A Simple, Optimal and Efficient Algorithm for Online Exp-Concave Optimization

Yi-Han Wang, Peng Zhao, Zhi-Hua Zhou @ LAMDA Group, Nanjing University, China

Download this poster
and presentation slides
from the homepage of
Yi-Han Wang.

The 39th Annual Conference on Learning Theory

Online eXp-concave Optimization (OXO)

Each round $t = 1, 2, \dots, T$:

- learner picks a decision $\mathbf{x}_t \in \mathcal{X} \subseteq \mathbb{R}^d$
- at the same time, environment picks convex loss $f_t : \mathcal{X} \to \mathbb{R}$
- the learner suffers $f_t(\mathbf{x}_t)$ and observes $\nabla f_t(\mathbf{x}_t)$

f_t is *α-exponentially-concave* ($\alpha > 0$), i.e., $\exp(-\alpha f(\cdot))$ is concave.

[Kivinen and Warmuth, COLT'99]

Objective: *regret* minimization $\quad \mathrm{Reg}_T \triangleq \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{u}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{u})$

Regret-Runtime Challenge in OXO

Online Newton Step (ONS) [Hazan, Kalai, Kale, Agarwal, COLT'06]

Initialize $\mathbf{x}_1 \in \mathcal{X}$, $A_0 = \epsilon I$.
Each round $t = 1, 2, \dots, T$:

- Play $\mathbf{x}_t$, observe $\nabla f_t(\mathbf{x}_t)$
- $A_t = A_{t-1} + \nabla f_t(\mathbf{x}_t)\nabla f_t(\mathbf{x}_t)^\top$
- $\boxed{\widehat{\mathbf{x}}_{t+1} = \mathbf{x}_t - \frac{1}{\gamma_0} A_t^{-1} \nabla f_t(\mathbf{x}_t)}$
- $\mathbf{x}_{t+1} = \boxed{\Pi_{\mathcal{X}}^{A_t}\left[\widehat{\mathbf{x}}_{t+1}\right]}$

Newton-style update with Hessian-like matrix

$$A_t = \epsilon I + \sum_{s=1}^{t} \nabla f_s(\mathbf{x}_s)\nabla f_s(\mathbf{x}_s)^\top$$

Mahalanobis projection

$$\Pi_{\mathcal{X}}^{A}[\mathbf{y}] \triangleq \min_{\mathbf{x}\in\mathcal{X}} (\mathbf{x}-\mathbf{y})^\top A(\mathbf{x}-\mathbf{y})$$

ONS achieves ***minimax optimal regret*** $O(d\log T)$. However, each ***Mahalanobis projection*** costs $\widetilde{O}(d^3)$ time even for simple domains including ***the unit ball***, leading to $\widetilde{O}(d^3 T)$ ***total runtime***.

Online Gradient Descent (OGD) achieves suboptimal regret $O(\sqrt{T})$ for OXO, but only takes $O(dT)$ runtime for the unit ball.

Online Quasi-Newton Steps (OQNS) [Mhammedi and Gatmiry, COLT'23]

OQNS circumvents ***Mahalanobis projections*** with ***log barrier***:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \mathrm{Approximate}\big((\nabla^2\Phi_t(\mathbf{x}_t))^{-1}\nabla\Phi_t(\mathbf{x}_t)\big) \text{ where } \Phi_t(\mathbf{x}) \triangleq \eta d \log\frac{1}{1-\|\mathbf{x}\|^2} + \frac{d+\eta}{2}\|\mathbf{x}\|^2 + \sum_{s=1}^{t}\left(\nabla f_s(\mathbf{x}_s)^\top\mathbf{x} + \frac{\gamma}{2}\left(\nabla f_s(\mathbf{x}_s)^\top(\mathbf{x}-\mathbf{x}_s)\right)^2\right).$$

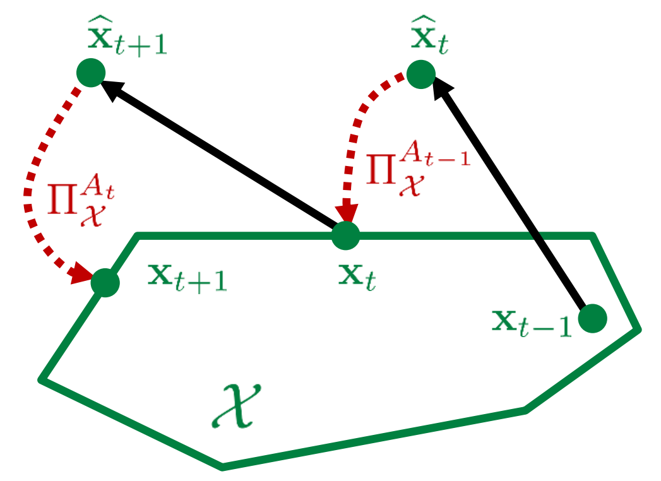
OQNS achieves minimax optimal regret $O(d\log T)$ and reduces runtime to $O(d^2 T\log T + d^3\sqrt{T\log T})$. However, OQNS is ***not*** "***ONS-like***" (does ***not*** apply to ***downstream results*** where ONS is backbone)!

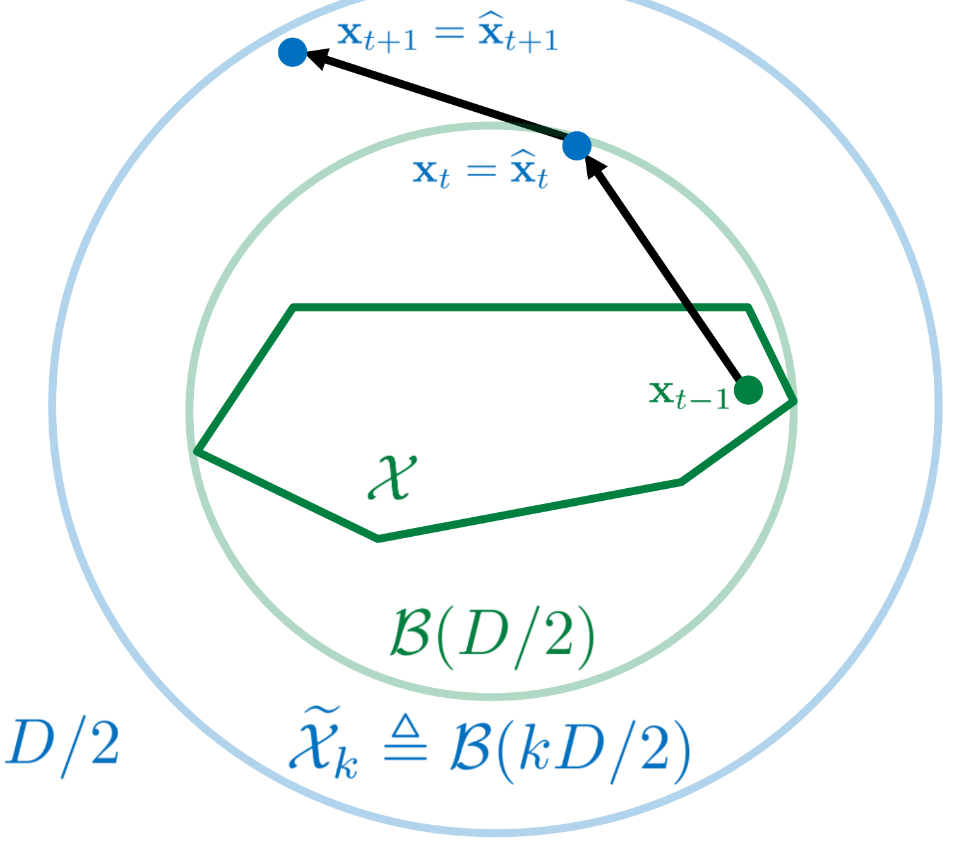
Our Results

Algorithm	Regret	Runtime	ONS-Like
OGD	$\sqrt{T}$	dT	N/A
ONS	$d\log T$	$d^2T + d^3T\log T$	**Yes**
OQNS	$d\log T$ (constants $\times 25$)	$d^2T\log T + d^3\sqrt{T\log T}$	**No**
LightONS	$d\log T$ (same constants as ONS)	$d^2T + d^3\sqrt{T\log T}$	**Yes**

LightONS achieves ① optimal regret, ② reduced runtime, ③ "***ONS-like***"

Simply changing ***1 line*** of ONS





from $\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}^{A_t}[\widehat{\mathbf{x}}_{t+1}]$

$$\text{to } \mathbf{x}_{t+1} = \begin{cases} \widehat{\mathbf{x}}_{t+1} & \text{if } \|\widehat{\mathbf{x}}_{t+1}\|_2 \le kD/2 \\ \Pi_{\mathcal{X}}^{A_t}[\widehat{\mathbf{x}}_{t+1}] & \text{otherwise} \end{cases}$$

achieves minimax optimal regret $O(d\log T)$ and reduces runtime to $O(d^2T + d^3\sqrt{T\log T})$. However, this is ***improper learning*** which ***may reduce the intrinsic difficulty*** of OXO. Example: for logistic regression, proper regret is $\Omega(de^G\log T)$, while improper regret reaches $O(d\log(GT))$.

[Hazan, Koren, Levy, COLT'14; Foster, Kale, Luo, Mohri, Sridharan, COLT'18]

LightONS (Our Method)

Initialize $\mathbf{y}_1 \in \widetilde{\mathcal{X}}_k$, $A_0 = \epsilon I$.
Each round $t = 1, 2, \dots, T$:

- $\boxed{\text{Play } \mathbf{x}_t = \Pi_{\mathcal{X}}[\mathbf{y}_t], \text{ compute } \nabla g_t(\mathbf{y}_t)}$ observe $\nabla f_t(\mathbf{x}_t)$,
- $A_t = A_{t-1} + \nabla g_t(\mathbf{y}_t)\nabla g_t(\mathbf{y}_t)^\top$
- $\widehat{\mathbf{y}}_{t+1} = \mathbf{y}_t - \frac{1}{\gamma_0} A_t^{-1}\nabla g_t(\mathbf{y}_t)$
- $\boxed{\mathbf{y}_{t+1} = \begin{cases} \widehat{\mathbf{y}}_{t+1} & \text{if } \|\widehat{\mathbf{y}}_{t+1}\|_2 \le kD/2 \\ \Pi_{\widetilde{\mathcal{X}}_k}^{A_t}[\widehat{\mathbf{y}}_{t+1}] & \text{otherwise} \end{cases}}$.

Domain conversion settles improper learning issue: Run ***deferred projections*** on ***surrogate loss***, then use ***cheap Euclidean projection*** to ensure proper learning.

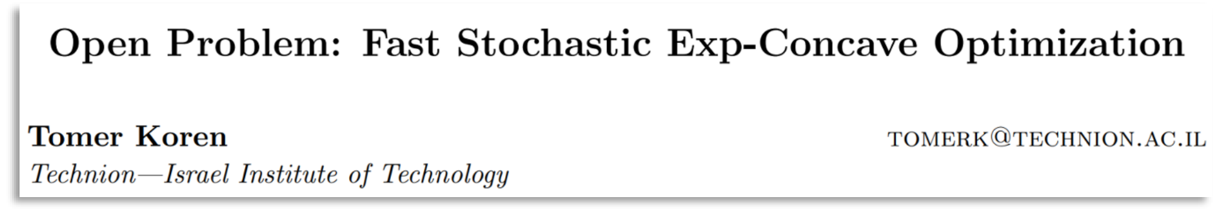
Surrogate loss: $g_t(\mathbf{y}) = \nabla f_t(\mathbf{x}_t)^\top\mathbf{y} + \frac{\max\{-\nabla f_t(\mathbf{x}_t)^\top(\mathbf{y}_t-\mathbf{x}_t),0\}}{\|\mathbf{y}_t-\mathbf{x}_t\|_2}\|\mathbf{y}-\Pi_{\mathcal{X}}[\mathbf{y}]\|_2$

True decision: $\mathbf{x}_t = \Pi_{\mathcal{X}}[\mathbf{y}_t]$

[Cutkosky and Orabona, COLT'18; Cutkosky, ICML'20]

Answering COLT'13 Open Problem

Consider Stochastic eXp-concave Optimization (SXO).



Open Problem: Fast Stochastic Exp-Concave Optimization

Tomer Koren TOMERK@TECHNION.AC.IL
Technion—Israel Institute of Technology

$\mathrm{ExcessRisk} \triangleq \mathbb{E}_f[f(\bar{\mathbf{x}}_T)] - \min_{\mathbf{u}\in\mathcal{X}} \mathbb{E}_f[f(\mathbf{u})]$ Objective: $\mathrm{ExcessRisk} \le \varepsilon$

With ***online-to-batch conversion (O2BC)***, ONS incurs runtime $\widetilde{O}(d^4/\varepsilon)$.

Open Problem. For SXO, *(i)* is it possible to achieve runtime $\widetilde{O}(d^2/\varepsilon)$? *(ii)* is it possible to perform any better than runtime $\widetilde{O}(d^4/\varepsilon)$?

With ***O2BC***, LightONS achieves runtime $\tilde{O}(d^3/\varepsilon)$ ***in expectation***, and ***with high probability***, thus answers part *(ii)* of the open problem.

Applications to Downstream Results

Compared with OQNS, LightONS is ***simple***, incurs small constants, most importantly, is "***ONS-like***" (applies to ***downstream results***).

① **Gradient-Norm Adaptivity**: ***problem-dependent*** regret sacles with $G_T \triangleq \sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_2^2 \le G^2T$. In benign cases, $G_T = o(T)$ yields much better regret. For ***unbounded online convex optimization***, this implies comparator-norm adaptivity, i.e., $\mathrm{Reg}_T \le \widetilde{O}(\|\mathbf{u}\|_2\sqrt{dG_T})$.
[Orabona, Cesa-Bianchi, Gentile, AISTATS'12; Cutkosky and Orabona, COLT'18]

② **Jointly Efficient Generalized Linear Bandits**: e.g., logistic bandits, K-arm, $0/1$-feedback, $\mathbb{P}(y_t = 0 \mid \mathbf{x}_t) = 1/(1+\exp(\mathbf{w}^\top\mathbf{x}_t))$. Challenge: *(a)* regret dependence on condition number $\kappa \propto \exp(D)$, *(b)* one pass.
[Zhang, Xu, Zhao, Sugiyama, NeurIPS'25]

③ **Memory-Efficient OXO**: only store top d' eigenvectors of the Hessian-like matrix A_t in ONS ($d' \ll d$), reduce memory from $O(d^2)$ to $O(d'd)$.
[Luo, Agarwal, Cesa-Bianchi, Langford, NIPS'16]

	Regret (ONS & LightONS)	Runtime (ONS)	Runtime (LightONS)	OQNS?
①	$O(d\log G_T)$	$\widetilde{O}(d^3T)$	$O(d^2T + d^3\sqrt{T}\log T)$	**Hardly**
②	$\widetilde{O}(d\sqrt{T} + \kappa d^2)$	$\widetilde{O}(d^2KT + d^3T)$	$\widetilde{O}(d^2KT + d^3 \cdot \min\{\sqrt{\kappa dT}\log\kappa, T\})$	**Hardly**
③	$O(d'\log T)$	$\widetilde{O}(d^3T)$	$\widetilde{O}(d'dT + d^3\sqrt{T})$	**Intricate**

Main References

- J. Kivinen, M. Warmuth. *Averaging expert predictions*. **COLT'99**.
- E. Hazan, A. Kalai, S. Kale, A. Agarwal. *Logarithmic regret algorithms for online convex optimization*. **COLT'06**.
- T. Koren. *Open problem: Fast stochastic exp-concave optimization*. **COLT'13**.
- Z. Mhammedi, K. Gatmiry. *Quasi-newton steps for efficient online exp-concave optimization*. **COLT'23**.
- A. Cutkosky, F. Orabona. *Black-box reductions for parameter-free online learning in Banach spaces*. **COLT'18**.
- A. Cutkosky. *Parameter-free, dynamic, and strongly-adaptive online learning*. **ICML'20**.