Online Convex Optimization with Stochastic Constraints

Ming Hsieh Department of Electrical Engineering

{ Hao Yu, Michael Neely, Xiaohan Wei } University of Southern California

USC Viterbi School of Engineering

1. OCO and Zinkevich's OGD

Online Convex Optimization (OCO) is a multi-round process of making decisions **without knowing what to optimize**

- An arbitrary sequence of convex loss functions $f^1(\mathbf{x}), f^2(\mathbf{x}), \ldots, f^t(\mathbf{x}), \ldots$ for each round.
- At each round t, choose $\mathbf{x}(t) \in \mathcal{X}$ without knowing $f^t(\mathbf{x})$ based only on previous $f^\tau(\cdot), \tau < t$.

Zinkevich's online gradient descent (OGD) chooses

$$\mathbf{x}(t+1) = \mathcal{P}_{\mathcal{X}}\left[\mathbf{x}(t) - \gamma \nabla f^t(\mathbf{x}(t))\right].$$

OCO performance metric: **regret** is accumulate loss difference between an algorithm and the optimal fixed decision in hindsight

$$\text{regret}(T) = \sum_{t=1}^{T} f^t(\mathbf{x}(t)) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f^t(\mathbf{x})$$

Zinkevich's OGD achieves $O(\sqrt{T})$ regret which is best possible without strong convexity.

2. OCO with Stochastic Constraints

Zinkevich's OGD is Optimistic:

- Existing OCO assumes **full knowledge of set $\mathcal{X}$ and low complexity of $\mathcal{P}_{\mathcal{X}}[\cdot]$.**
- Even if $\mathcal{X}$ is perfectly known, $\mathcal{P}_{\mathcal{X}}[\cdot]$ involved in Zinkevich's OGD is too expensive to compute for complicated $\mathcal{X}$, e.g., $\mathcal{X} = \{\mathbf{x} \in \mathcal{X}_0 : g_k(\mathbf{x}) \leq 0, \forall k \in \{1, 2, \ldots, m\}\}$.

We generalizes the conventional OCO to the setup with (i.i.d.) online functional constraints.

> Consider stochastic $\mathcal{X}$ given by
>
> $$\mathcal{X} = \{\mathbf{x} \in \mathcal{X}_0 : \mathbb{E}[g_k(\mathbf{x};\omega)] \leq 0, \forall k \in \{1, 2, \ldots, m\}\}$$
>
> where $\mathcal{X}_0$ is a simple known set and ω is i.i.d. from an unknown distribution. At each round t, the decision maker receives i.i.d. realizations $g_k^t(\mathbf{x}) = g_k(\mathbf{x};\omega(t))$ that are disclosed at round $t+1$ after $\mathbf{x}(t) \in \mathcal{X}_0$ is chosen.

Our Goal/Contribution:

- Avoid projection onto $\mathcal{X}$ and only use $\mathcal{P}_{\mathcal{X}_0}[\cdot]$ ($\mathcal{X}_0$ is simple).
- Solve online constraints, knowing each $g_k^t(\mathbf{x})$ after $t+1$.
- Achieve $O(\sqrt{T})$ regret and $O(\sqrt{T})$ constraint violations.

3. Our New Algorithm

- **Parameter:** External: $V > 0$ and $\alpha > 0$; Internal: introduce virtual queue $Q_k(t)$ for each stochastic constraint.
- **Initialization:** Set $\mathbf{x}(1) \in \mathcal{X}_0$ arbitrarily and $Q_k(1) = 0, \forall k$.
- At the end of each round $t = 1, 2, \ldots$, observe $f^t(\mathbf{x})$ and each $g_k^t(\mathbf{x})$ and update for the next round $t+1$ as follows:
  - Output $\mathbf{x}(t+1)$, decision for round $t+1$, via

$$\mathbf{x}(t+1) = \mathcal{P}_{\mathcal{X}_0}\left[\mathbf{x}(t) - \tfrac{1}{2\alpha}\mathbf{d}(t)\right] \quad \text{where} \quad \mathbf{d}(t) = V\nabla f^t(\mathbf{x}(t)) + \textstyle\sum_{k=1}^{m} Q_k(t)\nabla g_k^t(\mathbf{x}).$$

  - Update each internal parameter $Q_k(t+1)$ via

$$Q_k(t+1) = \max\left\{0, Q_k(t) + g_k^t(\mathbf{x}(t)) + [\nabla g_k^t(\mathbf{x}(t))]^{\mathsf{T}}[\mathbf{x}(t+1) - \mathbf{x}(t)]\right\}$$

4. Performance Guarantees

Let $\mathbf{x}^* = \mathcal{X}$ be the optimal fixed solution in hindsight.

If we choose $V = \sqrt{T}$ and $\alpha = T$ in our algorithm, then

Expected Performance

- Regret: $\mathbb{E}[\sum_{t=1}^{T} f^t(\mathbf{x}(t))] \leq \mathbb{E}[\sum_{t=1}^{T} f^t(\mathbf{x}^*)] + O(\sqrt{T})$.
- Constraint: $\mathbb{E}[\sum_{t=1}^{T} g_k^t(\mathbf{x}(t))] \leq O(\sqrt{T}), \forall k$

High Prob Performance

For any $0 < \lambda < 1$, we have

- $\Pr[\sum_{t=1}^{T} f^t(\mathbf{x}(t)) \leq \sum_{t=1}^{T} f^t(\mathbf{x}^*) + O(\sqrt{T}\log(T)\log^{1.5}(\frac{1}{\lambda}))] \geq 1-\lambda$
- $\Pr[\sum_{t=1}^{T} g_k^t(\mathbf{x}(t))] \leq O(\sqrt{T}\log(T)\log(\frac{1}{\lambda}))] \geq 1-\lambda, \forall k$

5. Special Case Problems

- **OCO with long term constraints** [Mahdavi'12JMLR] [Cotter'15COLT] [Jenatton'16ICML]
  - $g_k(\mathbf{x};\omega) \equiv g_k(\mathbf{x}), \forall k$.
  - Existing: $O(T^{\max\{\beta, 1-\beta\}})$ regret and $O(T^{1-\beta/2})$ constraint violations with $\beta \in (0,1)$.
  - Our alg: $O(\sqrt{T})$ regret and $O(\sqrt{T})$ constraint violations
- **Stochastic constrained convex opt** [Mahdavi'13NIPS][Lan'16]:
  - $f^t(\mathbf{x})$ i.i.d. generated (not arbitrarily time-varying).
  - Existing: offline (batch) solutions: high prob guarantee [Mahdavi'13NIPS] or expectation guarantee for problems with a single stochastic constraint [Lan'16].
  - Our alg: online and solve arbitrary # of stochastic constraints
- **Deterministic constrained convex opt** [Nedich'09JOTA]
  - $f^t(\mathbf{x}) \equiv f(\mathbf{x})$ and $g_k(\mathbf{x};\omega) \equiv g_k(\mathbf{x}), \forall k$.
  - Our alg is purely subgradient based and can solve non-smooth non-strongly-convex problems. Tied with other algs with the best convergence rates.

6. Exp: Job Scheduling in GEO-distributed Data Centers

- 100 servers clustered at 10 zones
- electricity prices vary across zones and time
- schedule online job arrivals
- serve all jobs and minimize electricity cost