$O(\sqrt{T})$ Static Regret and Instance Dependent Constraint Violation for Constrained Online Convex Optimization

Rahul Vaze[*]
School of Technology and Computer Science
Tata Institute of Fundamental Research, Mumbai
rahul.vaze@gmail.com

Abhishek Sinha
School of Technology and Computer Science
Tata Institute of Fundamental Research, Mumbai
abhishek.sinha@tifr.res.in

Abstract

The constrained version of the standard online convex optimization (OCO) framework, called COCO is considered, where on every round, a convex cost function and a convex constraint function are revealed to the learner after it chooses the action for that round. The objective is to simultaneously minimize the static regret and cumulative constraint violation (CCV). An algorithm is proposed that guarantees a static regret of $O(\sqrt{T})$ and a CCV of $\min\{\mathscr{V}, O(\sqrt{T}\log T)\}$, where $\mathscr{V}$ depends on the distance between the consecutively revealed constraint sets, the shape of constraint sets, dimension of action space and the diameter of the action space. When constraint sets have additional structure, $\mathscr{V} = O(1)$. Compared to the state of the art results, static regret of $O(\sqrt{T})$ and CCV of $O(\sqrt{T}\log T)$, that were universal, the new result on CCV is instance dependent, which is derived by exploiting the geometric properties of the constraint sets.

1 Introduction

In this paper, we consider the constrained version of the standard online convex optimization (OCO) framework, called constrained OCO or COCO. In COCO, on every round t, the online algorithm first chooses an admissible action $x_t \in \mathscr{X} \subset \mathbb{R}^d$, and then the adversary chooses a convex loss/cost function $f_t : \mathscr{X} \to \mathbb{R}$ and a constraint function of the form $g_t(x) \leq 0$, where $g_t : \mathscr{X} \to \mathbb{R}$ is a convex function. Since g_t's are revealed after the action x_t is chosen, an online algorithm need not necessarily take feasible actions on each round, and in addition to the static regret

$$\text{Regret}_{[1:T]} \equiv \sup_{\{f_t\}_{t=1}^T} \sup_{x^\star \in \mathscr{X}} \text{Regret}_T(x^\star), \text{ where } \text{Regret}_T(x^\star) \equiv \sum_{t=1}^T f_t(x_t) - \sum_{t=1}^T f_t(x^\star), \quad (1)$$

an additional metric of interest is the total cumulative constraint violation (CCV) defined as $\text{CCV}_{[1:T]} \equiv \sum_{t=1}^T \max(g_t(x_t), 0)$. Let $\mathscr{X}^\star$ be the feasible set consisting of all admissible actions that satisfy all constraints $g_t(x) \leq 0, t \in [T]$. Under the standard assumption that $\mathscr{X}^\star$ is not

[*]Both authors acknowledge support of the Department of Atomic Energy, Government of India, under project no. RTI4001 and Google India Research Award 2023.

empty (called the *feasibility assumption*), the goal is to design an online algorithm to simultaneously achieve a small regret (1) with respect to any admissible benchmark $x^\star \in \mathscr{X}^\star$ and a small CCV.

With constraint sets $\mathscr{G}_t = \{x \in \mathscr{X} : g_t(x) \leq 0\}$ being convex for all t, and the assumption $\mathscr{X}^\star = \cap_t \mathscr{G}_t \neq \varnothing$ implies that sets $S_t = \cap_{\tau=1}^t \mathscr{G}_\tau$ are convex and are nested, i.e. $S_t \subseteq S_{t-1}$ and $\mathscr{X}^\star \in S_t$ for all t. Essentially, set S_t's are sufficient to quantify the CCV.

1.1 Prior Work

Constrained OCO (COCO): (A) Time-invariant constraints: COCO with time-invariant constraints, *i.e.,* $g_t = g, \forall\, t$ [Yuan and Lamperski, 2018, Jenatton et al., 2016, Mahdavi et al., 2012, Yi et al., 2021] has been considered extensively, where functions g are assumed to be known to the algorithm *a priori*. The algorithm is allowed to take actions that are infeasible at any time to avoid the costly projection step of the vanilla projected OGD algorithm and the main objective was to design an *efficient* algorithm with a small regret and CCV while avoiding the explicit projection step.

(B) Time-varying constraints: The more difficult question is solving COCO problem when the constraint functions, *i.e.,* g_t's, change arbitrarily with time t. In this setting, all prior work on COCO made the feasibility assumption. One popular algorithm for solving COCO considered a Lagrangian function optimization that is updated using the primal and dual variables [Yu et al., 2017, Sun et al., 2017, Yi et al., 2023]. Alternatively, [Neely and Yu, 2017] and [Liakopoulos et al., 2019] used the drift-plus-penalty (DPP) framework [Neely, 2010] to solve the COCO, but which needed additional assumption, e.g. the Slater's condition in [Neely and Yu, 2017] and with weaker form of the feasibility assumption [Neely and Yu, 2017]'s.

[Guo et al., 2022] obtained the bounds similar to [Neely and Yu, 2017] but without assuming Slater's condition. However, the algorithm [Guo et al., 2022] was quite computationally intensive since it requires solving a convex optimization problem on each round. Finally, very recently, the state of the art guarantees on simultaneous bounds on regret $O(\sqrt{T})$ and CCV $O(\sqrt{T}\log T)$ for COCO were derived in [Sinha and Vaze, 2024] with a very simple algorithm that combines the loss function at time t and the CCV accrued till time t in a single loss function, and then executes the online gradient descent (OGD) algorithm on the single loss function with an adaptive step-size. Another extension of [Sinha and Vaze, 2024] can be found in [Lekeufack and Jordan, 2025] that considers COCO problem under predictions about f_t's and g_t's. See Remark 6 for comparison of this work with [Lekeufack and Jordan, 2025]. Please refer to Table 1 for a brief summary of the prior results.

The COCO problem has been considered in the *dynamic* setting as well [Chen and Giannakis, 2018, Cao and Liu, 2018, Vaze, 2022, Liu et al., 2022] where the benchmark $x^\star$ in (1) is replaced by $x_t^\star$ ($x_t^\star = \arg\min_x f_t(x)$) that is also allowed to change its actions over time. However, in this paper, we focus our entire attention on the static version. A special case of COCO is the online constraint satisfaction (OCS) problem that does not involve any cost function, *i.e.,* $f_t = 0,\ \forall t$, and the only object of interest is minimizing the CCV. The algorithm with state of the art guarantee for COCO [Sinha and Vaze, 2024] was shown to have a CCV of $O(\sqrt{T}\log T)$ for the OCS.

1.2 Convex Body Chasing Problem

A well-studied problem related to the COCO is the *nested convex body chasing (NCBC)* problem [Bansal et al., 2018, Argue et al., 2019, Bubeck et al., 2020], where at each round t, a convex set $\chi_t \subseteq \chi$ is revealed such that $\chi_t \subseteq \chi_{t-1}$, and $\chi_0 = \chi \subseteq \mathbb{R}^d$ is a convex, compact, and bounded set. The objective is to choose action $x_t \in \chi_t$ so as to minimize the total movement cost $C = \sum_{t=1}^T ||x_t - x_{t-1}||$, where $x_0 \in \chi$ is some fixed action. Best known-algorithms for NCBC [Bansal et al., 2018, Argue et al., 2019, Bubeck et al., 2020] choose x_t to be the centroid or Steiner point of χ_t, essentially well inside the newly revealed convex set in order to reduce the future movement cost. With COCO, such an approach does not appear useful because of the presence of cost functions f_t's whose minima could be towards the boundary of convex sets χ_t's.

1.3 Limitations of Prior Work

We explicitly show in Lemma 6 that the best known algorithm [Sinha and Vaze, 2024] (in terms of regret and up to log factors for CCV) for solving COCO suffers a CCV of $\Omega(\sqrt{T}\log T)$ even for 'simple' problem instances where $f_t = f$ and $g_t = g$ for all t and $d = 1$ dimension, for which ideally the CCV should be $O(1)$. The same is true for most other algorithms, where the main reason for their large CCV for simple instances is that all these algorithms treat minimizing the CCV as a regret minimization problem for functions g_t. What they fail to exploit is the geometry of the underlying nested convex sets S_t's that control the CCV.

1.4 Main open question

In comparison to the above discussed upper bounds, the best known simultaneous lower bound [Sinha and Vaze, 2024] for COCO is $\mathscr{R}_{[1:T]} = \Omega(\sqrt{d})$ and $\text{CCV}_{[1:T]} = \Omega(\sqrt{d})$, where d is the dimension of the action space $\mathscr{X}$. Without constraints, i.e., $g_t \equiv 0$ for all t, the lower bound on $\mathscr{R}_{[1:T]} = \Omega(\sqrt{T})$ [Hazan, 2012]. Thus, there is a fundamental gap between the lower and upper bound for the CCV, and the main open question for COCO is : *Is it possible to simultaneously achieve $\mathscr{R}_{[1:T]} = O(\sqrt{T})$ and $CCV_{[1:T]} = o(\sqrt{T})$ or $CCV_{[1:T]} = O(1)$* for COCO? Even though we do not fully resolve this question, in this paper, we make some meaningful progress by proposing an algorithm that exploits the geometry of the nested sets S_t's and show that it is possible to simultaneously achieve $\mathscr{R}_{[1:T]} = O(\sqrt{T})$ and $\text{CCV}_{[1:T]} = O(1)$ in certain cases, and for general case, give a bound on the CCV that depends on the shape of the convex sets S_t's while achieving $\mathscr{R}_{[1:T]} = O(\sqrt{T})$. In particular, the contributions of this paper are as follows.

1.5 Our Contributions

In this paper, we propose an algorithm (Algorithm 2) that tries to exploit the geometry of the nested convex sets S_t's. In particular, Algorithm 2 at time t, first takes an OGD step from the previous action x_{t-1} with respect to the most recently revealed loss function f_{t-1} with appropriate step-size to reach y_{t-1}, and then projects y_{t-1} onto the most recently revealed set S_{t-1} to get x_t, the action to be played at time t. Let F_t be the "projection" hyperplane passing through x_t that is perpendicular to $x_t - y_{t-1}$. For Algorithm 2, we derive the following guarantees.

- The regret of the Algorithm 2 is $O(\sqrt{T})$.
- The CCV for the Algorithm 2 takes the following form
  - When sets S_t's are structured, e.g. are spheres, or axis parallel cuboids/regular polygons, CCV is $O(1)$.
  - For the special case of $d = 2$, when projection hyperplanes F_t's progressively make increasing angles with respect to the first projection hyperplane F_1, the CCV is $O(1)$.
  - For general S_t's, the CCV is upper bounded by a quantity $\mathscr{V}$ that is a function of the distance between the consecutive sets S_t and S_{t+1} for all t, the shape of S_t's, dimension d and the diameter D. Since $\mathscr{V}$ depends on the shape of S_t's, there is no universal bound on $\mathscr{V}$, and the derived bound is instance dependent.
- As pointed out above, for general S_t's, there is no universal bound on the CCV of Algorithm 2. Thus, we propose an algorithm Switch that combines Algorithm 2 and the algorithm from [Sinha and Vaze, 2024] to provide a regret bound of $O(\sqrt{T})$ and a CCV that is minimum of $\mathscr{V}$ and $O(\sqrt{T}\log T)$. Thus, Switch provides a best of two worlds CCV guarantee, which is small if the sets S_t's are 'nice', while in the worst case it is at most $O(\sqrt{T}\log T)$.
- For the OCS problem, where $f_t = 0,\ \forall\ t$, we show that the CCV of Algorithm 2 is $O(1)$ compared to the CCV of $O(\sqrt{T}\log T)$ [Sinha and Vaze, 2024].

2 COCO Problem

On round t, the online policy first chooses an admissible action $x_t \in \mathscr{X} \subset \mathbb{R}^d$, and then the adversary chooses a convex cost function $f_t : \mathscr{X} \to \mathbb{R}$ and a constraint of the form $g_t(x) \leq 0$, where $g_t : \mathscr{X} \to \mathbb{R}$ is a convex function. Once the action x_t has been chosen, we let $\nabla f_t(x_t)$ and full function g_t or the set $\{x : g_t(x) \leq 0\}$ to be revealed, as is standard in the literature. We now state the standard assumptions made in the literature while studying the COCO problem [Guo et al., 2022, Yi et al., 2021, Neely and Yu, 2017, Sinha and Vaze, 2024].

Assumption 1 (Convexity) *$\mathscr{X} \subset \mathbb{R}^d$ is the admissible set that is closed, convex and has a finite Euclidean diameter D. The cost function $f_t : \mathscr{X} \mapsto \mathbb{R}$ and the constraint function $g_t : \mathscr{X} \mapsto \mathbb{R}$ are convex for all $t \geq 1$.*

Reference	Regret	CCV	Complexity per round
[Neely and Yu, 2017],	$O(\sqrt{T})$	$O(\sqrt{T})$	Conv-OPT, Slater's condition
[Liakopoulos et al., 2019]	$O(\sqrt{T})$	$O(\sqrt{T})$	Conv-OPT, Slater's condition
[Guo et al., 2022]	$O(\sqrt{T})$	$O(T^{\frac{3}{4}})$	Conv-OPT
[Yi et al., 2023]	$O(T^{\max(\beta,1-\beta)})$	$O(T^{1-\beta/2})$	Conv-OPT
[Sinha and Vaze, 2024]	$O(\sqrt{T})$	$O(\sqrt{T}\log T)$	Projection
This paper	$O(\sqrt{T})$	$O(\min\{\mathscr{V}, \sqrt{T}\log T\})$	Projection

Table 1: Summary of the results on COCO for arbitrary time-varying convex constraints and convex cost functions. In the above table, $0 \leq \beta \leq 1$ is an adjustable parameter. Conv-OPT refers to solving a constrained convex optimization problem on each round. Projection refers to the Euclidean projection operation on the convex set $\mathscr{X}$. The CCV bound for this paper is stated in terms of $\mathscr{V}$ which can be $O(1)$ or depends on the shape of convex sets S_t's.

Assumption 2 (Lipschitzness) *All cost functions $\{f_t\}_{t\geq 1}$ and the constraint functions $\{g_t\}_{t\geq 1}$'s are G-Lipschitz, i.e., for any $x, y \in \mathscr{X}$, we have $|f_t(x) - f_t(y)| \leq G||x-y||,\ |g_t(x) - g_t(y)| \leq G||x-y||,\ \forall t \geq 1$.*

Assumption 3 (Feasibility) *With $\mathscr{G}_t = \{x \in \mathscr{X} : g_t(x) \leq 0\}$, we assume that $\mathscr{X}^\star = \cap_{t=1}^T \mathscr{G}_t \neq \varnothing$. Any action $x^\star \in \mathscr{X}^\star$ is defined to be feasible.*

The feasibility assumption distinguishes the cost functions from the constraint functions and is common across all previous literature on COCO [Guo et al., 2022, Neely and Yu, 2017, Yu and Neely, 2016, Yuan and Lamperski, 2018, Yi et al., 2023, Liakopoulos et al., 2019, Sinha and Vaze, 2024].

For any real number z, we define $(z)^+ \equiv \max(0, z)$. Since g_t's are revealed after the action x_t is chosen, any online policy need not necessarily take feasible actions on each round. Thus in addition to the static[2] regret defined below

$$\text{Regret}_{[1:T]} \equiv \sup_{\{f_t\}_{t=1}^T} \sup_{x^\star \in \mathscr{X}^\star} \text{Regret}_{[1:T]}(x^\star), \quad \text{Regret}_{[1:T]}(x^\star) \equiv \sum_{t=1}^T f_t(x_t) - \sum_{t=1}^T f_t(x^\star) \tag{2}$$

where an additional obvious metric of interest is the total cumulative constraint violation (CCV) defined as $\text{CCV}_{[1:T]} = \sum_{t=1}^T (g_t(x_t))^+$. Under the standard assumption (Assumption 3) that $\mathscr{X}^\star$ is not empty, the goal is to design an online policy to simultaneously achieve a small regret with $x^\star \in \mathscr{X}^\star$ and a small CCV.

For simplicity, we define set

$$S_t = \cap_{\tau=1}^t \mathscr{G}_\tau, \tag{3}$$

where $\mathscr{G}_t$ is as defined in Assumption 3. All $\mathscr{G}_t$'s are convex and consequently, all S_t's are convex and are nested, i.e. $S_t \subseteq S_{t-1}$. Moreover, because of Assumption 3, each S_t is non-empty and in particular $\mathscr{X}^\star \in S_t$ for all t. After action x_t has been chosen, set S_t controls the constraint violation, which can be used to write an upper bound on the $\text{CCV}_{[1:T]}$ as follows.

Definition 4 *For a convex set χ and a point $x \notin \chi$, dist$(x, \chi) = \min_{y\in\chi} ||x-y||$.*

With G being the common Lipschitz constants for all g_t's, the constraint violation at time t,

$$(g_t(x_t))^+ \leq G\text{dist}(x_t, S_t), \text{ and } \text{CCV}_{[1:T]} \leq G\sum_{t=1}^T \text{dist}(x_t, S_t). \tag{4}$$

3 Algorithm from Sinha and Vaze [2024]

The best known algorithm (Algorithm 1) to solve COCO Sinha and Vaze [2024] (in terms of regret and up to log factors for CCV) was shown to have the following guarantee.

[2] The static-ness refers to the fixed benchmark using only one action $x^\star$ throughout the horizon of length T

Algorithm 1 Online Algorithm from Sinha and Vaze [2024]

Input: Sequence of convex cost functions $\{f_t\}_{t=1}^T$ and constraint functions $\{g_t\}_{t=1}^T$, $G =$ a common Lipschitz constant, $T =$ Horizon length, $D =$ Euclidean diameter of the admissible set $\mathscr{X}$, $\mathscr{P}_{\mathscr{X}}(\cdot) =$ Euclidean projection oracle on the set $\mathscr{X}$
Let $\beta = (2GD)^{-1}, V = 1, \lambda = \frac{1}{2\sqrt{T}}, \Phi(x) = \exp(\lambda x) - 1.$
Initialization: Set $x_1 = \mathbf{0}, \text{CCV}(0) = 0.$
For $t = 1 : T$
  Play x_t, observe f_t, g_t, incur a cost of $f_t(x_t)$ and constraint violation of $(g_t(x_t))^+$
  $\tilde{f}_t \leftarrow \beta f_t, \tilde{g}_t \leftarrow \beta \max(0, g_t).$
  $\text{CCV}(t) = \text{CCV}(t-1) + \tilde{g}_t(x_t).$
  Compute $\nabla_t = \nabla \hat{f}_t(x_t)$, where $\hat{f}_t(x) := V\tilde{f}_t(x) + \Phi'(\text{CCV}(t))\tilde{g}_t(x), \;\; t \geq 1.$
  $x_{t+1} = \mathscr{P}_{\mathscr{X}}(x_t - \eta_t \nabla_t)$, where $\quad \eta_t = \frac{\sqrt{2}D}{2\sqrt{\sum_{\tau=1}^t ||\nabla_\tau||_2^2}}$
EndFor

Theorem 5 *[Sinha and Vaze [2024]] Algorithm 1's Regret*$_{[1:T]} = O(\sqrt{T})$ *and CCV*$_{[1:T]} = O(\sqrt{T}\log T)$ *when* f_t, g_t *are convex.*

We next show that in fact the analysis of Sinha and Vaze [2024] is tight for the CCV even when $d = 1$ and $f_t(x) = f(x)$ and $g_t(x) = g(x)$ for all t. With finite diameter D and the fact that any $x^\star \in \mathscr{X}^\star$ belongs to all nested convex bodies S_t's, when $d = 1$, one expects that the CCV for any algorithm in this case will be $O(D)$. However, as we show next, Algorithm 1 does not effectively make use of geometric constraints imposed by nested convex bodies S_t's.

Lemma 6 *Even when* $d = 1$ *and* $f_t(x) = f(x)$ *and* $g_t(x) = g(x)$ *for all* t*, for Algorithm 1, its CCV*$_{[1:T]} = \Omega(\sqrt{T}\log T)$.

Proof: Input: Consider $d = 1$, and let $\mathscr{X} = [1, a], a > 2$. Moreover, let $f_t(x) = f(x)$ and $g_t(x) = g(x)$ for all t. Let $f(x) = cx^2$ for some (large) $c > 0$ and $g(x)$ be such that $G = \{x : g(x) \leq 0\} \subseteq [a/2, a]$ and let $|\nabla g(x)| \leq 1$ for all x.

Let $1 < x_1 < a/2$. Note that $\text{CCV}(t)$ (defined in Algorithm 1) is a non-decreasing function, and let $t^\star$ be the earliest time t such that $\Phi'(\text{CCV}(t))\nabla g(x) < -c$. For $f(x) = cx^2$, $\nabla f(x) \geq c$ for all $x > 1$. Thus, using Algorithm 1's definition, it follows that for all $t \leq t^\star$, $x_t < a/2$, since the derivative of f dominates the derivative of $\Phi'(\text{CCV}(t))g(x)$ until then.

Since $\Phi(x) = \exp(\lambda x) - 1$ with $\lambda = \frac{1}{2\sqrt{T}}$, and by definition $|\nabla g(x)| \leq 1$ for all x, thus, we have that by time $t^\star$, $\text{CCV}_{[1:t^\star]} = \Omega(\sqrt{T}\log T)$. Therefore, $\text{CCV}_{[1:T]} = \Omega(\sqrt{T}\log T)$. □

Essentially, Algorithm 1 is treating minimizing the CCV problem as regret minimization for function g similar to function f and this leads to its CCV of $\Omega(\sqrt{T}\log T)$. For any given input instance with $d = 1$, an alternate algorithm that chooses its actions following online gradient descent (OGD) projected on to the most recently revealed feasible set S_t achieves $O(\sqrt{T})$ regret (irrespective of the starting action x_1) and $O(D)$ CCV (since any $x^\star \in S_t$ for all t). We extend this intuition in the next section, and present an algorithm that exploits the geometry of the nested convex sets S_t for any d.

4 New Algorithm for solving COCO

In this section, we present a simple algorithm (Algorithm 2) for solving COCO. Algorithm 2 is essentially an online projected gradient algorithm (OGD), which first takes an OGD step from the previous action x_{t-1} with respect to the most recently revealed loss function f_{t-1} with appropriate step-size which is then projected onto S_{t-2} to reach y_{t-1}, and then projects y_{t-1} onto the most recently revealed set S_{t-1} to get x_t, the action to be played at time t. (3).

Remark 1 *Step 6 of Algorithm 2 might appear unnecessary, however, its useful for proving Theorem 12.*

Since Algorithm 2 is essentially an online projected gradient algorithm, similar to the classical result on OGD, next, we show that the regret of Algorithm 2 is $O(\sqrt{T})$.

Algorithm 2 Online Algorithm for COCO

Input: Sequence of convex cost functions $\{f_t\}_{t=1}^T$ and constraint functions $\{g_t\}_{t=1}^T$, G = a common Lipschitz constant, d dimension of the admissible set $\mathcal{X}$, step size $\eta_t = \frac{D}{G\sqrt{t}}$. D = Euclidean diameter of the admissible set $\mathcal{X}$, $\mathscr{P}_{\mathcal{X}}(\cdot)$ = Euclidean projection on the set $\mathcal{X}$,
Initialization: Set $x_1 \in \mathcal{X}$ arbitrarily, $\text{CCV}(0) = 0$.
For $t = 1 : T$
  Play x_t, observe f_t, g_t, incur a cost of $f_t(x_t)$ and constraint violation of $(g_t(x_t))^+$
  Set S_t as defined in (3)
  $y_t = \mathscr{P}_{S_{t-1}}(x_t - \eta_t \nabla f_t(x_t))$
  $x_{t+1} = \mathscr{P}_{S_t}(y_t)$
EndFor

Lemma 7 *The Regret$_{[1:T]}$ for Algorithm 2 is $O(\sqrt{T})$.*

Extension of Lemma 7 when f_t's are strongly convex which results in $\text{Regret}_{[1:T]} = O(\log T)$ for Algorithm 2 follows standard arguments Hazan [2012] and is omitted.

The real challenge is to bound the total CCV for Algorithm 2. Let x_t be the action played by Algorithm 2. Then by definition, $x_t \in S_{t-1}$. Moreover, from (4), the constraint violation at time t, $\text{CCV}(t) \leq G\text{dist}(x_t, S_t)$. The next action x_{t+1} chosen by Algorithm 2 belongs to S_t, however, it is obtained by first taking an OGD step from x_t to reach y_t and then projects y_t onto S_t. Since f_t's are arbitrary, the OGD step could be towards any direction, and thus, there is no direct relationship between x_{t+1} and x_t. Informally, $(x_1, x_2, \dots, x_T)$ is not a connected curve with any useful property. Thus, we take recourse in upper bounding the CCV via upper bounding the total movement cost M (defined below) between nested convex sets using projections.
The total constraint violation for Algorithm 2 is

$$\text{CCV}_{[1:t]} \leq G \sum_{\tau=1}^{t} \text{dist}(x_\tau, S_\tau) \overset{(a)}{\leq} G \sum_{\tau=1}^{t} ||x_\tau - b_\tau|| \overset{(b)}{=} GM_t, \tag{5}$$

where in (a) b_t is the projection of x_t onto S_t, i.e., $b_t = \mathscr{P}_{S_t}(x_t)$ and in (b) $M_t = \sum_{\tau=1}^t ||x_\tau - b_\tau||$ is defined to be the total movement cost on the instance $S_1, \dots, S_t$. The object of interest is $\mathbf{M_T}$.

5 Bounding the Total Movement Cost M_T for Algorithm 2

We start by considering structured problem instances where CCV of Algorithm 2 is $O(1)$, i.e., independent of T.

Lemma 8 *If all nested convex bodies $S_1 \supseteq S_2 \supseteq \cdots \supseteq S_T$ are spheres then $M_T \leq d^{3/2}D = O(1)$.*

Lemma 9 *If all nested convex bodies $S_1 \supseteq S_2 \supseteq \cdots \supseteq S_T$ are cuboids/regular polygons that are axis parallel to each other, then $M_T \leq d^{3/2}D = O(1)$.*

Interestingly, input instance where S_t's are axis-parallel cuboids has been used to derive the only known lower bound for COCO of $\text{Regret}_{[1:T]} = O(\sqrt{d})$ and $\text{CCV}_{[1:T]} = O(\sqrt{d})$ [Sinha and Vaze, 2024].

Remark 2 *Lemma 8 and 9 are first results of its kind in COCO, where even for nicely structured instances the previous best known guarantee is CCV$_{[1:T]} = O(\sqrt{T}\log T)$ [Sinha and Vaze, 2024] or CCV$_{[1:T]} = O(\sqrt{T})$ [Ferreira and Soares, 2025].*

Next, we show that similar $O(1)$ CCV guarantee can be obtained for Algorithm 2 with less structured input, however, only when $d = 2$.

5.1 Special case of $d = 2$

In this section, we show that if $d = 2$ (all convex sets S_t's lie in a plane) and the projections satisfy a monotonicity property depending on the problem instance, then we can bound the total CCV for Algorithm 2 independent of the time horizon T and consequently getting a $O(1)$ CCV.

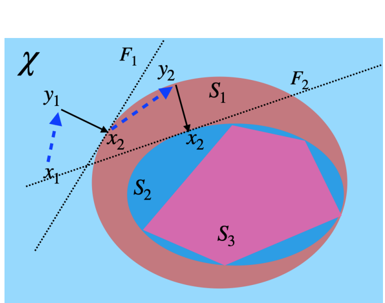


Figure 1: Definition of F_t's.

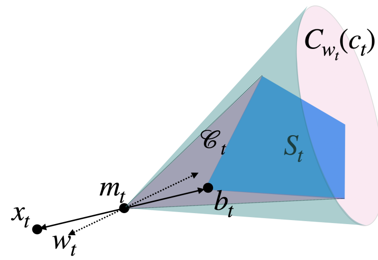


Figure 2: Figure representing the cone $C_{w_t}(c_t)$ that contains the convex hull of m_t and S_t with unit vector w_t..

Definition 10 *Recall from the definition of Algorithm 2,* $y_t = \mathscr{P}_{S_{t-1}}(x_t - \eta_t \nabla f_t(x_t))$ *and* $x_{t+1} = \mathscr{P}_{S_t}(y_t)$*. Let the hyperplane perpendicular to line segment* (y_t, x_{t+1}) *passing through* x_{t+1} *be* F_t*. Without loss of generality, we let* $y_t \notin S_t$*, since otherwise the projection is trivial. Essentially* F_t *is the projection hyperplane at time* t*. Let* $\mathscr{H}_t^+$ *denote the positive half plane corresponding to* F_t*, i.e.,* $\mathscr{H}_t^+ = \{z : z^T(y_t - x_{t+1}) \geq 0\}$*. Refer to Fig. 1. Let the angle between* F_1 *and* F_t *be* θ_t*.*

Definition 11 *The instance* $S_1 \supseteq S_2 \supseteq \cdots \supseteq S_T$ *is defined to be monotonic if* $\theta_2 \leq \theta_3 \leq \cdots \leq \theta_T$*.*

Theorem 12 *For* $d = 2$ *when the instance is monotonic,* $\text{CCV}_{[1:T]} = O(GD)$ *for Algorithm 2.*

Theorem 12 shows that CCV of Algorithm 2 is independent of T as long as the instance is monotonic when $d = 2$. It is worth noting that even under the monotonicity assumption it is non-trivial to upper bound the CCV since the successive angles made by F_t's with F_1 can increase arbitrarily slowly, making it difficult to control the total CCV. The proof is derived by using basic convex geometry results from Manselli and Pucci [1991] in combination with exploiting the definition of Algorithm 2 and the monotonicity condition.

Finally, in the next subsection, we upper bound M_T, and consequently the CCV for Algorithm 2, when the input has no structure other than S_t's being nested.

5.2 General Guarantee on CCV

In this subsection, we give a general bound on M_T of Algorithm 2 for any sequence of nested convex bodies which depends on the geometry of the nested convex bodies (instance dependent). To state the result we need the following preliminaries.

Following (5), $b_t = \mathscr{P}_{S_t}(x_t)$ where $x_t \in \partial S_{t-1}$, where ∂S is the boundary of convex set S. Without loss of generality, $x_t \notin S_t$ since otherwise the distance $||x_t - b_t|| = 0$. Let m_t be the mid-point of x_t and b_t, i.e. $m_t = \frac{x_t + b_t}{2}$.

Definition 13 *Let the convex hull of* $m_t \cup S_t$ *be* $\mathscr{C}_t$*. Let* w_t *be a unit vector such that there exists* $c_t > 0$ *such that the cone*

$$C_{w_t}(c_t) = \left\{ z \in \mathbb{R}^d : -w_t^T \frac{(z - m_t)}{||(z - m_t)||} \geq c_t \right\}$$

contains $\mathscr{C}_t$*. Since* S_t *is convex, such* $w_t, c_t > 0$ *exist. For example,* $w_t = b_t - x_t$ *is one such choice for which* $c_t > 0$ *since* $m_t \notin S_t$*. See Fig. 2 for a pictorial representation.*

Let $c^\star_{w_t,t} = \arg\max_{c_t} C_{w_t}(c_t)$*,* $c^\star_t = \max_{w_t} c^\star_{w_t,t}$*, and* $w^\star_t = \arg\max_{w_t} c^\star_{w_t,t}$*. Moreover, let* $c^\star = \min_t c^\star_t$*, where by definition,* $c^\star < 1$*.*

Essentially, $2\cos^{-1}(c^\star_t)$ is the angle width of $\mathscr{C}_t$ with respect to $w^\star_t$, i.e. each element of $\mathscr{C}_t$ makes an angle of at most $\cos^{-1}(c^\star_t)$ with $w^\star_t$.

Remark 3 *Note that* $c^\star_t$ *is only a function of the distance* $||x_t - b_t||$ *and the shape of* S_t*'s, in particular, the maximum width of* S_t *along the directions perpendicular to vector* $x_t - b_t$ $\forall\, t$ *which* *can be at most the diameter D. $c^\star_t$ decreases (increasing the "width" of cone $C_{w^\star_t}(c^\star_t)$) as $||x_t - b_t||$ decreases, but small $||x_t - b_t||$ also implies small violation at time t from* (5).

Remark 4 *$c^\star$ is instance dependent or algorithm dependent? For notational simplicity, we have defined $c^\star$ using x_t's (Algorithm 2 specific quantity) and its projection b_t on S_t. However, since x_t and x_{t-1} have no useful relation between them, x_t can be any arbitrary point on the boundary of S_{t-1}, and $c^\star$ is in effect defined with respect to arbitrary $x_t \in S_{t-1}$ making it an instance-dependent quantity.*

Lemma 14 *M_T for Algorithm 2 is at most $\frac{2V_d(d-1)}{V_{d-1}}\left(\frac{1}{c^\star}\right)^d D$, where V_d is the $(d-1)$-dimensional Lebesgue measure of the unit sphere in d dimensions.*

Proof Idea Projecting $x_t \in \partial S_{t-1}$ onto S_t to get $b_t = \mathscr{P}_{S_t}(x_t)$, the diameter of S_t is at most diameter of $S_{t-1} - ||x_t - b_t||$, however, only along the direction $b_t - x_t$. Since the shape of S_t is arbitrary, as a result, the diameter of S_t need not be smaller than the diameter of S_{t-1} along any pre-specified direction, which was the main idea used to derive Lemma 8. Thus, to prove Lemma 14 we relate the distance $||x_t - b_t||$ with the decrease in **mean width** of a convex body, that is defined as the expected width of the convex body along all the directions that are chosen uniformly randomly (formal definition is provided in Definition 34).

Note that $V_d/V_{d-1} = O(1/\sqrt{d})$. Thus, from Lemma 14 we get the following **main result** of the paper for Algorithm 2 combining Lemma 7 and Lemma 14.

Theorem 15 *Algorithm 2 has Regret$_{[1:T]} = O(\sqrt{T})$, and CCV$_{[1:T]} = O\left(\sqrt{d}\left(\frac{1}{c^\star}\right)^d D\right)$.*

Theorem 15 is an instance dependent result for the CCV, compared to the prior universal guarantees of $\tilde{O}(\sqrt{T})$ on the CCV. In particular, it exploits the geometric structure of the nested convex sets S_t's and derives an upper bound on the CCV that only depends on the 'shape' of S_t's via $c^\star$. Moreover, $c^\star$ is only a dimension (d) dependent quantity (independent of T) as long as the minimum distance between consecutive constraint sets is not function of T, since the diameter D is constant, whereas *all existing algorithms* will suffer from CCV of $\Omega(\sqrt{T})$ even in this case.

Remark 5 *One pertinent question at this time is: What is $c^\star$ and why should the CCV for a problem instance necessarily depend on it? $c^\star$ corresponds to the minimum angle width (via) of the problem instance, the angular width of the 'smallest' cone containing the newly revealed constraint sets. Angle width essentially depends on the width of the convex sets in directions perpendicular to the direction of projection, and controls the total CCV, since successive convex constraint sets are nested (lie inside each other), the smaller the angle width smaller is the room that an algorithm has to violate the constraints in future steps. Angle width also depends on the distance between x_t and S_t and is potentially large when $\mathrm{d}(x_t, S_t)$ is small and the diameter along the direction perpendicular to $x_t - b_t$ is large.*

$c^\star$ is a fundamental natural object that inherently captures the geometric difficulty in bounding the CCV. The core contribution of this paper is to formalize this by bringing in the **novel concept** *of connecting the reduction of* **average width** *of the convex constraint set to the total constraint violation, that entails non-trivial convex analysis. If $c^\star$ is in fact small (e.g. total CCV is $\Omega(\sqrt{T})$) for a problem instance then that problem instance does not have enough geometric features to extract via projections. To cover for such instances, we propose the Switch algorithm next to cap the CCV by $\tilde{O}(\sqrt{T})$.*

6 Algorithm Switch

Theorem 15 provides an instance dependent bound on the CCV, that is a function of $c^\star$. If $c^\star$ is small, CCV can be larger than $O(\sqrt{T}\log T)$, the CCV guarantee of Algorithm 1 [Sinha and Vaze, 2024]. Thus, next, we marry the two algorithms, Algorithm 1 and Algorithm 2, in Algorithm 3 to provide a **best of both results** as follows.

Theorem 16 Switch *(Algorithm 3) has regret Regret$_{[1:T]} = O(\sqrt{T})$, while CCV$_{[1:T]}$ = $\min\left\{O\left(\sqrt{d}\left(\frac{1}{c^\star}\right)^d D\right), O(\sqrt{T}\log T)\right\}$.*

Algorithm 3 Switch

Input: Sequence of convex cost functions $\{f_t\}_{t=1}^T$ and constraint functions $\{g_t\}_{t=1}^T$, G = a common Lipschitz constant, d dimension of the admissible set $\mathcal{X}$, D = Euclidean diameter of the admissible set $\mathcal{X}$, $\mathcal{P}_{\mathcal{X}}(\cdot)$ = Euclidean projection operator on the set $\mathcal{X}$,
Initialization: Set $x_1 \in \mathcal{X}$ arbitrarily, CCV$(0) = 0$.
For $t = 1 : T$
  If CCV$(t-1) \leq \sqrt{T} \log T$
    Follow Algorithm 2 and update CCV(t) = CCV$(t-1) + \max\{g_t(x_t), 0\}$.
  Else
    Follow Algorithm 1 with resetting CCV$(t-1) = 0$
  EndIf
EndFor

Algorithm Switch should be understood as the best of two worlds algorithm, where the two worlds correspond to one having nice convex sets S_t's that have CCV independent of T or $o(\sqrt{T})$ for Algorithm 2, while in the other, CCV of Algorithm 2 is large on its own, and the overall CCV is controlled by discontinuing the use of Algorithm 2 once its CCV reaches $\sqrt{T}\log T$ and switching to Algorithm 1 thereafter that has universal guarantee of $O(\sqrt{T}\log T)$ on its CCV.

7 OCS Problem

In [Sinha and Vaze, 2024], a special case of COCO, called the OCS problem, was introduced where $f_t \equiv 0$ for all t. Essentially, with OCS, only constraint satisfaction is the objective. In [Sinha and Vaze, 2024], Algorithm 1 was shown to have CCV of $O(\sqrt{T}\log T)$. Next, we show that Algorithm 2 has CCV of $O(1)$ for the OCS, a remarkable improvement.

Theorem 17 *For solving OCS, Algorithm 2 has CCV$_{[1:T]} = O\left(d^{d/2} D\right) = O(1)$.*

As discussed in [Sinha and Vaze, 2024], there are important applications of OCS, and it is important to find tight bounds on its CCV. Theorem 17 achieves this by showing that CCV of $O(1)$ can be achieved, where the constant depends only on the dimension of the action space and the diameter. This is a fundamental improvement compared to the CCV bound of $O(\sqrt{T}\log T)$ from [Sinha and Vaze, 2024]. Theorem 17 is derived by using the connection between the curve obtained by successive projections on nested convex sets and self-expanded curves (Definition 20) and then using a classical result on self-expanded curves from [Manselli and Pucci, 1991].

8 Experimental Results

In this section, we compare the performance of Algorithm 1 and Algorithm 2 experimentally. We start by simulating the performance of Algorithm 1 and Algorithm 2 on the input that was used to prove Lemma 6. Fig. 3 numerically verifies the claim of Lemma 6 that the CCV of Algorithm 1 is $\Omega(\sqrt{T}\log T)$, while the CCV of Algorithm 2 remains constant.

8.1 Synthetic Data

Next, we consider a more reasonable data setup to compare the performance of Algorithm 1 and Algorithm 2, where with $d = 10$, we let $f_t(x) = ||x - a_t||_1$, and a_t is a d-dimensional vector that is coordinate-wise uniformly distributed between $[-1, 1]$ and is independent across t. Similarly, we consider $g_t(x) = \max(0, w_t^T \cdot x - 0.1)$ where w_t is a d-dimensional vector that also is coordinate-wise uniformly distributed between $[-1, 1]$ and is independent across t. This choice ensures that $x = 0$ is feasible for all constraints, i.e., Assumption 3 is satisfied. In Figs. 4a and 4b, we plot the regret and CCV, respectively, for Algorithm 1 and Algorithm 2, and see that Algorithm 2 outperforms Algorithm 1 in both the regret and the CCV.

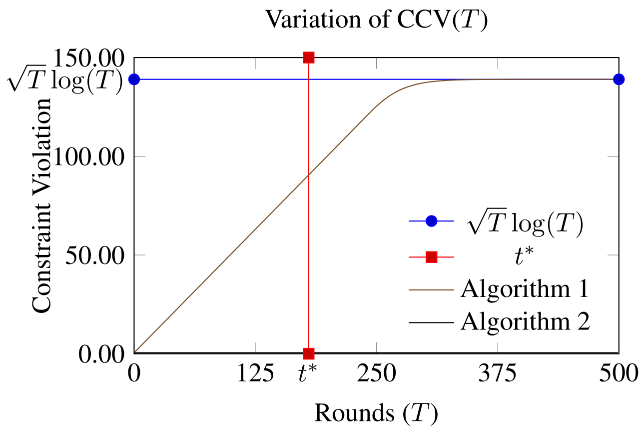
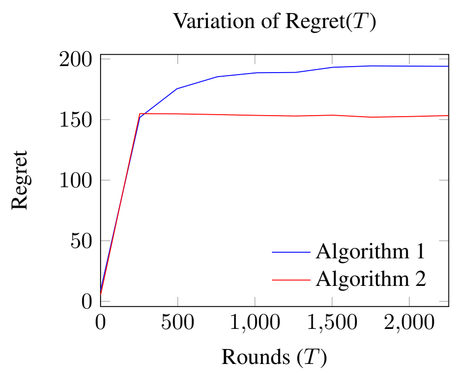
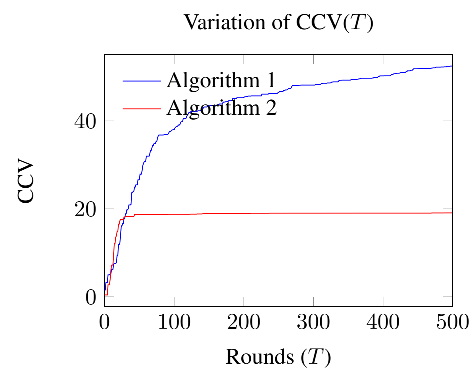


Figure 3: Regret and CCV comparison for input described in Lemma 6.



(a) Regret comparison of Algorithm 1 and Algorithm 2



(b) CCV comparison of Algorithm 1 and Algorithm 2

9 Conclusions

One fundamental open question for COCO is: whether it is possible to simultaneously achieve $\mathscr{R}_{[1:T]} = O(\sqrt{T})$ and $\text{CCV}_{[1:T]} = o(\sqrt{T})$ or $\text{CCV}_{[1:T]} = O(1)$. In this paper, we have made substantial progress towards answering this question by proposing an algorithm that exploits the geometric properties of the nested convex sets S_t's that effectively control the CCV. The state of the art algorithms [Sinha and Vaze, 2024, Ferreira and Soares, 2025] achieve a CCV of $\tilde{\Omega}(\sqrt{T})$ even for very simple instances as shown in Lemma 6, and conceptually different algorithms are needed to achieve CCV of $o(\sqrt{T})$. We propose one such algorithm and show that when the nested convex constraint sets are well structured, achieving a CCV of $O(1)$ is possible without losing out on $O(\sqrt{T})$ regret guarantee. We also derived a bound on the CCV for general problem instances, that is as a function of the shape of nested convex constraint sets and the distance between them, and the diameter.

In the absence of good lower bounds, the open question remains unresolved in general, however, this paper significantly improves the conceptual understanding of COCO problem by demonstrating that good algorithms need to exploit the geometry of the nested convex constraint sets.

References

Jianjun Yuan and Andrew Lamperski. Online convex optimization for cumulative constraints. *Advances in Neural Information Processing Systems*, 31, 2018.

Rodolphe Jenatton, Jim Huang, and Cédric Archambeau. Adaptive algorithms for online convex optimization with long-term constraints. In *International Conference on Machine Learning*, pages

402–411. PMLR, 2016.

Mehrdad Mahdavi, Rong Jin, and Tianbao Yang. Trading regret for efficiency: online convex optimization with long term constraints. *The Journal of Machine Learning Research*, 13(1):2503–2528, 2012.

Xinlei Yi, Xiuxian Li, Tao Yang, Lihua Xie, Tianyou Chai, and Karl Johansson. Regret and cumulative constraint violation analysis for online convex optimization with long term constraints. In *International Conference on Machine Learning*, pages 11998–12008. PMLR, 2021.

Hao Yu, Michael Neely, and Xiaohan Wei. Online convex optimization with stochastic constraints. *Advances in Neural Information Processing Systems*, 30, 2017.

Wen Sun, Debadeepta Dey, and Ashish Kapoor. Safety-aware algorithms for adversarial contextual bandit. In *International Conference on Machine Learning*, pages 3280–3288. PMLR, 2017.

Xinlei Yi, Xiuxian Li, Tao Yang, Lihua Xie, Yiguang Hong, Tianyou Chai, and Karl H Johansson. Distributed online convex optimization with adversarial constraints: Reduced cumulative constraint violation bounds under slater's condition. *arXiv preprint arXiv:2306.00149*, 2023.

Michael J Neely and Hao Yu. Online convex optimization with time-varying constraints. *arXiv preprint arXiv:1702.04783*, 2017.

Nikolaos Liakopoulos, Apostolos Destounis, Georgios Paschos, Thrasyvoulos Spyropoulos, and Panayotis Mertikopoulos. Cautious regret minimization: Online optimization with long-term budget constraints. In *International Conference on Machine Learning*, pages 3944–3952. PMLR, 2019.

Michael J Neely. Stochastic network optimization with application to communication and queueing systems. *Synthesis Lectures on Communication Networks*, 3(1):1–211, 2010.

Hengquan Guo, Xin Liu, Honghao Wei, and Lei Ying. Online convex optimization with hard constraints: Towards the best of two worlds and beyond. *Advances in Neural Information Processing Systems*, 35:36426–36439, 2022.

Abhishek Sinha and Rahul Vaze. Optimal algorithms for online convex optimization with adversarial constraints. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL `https://openreview.net/forum?id=TxffvJMnBy`.

Ricardo N. Ferreira and Cláudia Soares. Optimal bounds for adversarial constrained online convex optimization, 2025. URL `https://arxiv.org/abs/2503.13366`.

Jordan Lekeufack and Michael I. Jordan. An optimistic algorithm for online convex optimization with adversarial constraints, 2025. URL `https://arxiv.org/abs/2412.08060`.

Tianyi Chen and Georgios B Giannakis. Bandit convex optimization for scalable and dynamic iot management. *IEEE Internet of Things Journal*, 6(1):1276–1286, 2018.

Xuanyu Cao and KJ Ray Liu. Online convex optimization with time-varying constraints and bandit feedback. *IEEE Transactions on automatic control*, 64(7):2665–2680, 2018.

Rahul Vaze. On dynamic regret and constraint violations in constrained online convex optimization. In *2022 20th International Symposium on Modeling and Optimization in Mobile, Ad hoc, and Wireless Networks (WiOpt)*, pages 9–16, 2022. doi: 10.23919/WiOpt56218.2022.9930613.

Qingsong Liu, Wenfei Wu, Longbo Huang, and Zhixuan Fang. Simultaneously achieving sublinear regret and constraint violations for online convex optimization with time-varying constraints. *ACM SIGMETRICS Performance Evaluation Review*, 49(3):4–5, 2022.

Nikhil Bansal, Martin Böhm, Marek Eliáš, Grigorios Koumoutsos, and Seeun William Umboh. Nested convex bodies are chaseable. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1253–1260. SIAM, 2018.

C.J. Argue, Sébastien Bubeck, Michael B Cohen, Anupam Gupta, and Yin Tat Lee. A nearly-linear bound for chasing nested convex bodies. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 117–122. SIAM, 2019.

Sébastien Bubeck, Bo'az Klartag, Yin Tat Lee, Yuanzhi Li, and Mark Sellke. Chasing nested convex bodies nearly optimally. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1496–1508. SIAM, 2020.

Elad Hazan. The convex optimization approach to regret minimization. *Optimization for machine learning*, page 287, 2012.

Hao Yu and Michael J Neely. A low complexity algorithm with $o(\sqrt{T})$ regret and $o(1)$ constraint violations for online convex optimization with long term constraints. *arXiv preprint arXiv:1604.02218*, 2016.

Paolo Manselli and Carlo Pucci. Maximum length of steepest descent curves for quasi-convex functions. *Geometriae Dedicata*, 38(2):211–227, 1991.

Harold Gordon Eggleston. Convexity, 1966.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We provide complete theorem statements and proofs of all claims.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Our result crucially makes use of feasibility assumption (Assumption 3) which is universally used in the COCO literature. In the absence of good lower bounds, the problem considered in the paper question remains open in full generality.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We clearly state the assumptions under which our theoretical results hold.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [NA]

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [NA]

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [NA]

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [NA]

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.

- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [NA]

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: This paper deals with fundamental optimization theory and conform with NeurIPS Code of Ethics

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: This is a theoretical paper and the authors do not see any immediate direct societal impact of this paper.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This theoretical paper does not pose any such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper does not release new assets.

- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.

10 Comparison with [Lekeufack and Jordan, 2025]

Remark 6 *[Lekeufack and Jordan, 2025] consider the COCO problem when* **predictions** *about both cost functions f_t's and constraint functions g_t's are available. With predictions, they show that if predictions are perfect, $O(1)$ regret and CCV is achievable, while if the predictions are totally wrong, in the worst case the regret and CCV are at most as bad as the result of [Sinha and Vaze, 2024]. Intermediate range of results is also obtained depending on the quality of prediction. Essentially [Lekeufack and Jordan, 2025] use the prediction wrapper over the algorithm of [Sinha and Vaze, 2024] to derive their guarantee.*

In this paper, however, we are not assuming **any predictions***, and are solving the COCO problem with the worst case input, similar to* **all the prior work** *listed in Table 1. Moreover, the presented algorithm is conceptually different than [Sinha and Vaze, 2024], and for the first time shows that $O(1)$ or instance dependent CCV while having $O(\sqrt{T})$ regret is possible, which is not the case with prior work even for $d = 1$.*

Thus, the setting of [Lekeufack and Jordan, 2025] is completely different and not really comparable with our results.

11 Proof of Lemma 7

Proof: From the convexity of f_t's, for $x^\star$ satisfying Assumption (3), we have

$$f_t(x_t) - f_t(x^\star) \leq \nabla f_t^T(x_t - x^\star).$$

From the choice of Algorithm 2 for x_{t+1}, we have

$$\begin{aligned}
||x_{t+1} - x^\star||^2 &= ||\mathscr{P}_{S_t}(y_t) - x^\star||^2 \\
&\stackrel{(a)}{\leq} ||y_t - x^\star||^2, \\
&= ||\mathscr{P}_{S_{t-1}}(x_t - \eta_t \nabla f_t(x_t)) - x^\star||^2, \\
&\stackrel{(n)}{\leq} ||(x_t - \eta_t \nabla f_t^T(x_t)) - x^\star||^2,
\end{aligned}$$

where inequalities (a) and (b) follow since $x^\star \in S_t$ for all t. Hence

$$\begin{aligned}
||x_{t+1} - x^\star||^2 &\leq ||x_t - x^\star||^2 + \eta_t^2 ||\nabla f_t(x_t)||^2 - 2\eta_t \nabla f_t^T(x_t)(x_t - x^\star), \\
\nabla f_t^T(x_t)(x_t - x^\star) &\leq \frac{||x_t - x^\star||^2 - ||x_{t+1} - x^\star||^2}{\eta_t} + \eta_t G^2.
\end{aligned}$$

Summing this over $t = 1$ to T, we get

$$\begin{aligned}
2\sum_{t=1}^{T}(f_t(x_t) - f_t(x^\star)) &\leq \sum_{t=1}^{T} \nabla f_t^T(x_t - x^\star), \\
&\leq \sum_{t=1}^{T} \frac{||x_t - x^\star||^2 - ||x_{t+1} - x^\star||^2}{\eta_t} + \sum_{t=1}^{T} \eta_t G^2, \\
&\leq D^2 \frac{1}{\eta_T} + G^2 \sum_{t=1}^{T} \eta_t, \\
&\leq O(DG\sqrt{T}),
\end{aligned}$$

where the final inequality follows by choosing $\eta_t = \frac{D}{G\sqrt{t}}$. □

12 Proof of Lemma 8 and Lemma 9.

Proof: [Proof of Lemma 8] Recall the definition that $x_t \in \partial S_{t-1}, b_t = \mathscr{P}_{S_t}(x_t) \in S_t$ from (5). Let $||x_t - b_t|| = r$, then since all S_t's are spheres, at least along one of the d-orthogonal canonical basis vectors, diameter$(S_t) \leq$ diameter$(S_{t-1}) - \frac{r}{\sqrt{d}}$. Since the diameter along any of the d-axis is D, we get the answer. □ We would like to remark that the proof is short and elementary that should be seen as a strength. **Proof:** [Proof of Lemma 9] Proof is identical to Lemma 8. □

13 Preliminaries for Bounding the CCV in Theorem 12 and Theorem 17

Let $K_1, \dots, K_T$ be nested (i.e., $K_1 \supseteq K_2 \supset K_3 \supseteq \cdots \supseteq K_T$) bounded convex subsets of $\mathbb{R}^d$.

Definition 18 *If $\sigma_1 \in K_1$, and $\sigma_{t+1} = \mathscr{P}_{K_{t+1}}(\sigma_t)$, for $t = 1, \dots, T$. Then the curve*

$$\underline{\sigma} = \{(\sigma_1, \sigma_2), (\sigma_2, \sigma_3), \dots, (\sigma_{T-1}, \sigma_T)\}$$

is called the projection curve on $K_1, \dots, K_T$.

We are interested in upper bounding the quantity

$$\Sigma = \max_{\underline{\sigma}} \sum_{t=1}^{T-1} ||\sigma_t - \sigma_{t+1}||. \tag{6}$$

Lemma 19 *For a projection curve $\underline{\sigma}$, $\Sigma \leq d^{d/2}$diameter(K_1).*

To prove the result we need the following definition.

Definition 20 *A curve $\gamma : I \to \mathbb{R}^d$ is called self-expanded, if for every t where $\gamma'(t)$ exists, we have*

$$< \gamma'(t), \gamma(t) - \gamma(u) > \geq 0$$

for all $u \in I$ with $u \leq t$, where $< .,. >$ represents the inner product. In words, what this means is that γ starting in a point x_0 is self expanded, if for every $x \in \gamma$ for which there exists the tangent line T, the arc (sub-curve) (x_0, x) is contained in one of the two half-spaces, bounded by the hyperplane through x and orthogonal to T.

For self-expanded curves the following classical result is known.

Theorem 21 *Manselli and Pucci [1991] For any self-expanded curve γ belonging to a closed bounded convex set of $\mathbb{R}^d$ with diameter D, its total length is at most $O(d^{d/2}D)$.*

Proof: [Proof of Lemma 19] From Definition 18, the projection curve is

$$\underline{\sigma} = \{(\sigma_1, \sigma_2), (\sigma_2, \sigma_3), \dots, (\sigma_{T-1}, \sigma_T)\}.$$

Let the reverse curve be $\underline{r} = \{r_t\}_{t=0,\dots,T-2}$, where $r_t = (\sigma_{T-t}, \sigma_{T-t-1})$. Thus we are reading $\underline{\sigma}$ backwards and calling it $\underline{r}$. Note that since σ_t is the projection of σ_{t-1} on K_t, each piece-wise linear segment (σ_t, σ_{t+1}) is a straight line and hence differentiable except at the end points. Moreover, since each σ_t is obtained by projecting σ_{t-1} onto K_t and $K_{t+1} \subseteq K_t$, we have that the projection hyperplane F_t that passes through $\sigma_t = \mathscr{P}_{K_t}(\sigma_{t-1})$ and is perpendicular to $\sigma_t - \sigma_{t-1}$ separates the two sub curves $\{(\sigma_1, \sigma_2), (\sigma_2, \sigma_3), \dots, (\sigma_{t-1}, \sigma_t)\}$ and $\{(\sigma_t, \sigma_{t+1}), (\sigma_{t+1}, \sigma_{t+2}), \dots, (\sigma_{T-1}, \sigma_T)\}$.

Thus, we have that for each segment r_τ, at each point where it is differentiable, the curve $r_1, \dots r_{\tau-1}$ lies on one side of the hyperplane that passes through the point and is perpendicular to $r_{\tau+1}$. Thus, we conclude that curve $\underline{r}$ is self-expanded.

As a result, Theorem 21 implies that the length of $\underline{r}$ is at most $O(d^{d/2}\text{diameter}(K_1))$, and the result follows since the length of $\underline{r}$ is same as that of $\underline{\sigma}$ which is Σ. □

14 Proof of Theorem 12

Proof: Recall that $d = 2$, and the definition of F_t from Definition 10. Let the center be $\mathsf{c} = \mathscr{P}_{S_1}(x_1)$. Let t_{orth} be the earliest t for which $\angle(F_t, F_1) = \pi$.

Initialize $\kappa = 1$, $s(1) = 1$, $\tau(1) = 1$.

BeginProcedure Step 1:Definition of Phase κ. Consider

$$\tau(\kappa) = \arg \max_{s(\kappa) < t \leq t_{\text{orth}}, \angle(F_{s(\kappa)}, F_t) \leq \pi/4} t.$$

If there is no such $\tau(\kappa)$,

Phase κ ends, define Phase κ as **Empty**, $s(\kappa+1) = \tau(\kappa)+1$.

Else If

$\angle(F_{\tau(\kappa)}, F_1) = \pi$ Exit

Else If

$s(\kappa+1) = \tau(\kappa)$

End If

Increment $\kappa = \kappa + 1$, and Go to Step 1.

EndProcedure

Example 22 *To better understand the definition of phases, consider Fig. 5, where the largest* t *for which the angle between* F_t *and* F_1 *is at most* $\pi/4$ *is* 3*. Thus,* $\tau(1) = 3$*, i.e., phase* 1 *explores till time* $t = 3$ *and phase* 1 *ends. The starting hyperplane to consider in phase* 2 *is* $s(2) = 3$ *and given that angle between* F_3 *and and the next hyperplane* F_4 *is more than* $\pi/4$*, phase* 2 *is empty and phase* 2 *ends by exploring till* $t = 4$*. The starting hyperplane to consider in phase* 3 *is* $s(3) = 4$ *and the process goes on. The first time* t *such that the angle between* F_1 *and* F_t *is* π *is* $t = 6$*, and thus* $t_{orth} = 6$*, and the process stops at time* $t = 6$*. This also implies that* $S_6 \subset F_1$*. Since* S_t*'s are nested, for all* $t \geq 6$*,* $S_t \subset F_1$*. Hence the total CCV after* $t \geq t_{orth}$ *is at most* GD*.*

The main idea with defining phases, is to partition the whole space into empty and non-empty regions, where in each non-empty region, the starting and ending hyperplanes have an angle to at most $\pi/4$, while in an empty phase the starting and ending hyperplanes have an angle of at least $\pi/4$. Thus, we get the following simple result.

Lemma 23 *For* $d = 2$*, there can be at most* 4 *non-empty and* 4 *empty phases.*

Proof is immediate from the definition of the phases, since any consecutively occurring non-empty and empty phase exhausts an angle of at least $\pi/4$.

Remark 7 *Since we are in* $d = 2$ *dimensions, for all* $t \geq t_{orth}$*, the movement is along the hyperplane* F_1 *and thus the resulting constraint violation after time* $t \geq t_{orth}$ *is at most* GD*. Thus, in the phase definition above, we have only considered time till* t_{orth} *and we only need to upper bound the CCV till time* t_{orth}*.*

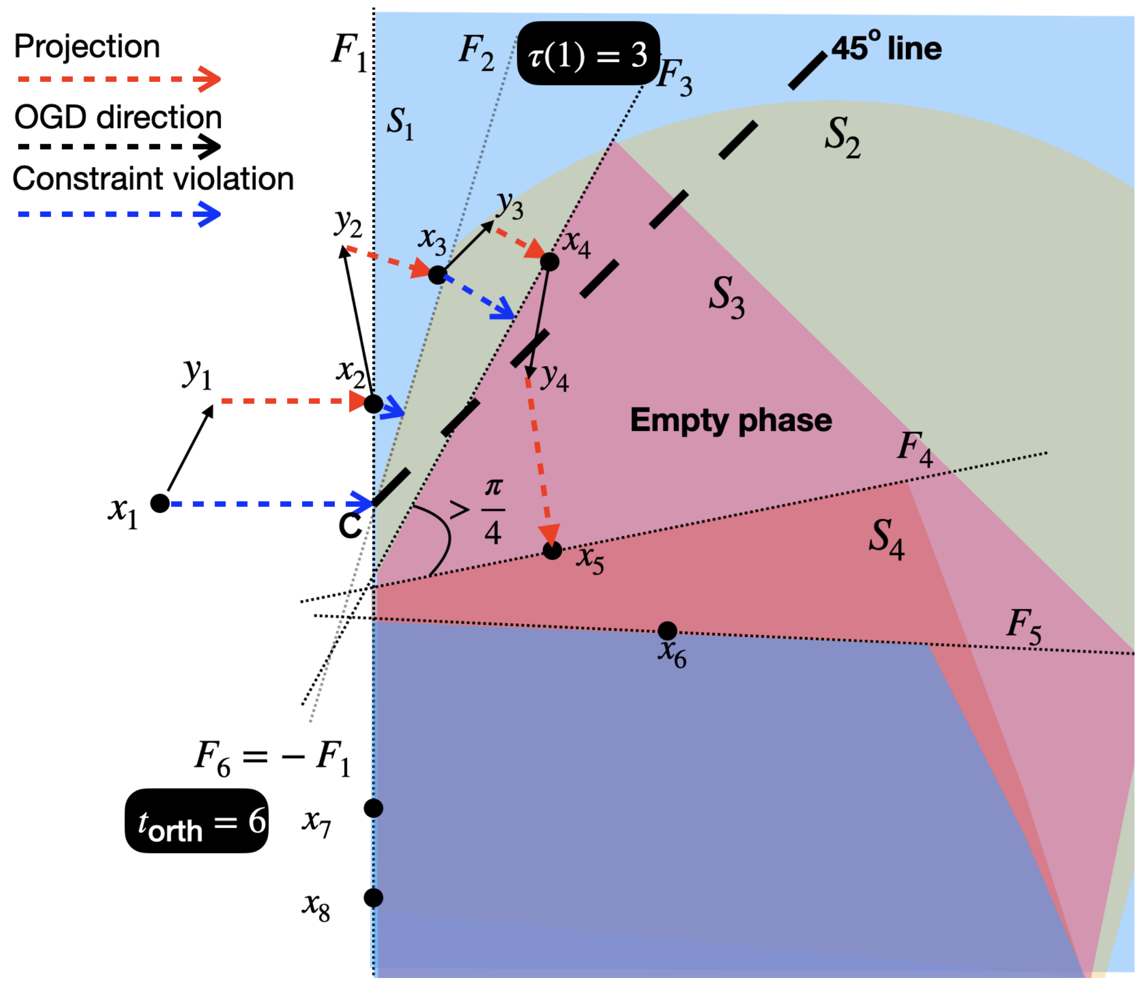


Figure 5: Figure corresponding to Example 22.

We next define the following required quantities.

Definition 24 *With respect to the quantities defined for Algorithm 2, let for a non-empty phase κ*

$$r_{\max}(\kappa) = \max_{s(\kappa)<t\leq\tau(\kappa)} ||y_t - \mathsf{c}|| \text{ and } t^\star(\kappa) = \arg \max_{s(\kappa)<t\leq\tau(\kappa)}^{T} ||y_t - \mathsf{c}||.$$

$t^\star(\kappa)$ is the time index belonging to phase κ for which y_t is the farthest.

Definition 25 *A non-empty phase κ consists of time slots $\mathscr{T}(\kappa) = [\tau(\kappa-1), \tau(\kappa)]$ and the angle $\angle(F_{t_1}, F_{t_2}) \leq \pi/4$ for all $t_1, t_2 \in \mathscr{T}(\kappa)$. Using Definition 24, we partition $\mathscr{T}(\kappa)$ as $\mathscr{T}(\kappa) = \mathscr{T}^-(\kappa) \cup \mathscr{T}^+(\kappa)$, where $\mathscr{T}^-(\kappa) = [\tau(\kappa-1)+1, t^\star(\kappa)+1]$ and $\mathscr{T}^+(\kappa) = [t^\star(\kappa)+2, \tau(\kappa)]$.*

Thus, $\mathscr{T}(\kappa)$ and $\mathscr{T}(\kappa+1)$ have one common time slot.

Definition 26 *[Definition of $z_t(\kappa)$ for $t \in \mathscr{T}^-(\kappa)$]. Let $z_{t^\star(\kappa)+1} = x_{t^\star(\kappa)+1}$. For $t \in \mathscr{T}^-(\kappa)\backslash t^\star(\kappa)+1$, define $z_t(\kappa)$ inductively as follows. $z_t(\kappa)$ is the pre-image of $z_{t+1}(\kappa)$ on F_{t-1} such that the projection of $z_t(\kappa)$ on F_t is $z_{t+1}(\kappa)$.*

Definition 27 *[Definition of $z_t(\kappa)$ for $t \in \mathscr{T}^+(\kappa)$]. For $t \in \mathscr{T}^+(\kappa)$, define $z_t(\kappa)$ inductively as follows. $z_t(\kappa)$ is the projection of $z_{t-1}(\kappa)$ on F_{t-1}.*

See Fig. 6 for a visual illustration of $t^\star(\kappa)$ and $z_t(\kappa)$.

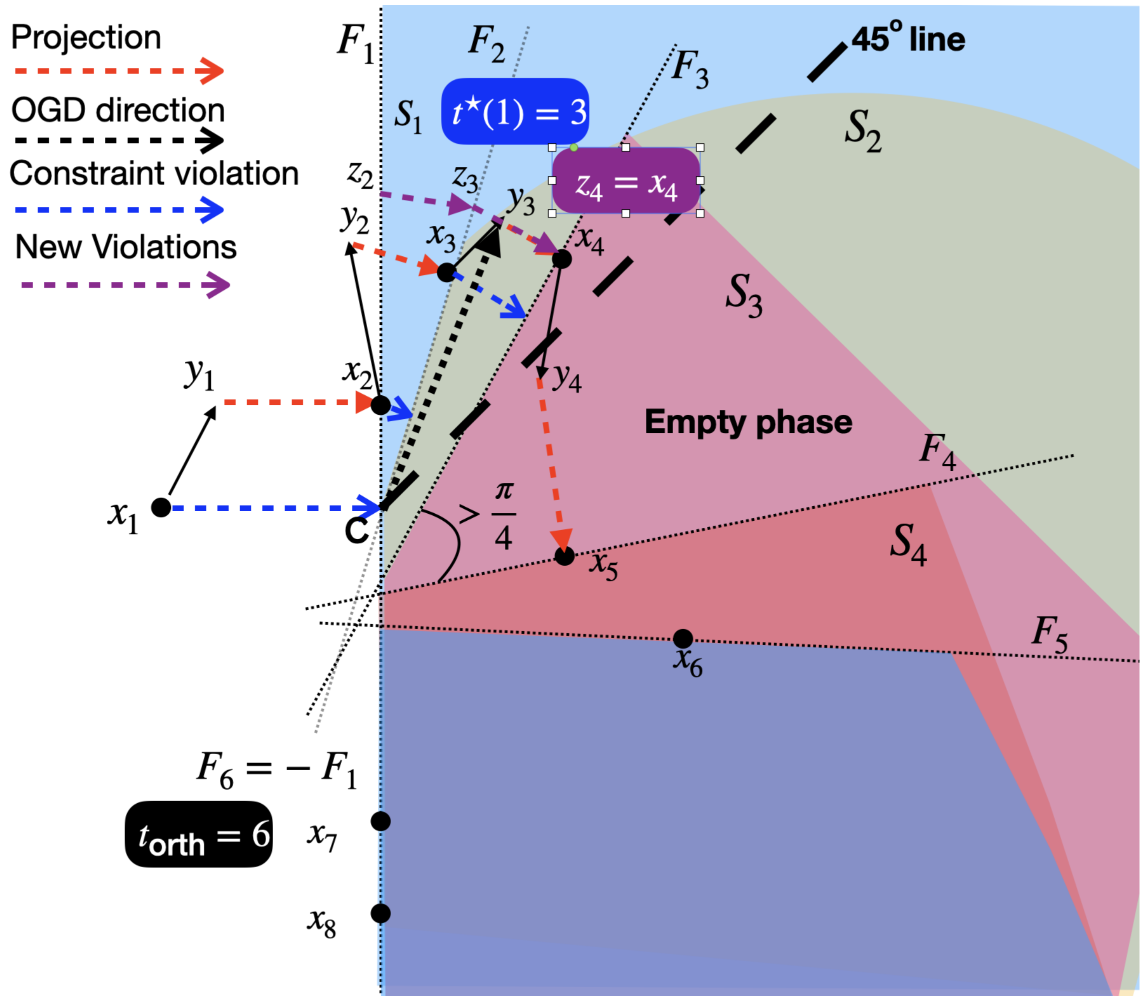


Figure 6: Illustration of definition of $z_t(\kappa)$ for $t \in \mathscr{T}(\kappa)$. In this example, for phase 1, $t^\star(1) = 3$ since the distance of y_3 from c is the farthest for phase 1 that consists of time slots $\mathscr{T}(1) = \{2, 3\}$. Hence $z_{t^\star(1)+1}(1) = x_4$. For $t \in \mathscr{T}(1) \backslash t^\star(1) + 1$, $z_t(1)$ are such $z_{t+1}(1)$ is a projection of $z_t(1)$ onto F_t.

The main idea behind defining $z_t(\kappa)$'s is as follows. For each non-empty phase, we will construct a projection curve (Definition 18) using points z_k such that the length of the projection curve upper bounds the CCV of Algorithm 2 (shown in Lemma 33), and then use Lemma 19 to upper bound the length of the projection curve.

Definition 28 *[Definition of S'_t for a non-empty phase κ:] $S'_{t^\star(\kappa)+1} = S_{t^\star(\kappa)+1}$. For $t \in \mathscr{T}^-(\kappa) \backslash t^\star(\kappa) + 1$, S'_t is the convex hull of $z_{t+1}(\kappa) \cup S_t \cup S'_{t+1}(\kappa)$. For $t \in \mathscr{T}^+(\kappa)$, $S'_t = S_t$. See Fig. 7.*

Lemma 29 *For a non-empty phase κ, for any $t \in \mathscr{T}(\kappa)$, $S'_{t+1} \subseteq S'_t$, i.e. they are nested.*

Definition 30 *For a non-empty phase, $\chi(\kappa) = S'_{\tau(\kappa-1)} \cap \mathscr{H}^+_{\tau(\kappa)}$, where $\mathscr{H}^+_{\tau(\kappa)}$ has been defined in Definition 10.*

Definition 31 *[New Violations for $t \in \mathscr{T}(\kappa)$:] For a non-empty phase κ, for $t \in \mathscr{T}(\kappa) \backslash \tau(\kappa - 1)$, let*

$$v_t(\kappa) = ||z_t(\kappa) - z_{t-1}(\kappa)||.$$

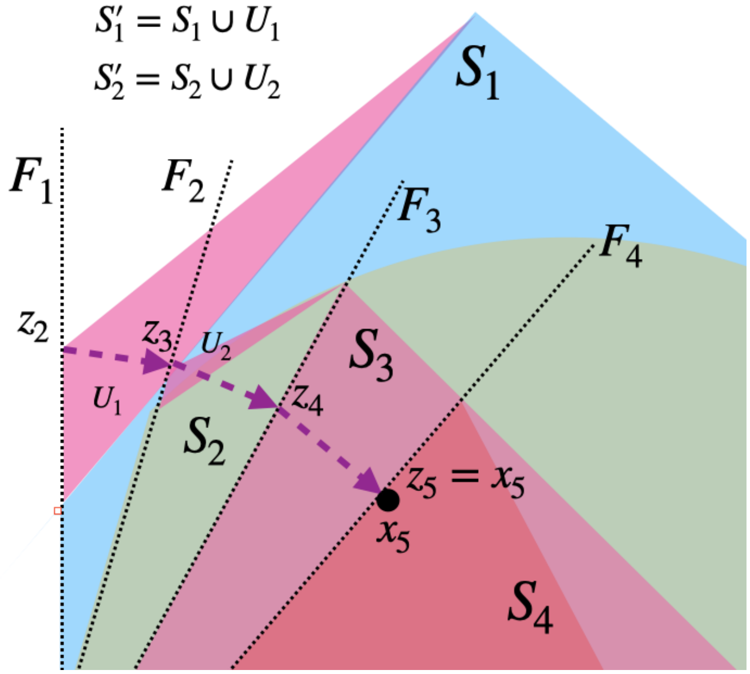


Figure 7: Definition of S_t's where U_t are the extra regions that are added to S_t to get S_t'.

Lemma 32 *For each non-empty phase κ, all $z_t(\kappa)$'s for $t \in \mathscr{T}(\kappa)$ belongs to $\mathscr{B}(\mathsf{c}, \sqrt{2}D)$, where $\mathscr{B}(c, r)$ is a ball with radius r centered at c. In other words, $\chi(\kappa) \subseteq \mathscr{B}(\mathsf{c}, \sqrt{2}D)$.*

Proof: Recall that for a non-empty phase κ, $\mathscr{T}(\kappa) = \mathscr{T}^-(\kappa) \cup \mathscr{T}^+(\kappa)$. We first argue about $t \in \mathscr{T}^-(\kappa)$. By definition, $z_{t^\star(\kappa)+1} = x_{t^\star(\kappa)+1}$ and $x_{t^\star(\kappa)+1} \in S_{t^\star(\kappa)}$. Thus, $z_{t^\star(\kappa)+1} \in \mathscr{B}(\mathsf{c}, \sqrt{2}D)$. Next we argue for $t \in \mathscr{T}^-(\kappa) \backslash t^\star(\kappa)+1$. Recall that the diameter of $\mathscr{X}$ is D, and the fact that $y_t \in S_{t-1}$ from Algorithm 2. Thus, for any non-empty phase κ, the distance from c to the farthest y_t belonging to the phase κ is at most D, i.e., $r_{\max}(\kappa) \leq D$. Let the pre-image of $z_{t^\star(\kappa)+1}(\kappa)$ onto $F_{s(\kappa)}$ (the base hyperplane with respect to which all hyperplanes have an angle of at most $\pi/4$ in phase κ) be $p(\kappa)$ such that projection of $p(\kappa)$ onto $F_{s(\kappa)}$ is $z_{t^\star(\kappa)+1}(\kappa)$. From the definition of any non-empty phase, the angle between $F_{s(\kappa)}$ and F_t for $t \in \mathscr{T}(\kappa)$ is at most $\pi/4$. Thus, the distance of $p(\kappa)$ from c is at most $\sqrt{2}D$.

Consider the 'triangle' $\Pi(\kappa)$ that is the convex hull of c, $z_{t^\star(\kappa)+1}(\kappa)$ and $p(\kappa)$. Given that the angle between $F_{t^\star(\kappa)}$ and $F_{t^\star(\kappa)-1}$ is at most $\pi/4$, the argument above implies that $z_t(\kappa) \in \Pi(\kappa)$ for $t = t^\star(\kappa)$. For $t = t^\star(\kappa) - 1$, $z_t(\kappa) \in F_{t-1}$ is the projection of $z_{t-1}(\kappa)$ onto S'_{t-1}. This implies that the distance of $z_t(\kappa)$ (for $t = t^\star(\kappa) - 1$) from c is at most

$$\frac{D}{\cos(\alpha_{t,t^\star(\kappa)})\cos(\alpha_{t^\star(\kappa),t^\star(\kappa)+1})},$$

where α_{t_1,t_2} is the angle between F_{t_1} and F_{t_2}. From the monotonicity of angles θ_t (Definition 11), and the definition of a non-empty phase, we have that $\alpha_{t,t^\star(\kappa)} + \alpha_{t^\star(\kappa),t^\star(\kappa)+1} \leq \pi/4$ and $\alpha_{t,t^\star(\kappa)} \geq 0, \alpha_{t^\star(\kappa),t^\star(\kappa)+1} \geq 0$. Next, we appeal to the identity

$$\cos(A + B) \leq \cos(A)\cos(B) \tag{7}$$

where $A + B \leq \pi/4$, to claim that $z_t(\kappa) \in \Pi(\kappa)$ for $t = t^\star(\kappa) - 1$.

Iteratively using this argument while invoking the identity (7) gives the result that for any $t \in \mathscr{T}^-(\kappa)$, we have that $z_t(\kappa)$ belongs to $\Pi(\kappa)$. Since $\Pi(\kappa) \subseteq \mathscr{B}(\mathsf{c}, \sqrt{2}D)$, we have the claim for all $t \in \mathscr{T}^-(\kappa)$.

By definition $z_t(\kappa)$ for $t \in \mathscr{T}^+(\kappa)$ belong to $S_{t-1} \subseteq S_1$. Thus, their distance from c is at most D. $\square$

Lemma 33 *For each non-empty phase κ, and for $t \in \mathscr{T}(\kappa)$ the violation $v_t(\kappa) \geq$ dist(x_t, S_t), where dist(x_t, S_t) is the original violation.*

Proof: By construction of any non-empty phase κ, for $t \in \mathscr{T}(\kappa)$ both $x_t(\kappa)$ and $z_t(\kappa)$ belong to F_{t-1}. Moreover, by construction, the distance of $z_t(\kappa)$ from c is at least as much as the distance of x_t from c. Thus, using the monotonicity property of angles θ_t (Definition 11) we get the result. See Fig. 6 for a visual illustration. $\square$

For each non-empty phase κ, by definition, the curve defined by sequence $z_t(\kappa)$ for $t \in \mathscr{T}(\kappa)$ is a projection curve (Definition 18) on sets $S'_t(\kappa)$ (note that $S'_t(\kappa)$'s are nested from Lemma 29). Moreover, for all $t \in \mathscr{T}(\kappa)$, set $S'_t(\kappa) \subset \chi(\kappa)$ which is a bounded convex set. Thus, for $d = 2$ from Lemma 19 the length of curve $\underline{z}(\kappa) = \{(z_t(\kappa), z_{t+1}(\kappa))\}_{t \in \mathscr{T}(\kappa)}$

$$\sum_{t \in \mathscr{T}(\kappa)} v_t(\kappa) \leq 2\text{diameter}(\chi(\kappa)). \tag{8}$$

By definition, the number of non-empty phases till time t_{orth} is at most 4. Moreover, in each non-empty phase $\chi(\kappa) \subseteq \mathscr{B}(\mathsf{c}, \sqrt{2}D)$ from Lemma 32 .

Thus, from (8), we have that

$$\begin{aligned}\sum_{\text{Phase } \kappa \text{ is non-empty}} \sum_{t \in \mathscr{T}(\kappa)} v_t(\kappa) &\leq \sum_{\text{Phase } \kappa \text{ is non-empty}} 2 \text{ diameter}(\chi(\kappa)) \\ &\leq 8 \text{ diameter}(\mathscr{B}(\mathsf{c}, \sqrt{2}D)) \leq O(D). \end{aligned} \tag{9}$$

Using Lemma 33, we get

$$\sum_{\text{Phase } \kappa \text{ is non-empty}} \sum_{t \in \mathscr{T}(\kappa)} \text{dist}(x_t, S_t) \leq O(D). \tag{10}$$

For any empty phase, the constraint violation is the length of line segment $(x_t, \mathscr{P}_{S_t}(x_t))$ (Algorithm 2) crossing it is a straight line whose length is at most $O(D)$. Moreover, the total number of empty phases (Lemma 23) is a constant. Thus, the length of the curve $(x_t, \mathscr{P}_{S_t}(x_t))$ for Algorithm 2 corresponding to all empty phases is at $O(D)$.

Recall from (4) that the CCV is at most G times dist(x_t, S_t). Thus, from (10) we get that the total violation incurred by Algorithm 2 corresponding to non-empty phases is at most $O(GD)$, while corresponding to empty phases is at $O(GD)$. Finally, accounting for the very first violation dist$(x_1, S_1) \leq D$ and the fact that the CCV after time $t \geq t_{\text{orth}}$ (Remark 7) is at most GD, we get that the total constraint violation CCV$_{[1:T]}$ for Algorithm 2 is at most $O(GD)$.

$\square$

15 Proof of Theorem 14

Proof: We need the following preliminaries.

Definition 34 *Let K be a non-empty convex bounded set in $\mathbb{R}^d$. Let u be a unit vector, and ℓ_u a line through the origin parallel to u. Let K_u be the orthogonal projection of K onto ℓ_u, with length $|K_u|$. The mean width of K is defined as*

$$W(K) = \frac{1}{V_d} \int_{\mathbb{S}_1^d} |K_u| du, \tag{11}$$

where $\mathbb{S}_1^d$ is the unit sphere in d dimensions and V_d its $(d-1)$-dimensional Lebesgue measure.

The following is immediate.

$$0 \leq W(K) \leq \text{diameter}(K). \tag{12}$$

Lemma 35 *Eggleston [1966] For $d = 2$,*

$$W(K) = \frac{\textit{Perimeter}(K)}{\pi}.$$

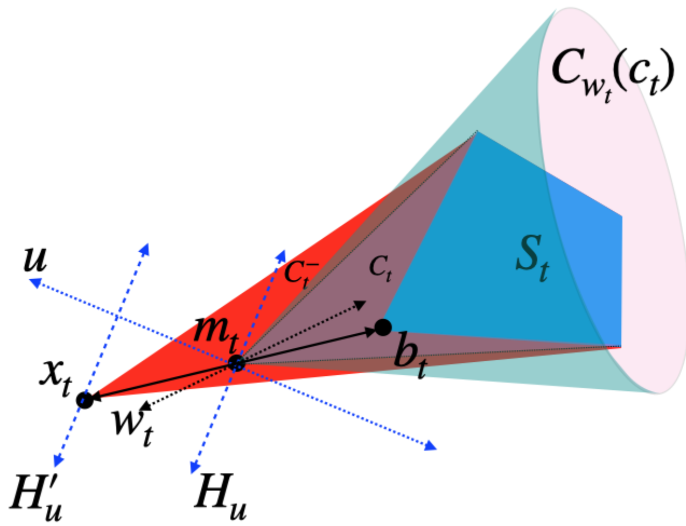


Figure 8: Figure representing the cone $C_{w_t}(c_t)$ that contains the convex hull of m_t and S_t with respect to the unit vector w_t. u is a unit vector perpendicular to H_u an hyperplane that is a supporting hyperplane C_t at m_t such that $\mathscr{C}_t \cap H_u = \{m_t\}$ and $u^T(x_t - m_t) \geq 0$

Lemma 35 implies that $W(K) \neq W(K_1) + W(K_2)$ even if $K_1 \cup K_2 = K$ and $K_1 \cap K_2 = \phi$.

Recall from (5) that $x_t \in \partial S_{t-1}$ and b_t is the projection of x_t onto S_t, and m_t is the mid-point of x_t and b_t, i.e. $m_t = \frac{x_t+b_t}{2}$. Moreover, the convex sets S_t's are nested, i.e., $S_1 \supseteq S_2 \supseteq \cdots \supseteq S_T$. To prove Theorem 14 we will bound the rate at which $W(S_t)$ (Definition 34) decreases as a function of the length $||x_t - b_t||$.

From Definition 13, recall that $\mathscr{C}_t$ is the convex hull of $m_t \cup S_t$. We also need to define $\mathscr{C}_t^-$ as the convex hull of $x_t \cup S_t$. Since $S_t \subseteq \mathscr{C}_t$ and $\mathscr{C}_t^- \subseteq S_{t-1}$ (since S_{t-1} is convex and $x_t \in S_{t-1}$), we have

$$W(S_t) - W(S_{t-1}) \leq W(\mathscr{C}_t) - W(\mathscr{C}_t^-). \tag{13}$$

Definition 36 $\Delta_t = W(\mathscr{C}_t) - W(\mathscr{C}_t^-)$.

The main ingredient of the proof is the following Lemma that bounds Δ_t whose proof is provided after completing the proof of Theorem 14.

Lemma 37

$$\Delta_t \leq -V_{d-1}\frac{||x_t - b_t||}{2V_d(d-1)}(c_t^\star)^d,$$

where $c_t^\star$ has been defined in Definition 13.

Recalling that $c^\star = \min_t c_t^\star$ from Definition 13, and combining Lemma 37 with (12) and (13), we get that

$$\sum_{t=1}^{T} ||x_t - b_t|| \leq \frac{2V_d(d-1)}{V_{d-1}}\left(\frac{1}{c^\star}\right)^d \text{diameter}(S_1),$$

since $S_1 \supseteq S_2 \supseteq \cdots \supseteq S_T$. Recalling that diameter$(S_1) \leq D$, Theorem 14 follows. □

Proof: [Proof of Lemma 37]

Let H_u be the hyperplane perpendicular to vector u. Let $\mathscr{U}_0$ be the set of unit vectors u such that hyperplanes H_u are supporting hyperplanes to $\mathscr{C}_t$ at point m_t such that $\mathscr{C}_t \cap H_u = \{m_t\}$ and $u^T(x_t - m_t) \geq 0$. See Fig. 8 for reference.

Since b_t is a projection of x_t onto S_t, and m_t is the mid-point of x_t, b_t, for $u \in \mathscr{U}_0$, the hyperplane H'_u containing x_t and parallel to H_u is a supporting hyperplane for $\mathscr{C}_t^-$.

Thus, using the definition of K_u from (11),

$$\Delta_t \leq \frac{1}{V_d}\int_{\mathscr{U}_0}(|\mathscr{C}_{t,u}| - |\mathscr{C}^-_{t,u}|)du = -\frac{||x_t - b_t||}{2V_d}\int_{\mathscr{U}_0} u^T \frac{(x_t - m_t)}{||x_t - m_t||}\, du, \tag{14}$$

since $||x_t - m_t|| = ||x_t - b_t||/2$.

Recall the definition of $C_{w_t^\star}(c_t^\star)$ from Definition 13 which implies that the convex hull of m_t and $S_t, \mathscr{C}_t$ is contained in $C_{w_t^\star}(c_t^\star)$. Next, we consider $\mathscr{U}_1$ the set of unit vectors u such that hyperplanes H_u are supporting hyperplanes to $C_{w_t^\star}(c_t^\star)$ at point m_t such that $u^T(x_t - m_t) \geq 0$. By definition $\mathscr{C}_t \subseteq C_{w_t^\star}(c_t^\star)$, it follows that $\mathscr{U}_1 \subset \mathscr{U}_0$.

Thus, from (14)

$$\Delta_t \leq -\frac{||x_t - b_t||}{2V_d}\int_{\mathscr{U}_1} u^T.\frac{(x_t - m_t)}{||x_t - m_t||}du \tag{15}$$

Recalling the definition of $w_t^\star$ (Definition 13), vector $u \in \mathscr{U}_1$ can be written as

$$u = \lambda u_\perp + \sqrt{1-\lambda^2}w_t^\star,$$

where $u_\perp^T w_t^\star = 0$, $|u_\perp| = 1$ and since $u \in \mathscr{U}_1$

$$0 \leq \lambda = \sqrt{1 - (u^T w_t^\star)} = u^T u_\perp \leq c_t^\star.$$

Let $\mathscr{S}_\perp = \{u_\perp : |u_\perp| = 1, u_\perp^T w_t^\star = 0\}$. Let $du_\perp$ be the $(n-2)$-dimensional Lebesgue measure of $\mathscr{S}_\perp$.

It is easy to verify that $du = \lambda^{d-2}(1-\lambda^2)^{-1/2}d\lambda du_\perp$ and hence from (15)

$$\Delta_t \leq -\frac{||x_t - b_t||}{V_d}\int_0^{c_t^\star}\lambda^{d-2}(1-\lambda^2)^{-1/2}d\lambda\int_{\mathscr{S}_\perp}(\lambda u_\perp + \sqrt{1-\lambda^2}w_t^\star)^T\frac{(x_t - m_t)}{||x_t - m_t||}du_\perp. \tag{16}$$

Note that $\int_{du_\perp} u_\perp du_\perp = 0$. Thus,

$$\begin{aligned}
\Delta_t &= -\frac{||x_t - b_t||}{2V_d}\frac{(w_t^\star)^T(x_t - m_t)}{||x_t - m_t||}\int_0^{c_t^\star}\lambda^{d-2}(1-\lambda^2)^{-1/2}\sqrt{1-\lambda^2}\, d\lambda\int_{\mathscr{S}_\perp}du_\perp,\\
&\overset{(a)}{\leq} -V_{d-1}\frac{||x_t - b_t||}{2V_d}\frac{(w_t^\star)^T(x_t - m_t)}{||x_t - m_t||}\int_0^{c_t^\star}\lambda^{d-2}\, d\lambda,\\
&\overset{(b)}{\leq} -V_{d-1}\frac{||x_t - b_t||}{2V_d(d-1)}c_t^\star(c_t^\star)^{d-1},\\
&= -V_{d-1}\frac{||x_t - b_t||}{2V_d(d-1)}(c_t^\star)^d,
\end{aligned} \tag{17}$$

where (a) follows since $\int_{\mathscr{S}_\perp} du_\perp = V_{d-1}$ by definition, (b) follows since $\frac{(w_t^\star)^T(x_t - m_t)}{||x_t - m_t||} \geq c_t^\star$ from Definition 13.

□

16 Proof of Theorem 16

Proof: Since CCV(t) is a monotone non-decreasing function, let $t_{\min}$ be the largest time until which Algorithm 2 is followed by Switch. The regret guarantee is easy to prove. From Theorem 15, regret until time $t_{\min}$ is at most $O(\sqrt{t_{\min}})$. Moreover, starting from time $t_{\min}$ till T, from Theorem 5, the regret of Algorithm 1 is at most $O(\sqrt{T - t_{\min}})$. Thus, the overall regret for Switch is at most $O(\sqrt{T})$.

For the CCV, with Switch, until time $t_{\min}$, CCV$(t_{\min}) \leq \sqrt{T}\log T$. At time $t_{\min}$, Switch starts to use Algorithm 1 which has the following appealing property from (8) Sinha and Vaze [2024] that for any $t \geq t_{\min}$ where at time $t_{\min}$ Algorithm 1 was started to be used with resetting $\text{CCV}(t_{\min}) = 0$. For any $t \geq t_{\min}$

$$\Phi(\text{CCV}(t)) + \text{Regret}_t(x^\star) \leq \sqrt{\sum_{\tau=t_{\min}}^{t} \big(\Phi'(\text{CCV}(\tau))\big)^2} + \sqrt{t - t_{\min}}. \tag{18}$$

where $\beta = (2GD)^{-1}, V = 1, \lambda = \frac{1}{2\sqrt{T}}, \Phi(x) = \exp(\lambda x) - 1$, and $\lambda = \frac{1}{2\sqrt{T}}$. We trivially have $\text{Regret}_t(x^\star) \geq -\frac{Dt}{2D} \geq -\frac{t}{2}$. Hence, from (18), we have that for any $\lambda = \frac{1}{2\sqrt{T}}$ and any $t \geq t_{\min}$

$$\text{CCV}_{[t_{\min},T]} \leq 4GD \ln\!\big(2\big(1+2T\big)\big)\sqrt{T}.$$

Since as argued before, with Switch, $\text{CCV}(t_{\min}) \leq \sqrt{T}\log T$, we get that $\text{CCV}_{[1:T]} \leq O(\sqrt{T}\log T)$. □

17 Proof of Theorem 17

Clearly, with $f_t \equiv 0$ for all t, with Algorithm 2, $y_t = x_t$ and the successive x_t's are such that $x_{t+1} = \mathscr{P}_{S_t}(x_t)$. Thus, essentially, the curve $\underline{x} = (x_1, x_2), (x_2, x_3), \ldots, (x_{T-1}, x_T)$ formed by Algorithm 2 for OCS is a projection curve (Definition 18) on $S_1 \supseteq, \ldots, \supseteq S_T$ and the result follows from Lemma 19 and the fact that $\text{diameter}(S_1) \leq D$.