Online Convex Optimization First-Order Algorithms and Second-Order Methods

Zhihao Jin

May 23, 2025

Table of Contents

Preliminaries:The Online Convex Optimization Setting

- In OCO, an online player iteratively makes decisions. At the time of each decision, the outcome or outcomes associated with it are unknown to the player.
- After committing to a decision, the decision maker suffers a loss: every possible decision incurs a (possibly different) loss. These losses are unknown to the decision maker beforehand.
- The online convex optimization (OCO) framework models the decision set as a convex set in Euclidean space denoted as $\mathcal{K} \subseteq \mathbb{R}^n$. The costs are modeled as bounded convex functions over $\mathcal{K}$.

Preliminaries:The Online Convex Optimization Setting

The OCO framework can be seen as a structured repeated game. The protocol of this learning framework is as follows:

- At iteration t, the online player chooses $\mathbf{x}_t \in \mathcal{K}$.
- After the player has committed to this choice, a convex cost function $f_t \in \mathcal{F} : \mathcal{K} \mapsto \mathbb{R}$ is revealed.$\mathcal{F}$ is the bounded family of cost functions available to the adversary.
- The cost incurred by the online player is $f_t(\mathbf{x}_t)$, the value of the cost function for the choice $\mathbf{x}_t$.
- Let T denote the total number of game iterations.

Preliminaries:The Online Convex Optimization Setting

- The appropriate performance metric comes from game theory: define the regret of the decision maker to be the difference between the total cost she has incurred and that of the best fixed decision in hindsight.
- Let $\mathcal{A}$ be an algorithm for OCO, which maps a certain game history to a decision in the decision set:

$$\mathbf{x}_t^{\mathcal{A}} = \mathcal{A}\left(f_1, \ldots, f_{t-1}\right) \in \mathcal{K}.$$

define the regret of $\mathcal{A}$ after T iterations as:

$$\text{Regret}_T(\mathcal{A}) = \sup_{\{f_1,\ldots,f_T\}\subseteq\mathcal{F}} \left\{ \sum_{t=1}^{T} f_t\left(\mathbf{x}_t^{\mathcal{A}}\right) - \min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x}) \right\}.$$

Preliminaries:The Online Convex Optimization Setting

- An algorithm performs well if its regret is sublinear as a function of T (i.e. $\mathrm{Regret}_T(\mathcal{A}) = o(T)$), since this implies that on average, the algorithm performs as well as the best fixed strategy in hindsight.
- The running time of an algorithm for OCO is defined to be the worstcase expected time to produce $\mathbf{x}_t$, for an iteration $t \in [T]$ in a T-iteration repeated game.
- Typically, the running time will depend on n (the dimensionality of the decision set $\mathcal{K}$), T (the total number of game iterations), and the parameters of the cost functions and underlying convex set.

Preliminaries:The experts problem

- The decision maker has to choose among the advice of n given experts. After making a choice, a loss between zero and 1 is incurred. At each iteration, the costs of the various experts are arbitrary.The goal of the decision maker is to do as well as the best expert in hindsight.
- The set of decisions is the set of all distributions over n elements (experts); that is, the n-dimensional simplex $\mathcal{K} = \Delta_n = \{\mathbf{x} \in \mathbb{R}^n, \sum_i \mathbf{x}_i = 1, \mathbf{x}_i \geq 0\}$.
- Let the cost of the i th expert at iteration t be $\mathbf{g}_t(i)$, and let $\mathbf{g}_t$ be the cost vector of all n experts. Then the cost function is the expected cost of choosing an expert according to distribution $\mathbf{x}$, and it is given by the linear function $f_t(\mathbf{x}) = \mathbf{g}_t^\top \mathbf{x}$.

Table of Contents

Intro

- Describe and analyze the most simple and basic algorithms for online convex optimization,which are also surprisingly useful and applicable in practice.
- The goal of the algorithms introduced next is to minimize regret, rather than the optimization error (which is ill-defined in an online setting).
- $$\text{Regret}_T = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x})$$

Intro

Table: Attainable asymptotic regret bounds for loss function classes.

	α-strongly convex	β-smooth	δ-exp-concave
Upper bound	$\frac{1}{\alpha}\log T$	$\sqrt{T}$	$\frac{n}{\delta}\log T$
Lower bound	$\frac{1}{\alpha}\log T$	$\sqrt{T}$	$\frac{n}{\delta}\log T$
Average regret	$\frac{\log T}{\alpha T}$	$\frac{1}{\sqrt{T}}$	$\frac{n\log T}{\delta T}$

Online Gradient Descent

Algorithm 8 online gradient descent

1: Input: convex set $\mathcal{K}$, T, $\mathbf{x}_1 \in \mathcal{K}$, step sizes $\{\eta_t\}$
2: **for** $t = 1$ to T **do**
3: Play $\mathbf{x}_t$ and observe cost $f_t(\mathbf{x}_t)$.
4: Update and project:

$$\mathbf{y}_{t+1} = \mathbf{x}_t - \eta_t \nabla f_t(\mathbf{x}_t)$$
$$\mathbf{x}_{t+1} = \prod_{\mathcal{K}}(\mathbf{y}_{t+1})$$

5: **end for**

Online Gradient Descent

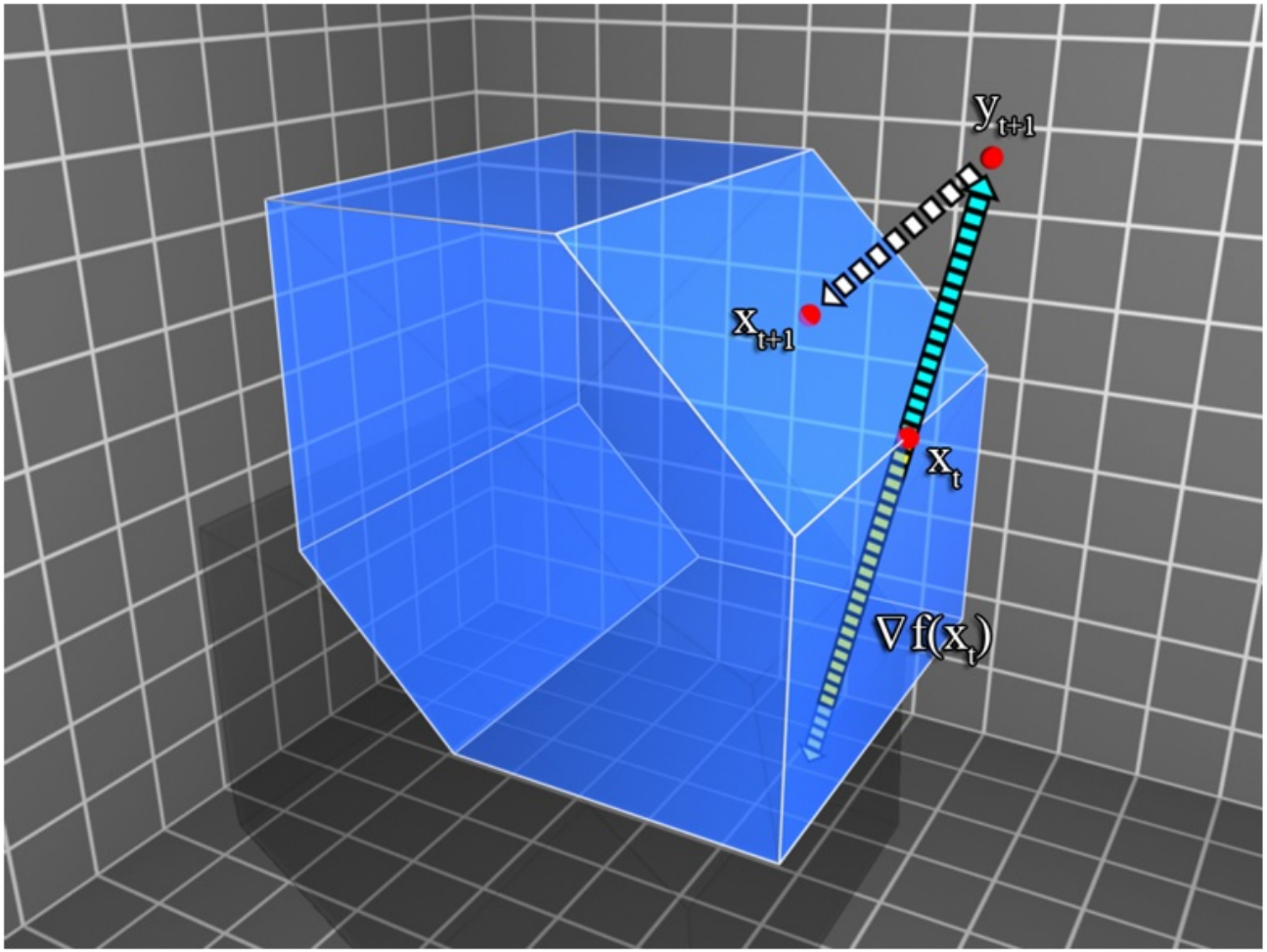

Figure 3.1: OGD: the iterate $\mathbf{x}_{t+1}$ is derived by advancing $\mathbf{x}_t$ in the direction of the current gradient ∇_t, and projecting back into $\mathcal{K}$

Theorem 3.1

Theorem

Online gradient descent with step sizes $\left\{\eta_t = \frac{D}{G\sqrt{t}}, t \in [T]\right\}$ *guarantees the following for all* $T \geq 1$ *:*

$$\text{Regret}_T = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}^\star \in \mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x}^\star) \leq \frac{3}{2} GD\sqrt{T}.$$

Denote by D *an upper bound on the diameter of* $\mathcal{K}$ *:*
$\forall \mathbf{x}, \mathbf{y} \in \mathcal{K}, \|\mathbf{x} - \mathbf{y}\| \leq D$.
Denote by $G > 0$ *an upper bound on the norm of the subgradients of* f *over* $\mathcal{K}$*, i.e.,* $\|\nabla f(\mathbf{x})\| \leq G$ *for all* $\mathbf{x} \in \mathcal{K}$.

Proof of theorem 3.1

Proof.

- Let $\mathbf{x}^\star \in \arg\min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x})$. Define $\nabla_t \overset{\text{def}}{=} \nabla f_t(\mathbf{x}_t)$. By convexity
  $f_t(\mathbf{x}_t) - f_t(\mathbf{x}^\star) \leq \nabla_t^\top (\mathbf{x}_t - \mathbf{x}^\star)$
- $\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 = \|\prod_{\mathcal{K}} (\mathbf{x}_t - \eta_t \nabla_t) - \mathbf{x}^\star\|^2 \leq \|\mathbf{x}_t - \eta_t \nabla_t - \mathbf{x}^\star\|^2$
- $$\|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2 \leq \|\mathbf{x}_t - \mathbf{x}^\star\|^2 + \eta_t^2 \|\nabla_t\|^2 - 2\eta_t \nabla_t^\top (\mathbf{x}_t - \mathbf{x}^\star)$$
  $$2\nabla_t^\top (\mathbf{x}_t - \mathbf{x}^\star) \leq \frac{\|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2}{\eta_t} + \eta_t G^2$$

Proof of theorem 3.1

Proof.

Summing from $t = 1$ to T, and setting $\eta_t = \frac{D}{G\sqrt{t}}$ (with $\frac{1}{\eta_0} \overset{\text{def}}{=} 0$):

$$
\begin{aligned}
& 2\left(\sum_{t=1}^{T} f_t(\mathbf{x}_t) - f_t(\mathbf{x}^\star)\right) \leq 2\sum_{t=1}^{T} \nabla_t^\top (\mathbf{x}_t - \mathbf{x}^\star) \\
& \leq \sum_{t=1}^{T} \frac{\|\mathbf{x}_t - \mathbf{x}^\star\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^\star\|^2}{\eta_t} + G^2 \sum_{t=1}^{T} \eta_t \\
& \leq \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}^\star\|^2 \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right) + G^2 \sum_{t=1}^{T} \eta_t \qquad \frac{1}{\eta_0} \overset{\text{def}}{=} 0 \\
& \leq D^2 \sum_{t=1}^{T} \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right) + G^2 \sum_{t=1}^{T} \eta_t \qquad \|\mathbf{x}_{T+1} - \mathbf{x}^*\|^2 \geq 0 \\
& \leq D^2 \frac{1}{\eta_T} + G^2 \sum_{t=1}^{T} \eta_t \\
& \leq 3DG\sqrt{T} \qquad \text{telescoping series}
\end{aligned}
$$

□

Theorem 3.2

Theorem

Any algorithm for online convex optimization incurs $\Omega(DG\sqrt{T})$ regret in the worst case. This is true even if the cost functions are generated from a fixed stationary distribution.

- Introduce a seemingly sophisticated and obviously general framework for learning and prediction, as well as a linear-time algorithm for the most general case, complete with tight regret bounds.
- For important classes of loss functions significantly better regret bounds are possible.

Theorem 3.3

Theorem

For α-strongly convex loss functions, online gradient descent with step sizes $\eta_t = \frac{1}{\alpha t}$ achieves the following guarantee for all $T \geq 1$

$$\text{Regret}_T \leq \frac{G^2}{2\alpha}(1 + \log T).$$

Proof.

- Let $\mathbf{x}^\star \in \arg\min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x})$.Define $\nabla_t \overset{\text{def}}{=} \nabla f_t(\mathbf{x}_t)$.
- $2\left(f_t(\mathbf{x}_t) - f_t(\mathbf{x}^\star)\right) \leq 2\nabla_t^\top(\mathbf{x}_t - \mathbf{x}^\star) - \alpha\left\|\mathbf{x}^\star - \mathbf{x}_t\right\|^2$
- $\left\|\mathbf{x}_{t+1} - \mathbf{x}^\star\right\|^2 = \left\|\prod_{\mathcal{K}}(\mathbf{x}_t - \eta_t\nabla_t) - \mathbf{x}^\star\right\|^2 \leq \left\|\mathbf{x}_t - \eta_t\nabla_t - \mathbf{x}^\star\right\|^2$
- $\left\|\mathbf{x}_{t+1} - \mathbf{x}^\star\right\|^2 \leq \left\|\mathbf{x}_t - \mathbf{x}^\star\right\|^2 + \eta_t^2\left\|\nabla_t\right\|^2 - 2\eta_t\nabla_t^\top(\mathbf{x}_t - \mathbf{x}^\star)$
- $2\nabla_t^\top(\mathbf{x}_t - \mathbf{x}^\star) \leq \frac{\left\|\mathbf{x}_t - \mathbf{x}^\star\right\|^2 - \left\|\mathbf{x}_{t+1} - \mathbf{x}^\star\right\|^2}{\eta_t} + \eta_t G^2$.

Proof of theorem 3.3

Proof.

Summing from $t = 1$ to T, and setting $\eta_t = \frac{1}{\alpha t}$ (define $\frac{1}{\eta_0} \overset{\text{def}}{=} 0$):

$$
\begin{aligned}
& 2 \sum_{t=1}^{T} \left(f_t \left(\mathbf{x}_t \right) - f_t \left(\mathbf{x}^{\star} \right) \right) \\
\leq & \sum_{t=1}^{T} \left\| \mathbf{x}_t - \mathbf{x}^{\star} \right\|^2 \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} - \alpha \right) + G^2 \sum_{t=1}^{T} \eta_t \\
& \text{since } \frac{1}{\eta_0} \overset{\text{def}}{=} 0, \left\| \mathbf{x}_{T+1} - \mathbf{x}^* \right\|^2 \geq 0 \\
= & 0 + G^2 \sum_{t=1}^{T} \frac{1}{\alpha t} \\
\leq & \frac{G^2}{\alpha} (1 + \log T)
\end{aligned}
$$

Application: Stochastic Gradient Descent

- A special case of Online Convex Optimization is the well-studied setting of stochastic optimization.
- Unlike standard offline optimization, the optimizer is given access to a noisy gradient oracle, defined by

$$\mathcal{O}(\mathbf{x}) \overset{\text{def}}{=} \tilde{\nabla}_{\mathbf{x}} \quad \text{s.t.} \quad \mathbf{E}\left[\tilde{\nabla}_{\mathbf{x}}\right] = \nabla f(\mathbf{x}), \mathbf{E}\left[\left\|\tilde{\nabla}_{\mathbf{x}}\right\|^2\right] \leq G^2$$

- Applying the OGD algorithm over a sequence of linear functions that are defined by the noisy gradient oracle at consecutive points, and finally returning the average of all points along the way,then obtain the stochastic gradient descent algorithm.

Application: Stochastic Gradient Descent

Algorithm 9 stochastic gradient descent

1: Input: $\mathcal{O}$,$\mathcal{K}$, T, $\mathbf{x}_1 \in \mathcal{K}$, step sizes $\{\eta_t\}$
2: **for** $t = 1$ to T **do**
3: Let $\tilde{\nabla}_t = \mathcal{O}(\mathbf{x}_t)$
4: Update and project:

$$\mathbf{y}_{t+1} = \mathbf{x}_t - \eta_t \tilde{\nabla}_t$$

$$\mathbf{x}_{t+1} = \prod_{\mathcal{K}}(\mathbf{y}_{t+1})$$

5: **end for**
6: **return** $\bar{\mathbf{x}}_T \stackrel{\text{def}}{=} \frac{1}{T}\sum_{t=1}^{T} \mathbf{x}_t$

Theorem 3.4

Regret bounds for OCO translate to convergence rates for stochastic optimization.

Theorem

Algorithm 9 with step sizes $\eta_t = \frac{D}{G\sqrt{t}}$ *guarantees*

$$\mathbf{E}\left[f(\overline{\mathbf{x}}_T)\right] \leq \min_{\mathbf{x}^\star \in \mathcal{K}} f(\mathbf{x}^\star) + \frac{3GD}{2\sqrt{T}}.$$

Proof of theorem 3.4

Proof.

- Define the linear functions $f_t(\mathbf{x}) \overset{\text{def}}{=} \tilde{\nabla}_t^\top \mathbf{x}$.
- $$\begin{aligned}
&\mathbf{E}\left[f(\overline{\mathbf{x}}_T)\right] - f(\mathbf{x}^\star) \\
&\leq \mathbf{E}\left[\tfrac{1}{T}\textstyle\sum_t f(\mathbf{x}_t)\right] - f(\mathbf{x}^\star) && \text{convexity of } f \text{ (Jensen)} \\
&\leq \tfrac{1}{T}\mathbf{E}\left[\textstyle\sum_t \nabla f(\mathbf{x}_t)^\top (\mathbf{x}_t - \mathbf{x}^\star)\right] && \text{convexity again} \\
&= \tfrac{1}{T}\mathbf{E}\left[\textstyle\sum_t \tilde{\nabla}_t^\top (\mathbf{x}_t - \mathbf{x}^\star)\right] && \text{noisy gradient estimator} \\
&= \tfrac{1}{T}\mathbf{E}\left[\textstyle\sum_t f_t(\mathbf{x}_t) - f_t(\mathbf{x}^\star)\right] && \text{Algorithm 9, line (3)} \\
&\leq \frac{\text{Regret}}{T} \\
&\leq \frac{3GD}{2\sqrt{T}} && \text{definition}
\end{aligned}$$

Table of Contents

Exp-Concave Functions

- Consider $f_t(\mathbf{x}) = -\log\left(\mathbf{r}_t^\top \mathbf{x}\right)$ is not strongly convex.The Hessian of this function is given by

$$\nabla^2 f_t(\mathbf{x}) = \frac{\mathbf{r}_t \mathbf{r}_t^\top}{\left(\mathbf{r}_t^\top \mathbf{x}\right)^2}$$

  which is a rank one matrix.
- An important observation is that this Hessian is large in the direction of the gradient. This property is called exp-concavity.

Exp-Concave Functions

Definition

A convex function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is defined to be α-expconcave over $\mathcal{K} \subseteq \mathbb{R}^n$ if the function g is concave, where $g : \mathcal{K} \mapsto \mathbb{R}$ is defined as

$$g(\mathbf{x}) = e^{-\alpha f(\mathbf{x})}.$$

Exp-Concave Functions

Lemma

A twice-differentiable function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is α-exp-concave at $\mathbf{x}$ if and only if

$$\nabla^2 f(\mathbf{x}) \succcurlyeq \alpha \nabla f(\mathbf{x}) \nabla f(\mathbf{x})^\top.$$

Lemma

Let $f : \mathcal{K} \to \mathbb{R}$ be an α-exp-concave function, and D, G denote the diameter of $\mathcal{K}$ and a bound on the (sub)gradients of f respectively. The following holds for all $\gamma \leq \frac{1}{2} \min\left\{\frac{1}{GD}, \alpha\right\}$ and all $\mathbf{x}, \mathbf{y} \in \mathcal{K}$:

$$f(\mathbf{x}) \geq f(\mathbf{y}) + \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}) + \frac{\gamma}{2} (\mathbf{x} - \mathbf{y})^\top \nabla f(\mathbf{y}) \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}).$$

The Online Newton Step Algorithm

- Consider a quasi-Newton approach, i.e., an online convex optimization algorithm that approximates the second derivative, or Hessian in more than one dimension.
- At each iteration, online Newton step algorithm chooses a vector that is the projection of the sum of the vector chosen at the previous iteration and an additional vector.
- Since adding a multiple of the Newton vector $A_t^{-1}\nabla_t$ to the current vector may result in a point outside the convex set, an additional projection step is required to obtain $\mathbf{x}_t$, the decision at time t.
- This projection is different than the standard Euclidean projection used by online gradient descent. It is the projection according to the norm defined by the matrix A_t, rather than the Euclidean norm.

The Online Newton Step Algorithm

Algorithm 12 online Newton step

Input: convex set $\mathcal{K}$, T, $\mathbf{x}_1 \in \mathcal{K} \subseteq \mathbb{R}^n$, parameters $\gamma, \varepsilon > 0$, $A_0 = \varepsilon \mathbf{I}_n$

for $t = 1$ to T **do**

  Play $\mathbf{x}_t$ and observe cost $f_t(\mathbf{x}_t)$.

  Rank-1 update: $A_t = A_{t-1} + \nabla_t \nabla_t^\top$

  Newton step and generalized projection:

$$\mathbf{y}_{t+1} = \mathbf{x}_t - \frac{1}{\gamma} A_t^{-1} \nabla_t$$

$$\mathbf{x}_{t+1} = \prod_{\mathcal{K}}^{A_t} (\mathbf{y}_{t+1}) = \arg\min_{\mathbf{x} \in \mathcal{K}} \left\{ \|\mathbf{y}_{t+1} - \mathbf{x}\|_{A_t}^2 \right\}$$

end for

Theorem 4.2

Theorem

Algorithm 12 with parameters $\gamma = \frac{1}{2}\min\left\{\frac{1}{GD}, \alpha\right\}, \varepsilon = \frac{1}{\gamma^2 D^2}$ *and* $T \geq 4$ *guarantees*

$$\textit{Regret}_T \leq 2\left(\frac{1}{\alpha} + GD\right) n \log T.$$

Lemma

The regret of online Newton step is bounded by

$$\text{Regret}_T(\textit{ONS}) \leq \left(\frac{1}{\alpha} + GD\right)\left(\sum_{t=1}^{T} \nabla_t^\top A_t^{-1} \nabla_t + 1\right).$$

Proof of lemma

Proof.

- Let $\mathbf{x}^\star \in \arg\min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x})$ be the best decision in hindsight.
- For $\gamma = \frac{1}{2}\min\left\{\frac{1}{GD}, \alpha\right\}, f_t(\mathbf{x}_t) - f_t(\mathbf{x}^\star) \leq R_t$, where define $R_t \stackrel{\text{def}}{=} \nabla_t^\top (\mathbf{x}_t - \mathbf{x}^\star) - \frac{\gamma}{2}(\mathbf{x}^\star - \mathbf{x}_t)^\top \nabla_t \nabla_t^\top (\mathbf{x}^\star - \mathbf{x}_t)$.
- $$\mathbf{y}_{t+1} - \mathbf{x}^\star = \mathbf{x}_t - \mathbf{x}^\star - \frac{1}{\gamma} A_t^{-1} \nabla_t, \text{ and}$$
$$A_t(\mathbf{y}_{t+1} - \mathbf{x}^\star) = A_t(\mathbf{x}_t - \mathbf{x}^\star) - \frac{1}{\gamma}\nabla_t$$
- $$(\mathbf{y}_{t+1} - \mathbf{x}^\star)^\top A_t (\mathbf{y}_{t+1} - \mathbf{x}^\star) =$$
$$(\mathbf{x}_t - \mathbf{x}^\star)^\top A_t (\mathbf{x}_t - \mathbf{x}^\star) - \frac{2}{\gamma}\nabla_t^\top (\mathbf{x}_t - \mathbf{x}^\star) + \frac{1}{\gamma^2}\nabla_t^\top A_t^{-1} \nabla_t.$$

Proof of lemma

Proof.

- $$\begin{aligned}
&\left(\mathbf{y}_{t+1}-\mathbf{x}^{\star}\right)^{\top} A_t\left(\mathbf{y}_{t+1}-\mathbf{x}^{\star}\right)=\left\|\mathbf{y}_{t+1}-\mathbf{x}^{\star}\right\|_{A_t}^2 \\
&\geq\left\|\mathbf{x}_{t+1}-\mathbf{x}^{\star}\right\|_{A_t}^2 \\
&=\left(\mathbf{x}_{t+1}-\mathbf{x}^{\star}\right)^{\top} A_t\left(\mathbf{x}_{t+1}-\mathbf{x}^{\star}\right).
\end{aligned}$$

- $$\begin{aligned}
\nabla_t^{\top}\left(\mathbf{x}_t-\mathbf{x}^{\star}\right) \leq & \frac{1}{2\gamma} \nabla_t^{\top} A_t^{-1} \nabla_t+\frac{\gamma}{2}\left(\mathbf{x}_t-\mathbf{x}^{\star}\right)^{\top} A_t\left(\mathbf{x}_t-\mathbf{x}^{\star}\right) \\
& -\frac{\gamma}{2}\left(\mathbf{x}_{t+1}-\mathbf{x}^{\star}\right)^{\top} A_t\left(\mathbf{x}_{t+1}-\mathbf{x}^{\star}\right)
\end{aligned}$$

Proof of lemma

Proof.

- $$\sum_{t=1}^{T} \nabla_t^\top (\mathbf{x}_t - \mathbf{x}^\star) \leq \frac{1}{2\gamma} \sum_{t=1}^{T} \nabla_t^\top A_t^{-1} \nabla_t + \frac{\gamma}{2} (\mathbf{x}_1 - \mathbf{x}^\star)^\top A_1 (\mathbf{x}_1 - \mathbf{x}^\star)$$
$$+ \frac{\gamma}{2} \sum_{t=2}^{T} (\mathbf{x}_t - \mathbf{x}^\star)^\top (A_t - A_{t-1}) (\mathbf{x}_t - \mathbf{x}^\star)$$
$$- \frac{\gamma}{2} (\mathbf{x}_{T+1} - \mathbf{x}^\star)^\top A_T (\mathbf{x}_{T+1} - \mathbf{x}^\star)$$
$$\leq \frac{1}{2\gamma} \sum_{t=1}^{T} \nabla_t^\top A_t^{-1} \nabla_t + \frac{\gamma}{2} \sum_{t=1}^{T} (\mathbf{x}_t - \mathbf{x}^\star)^\top \nabla_t \nabla_t^\top (\mathbf{x}_t - \mathbf{x}^\star)$$
$$+ \frac{\gamma}{2} (\mathbf{x}_1 - \mathbf{x}^\star)^\top \left(A_1 - \nabla_1 \nabla_1^\top \right) (\mathbf{x}_1 - \mathbf{x}^\star)$$

Proof of lemma

Proof.

- $$\sum_{t=1}^{T} R_t \leq \frac{1}{2\gamma} \sum_{t=1}^{T} \nabla_t^\top A_t^{-1} \nabla_t + \frac{\gamma}{2} \left(\mathbf{x}_1 - \mathbf{x}^\star\right)^\top \left(A_1 - \nabla_1 \nabla_1^\top\right) \left(\mathbf{x}_1 - \mathbf{x}^\star\right)$$
- Use the algorithm parameters $A_1 - \nabla_1 \nabla_1^\top = \varepsilon \mathbf{I}_n, \varepsilon = \frac{1}{\gamma^2 D^2}$ and $\left\|\mathbf{x}_1 - \mathbf{x}^\star\right\|^2 \leq D^2$.
- $$\begin{aligned} \text{Regret}_T(ONS) &\leq \sum_{t=1}^{T} R_t \leq \frac{1}{2\gamma} \sum_{t=1}^{T} \nabla_t^\top A_t^{-1} \nabla_t + \frac{\gamma}{2} D^2 \varepsilon \\ &\leq \frac{1}{2\gamma} \sum_{t=1}^{T} \nabla_t^\top A_t^{-1} \nabla_t + \frac{1}{2\gamma} \end{aligned}$$

Proof of theorem 4.2

Proof.

- $\nabla_t^\top A_t^{-1} \nabla_t = A_t^{-1} \bullet \nabla_t \nabla_t^\top = A_t^{-1} \bullet (A_t - A_{t-1})$, where for matrices $A, B \in \mathbb{R}^{n \times n}$ we denote by $A \bullet B = \sum_{i=1}^n \sum_{j=1}^n A_{ij} B_{ij} = \mathrm{Tr}\left(AB^\top\right)$.
- For positive semidefinite matrices, i.e., $A^{-1} \bullet (A - B) \leq \log \frac{|A|}{|B|}$, where $|A|$ denotes the determinant of the matrix A.
- $$\begin{aligned}\sum_{t=1}^T \nabla_t^\top A_t^{-1} \nabla_t &= \sum_{t=1}^T A_t^{-1} \bullet \nabla_t \nabla_t^\top \\ &= \sum_{t=1}^T A_t^{-1} \bullet (A_t - A_{t-1}) \\ &\leq \sum_{t=1}^T \log \frac{|A_t|}{|A_{t-1}|} = \log \frac{|A_T|}{|A_0|}\end{aligned}$$

Proof of theorem 4.2

Proof.

- Since $A_T = \sum_{t=1}^{T} \nabla_t \nabla_t^\top + \varepsilon I_n$ and $\|\nabla_t\| \leq G$, the largest eigenvalue of A_T is at most $TG^2 + \varepsilon$. Hence the determinant of A_T can be bounded by $|A_T| \leq \left(TG^2 + \varepsilon\right)^n$.
- Recalling that $\varepsilon = \frac{1}{\gamma^2 D^2}$ and $\gamma = \frac{1}{2}\min\left\{\frac{1}{GD}, \alpha\right\}$, for $T > 4$,

$$\sum_{t=1}^{T} \nabla_t^\top A_t^{-1} \nabla_t \leq \log\left(\frac{TG^2 + \varepsilon}{\varepsilon}\right)^n \leq n \log\left(TG^2\gamma^2 D^2 + 1\right) \leq n \log T$$

- $$\text{Regret}_T(\text{ONS}) \leq \left(\frac{1}{\alpha} + GD\right)(n \log T + 1),$$

  which implies the theorem for $n > 1, T \geq 4$.