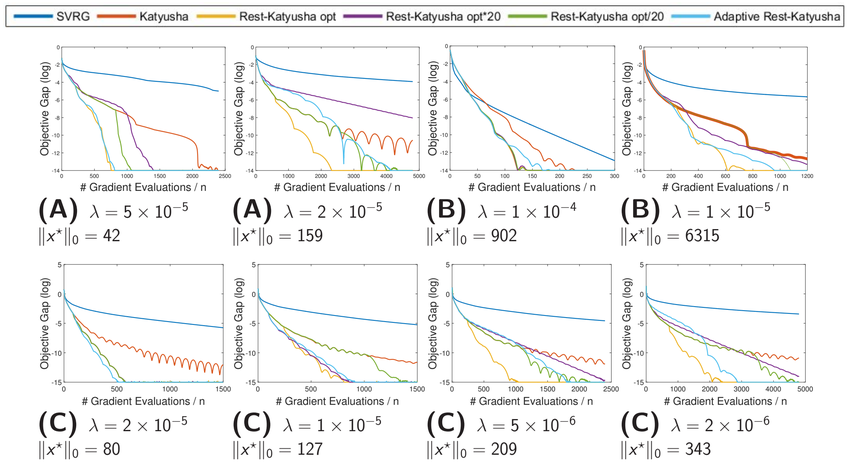
$$x^\star \in \arg\min_{x\in\mathbb{R}^d} \left\{ F(x) := \frac{1}{2n}\|Ax - b\|_2^2 + \lambda\|x\|_1 \right\}. \quad (8)$$

Table: Datasets for the experiments and minibatch size we adopt for the algorithms

| Data set | Size ($n$, $d$) | Minibatch size. |
|---|---|---|
| **(A)** Madelon | (2000, 500) | 1 |
| **(B)** RCV1 | (20242, 47236) | 80 |
| **(C)** Reged | (500, 999) | 1 |

- For Lasso, **sparser the solution is, faster our algorithm converges!**

Figure: Lasso experiments (We denote "Rest-Kat Opt" → optimal $\mu_c$ input, "Rest-Kat Opt*20" → 20 times overestimation , "Rest-Kat Opt/20" → 20 times underestimation.)




J.Tang@ed.ac.uk ;Work supported by H2020-MSCA-ITN Machine Sensing Training Network (MacSeNet), project number 642685.