Proceedings of Machine Learning Research vol 313:1–29, 2026 37th International Conference on Algorithmic Learning Theory

Universal Dynamic Regret and Constraint Violation Bounds for Constrained Online Convex Optimization

Subhamon Supantha[*] SUBHAMON@MINDS.IITB.AC.IN
Indian Institute of Technology, Bombay, India

Abhishek Sinha ABHISHEK.SINHA@TIFR.RES.IN
Tata Institute of Fundamental Research, Mumbai, India

Editors: Matus Telgarsky and Jonathan Ullman

Abstract

We consider a generalization of the celebrated Online Convex Optimization (OCO) framework with adversarial online constraints. In this problem, an online learner interacts with an adversary sequentially over multiple rounds. At the beginning of each round, the learner chooses an action from a convex decision set. After that, the adversary reveals a convex cost function and a convex constraint function. The goal of the learner is to minimize the cumulative cost while satisfying the constraints as tightly as possible. We present two efficient algorithms with simple modular structures that give universal dynamic regret and cumulative constraint violation bounds, improving upon state-of-the-art results. While the first algorithm, which achieves the optimal regret bound, involves projection onto the constraint sets, the second algorithm is projection-free and achieves better violation bounds in rapidly varying environments. Our results hold in the most general case when both the cost and constraint functions are chosen arbitrarily, and the constraint functions need not contain any common feasible point. We establish these results by introducing a general framework that reduces the constrained learning problem to an instance of the standard OCO problem with specially constructed surrogate cost functions.

Keywords: Learning with constraints, Universal dynamic regret, Constraint violation bounds

1. Introduction

Online Convex Optimization (OCO) is a standard framework for studying sequential decision-making under uncertainty. In this framework, an online learner selects an action x_t from a convex decision set $\mathcal{X} \subseteq \mathbb{R}^d$, for T rounds ($t = 1, 2, \ldots, T$). At the end of each round t, an adversary reveals a convex cost function $f_t : \mathcal{X} \mapsto \mathbb{R}$. Consequently, the learner incurs a cost of $f_t(x_t)$ for round t. The goal of the learner is to choose actions sequentially to minimize the cumulative cost $\sum_{t=1}^{T} f_t(x_t)$ over the horizon of length T (Hazan, 2022; Orabona, 2019). Since Zinkevich (2003) introduced the ubiquitous Online Gradient Descent (OGD) algorithm and showed that it achieves the minimax optimal $O(\sqrt{T})$ static regret, many improved algorithms have been proposed in the literature with refined analyses and sharper performance guarantees. See Hazan (2022) and Orabona (2019) for excellent textbook treatments of this topic.

A substantial generalization of OCO called Constrained Online Convex Optimization (COCO), which involves additional time-varying constraints, has recently attracted sufficient attention from the learning community (Mahdavi et al., 2012; Cao and Liu, 2018; Neely and Yu, 2017; Guo et al., 2022; Sinha and Vaze, 2024; Lekeufack and Jordan, 2024). COCO arises in many settings, including AI safety (Wachi et al., 2024; Hua et al., 2025), obstacle avoidance in robotics (Snyder

[*]This work was initiated while the first author was working as a summer intern at TIFR, Mumbai.

et al., 2023), Bandits with Knapsacks (Immorlica et al., 2022), and pay-per-click ad markets with budget constraints (Liakopoulos et al., 2019). In this problem, after the online learner takes its action $x_t \in \mathcal{X}$ at round t, in addition to the cost function f_t, a constraint function $g_t : \mathcal{X} \mapsto \mathbb{R}$ is revealed to the learner. The function g_t encodes an instantaneous constraint of the form $g_t(x) \leq 0$. The online learner is ideally expected to have played an action x_t such that it satisfies the constraint *i.e.,* $g_t(x_t) \leq 0, \ \forall t$. However, this is not always possible as g_t is adversarially chosen and revealed only *after* the learner chooses its action x_t. The learner is thus allowed to violate the constraints to some extent, and the goal is to design an online algorithm that simultaneously minimizes the regret for the cost functions and the cumulative constraint violations (CCV) for the constraint functions.

To the best of our knowledge, all previous works on the COCO problem, including Sinha and Vaze (2024); Guo et al. (2022); Cao and Liu (2018); Neely and Yu (2017), make the following strong assumption on the choice of the constraint functions:

Assumption 1 (Common Feasibility) *There exists a fixed action $x^\star \in \mathcal{X}$ which satisfies every constraints,* i.e., $g_t(x^\star) \leq 0, \forall t \geq 1$.

The common feasibility assumption is substantially restrictive as it limits the flexibility of the adversary in choosing the constraint functions. In practice, if the environment is rapidly varying with time, this assumption may no longer hold and hence, the theoretical guarantees derived under this assumption become vacuous. Dropping Assumption 1 fundamentally changes the nature of the problem. In particular, in the absence of a single action that satisfies all constraints simultaneously, static regret is no longer a meaningful performance metric.

In this paper, we drop Assumption 1 altogether and focus on the stronger objective of guaranteeing universal dynamic regret bounds against *any* feasible comparator sequence. Dynamic regret generalizes the classic static regret metric. While static regret compares the performance of an online algorithm with a *fixed* benchmark, in dynamic regret, we measure the online algorithm's performance against a class of time-varying benchmark sequences. Two different variants of dynamic regret metrics are widely studied in the literature - the more conservative *worst-case* dynamic regret and the more flexible *universal* dynamic regret. Guaranteeing universal dynamic regret bounds is technically more challenging as, unlike worst-case dynamic regret, where the path-length of the comparator sequence can be estimated (*e.g.,* using the doubling trick), minimizing universal dynamic regret requires designing algorithms which are oblivious to the comparator path-length. See Section 2 for details.

In this paper, we propose a general framework, given by Algorithm 1, that reduces the constrained online optimization problem to the standard OCO problem. COCO algorithms constructed within this framework only differ in the way the surrogate costs are constructed and the choice of the OCO subroutines used to minimize the surrogate cost functions.

Algorithm 1 Algorithmic Template for COCO

Require: A generic OCO subroutine $\mathcal{A}^{\text{OCO}}$ with a universal dynamic regret bound

1: **for** $t = 1 : T$ **do**
2: From the first $t-1$ cost and constraint functions, construct a surrogate cost function $\hat{f}_{t-1}$.
3: Choose action x_t by running $\mathcal{A}^{\text{OCO}}$ on the surrogate cost functions $\{\hat{f}_\tau\}_{\tau=1}^{t-1}$.
4: Receive new cost function f_t and the constraint function g_t.
5: **end for**

We emphasize the online nature of Algorithm 1: the surrogate cost function $\hat{f}_{t-1}$ and the action x_t depend only on the past $t-1$ cost and constraint functions $\{f_\tau, g_\tau\}_{\tau=1}^{t-1}$. The cost function f_t and the constraint function g_t are revealed only after x_t is played. Consequently, the cost f_t, the constraint g_t, and hence, the surrogate cost $\hat{f}_t$ may depend on the learner's action x_t.

Our Contribution

1. We present two online algorithms for COCO with different performance and computational tradeoffs. While the first algorithm guarantees a minimax optimal universal dynamic regret bound, the second algorithm achieves a better CCV bound in rapidly varying environments. Furthermore, the first algorithm naturally generalizes to *quasiconvex* constraint functions, beyond the convex setting considered in most prior work.

2. From a computational viewpoint, while the first algorithm requires the full knowledge of the constraint functions and involves a projection step, the second algorithm is lightweight and requires only first-order gradient information.

3. Unlike prior works, our results hold without the restrictive common feasibility assumption (Assumption 1). Both algorithms follow a conceptually simple template of running a standard OCO subroutine on a sequence of *surrogate* cost functions, as outlined in Algorithm 1.

4. As an intermediate technical result, in Section 4.1, we present a new Lipschitz-adaptive algorithm, called AHAG, for the standard OCO problem. This algorithm achieves $\tilde{\mathcal{O}}(\sqrt{(1+\mathcal{P}_T)T})$ universal dynamic regret without *a priori* knowledge of the common Lipschitz constant of the cost functions. This result might be of independent interest.

1.1. Related Works

COCO with Static Regret Guarantees: Static regret metric in the context of COCO uses a fixed benchmark which is assumed to be feasible for all constraints (see Assumption 1). Early works on COCO focused on the setting where the constraint functions are fixed throughout. The main motivation behind this line of work is to design computationally efficient first-order algorithms that avoid projecting onto the fixed constraint set (Mahdavi et al., 2012; Jenatton et al., 2016; Yu and Neely, 2020; Yuan and Lamperski, 2018). Mannor et al. (2009); Yu et al. (2017); Sinha (2024) considered the problem with stochastic constraints in both full and bandit feedback settings. The more difficult problem of COCO with time-varying adversarial constraints was considered by Neely and Yu (2017); Yi et al. (2022). These papers additionally assume *Slater's condition*, which is more restrictive than common feasibility. Slater's condition assumes that all constraints are strictly satisfied by a positive margin $\eta > 0$ by a fixed benchmark. Using only the common feasibility assumption, Guo et al. (2022) proposed an online primal-dual algorithm that achieves $O(\sqrt{T})$ static regret and $O(T^{3/4})$ CCV. More recently, Sinha and Vaze (2024) proposed the first minimax-optimal COCO algorithm that yields $O(\sqrt{T})$ static regret and $\tilde{O}(\sqrt{T})$ CCV. Their algorithm uses a Lipschitz-adaptive variant of the online gradient descent (OGD) subroutine on a sequence of surrogate cost functions which are constructed by linearly combining the cost and constraint functions via an exponential Lyapunov potential. Lekeufack and Jordan (2024) extended their algorithm to the optimistic setting where the learner has access to unreliable predictions for the cost and constraint functions. Further works have provided more refined CCV guarantees (Sinha and Vaze, 2025; Vaze and Sinha, 2025).

Dynamic Regret bounds in OCO: In the dynamic regret metric, the performance of an online policy is compared against a sequence of time-varying benchmarks. The seminal work of Zinkevich (2003) established that the standard OGD policy also achieves $\mathcal{O}((1+\mathcal{P}_T)\sqrt{T})$ universal dynamic regret for the standard OCO problem, where $\mathcal{P}_T$ is the path-length of the benchmark sequence. However, this fell short of the $\Omega(\sqrt{(1+\mathcal{P}_T)T})$ lower bound established by Zhang et al. (2018), who further proposed a novel algorithm, called ADER, which matched the minimax lower bound. Their algorithm employed the Hedge policy (Cesa-Bianchi and Lugosi, 2006) to track experts, each running an instance of OGD with their own guesses of the path length. The ADER algorithm, however, needs the knowledge of the upper bound of the ℓ_∞ norms of the loss functions. Zhao et al. (2020, 2024) subsequently improved this bound by devising an ensemble learning algorithm which adapts to the smoothness and gradient variation of the loss functions. Yang et al. (2016) showed that OGD achieves $\mathcal{O}(\sqrt{(1+\mathcal{P}_T^\star)T})$ worst-case dynamic regret in the general case, where $\mathcal{P}_T^\star$ is the path-length corresponding to the minimizers of the cost functions. Under additional structural assumptions, tighter worst-case dynamic regret bounds have now been established (Yang et al., 2016; Mokhtari et al., 2016; Zhang et al., 2017; Besbes et al., 2015; Zhang et al., 2022; Chen et al., 2017).

COCO with Dynamic Regret Guarantees: Coming back to the COCO problem, for time-invariant constraint functions, Yi et al. (2021) achieved $\mathcal{O}(\sqrt{(1+\mathcal{P}_T^\star)T})$ worst-case dynamic regret and $O(\sqrt{T})$ CCV, which was later improved to $O(\sqrt{(1+\mathcal{P}_T^\star)T})$ regret and $O(\log T)$ CCV by Guo et al. (2022). In case of time-varying adversarial constraints, Guo et al. (2022) achieved $\mathcal{O}(\sqrt{(1+\mathcal{P}_T)T})$ universal dynamic regret and $O(T^{\frac{3}{4}})$ CCV under the Common Feasibility assumption (Assumption 1). Under both Assumption 1 and known path length, Lekeufack and Jordan (2024) proposed an algorithm with worst case dynamic regret $O(\sqrt{(1+\mathcal{P}_T)T})$ and $\tilde{O}(\sqrt{T})$ CCV guarantees. Liu et al. (2021) proposed an algorithm with a worst-case dynamic regret bound of $O(\sqrt{(1+\mathcal{P}_T^\star)T})$ and $O(\max(\sqrt{T}, V_g))$ CCV, where $V_g \equiv \sum_{t=1}^{T} \sup_{x\in\mathcal{X}} ||g_t(x) - g_{t-1}(x)||_2$. A major drawback of this result is that V_g can be linear in T even for very simple problem instances (*e.g.,* when the constraints are chosen alternatively from two functions g_1, g_2 where $g_1 \neq g_2$.) Table 1 compares and contrasts the results from the previous works with respect to ours.

Convex Body Chasing: The COCO problem is closely related to the classic Convex Body Chasing (CBC) problem (Bubeck et al., 2020; Friedman and Linial, 1993; Sellke, 2020). In CBC, at each round $t \geq 1$, the adversary first reveals a convex set $\mathcal{X}_t \subseteq \mathbb{R}^d$ and the learner is required to choose a feasible point $x_t \in \mathcal{X}_t$. The cost is measured by the total movement cost: $\sum_{t=2}^{T} ||x_t - x_{t-1}||$. The performance of an online policy is measured in terms of the *competitive ratio* metric which is obtained by taking the ratio of the cost of the online algorithm and the cost of the optimal offline policy. Argue et al. (2021) presents an online algorithm that achieves $O(\min(d, \sqrt{d \log T}))$ competitive ratio for d-dimensional decision space. COCO can be considered as a variant of CBC involving both cost and constraint functions, where the algorithm moves first in each round. Furthermore, in COCO, we are interested in additive performance guarantees (Regret, CCV), as opposed to multiplicative measures such as competitive ratio. See Sinha and Vaze (2024, Section A.8) for further discussion on the connection between COCO and CBC.

Reference	Regret	Type	CCV	Complexity	Assumptions
Yi et al. (2021)	$\mathcal{O}(\sqrt{(1+\mathcal{P}_T(u_{1:T}))T})$	**UD**	$\mathcal{O}(\sqrt{T})$	CONV-OPT	Fixed Constr.
Guo et al. (2022)	$\mathcal{O}((1+\mathcal{P}_T(u_{1:T}))\sqrt{T})$	**UD**	$\mathcal{O}(T^{3/4})$	CONV-OPT	**CF**
Lekeufack and Jordan (2024)	$\mathcal{O}(\sqrt{(1+\mathcal{P}_T)T})$	**WD**	$\tilde{\mathcal{O}}(\sqrt{(1+\mathcal{P}_T)T})$	GRAD EVAL	**CF**, Known $\mathcal{P}_T$
Cao and Liu (2018)	$\mathcal{O}(\sqrt{(1+\mathcal{P}_T^\star)T})$	**WD**	$\mathcal{O}(T^{3/4}(1+\mathcal{P}_T^\star)^{1/4})$	CONV-OPT	Known $\mathcal{P}_T^\star$
Liu et al. (2021)	$\mathcal{O}(\sqrt{(1+\mathcal{P}_T^\star)T})$	**WD**	$\mathcal{O}(\max(\sqrt{T}, V_g))$	CONV-OPT	Known $\mathcal{P}_T^\star$
This paper (Algorithm 2)	$\mathcal{O}(\sqrt{(1+\mathcal{P}_T(u_{1:T}))T})$	**UD**	$\mathcal{O}(\sqrt{(1+\mathcal{P}_T^\star)T})$	PROJ	—
This paper (Algorithm 5)	$\mathcal{O}((1+\mathcal{P}_T(u_{1:T}))\sqrt{T})$	**UD**	$\mathcal{O}(\max(T^{3/4}, \sqrt{T(1+\mathbb{P}_T^\star)}))$	GRAD EVAL	—

Table 1: Summary of the results on the COCO problem with dynamic regret and CCV bounds. Abbreviations have the following meanings- PROJ: Euclidean projection on the time-varying constraint set, CONV-OPT: Solving a convex optimization problem over the decision set, GRAD EVAL: Evaluating the gradients of the cost and constraint functions, **UD**: Universal Dynamic Regret, $u_{1:T}$: comparator action sequence for universal dynamic regret, **WD**: Worst-case Dynamic Regret, **CF**: Assumption 1 (Common feasibility), $\mathcal{P}_T$: Known path length, $\mathcal{P}_T(u_{1:T})$: path length of any arbitrary feasible comparator sequence, $\mathcal{P}_T^\star$: path length corresponding to the feasible minimizers of the constraint functions, $\mathbb{P}_T^\star$: minimum feasible path length.

2. Problem Statement and Performance Metrics

We consider a sequential game played between an online learner and an adversary for T rounds. Let $\mathcal{X} \subseteq \mathbb{R}^d$ be a closed and convex decision set. For simplicity, we assume that the diameter of the decision set is finite[1] and upper-bounded by D. On the t^{th} round, the online learner plays an action x_t from the decision set $\mathcal{X}$. Upon observing the learner's action, the adversary reveals a convex *cost function* $f_t : \mathcal{X} \mapsto \mathbb{R}$ and a closed and convex *constraint function* $g_t : \mathcal{X} \mapsto \mathbb{R}$. The constraint function encodes a constraint of the form $g_t(x) \leq 0$. As a result, the learner incurs a cost of $f_t(x_t)$ and a constraint violation of $\max(0, g_t(x_t)) \equiv g_t^+(x_t)$ on round t. The objective of the learner is to choose a sequence of actions so as to *simultaneously* minimize the cumulative cost $\sum_{t=1}^T f_t(x_t)$ and the cumulative constraint violation (CCV) $\sum_{t=1}^T g_t^+(x_t)$. All cost and constraint functions are assumed to be G-Lipschitz with respect to a norm $||\cdot||$, which will be taken to be the standard Euclidean norm unless specified otherwise. The cost and constraint functions could be otherwise arbitrary and we do not impose any further restrictions on them. In particular, we do not make the common feasibility assumption (Assumption 1), which has been a common limiting factor in previous works.

Remark 1 *In the special case when all constraint functions are identically equal to zero, we recover the standard Online Convex Optimization (OCO) framework (Hazan, 2022).*

In the following, we introduce several definitions for the COCO problem that will be used extensively throughout the paper. These straightforwardly generalize the corresponding definitions in OCO.

Definition 2 (Comparator Sequence) *A comparator (*a.k.a. *benchmark) sequence denoted by* $u_{1:T} \equiv \{u_1, u_2, \ldots, u_T\}$ *is a sequence of actions in the decision set* $\mathcal{X}$ *with which the performance of an online algorithm is compared.*

1. This assumption can be relaxed. See Remark 17.

As customary in the literature, the performance bounds of the proposed online algorithms will be given in terms of the *path length* of the comparator sequence, defined below.

Definition 3 (Path Length) *The path length of a comparator sequence $u_{1:T}$ is defined to be*

$$\mathcal{P}_T(u_{1:T}) = \sum_{t=2}^{T} ||u_t - u_{t-1}||. \tag{1}$$

Intuitively, path length captures the degree of temporal variation of a comparator sequence.

Definition 4 (Feasible Set) *The feasible set $\mathcal{X}_t^\star$ for round t is defined to be the subset of the decision set $\mathcal{X}$ for which the t^{th} constraint is satisfied,* i.e.,

$$\mathcal{X}_t^\star := \{x \in \mathcal{X} : \, g_t(x) \leq 0\}.$$

Since the constraint functions are assumed to be closed and convex, the feasible sets are also closed and convex. Without any loss of generality, we assume the feasible sets to be non-empty.

Definition 5 (Feasible Comparators) *A comparator sequence $u_{1:T}$ is said to be feasible if each of its element satisfies the corresponding constraint,* i.e.,

$$u_t \in \mathcal{X}_t^\star, \forall t \geq 1.$$

Definition 6 (Cumulative Constraint Violation (CCV)) *The cumulative constraint violation (CCV) at the end of round T is defined as:*

$$Q(T) = \sum_{t=1}^{T} (g_t(x_t))^+. \tag{2}$$

Definition 7 (Universal Dynamic Regret) *For any sequence of cost functions $f_{1:T}$ and any feasible comparator sequence $u_{1:T}$, the universal dynamic regret for an action sequence $x_{1:T}$ is defined as:*

$$\text{UD-REGRET}\big(f_{1:T}; u_{1:T}\big) = \sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(u_t) \tag{3}$$

Note that the action sequence $x_{1:T}$ need not be feasible and can have a strictly positive CCV (2). Since a universal dynamic regret bound must hold for all feasible comparator sequences, which could possibly have widely varying path lengths, the algorithm must be oblivious of the comparator sequence and, specifically, of the path length of the comparator. In other words, the algorithm cannot tune its internal parameters according to the path lengths of any specific comparator sequence.

Remark 8 *Under the Common Feasibility assumption (Assumption 1), there exists a common feasible action $x^\star \in \cap_{t=1}^{T} \mathcal{X}_t^\star$. Hence, choosing this comparator every round,* i.e., $u_t = x^\star, \forall t$, *we recover the static regret metric considered in earlier works (Sinha and Vaze, 2024; Guo et al., 2022; Mahdavi et al., 2012).*

Definition 9 (Worst-case Dynamic Regret) *For any sequence of cost functions $f_1, \ldots, f_T$, the worst-case dynamic regret is defined as*

$$\text{WD-REGRET}(f_{1:T}) = \sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} \min_{x_t^\star \in \mathcal{X}_t^\star} f_t(x_t^\star). \tag{4}$$

In other words, the worst-case dynamic regret metric considers a restricted class of benchmark sequences, each of which consists of instantaneous minimizers of the cost functions over the corresponding feasible sets. The minimum path length among all such benchmark sequences is denoted by $\mathcal{P}_T^\star$.

Remark 10 *Unlike the universal dynamic regret, for the worst-case dynamic regret metric, we can assume, without much loss of generality, that a tight upper-bound to the path-length $\mathcal{P}_T^\star$ is known. This is because, in this case, the path-length can always be estimated within a constant factor via the standard doubling trick (Guo et al., 2022; Cao and Liu, 2018).*

Comparison between Universal and Worst-Case Dynamic Regret Metrics: We now briefly explain the operational advantage of having a universal dynamic regret bound over a worst-case dynamic regret bound. Recall that our final objective is to select actions so as to minimize the cumulative cost $\sum_{t=1}^{T} f_t(x_t)$. A typical no-regret online algorithm, while competing against a feasible benchmark $u_{1:T}$, achieves a cumulative cost of

$$\sum_{t=1}^{T} f_t(x_t) \leq \underbrace{\sum_{t=1}^{T} f_t(u_t)}_{\text{term-I}} + \underbrace{\psi_T(\mathcal{P}_T(u_{1:T}))}_{\text{term-II}}, \tag{5}$$

where $\psi_T(\cdot)$ is some algorithm-dependent non-decreasing function of the path length of the comparators. To obtain the tightest possible guarantee on the cumulative cost, one must minimize the RHS of (5). In case of worst-case dynamic regret, the comparator sequence is fixed to be $x_t^\star = \arg\min_{x \in \mathcal{X}_t^\star} f_t(x), t \geq 1$, which minimizes only term-I in the upper bound (5). However, the corresponding path length $\mathcal{P}_T^\star$ of this sequence may be large, resulting in a large value of term-II, and hence, a suboptimal bound on the cumulative cost. In contrast, under a universal dynamic regret guarantee, the bound (5) holds uniformly over *all* feasible comparator sequences. This allows one to select a feasible benchmark sequence $u_{1:T}^\star$ that optimally balances the tradeoff between term-I and term-II, thereby yielding a strictly tighter cumulative cost bound. Similar observations have also been made in Zhang et al. (2018).

3. COCO Algorithm with Full Constraint Feedback

In this Section, we present our first COCO algorithm, Algorithm 2, that simultaneously guarantees a minimax optimal universal dynamic regret bound and a competitive CCV. Algorithm 2 is based on transforming the original cost functions so that infeasible actions are penalized in a geometrically meaningful way. The key idea is to add a distance-based penalty to the cost functions that ensures all global minimizers of the transformed cost function lie within the current feasible set, while preserving convexity and Lipschitz continuity.

For any closed and convex set $S \subseteq \mathcal{X}$, let $\mathsf{dist}(x, S) \equiv \min_{y \in S} ||x - y||$ denote the minimum distance from a point $x \in \mathcal{X}$ to the set S, where the distance is measured with respect to the norm $|| \cdot ||$. We now define the following auxiliary cost functions $\tilde{f}_t : \mathcal{X} \mapsto \mathbb{R}, t \geq 1$:

$$\tilde{f}_t(x) := f_t(x) + 2G\mathsf{dist}(x, \mathcal{X}_t^*), \quad x \in \mathcal{X}, \tag{6}$$

where, we recall, G is an upper bound to the Lipschitz constant of the cost and constraint functions. Intuitively, the auxiliary cost function adds a penalty to the actions that are away from the current feasible set $\mathcal{X}_t^\star$. See Figure 7.1 in the Appendix for illustrative examples. The auxiliary cost functions will be used to construct the surrogate cost functions for which we would run a no-regret algorithm, as outlined in Algorithm 1.

Since the function $\mathsf{dist}(\cdot, S)$ is known to be convex and 1-Lipschitz (Boyd and Vandenberghe, 2004, Example 3.16), it follows that the auxiliary cost function $\tilde{f}_t$ is convex and $3G$-Lipschitz for any $t \geq 1$. The following two lemmas underscore two important properties of the auxiliary cost functions, which will be useful later in the analysis.

Lemma 11 *For any feasible comparator sequence $u_{1:T}$ and any decision sequence $x_{1:T}$, we have* $\text{UD-REGRET}\big(f_{1:T}; u_{1:T}\big) \leq \text{UD-REGRET}\big(\tilde{f}_{1:T}; u_{1:T}\big)$, i.e.,

$$\sum_{t=1}^{T} \big(f_t(x_t) - f_t(u_t)\big) \leq \sum_{t=1}^{T} \big(\tilde{f}_t(x_t) - \tilde{f}_t(u_t)\big).$$

Proof

Definition (6) implies that for any $x_t \in \mathcal{X}$, we have $\tilde{f}_t(x_t) \geq f_t(x_t)$. Furthermore, since $u_{1:T}$ is a feasible sequence, we have $u_t \in \mathcal{X}_t^\star, \forall t \geq 1$. This yields $\mathsf{dist}(u_t, \mathcal{X}_t^\star) = 0$, and hence, $\tilde{f}_t(u_t) = f_t(u_t), \forall t \geq 1$. Combining the above two relations, the result follows. ■

The next result states that for any round $t \geq 1$, the minima of the auxiliary function $\tilde{f}_t$ over the entire decision set $\mathcal{X}$ lies in the feasible set $\mathcal{X}_t^\star$. See Section 7.1 in the Appendix for a visual illustration of this property.

Lemma 12 *For any round $t \geq 1$, we have*

$$\arg\min_{x \in \mathcal{X}} \tilde{f}_t(x) \in \mathcal{X}_t^\star.$$

Proof We prove the claim by means of contradiction by showing that any minimizer of $\tilde{f}_t$ lying outside $\mathcal{X}_t^\star$ can be strictly improved by projecting it onto $\mathcal{X}_t^\star$.

Let $u^\star \in \arg\min_{x \in \mathcal{X}} \tilde{f}_t(x)$, and for the sake of contradiction, assume that $u^\star \notin \mathcal{X}_t^\star$. Let $v^\star := \mathsf{Proj}_{\mathcal{X}_t^\star}(u^\star)$ be the projection of $u^\star$ on to the set $\mathcal{X}_t^\star$, *i.e.,*

$$v^\star = \arg\min_{z \in \mathcal{X}_t^\star} ||z - u^\star||.$$

Since the set $\mathcal{X}_t^\star$ is closed and convex, the existence and uniqueness of the projection $v^\star$ are guaranteed by the Projection Theorem (Bertsekas et al., 2003, Proposition 2.2.1). Thus we have the following series of inequalities:

$$\begin{aligned}
\tilde{f}_t(u^\star) &\overset{(a)}{=} f_t(u^\star) + 2G||v^\star - u^\star|| \\
&\overset{(b)}{\geq} f_t(u^\star) + G||v^\star - u^\star|| + |f_t(v^\star) - f_t(u^\star)| \\
&\geq f_t(u^\star) + G||v^\star - u^\star|| + f_t(v^\star) - f_t(u^\star) \\
&= f_t(v^\star) + G||v^\star - u^\star|| \\
&\overset{(c)}{>} f_t(v^\star) \\
&\overset{(d)}{=} \tilde{f}_t(v^\star),
\end{aligned}$$

where (a) follows from Definition (6), (b) follows from the G-Lipschitzness of the cost functions, which yields $|f_t(v^\star) - f_t(u^\star)| \leq G||v^\star - u^\star||$, (c) follows from our assumption that $u^\star \notin \mathcal{X}_t^\star$ and hence, $||v^\star - u^\star|| > 0$, and finally, (d) follows from Definition (6) using the fact that $v^\star \in \mathcal{X}_t^\star$. Hence, we have $\tilde{f}_t(u^\star) > \tilde{f}_t(v^\star)$. Since $u^\star$ is defined to be a minimizer of the auxiliary function $\tilde{f}_t$ over the entire decision set $\mathcal{X}$, the above constitutes a contradiction and concludes the proof. ■

We now construct the t^{th} surrogate cost function $\hat{f}_t : \mathcal{X} \mapsto \mathbb{R}$ by adding the auxiliary cost function to the non-negative part of the constraint function, *i.e.,*

$$\hat{f}_t(x) = g_t^+(x) + \tilde{f}_t(x), \;\; t \geq 1. \tag{7}$$

Clearly, the surrogate cost function $\hat{f}_t$ is convex and $4G$-Lipschitz as its Lipschitz constant $\mathsf{Lip}(\hat{f}_t)$ can be upper bounded as:

$$\mathsf{Lip}(\hat{f}_t) \leq \mathsf{Lip}(g_t^+) + \mathsf{Lip}(\tilde{f}_t) \leq G + \mathsf{Lip}(f_t) + 2G\mathsf{Lip}(\mathsf{dist}(\cdot)) \leq 4G.$$

As stated earlier, our proposed algorithms follow the generic algorithmic template given by Algorithm 1. For Algorithm 2, we use the existing ADER algorithm, proposed by Zhang et al. (2018), as our OCO subroutine ($\mathcal{A}^{\text{OCO}}$) for minimizing the surrogate cost. In our analysis, we will need only the final universal dynamic regret bound for ADER as given below.

Theorem 13 ((Zhang et al., 2018)) *For any sequence of G-Lipschitz convex cost functions $\{\hat{f}_t\}_{t=1}^T$ defined on a convex domain with diameter D, the* ADER *algorithm achieves the following universal dynamic regret bound for any benchmark sequence $u_{1:T}$:*

$$\text{UD-REGRET}(\hat{f}_{1:T}; u_{1:T}) \leq \gamma\sqrt{T(1 + \mathcal{P}_T(u_{1:T}))},$$

where $\gamma = O(GD \log\log T)$.

In the following section, we will present a more general dynamic regret minimizing algorithm for problems with unbounded Lipschitz constants. Hence, we refer the reader to Zhang et al. (2018) for details on the ADER algorithm. We now state the main result of this section.

Theorem 14 (Universal Dynamic Regret and CCV for Algorithm 2) *Algorithm 2 achieves the following universal dynamic regret and CCV bounds:*

$$\text{UD-REGRET}\big(f_{1:T}; u_{1:T}\big) \leq \zeta GD\sqrt{(1 + \mathcal{P}_T(u_{1:T}))T}, \;\; \text{CCV}_T \leq \zeta GD\sqrt{(1 + \mathcal{P}_T^\star)T},$$

where $u_{1:T}$ is any feasible comparator sequence, $\mathcal{P}_T^\star$ is the path length of the worst-case feasible comparator sequence given by $x_t^\star = \arg\min_{x \in \mathcal{X}_t^\star} f_t(x), \; t \geq 1$, and $\zeta = O(\log\log T)$.

Algorithm 2 The First COCO Algorithm

1: Run Algorithm 1 with $\mathcal{A}^{\text{OCO}} = \text{ADER}$ on the surrogate cost functions defined below:

$$\hat{f}_t(x) = f_t(x) + g_t^+(x) + 2G\text{dist}(x, \mathcal{X}_t^\star),\ x \in \mathcal{X};\ \ t \geq 1. \tag{8}$$

Proof The proof follows via a regret decomposition argument that simultaneously upper bounds the cumulative constraint violation and the universal dynamic regret using the regret guarantee of the OCO subroutine.

For any feasible comparator sequence $u_{1:T}$, we have the following dynamic regret decomposition equality:

$$\begin{aligned} Q(T) + \text{UD-REGRET}_T\big(\tilde{f}_{1:T}; u_{1:T}\big) &= Q(T) + \sum_{t=1}^{T}\big(\tilde{f}_t(x_t) - \tilde{f}_t(u_t)\big) \\ &= \sum_{t=1}^{T} g_t^+(x_t) + \sum_{t=1}^{T}\big(\tilde{f}_t(x_t) - \tilde{f}_t(u_t)\big) \\ &\overset{(a)}{=} \sum_{t=1}^{T}\big(\hat{f}_t(x_t) - \hat{f}_t(u_t)\big) \\ &= \text{UD-REGRET}_T\big(\hat{f}_{1:T}; u_{1:T}\big), \end{aligned} \tag{9}$$

where in step (a), we have used Definition (7) for the surrogate costs along with the feasibility of the comparator sequence. Finally, using Theorem 13, the RHS of inequality (9) can be upper bounded as:

$$\text{UD-REGRET}_T\big(\hat{f}_{1:T}; u_{1:T}\big) \leq \gamma\sqrt{(1 + \mathcal{P}_T(u_{1:T})T}, \tag{10}$$

where $\mathcal{P}_T(u_{1:T})$ is the path length of the benchmark sequence $u_{1:T}$, and $\gamma = O(GD \log\log T)$.

Bounding CCV: Consider the comparator sequence $u_t := x_t^\star,\ \forall t \in [T]$, where $x_t^\star$ is the global minimizer of the auxiliary cost function $\tilde{f}_t$, *i.e.,* $x_t^\star = \arg\min_{x\in\mathcal{X}} \tilde{f}_t(x)$. By Lemma 12, the comparator sequence $x_{1:T}^\star$ is feasible. Furthermore, for any $t \geq 1$, we have

$$u_t = x_t^\star = \arg\min_{x\in\mathcal{X}} \tilde{f}_t(x) \overset{(a)}{=} \arg\min_{x\in\mathcal{X}_t^\star} \tilde{f}_t(x) \overset{(b)}{=} \arg\min_{x\in\mathcal{X}_t^\star} f_t(x), \tag{11}$$

where (a) follows from Lemma 12 and (b) follows from the fact that $f_t(x) = \tilde{f}_t(x), \forall x \in \mathcal{X}_t^\star$. Eqn. (11) shows that the comparator sequence $x_{1:T}^\star$ coincides with the worst-case benchmark used in Definition 9. Let $\mathcal{P}_T^\star$ denote the minimum path length corresponding to the worst-case benchmarks.

Since, by definition, $x_t^\star$ is a global minimizer of $\tilde{f}_t$, we have $\big(\tilde{f}_t(x_t) - \tilde{f}_t(x_t^\star)\big) \geq 0, \forall t$. This implies that $\text{UD-REGRET}\big(\tilde{f}_{1:T}; x_{1:T}^\star\big) \geq 0$. Hence, from the dynamic regret decomposition inequality (9), we obtain the following upper bound to CCV:

$$Q(T) \leq \text{UD-REGRET}_T\big(\hat{f}_{1:T}; x_{1:T}^\star\big) \leq \gamma\sqrt{(1 + \mathcal{P}_T^\star)T}.$$

Bounding Regret: Finally, let $u_{1:T}$ be any feasible comparator sequence. Hence, we have

$$\begin{aligned}\text{UD-Regret}_T\big(f_{1:T}; u_{1:T}\big) &\overset{(a)}{\leq} \text{UD-Regret}_T\big(\tilde{f}_{1:T}; u_{1:T}\big) \\ &\overset{(b)}{\leq} \text{UD-Regret}_T\big(\hat{f}_{1:T}; u_{1:T}\big) \\ &\overset{(c)}{\leq} \gamma\sqrt{(1+\mathcal{P}_T(u_{1:T}))T}.\end{aligned}$$

where (a) follows from Lemma 11, (b) follows from Eqn. (9), where we have used the non-negativity of $Q(T)$, and (c) follows from Eqn. (10). ■

Remark 15 *It is interesting to note that the above proof remains unaltered even when the* $\mathsf{dist}(\cdot, \mathcal{X}_t^\star)$ *term in the auxiliary cost function* (6) *is defined with respect to* any *arbitrary norm. This allows one to obtain tighter regret bounds with improved dependence on the ambient dimension by exploiting the geometry of the cost and constraint functions.*

Remark 16 *The G-Lipschitzness of the constraint functions imply that $g_t^+(x) \leq G\mathsf{dist}(\cdot, \mathcal{X}_t^\star), \forall x \in \mathcal{X}$. Hence, intuitively, both $g_t^+(\cdot)$ and $\mathsf{dist}(\cdot, \mathcal{X}_t^\star)$ measure some form of "distance" to the current feasible set $\mathcal{X}_t^\star$. This raises a natural question:* Can we drop the last term in Eqn. (8) involving the $\mathsf{dist}(\cdot,\cdot)$ function and simply use $f_t + g_t^+(x)$ as the surrogate cost function? *The answer to this question turns out to be negative. See Appendix 7.2 for a counter-example.*

Remark 17 *Algorithm 2 can be made parameter-free w.r.t. the horizon-length T by using the standard doubling trick. On the other hand,* ADER *uses the knowledge of the diameter of the decision set (D). Hence, it is not a priori clear how to make the algorithm parameter-free w.r.t. D. However, Jacobsen and Cutkosky (2022, Appendix J, Algorithm 6 and Lemma 10) give a simple reduction scheme that transforms any online algorithm with a dynamic regret bound on a bounded domain into one achieving a similar bound on the unbounded domain. This reduction uses an additional parameter-free 1 dimensional OCO subroutine (which can be constructed,* e.g., *using the coin-betting scheme of Orabona (2019, Chapter 9)). Using this scheme, the OCO subroutine* ADER*, and hence, Algorithm 2 can be made parameter-free with respect to D in a black-box fashion.*

Extension to quasiconvex constraint functions: Algorithm 2 can be extended to the setting where the constraint functions are assumed to be only *quasiconvex* and G-Lipschitz, rather than convex and G-Lipschitz, as assumed earlier. Recall that a function $g : \mathcal{X} \to \mathbb{R}$ is called *quasiconvex* if its domain is convex and all its sublevel sets, given by

$$S_\alpha = \{x \in \mathcal{X} : g(x) \leq \alpha\}, \ \alpha \in \mathbb{R},$$

are convex. Clearly, all convex functions are quasiconvex. See Boyd and Vandenberghe (2004, Section 3.4) for standard examples of non-convex yet quasiconvex functions.

Given a quasiconvex constraint function g_t, let $\mathcal{X}_t^\star = \{x \in \mathcal{X} : g_t(x) \leq 0\}$ denote its feasible set. Since $\mathcal{X}_t^\star$ is convex, we now define a new G-Lipschitz convex constraint function $h_t : \mathcal{X} \mapsto \mathbb{R}_+$ as follows:

$$h_t(x) = G\mathsf{dist}(x, \mathcal{X}_t^\star), \ x \in \mathcal{X}.$$

Clearly, g_t and h_t have the same feasible set. We then run Algorithm 2 using the original cost function f_t and the new convex constraint function h_t for all $t \geq 1$. Using the Lipschitz continuity of g_t, we have $g_t^+ \leq h_t(x), \forall x \in \mathcal{X}$. Consequently, the regret and cumulative constraint violation bounds established in Theorem 14 continue to hold in this quasiconvex setting.

Computational aspects: Note that the universal dynamic regret minimizer subroutine ADER requires only a sub-gradient of the surrogate cost function as an input on every round. From Eqn. (8), it follows that in order to compute a sub-gradient of the surrogate cost function $\hat{f}_t$, we need to compute sub-gradients of the cost and constraint functions and the gradient of the function $\mathsf{dist}(\cdot, \mathcal{X}_t^\star)$. To compute the latter, let $\mathsf{Proj}_{\mathcal{X}_t}(x_t)$ denote the Euclidean projection of x_t onto the feasible set $\mathcal{X}_t^\star$. Then we have

$$\nabla_x \mathsf{dist}(x, \mathcal{X}_t^\star)|_{x=x_t} = \nabla_x \min_{y \in \mathcal{X}_t^\star} ||x - y|||_{x=x_t} = \frac{x_t - \mathsf{Proj}_{\mathcal{X}_t}(x_t)}{||x_t - \mathsf{Proj}_{\mathcal{X}_t}(x_t)||},$$

which is precisely the unit vector in the direction from x_t to its projection on to $\mathcal{X}_t^\star$. Thus the main computational bottleneck for running Algorithm 2 is the computation of projections on the time-varying feasible sets $\mathcal{X}_t^\star$'s on each round. Note that the universal dynamic regret minimizer subroutine (ADER) also involves a projection onto the (fixed) decision set $\mathcal{X}$. However, in most practical cases, $\mathcal{X}$ has simple forms (*e.g.,* Euclidean box, d-dimensional ball), which enable fast computation of this projection step. Hence, the run time of Algorithm 2 is primarily dominated by the former projection step.

4. COCO with First-order Feedback

While Algorithm 2 achieves the optimal dynamic regret guarantee, projecting onto time-varying sets may be computationally prohibitive in high-dimensional settings. This motivates us to propose a new *projection-free* algorithm, Algorithm 5, that relies only on the first-order information of the cost and constraint functions. Compared to the previous algorithm, Algorithm 5 avoids the costly projection step onto the time-varying feasible sets altogether. Although it features a sub-optimal dependence on the path length in its regret bound, we will see that Algorithm 5 achieves a tighter constraint violation guarantee in rapidly varying environments. Readers willing to assume the adaptive regret bound in Theorem 19 can skip Section 4.1 and proceed directly to Section 4.2 without much loss of continuity.

4.1. Preliminaries: Universal Dynamic Regret with Unbounded Lipschitz Constants

Algorithm 5 uses the common template as given by Algorithm 1. However, as we will see in the sequel, the Lipschitz constants of the surrogate cost functions in this case cannot be upper bounded *a priori* as their growth depends on the past actions of the algorithm. This must be contrasted with the surrogate cost functions (7) used by Algorithm 2, which are $4G$-Lipschitz irrespective of the history. Hence, as a prerequisite, we first design an adaptive OCO subroutine, called AHAG, that yields a universal dynamic regret bound for any sequence of convex cost functions with *unbounded* Lipschitz constants. This is achieved by combining logarithmically many base policies, each running an instance of the AdaGrad sub-routine (Duchi et al., 2011) with different path-length estimates, with the AdaHedge algorithm (De Rooij et al., 2014). See Algorithm 3 for the pseudocode of AdaGrad. Note that ADER achieves the minimax-optimal $\mathcal{O}(\sqrt{(1+\mathcal{P}_T)T})$ universal dynamic regret bound. However, ADER requires the range and the gradient norms of all cost functions to be bounded by a constant (Zhang et al., 2018, Assumptions 1 and 2), and hence, it cannot be applied in our setting. To the best of our knowledge, the AHAG subroutine, described in Algorithm 4, is new and might be of independent interest.

In Theorem 18, we give two different adaptive dynamic regret bounds for the AdaGrad subroutine, corresponding to known and unknown path lengths. These results will be tightened later in Theorem

19 using multiple experts. While the dynamic regret of the AdaGrad algorithm for strongly-convex loss functions has been investigated in the literature (Nazari and Khorram, 2024), its dynamic regret bound for the basic convex case still remains under-explored.

Algorithm 3 AdaGrad: Online Gradient Descent with Adaptive step sizes

1: **Input** : Convex decision set $\mathcal{X}$, sequence of convex cost functions $\{\hat{f}_t\}_{t=1}^T$, $\text{diam}(\mathcal{X}) = D$, $\text{Proj}_{\mathcal{X}}(\cdot)$ = Euclidean projection on the convex set $\mathcal{X}$, non-increasing step sizes $\{\eta_t\}_{t\geq 1}$.
2: **Initialize** : $x_1 \in \mathcal{X}$ arbitrarily.
3: **for** $t = 1 : T$ **do**
4: Play x_t and compute $\nabla_t = \nabla \hat{f}_t(x_t)$
5: Set $x_{t+1} = \text{Proj}_{\mathcal{X}}(x_t - \eta_t \nabla_t)$
6: **end for**

Theorem 18 (Dynamic Regret Bounds for AdaGrad) *The AdaGrad subroutine, given in Algorithm 3, achieves the following adaptive **dynamic regret** bound, denoted by* D-REGRET$(\hat{f}_{1:T}; \mathcal{P}_T)$, *for any sequence of convex cost functions $\hat{f}_{1:T}$ and any comparator sequence $u_{1:T}$ whose path length is known to be at most $\mathcal{P}_T$, using an adaptive learning rate sequence $\eta_t = \frac{(D+1)\sqrt{1+\mathcal{P}_T}}{\sqrt{2\sum_{\tau=1}^t ||\nabla_\tau||^2}}$, $t \geq 1$:*

$$\text{D-REGRET}(\hat{f}_{1:T}; \mathcal{P}_T) \leq (D+1)\sqrt{2(1+\mathcal{P}_T)}\sqrt{\sum_{t=1}^T ||\nabla_t||^2}. \tag{12}$$

AdaGrad *also achieves the following **universal dynamic regret** bound, using the path-length-independent adaptive step size sequence $\eta_t = \frac{(D+1)}{\sqrt{2\sum_{\tau=1}^t ||\nabla_\tau||^2}}$, $t \geq 1$:*

$$\text{UD-REGRET}(\hat{f}_{1:T}; u_{1:T}) \leq \sqrt{2}(D+1)\big(1+\mathcal{P}_T(u_{1:T})\big)\sqrt{\sum_{t=1}^T ||\nabla_t||^2}, \tag{13}$$

where $\nabla_t \equiv \nabla \hat{f}_t(x_t), \forall t \geq 1$ and $\text{diam}(\mathcal{X}) = D$.

Notably, Theorem 18 holds without assuming any uniform upper bound for the Lipschitz constants of the cost functions. Our proof of Theorem 18 extends the static regret analysis of the AdaGrad algorithm from Orabona (2019, Theorem 4.14) to the dynamic regret setting. See Appendix 7.3 for the details.

4.1.1. TIGHTER UNIVERSAL DYNAMIC REGRET BOUND WITH MULTIPLE EXPERTS

Theorem 18 states that the standard AdaGrad subroutine achieves a suboptimal universal dynamic regret bound proportional to the path length of the comparator sequence. However, the same theorem also shows that if the path length of the comparator $u_{1:T}$ is known, AdaGrad (with a different step size sequence) achieves a tighter dynamic regret bound, which scales with the square root of the path length. In this section, we obtain an optimal universal dynamic regret bound by combining different AdaGrad sub-routines, each corresponding to different path length estimates, through the AdaHedge algorithm (Orabona, 2019, Section 7.6), improving the dependence on the path length to

$\sqrt{1 + \mathcal{P}_T(u_{1:T})}$. The AdaHedge algorithm, introduced by De Rooij et al. (2014), is a generalization of the celebrated Hedge algorithm for the classic Prediction with Expert advice problem (Cesa-Bianchi and Lugosi, 2006). AdaHedge is adaptive to the Lipschitz constants of the cost functions in the sense that, unlike Hedge, it does not need to know a uniform upper bound to the Lipschitz constants of the costs *a priori*. See Theorem 22 in the Appendix for the regret bound for the AdaHedge algorithm. Using ideas similar to Zhang et al. (2018), our AHAG subroutine, described in Algorithm 4, runs an expert tracking AdaHedge algorithm, where each expert runs an instance of AdaGrad subroutine with step sizes tuned to their own guess of the path length.

Algorithm 4 AHAG: AdaHedge with AdaGrad Algorithm

1: **Initialize:** $N = \lceil \frac{1}{2} \log_2(1 + DT) \rceil + 1$ experts. Expert $i \in [N]$ runs AdaGrad (Algorithm 3) with the following adaptive step size sequence

$$\eta_t^i = \frac{(D+1)2^{i-1}}{\sqrt{2 \sum_{\tau=1}^{t} ||\nabla f_\tau(x_\tau^i)||^2}}, \quad t \geq 1. \tag{14}$$

Initialize the weight vector to the uniform distribution $w_1 \leftarrow [1/N, \dots, 1/N]$; Choose the initial actions of the experts $(x_1^i, \forall i)$ arbitrarily from $\mathcal{X}$

2: **for** $t = 1 : T$ **do**

3: Play a convex combination of experts actions with the coefficients given by the weight vector w_t :

$$x_t = \sum_{i=1}^{N} w_t^i x_t^i.$$

4: Recieve the current cost function f_t (only gradient information suffices).

5: **for** experts $i = 1$ to N **do**

6: Pass f_t to their AdaGrad subroutine (Algorithm 3) with learning rate η_t^i

7: Receive the next action x_{t+1}^i

8: Compute the loss of the i^{th} expert

$$h_{t+1}^i = \big(\langle \nabla f_t(x_{t+1}^i), x_{t+1}^i \rangle, \ i \in [N]\big)$$

9: **end for**

10: Pass loss vector $h_{t+1} = \big(h_{t+1}^i, i \in [N]\big)$ to AdaHedge and obtain the updated weight vector w_{t+1}.

11: **end for**

AHAG is a first-order algorithm, and its main computational bottleneck is the computation of N gradients in each round (line 8). The following theorem gives a universal dynamic regret bound for AHAG, which will be used as the no-regret subroutine $\mathcal{A}^{\text{OCO}}$ in Algorithm 5.

Theorem 19 *The* AHAG *algorithm, given in Algorithm 4, achieves the following Lipschitz-adaptive universal dynamic regret bound for any cost functions* $f_{1:T}$ *and any benchmark sequence* $u_{1:T}$:

$$\text{UD-REGRET}(f_{1:T}; u_{1:T}) \leq \gamma\sqrt{1+\mathcal{P}_T(u_{1:T})}\sqrt{\sum_{t=1}^{T}||\nabla_t||^2}, \tag{15}$$

where $\nabla_t \equiv \nabla f_t(x_t), \forall t \geq 1$, *and* $\gamma \equiv O(D \log\log T)$, *where the decision set* $\mathcal{X}$ *is bounded within an Euclidean ball of radius at most* $D/2$.

4.2. The Second COCO Algorithm: Design and Analysis

In this section, we present our second COCO algorithm that achieves competitive performance for any convex and bounded decision set while utilizing only first-order gradient feedback. The pseudocode of the proposed algorithm is given below in Algorithm 5.

Algorithm 5 The Second COCO Algorithm (Projection-free)

1: **Parameters:** $V = \tilde{O}(GD\sqrt{T}), Q(0) = 0$, Quadratic Lyapunov function $\Phi(x) = x^2$.

2: Run Algorithm 1 with $\mathcal{A}^{\text{OCO}} =$ AHAG (Algorithm 4), with the surrogate cost functions defined as follows:

$$\hat{f}_t(x) = Vf_t(x) + \Phi'(Q(t))g_t^+(x),\ x \in \mathcal{X};\ \ t \geq 1, \tag{16}$$

The variable $Q(t)$ denotes the cumulative CCV up to round t, which evolves as follows:

$$Q(t) = Q(t-1) + g_t^+(x_t),\ \ t \geq 1. \tag{17}$$

Analysis: Let $\Phi : \mathbb{R} \mapsto \mathbb{R}$ be a non-decreasing convex function ($\Phi(0) = 0$), which will be used as a Lyapunov (*a.k.a.* potential) function in our analysis. Since the function $\Phi(\cdot)$ is convex, for any two real numbers x, y, we have $\Phi(x) - \Phi(y) \leq \Phi'(x)(x - y)$. Let the variable $Q(t)$ denote the CCV up to round t, which evolves as $Q(t) = Q(t-1) + g_t^+(x_t)$. Thus, we can upper bound the increase in the Lyapunov function on round t as:

$$\Phi\big(Q(t)\big) - \Phi\big(Q(t-1)\big) \leq \Phi'(Q(t))g_t^+(x_t). \tag{18}$$

Let $u_{1:T}$ be any feasible comparator sequence and V be a positive parameter to be fixed later. Adding $V\big(f_t(x_t) - f_t(u_t)\big)$ to both sides of inequality (18) and summing over $t = 1$ to T, we obtain

$$\Phi\big(Q(T)\big) + V\sum_{t=1}^{T}\Big(f_t(x_t) - f_t(u_t)\Big) \leq \sum_{t=1}^{T}\Phi'(Q(t))g_t^+(x_t) + V\sum_{t=1}^{T}\Big(f_t(x_t) - f_t(u_t)\Big).$$

Using the feasibility of the comparator sequence, we have $g_t^+(u_t) = 0$ for $1 \leq t \leq T$. Hence, the previous inequality yields the following:

$$
\begin{aligned}
&\Phi\big(Q(T)\big) + V \sum_{t=1}^{T}\Big(f_t(x_t) - f_t(u_t)\Big) \\
\leq\ & \sum_{t=1}^{T}\Big(V f_t(x_t) + \Phi'(Q(t)) g_t^{+}(x_t)\Big) - \sum_{t=1}^{T}\Big(V f_t(u_t) + \Phi'(Q(t)) g_t^{+}(u_t)\Big) \\
=\ & \sum_{t=1}^{T}\Big(\hat{f}_t(x_t) - \hat{f}_t(u_t)\Big),
\end{aligned}
$$

where the surrogate functions $\{\hat{f}_t\}_{t\geq 1}$ have been defined in Eqn. (16). Hence, using the definition of universal dynamic regret (3), we obtain the following *dynamic regret decomposition inequality*:

$$
\Phi(Q(T)) + V\ \text{UD-REGRET}(f_{1:T}; u_{1:T}) \leq \text{UD-REGRET}(\hat{f}_{1:T}; u_{1:T}), \tag{19}
$$

Inequality (19) plays a key role in the analysis by relating the cumulative constraint violation with the universal dynamic regret through the Lyapunov function.

Since the surrogate cost function $\hat{f}_t$, defined in Eqn. (16), involves a time-varying factor $\Phi'(Q(t))$, its Lipschitz constant cannot be bounded *a priori* as the CCV $Q(t)$ could potentially grow indefinitely. Because of this, Algorithm 5 passes the surrogate costs to the Lipschitz-adaptive AHAG algorithm, described in Algorithm 4, which does not require any fixed upper bound to the Lipschitz constants of the cost functions. Plugging in the universal dynamic regret bound from Theorem 19 for the surrogate cost functions, the regret decomposition inequality (19) yields the following:

$$
\Phi(Q(T)) + V\ \text{UD-REGRET}(f_{1:T}; u_{1:T}) \leq \gamma\sqrt{1+\mathcal{P}_T(u_{1:T})}\sqrt{\sum_{t=1}^{T}||\nabla_t||^2}, \tag{20}
$$

where $\nabla_t = \nabla \hat{f}_t(x_t) = V\nabla f_t(x_t) + \Phi'(Q(t))\nabla g_t^{+}(x_t)$ is a (sub)-gradient of the surrogate cost function at x_t. Furthermore, the term appearing on the RHS can be upper-bounded as follows:

$$
\begin{aligned}
\sqrt{\sum_{t=1}^{T}||\nabla_t||^2} &\overset{(a)}{\leq} \sqrt{\sum_{t=1}^{T} G^2\,(V + \Phi'(Q(t)))^2} \\
&\overset{(b)}{\leq} G\sqrt{\sum_{t=1}^{T} 2V^2 + \sum_{t=1}^{T} 2(\Phi'(Q(t)))^2} \\
&\overset{(c)}{\leq} \sqrt{2}GV\sqrt{T} + \sqrt{2}G\sqrt{\sum_{t=1}^{T}(\Phi'(Q(t)))^2} \\
&\overset{(d)}{\leq} \sqrt{2}GV\sqrt{T} + \sqrt{2}G\Phi'(Q(T))\sqrt{T},
\end{aligned}
$$

where inequality (a) follows from the G-Lipschitzness of the cost and constraint functions and using triangle inequality, (b) follows from the fact that $(x+y)^2 \leq 2x^2 + 2y^2$, (c) follows from the fact that for any two non-negative real numbers x, y, we have $\sqrt{x+y} \leq \sqrt{x} + \sqrt{y}$, and (d) follows from the monotonicity of the $Q(t)$ variables and the convexity of the $\Phi(\cdot)$ function. Plugging in the above bound in (20), we obtain

$$\begin{aligned}\Phi(Q(T)) + V\ \text{UD-Regret}(f_{1:T}; u_{1:T}) \quad \leq \quad & \gamma\sqrt{1+\mathcal{P}_T(u_{1:T})}\Phi'(Q(T))\sqrt{T} \\ & +\gamma\sqrt{1+\mathcal{P}_T(u_{1:T})}V\sqrt{T}, \end{aligned} \tag{21}$$

where we have redefined the constant $\gamma \leftarrow \gamma G\sqrt{2}$. Finally, choosing $\Phi(x)$ to be the quadratic potential function, $\Phi(x) = x^2$, Eqn. (21) simplifies to

$$Q^2(T) + V\ \text{UD-REGRET}(f_{1:T}; u_{1:T}) \leq 2\gamma\sqrt{1+\mathcal{P}_T(u_{1:T})}\,Q(T)\sqrt{T} + \gamma\sqrt{1+\mathcal{P}_T(u_{1:T})}V\sqrt{T}. \tag{22}$$

We now solve inequality (22) for obtaining bounds for CCV ($Q(T)$) and UD-REGRET$(f_{1:T}; u_{1:T})$.

Bounding CCV: Since the cost functions are G-Lipschitz, we have

$$|f_t(x_t) - f_t(u_t)| \leq G||x_t - u_t|| \leq GD,\ \forall t \implies \text{UD-REGRET}(f_{1:T}; u_{1:T}) \geq -GDT.$$

Plugging in the above bound into inequality (22), we obtain

$$Q^2(T) - 2\gamma\sqrt{T\big(1+\mathcal{P}_T(u_{1:T})\big)}Q(T) - \gamma V\sqrt{T\big(1+\mathcal{P}_T(u_{1:T})\big)} - VGDT \leq 0.$$

Solving the above quadratic inequality in $Q(T)$, we obtain:

$$\begin{aligned} 2Q(T) \quad \leq \quad & 2\gamma\sqrt{T\big(1+\mathcal{P}_T(u_{1:T})\big)} + \bigg(\Big(2\gamma\sqrt{T(1+\mathcal{P}_T(u_{1:T})}\Big)^2 \\ + \quad & 4\gamma V\sqrt{T\big(1+\mathcal{P}_T(u_{1:T})\big)} + 4VGDT\bigg)^{\frac{1}{2}}. \end{aligned}$$

Using the fact that for $x \geq 0, y \geq 0$, we have $\sqrt{x+y} \leq \sqrt{x} + \sqrt{y}$, the above inequality yields the following CCV bound:

$$Q(T) \leq\ 2\gamma\sqrt{T(1+\mathcal{P}_T(u_{1:T}))} + \frac{1}{2}(4\gamma V\sqrt{T(1+\mathcal{P}_T(u_{1:T}))})^{\frac{1}{2}} + \frac{1}{2}(4VGDT)^{\frac{1}{2}}. \tag{23}$$

Since the above CCV bound holds for any feasible comparator sequence, the smallest upper bound is obtained by choosing a feasible comparator $u^\star_{1:T}$ having the shortest path length $\mathbb{P}^\star_T$.

Bounding Regret: From inequality (22), we obtain the following bound for the universal dynamic regret

$$\begin{aligned} V\ \text{UD-Regret}(f_{1:T}; u_{1:T}) &\leq 2\gamma\sqrt{T\big(1+\mathcal{P}_T(u_{1:T})\big)}Q(T) - Q^2(T) + \gamma V\sqrt{T\big(1+\mathcal{P}_T(u_{1:T})\big)} \\ &\overset{(a)}{\leq} \gamma^2(1+\mathcal{P}_T(u_{1:T}))T + \gamma V\sqrt{T\big(1+\mathcal{P}_T(u_{1:T})\big)}, \end{aligned} \tag{24}$$

where (a) follows from the fact that $2aQ(T) - Q^2(T) \leq a^2, \forall Q(T) \in \mathbb{R}$. Finally, choosing the parameter $V = \gamma T^{\frac{1}{2}}$, we arrive at our main result of this section.

Theorem 20 (Universal Dynamic Regret and CCV for Algorithm 5) *Algorithm 5 achieves the following universal dynamic regret and CCV bounds:*

$$\text{UD-REGRET}(f_{1:T}; u_{1:T}) \leq \zeta GD\big(1 + \mathcal{P}_T(u_{1:T})\big)\sqrt{T}, \ \text{CCV}_T \leq \zeta GD(T^{3/4} + \sqrt{T(1 + \mathbb{P}_T^\star)}),$$

where $u_{1:T}$ is any feasible comparator sequence, $\mathbb{P}_T^\star$ is the minimum path length among all feasible comparators, i.e.,

$$\mathbb{P}_T^\star = \min_{y_1,\ldots,y_T} \sum_{t=2}^{T} ||y_t - y_{t-1}||_2, \quad y_t \in \mathcal{X}_t^\star, \ \ t \geq 1,$$

and $\zeta = O(\log\log T)$.

Theorem 20 upper bounds CCV in terms of the shortest possible feasible path length $\mathbb{P}_T^\star$, which captures the intrinsic difficulty of satisfying the time-varying constraints.

Remark 21 *If the common feasibility assumption (Assumption 1) holds, we immediately have $\mathbb{P}_T^\star = 0$. This yields* UD-REGRET $= O((1 + \mathcal{P}_T(u_{1:T}))\sqrt{T})$, $CCV_T = \tilde{O}(T^{3/4})$. *These bounds generalize prior results of Guo et al. (2022, Theorem 3) and Cao and Liu (2018, Section III). Although both the previous algorithms need to know the full constraint functions, Algorithm 5 is computationally efficient as it is projection-free and only requires first-order feedback.*

Comparison between Algorithm 2 and Algorithm 5: Comparing the statements of Theorem 14 and Theorem 20, we observe that the first algorithm offers an improved dependence on the path length in the universal dynamic regret bound ($\tilde{O}(\sqrt{1 + \mathcal{P}_T(u_{1:T})})$ vs $\tilde{O}((1 + \mathcal{P}_T(u_{1:T})))$. On the other hand, for problems where the cost functions change rapidly with time, the second algorithm yields a better CCV bound. To see this, recall that $\mathcal{P}_T^\star$ denotes the path length generated by the minimizers of the cost functions and $\mathbb{P}_T^\star$ is the shortest feasible path length, which is independent of the cost functions. Now, if $\mathbb{P}_T^\star = \Omega(\sqrt{T})$, the second algorithm yields $O(\sqrt{T(1 + \mathbb{P}_T^\star)})$ CCV which could be substantially smaller than the $O(\sqrt{T(1 + \mathcal{P}_T^\star)})$ CCV bound guaranteed by Algorithm 2. On the other hand, in terms of computational complexity, the Algorithm 5 is more efficient than Algorithm 2 as the former needs only the gradient information and avoids the costly projection step.

5. Conclusion

In this paper, we propose two different online algorithms for COCO, each achieving sublinear universal dynamic regret and cumulative constraint violation bounds. These bounds improve upon state-of-the-art results and remove the restrictive common feasibility assumption made in the literature. Our proposed algorithms have a modular structure and come with a simple and streamlined analysis using regret decomposition inequality. An important open direction is to establish simultaneous lower bounds on the universal dynamic regret and cumulative constraint violation. It will also be interesting to extend the proposed framework to bandit feedback settings.

6. Acknowledgment

We thank the anonymous reviewers for their constructive feedback, which helped improve the clarity of the presentation. This work was supported in part by the Department of Atomic Energy, Government of India, under project no. RTI4001 and in part by a Google India Faculty Research Award.

REFERENCES

C. J. Argue, Anupam Gupta, Ziye Tang, and Guru Guruganesh. Chasing convex bodies with linear competitive ratio. *J. ACM*, 68(5), August 2021. ISSN 0004-5411. doi: 10.1145/3450349. URL https://doi.org/10.1145/3450349.

Dimitri Bertsekas, Angelia Nedic, and Asuman Ozdaglar. *Convex analysis and optimization*, volume 1. Athena Scientific, 2003.

Omar Besbes, Yonatan Gur, and Assaf Zeevi. Non-stationary stochastic optimization. *Operations research*, 63(5):1227–1244, 2015.

Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

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.

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.

Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

Xi Chen, Yining Wang, and Yu-Xiang Wang. Non-stationary stochastic optimization under $l\_\{p, q\}$-variation measures. *arXiv preprint arXiv:1708.03020*, 2017.

Steven De Rooij, Tim Van Erven, Peter D Grünwald, and Wouter M Koolen. Follow the leader if you can, hedge if you must. *The Journal of Machine Learning Research*, 15(1):1281–1316, 2014.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

Joel Friedman and Nathan Linial. On convex body chasing. *Discrete & Computational Geometry*, 9 (3):293–321, 1993.

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.

Elad Hazan. *Introduction to online convex optimization*. MIT Press, 2022.

Tim Tian Hua, James Baskerville, Henri Lemoine, Mia Hopman, Aryan Bhatt, and Tyler Tracy. Combining cost-constrained runtime monitors for ai safety. *arXiv preprint arXiv:2507.15886*, 2025.

Nicole Immorlica, Karthik Sankararaman, Robert Schapire, and Aleksandrs Slivkins. Adversarial bandits with knapsacks. *Journal of the ACM*, 69(6):1–47, 2022.

Andrew Jacobsen and Ashok Cutkosky. Parameter-free mirror descent. In *Conference on Learning Theory*, pages 4160–4211. PMLR, 2022.

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.

Jordan Lekeufack and Michael I Jordan. An optimistic algorithm for online convex optimization with adversarial constraints. *arXiv preprint arXiv:2412.08060*, 2024.

Nikolaos Liakopoulos, Apostolos Destounis, Georgios Paschos, Thrasyvoulos Spyropoulos, and Panayotis Mertikopoulos. Cautious regret minimization: Online optimization with long-term budget constraints. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3944–3952. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/liakopoulos19a.html.

Qingsong Liu, Wenfei Wu, Longbo Huang, and Zhixuan Fang. Simultaneously achieving sublinear regret and constraint violations for online convex optimization with time-varying constraints. *Performance Evaluation*, 152:102240, 2021.

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.

Shie Mannor, John N Tsitsiklis, and Jia Yuan Yu. Online learning with sample path constraints. *Journal of Machine Learning Research*, 10(3), 2009.

Aryan Mokhtari, Shahin Shahrampour, Ali Jadbabaie, and Alejandro Ribeiro. Online optimization in dynamic environments: Improved regret rates for strongly convex problems. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 7195–7201. IEEE, 2016.

Parvin Nazari and Esmaile Khorram. Dynamic regret of adaptive gradient methods for strongly convex problems. *Optimization*, 73(3):517–543, 2024.

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

Francesco Orabona. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213*, 2019.

Mark Sellke. Chasing convex bodies optimally. In *Proceedings of the Thirty-First Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '20, page 1509–1518, USA, 2020. Society for Industrial and Applied Mathematics.

Abhishek Sinha. BanditQ - Fair Bandits with Guaranteed Rewards. In *Uncertainty in Artificial Intelligence*. PMLR, 2024.

Abhishek Sinha and Rahul Vaze. Optimal algorithms for online convex optimization with adversarial constraints. *Advances in Neural Information Processing Systems*, 37:41274–41302, 2024.

Abhishek Sinha and Rahul Vaze. Beyond $\tilde{O}(\sqrt{t})$ constraint violation for online convex optimization with adversarial constraints. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=yK4Xu7DDd6.

David Snyder, Meghan Booker, Nathaniel Simon, Wenhan Xia, Daniel Suo, Elad Hazan, and Anirudha Majumdar. Online learning for obstacle avoidance. In *Conference on Robot Learning*, pages 2926–2954. PMLR, 2023.

Rahul Vaze and Abhishek Sinha. $O(\sqrt{T})$ static regret and instance dependent constraint violation for constrained online convex optimization. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=YmbQ0qnQ76.

Akifumi Wachi, Xun Shen, and Yanan Sui. A survey of constraint formulations in safe reinforcement learning. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, IJCAI '24, 2024. ISBN 978-1-956792-04-1. doi: 10.24963/ijcai.2024/913. URL https://doi.org/10.24963/ijcai.2024/913.

Tianbao Yang, Lijun Zhang, Rong Jin, and Jinfeng Yi. Tracking slowly moving clairvoyant: Optimal dynamic regret of online learning with true and noisy gradient. In *International Conference on Machine Learning*, pages 449–457. PMLR, 2016.

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.

Xinlei Yi, Xiuxian Li, Tao Yang, Lihua Xie, Tianyou Chai, and H Karl. Regret and cumulative constraint violation analysis for distributed online constrained convex optimization. *IEEE Transactions on Automatic Control*, 2022.

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. *Journal of Machine Learning Research*, 21(1):1–24, 2020.

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

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

Lijun Zhang, Tianbao Yang, Jinfeng Yi, Rong Jin, and Zhi-Hua Zhou. Improved dynamic regret for non-degenerate functions. *Advances in Neural Information Processing Systems*, 30, 2017.

Lijun Zhang, Shiyin Lu, and Zhi-Hua Zhou. Adaptive online learning in dynamic environments. *Advances in neural information processing systems*, 31, 2018.

Lijun Zhang, Guanghui Wang, Jinfeng Yi, and Tianbao Yang. A simple yet universal strategy for online convex optimization. In *International Conference on Machine Learning*, pages 26605–26623. PMLR, 2022.

Peng Zhao, Yu-Jie Zhang, Lijun Zhang, and Zhi-Hua Zhou. Dynamic regret of convex and smooth functions. *Advances in Neural Information Processing Systems*, 33:12510–12520, 2020.

Peng Zhao, Yu-Jie Zhang, Lijun Zhang, and Zhi-Hua Zhou. Adaptivity and non-stationarity: Problem-dependent dynamic regret for online convex optimization. *Journal of Machine Learning Research*, 25(98):1–52, 2024.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th international conference on machine learning (icml-03)*, pages 928–936, 2003.

7. Appendix

7.1. Visualizing the transformed functions

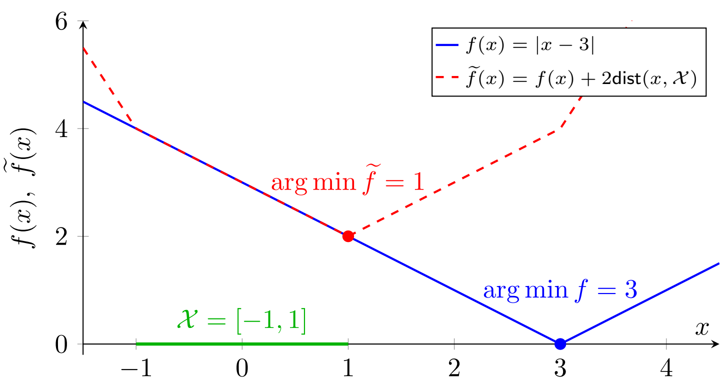

Figure 1: Illustration of the transformation $f \rightsquigarrow \widetilde{f}$. Here $f(x) = |x-3|$, $\mathcal{X} = [-1,1], G = 1$, and $\widetilde{f}(x) = f(x) + 2\,\mathsf{dist}(x, \mathcal{X})$. After the transformation, the original minimizer at $x = 3$ is sent to $x = 1$, which lies at the boundary of $\mathcal{X}$.

Function of one variable: Figure 1 below illustrates a single variable convex function $f(x) = |x-3|, x \in \mathbb{R}$ and the associated auxiliary cost function $\tilde{f}(x)$. The feasible set is chosen to be the closed interval $\mathcal{X} = [-1, 1]$. From the figure, we observe that while the unconstrained minimum value of the function f is attained at $x = 3$, the unconstrained minimum of the auxiliary function $\tilde{f}$ is attained at $x = 1$, which is *in* the feasible set. This verifies the conclusion from Lemma 12.

Function of two variables: Figure 2 illustrates a similar observation for a convex function with two variables defined as $f(x) = \sqrt{(x-3)^2 + y^2},\ x, y \in \mathbb{R}$, with the convex feasible set $S = \{(x,y) \in \mathbb{R}^2 : x^2 + y^2 \leq 1\}$. The global minimum of the auxiliary function is shifted to the feasible point $(1,0)$ from the global minimum of the original function located at $(3,0)$.

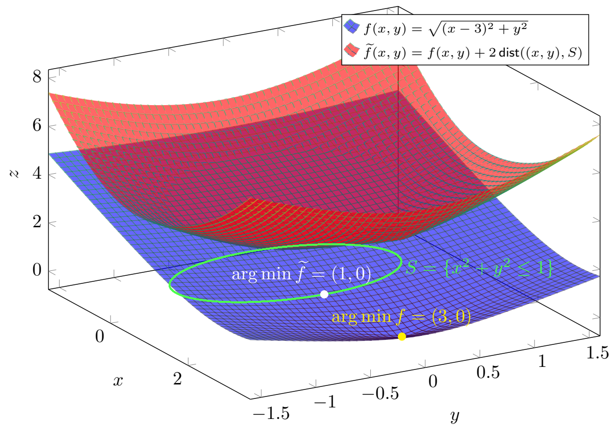

Figure 2: Illustration of the transformation $f \rightsquigarrow \widetilde{f}$. Here $f(x,y) = \sqrt{(x-3)^2 + y^2}, G = 1$, feasible set $S = \{(x,y) : x^2 + y^2 \leq 1\}$, and $\widetilde{f}(x,y) = f(x,y) + 2\,\mathrm{dist}((x,y),S)$. After the transformation, the original minimizer at $(3,0)$ is sent to $(1,0)$ which lies at the boundary of S.

7.2. Necessity of the $\mathsf{dist}(\cdot, \mathcal{X}_t^\star)$ term in the definition of the surrogate function (8)

In this section, we give a simple counterexample that shows that running *any* no-regret OCO algorithm on the surrogate cost functions $\hat{f}_t := f_t + g_t^+, t \geq 1$, may lead to a *linear* CCV. Consider the following one-dimensional time-invariant problem where the decision set is $\mathcal{X} = [-1,1]$ and the cost and constraint functions are given by

$$f_t(x) = \big(x - \frac{1}{2}\big)^2, \text{ and } g_t(x) = 0.2x, \; t \geq 1.$$

Clearly, all feasible sets are identical and given by $\mathcal{X}_t^\star = [-1,0],\; t \geq 1$. The minimizer $x^\star$ of the surrogate cost function can be found by setting the derivative of $\hat{f}_t(x) \equiv \big(x - \frac{1}{2}\big)^2 + 0.2x^+$ to zero, yielding $x^\star = 2/5$. This implies that the iterates of *any* no-regret policy must converge to the point $x^\star = 2/5$ as T gets large. Hence, after T rounds, the CCV is approximately $\approx 0.2 \times \frac{2}{5} \times T = 0.08T$, which grows *linearly* with T. See Figure 3 for an illustration.

The reason behind the above issue is that the magnitude of the gradient of the constraint function is negligible compared to that of the cost function. Algorithm 2 avoids this issue by adding the $\mathsf{dist}(\cdot)$ function to the surrogate cost. In this case, the surrogate cost becomes

$$\hat{f}_t(x) \equiv \big(x - \frac{1}{2}\big)^2 + 0.2x^+ + 6x^+, \;\; \forall t \geq 1.$$

It can be verified that the unconstrained minimum of the above surrogate cost function lies at $x^\star = 0 \in \mathcal{X}_t^\star$ (see Lemma 12), leading to a sublinear CCV bound.

7.3. Proof of Theorem 18

Define the sequence $y_{t+1} := x_t - \eta_t \nabla_t, \; \forall t \geq 1$, where $\{\eta_t\}_{t\geq 1}$ is any non-increasing adaptive step size sequence. From Algorithm 3, we have $x_{t+1} = \mathsf{Proj}_{\mathcal{X}}(y_{t+1}), \; \forall t \geq 1$. Let $x_{1:T}^\star$ be any

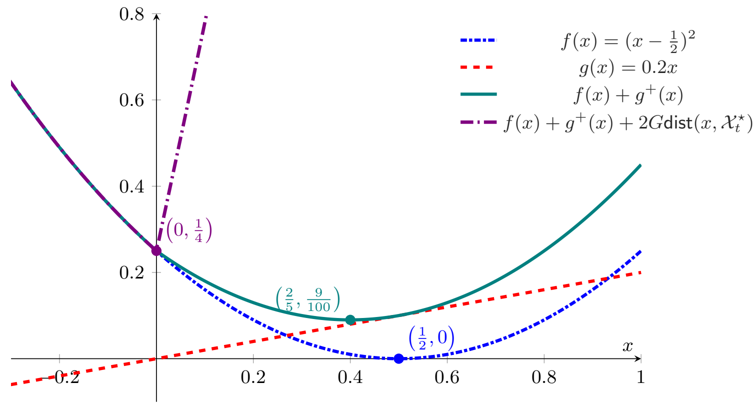

Figure 3: Illustrating various original and the surrogate cost functions and their minima

comparator sequence with $x_t^\star \in \mathcal{X}, \forall t \in [T]$. Then we have

$$||x_{t+1} - x_t^\star||^2 \overset{(a)}{\leq} ||y_{t+1} - x_t^\star||^2 = ||x_t - x_t^\star||^2 + \eta_t^2||\nabla_t||^2 - 2\eta_t \nabla_t^\top (x_t - x_t^\star),$$

where inequality (a) follows from the non-expansive property of Euclidean projection and the second equality follows from the definition of y_{t+1}. Rearranging the above inequality, we have:

$$2\nabla_t^\top (x_t - x_t^\star) \leq \frac{||x_t - x_t^\star||^2 - ||x_{t+1} - x_t^\star||^2}{\eta_t} + \eta_t ||\nabla_t||^2. \tag{25}$$

Using the convexity of the cost functions and summing the above inequalities for $1 \leq t \leq T$, we obtain:

$$2\sum_{t=1}^{T} \left(\hat{f}_t(x_t) - \hat{f}_t(x_t^\star)\right) \leq 2\sum_{t=1}^{T} \nabla_t^\top (x_t - x_t^\star) \leq \underbrace{\sum_{t=1}^{T} \frac{||x_t - x_t^\star||^2 - ||x_{t+1} - x_t^\star||^2}{\eta_t}}_{(A)} + \sum_{t=1}^{T} \eta_t ||\nabla_t||^2. \tag{26}$$

Next we simplify term (A) in Eqn. (26). We have

$$\begin{aligned} (A) &= \frac{||x_1^\star - x_1||^2}{\eta_1} - \frac{||x_T^\star - x_{T+1}||^2}{\eta_T} + \sum_{t=1}^{T-1} \frac{||x_{t+1} - x_{t+1}^\star||^2}{\eta_{t+1}} - \frac{||x_{t+1} - x_t^\star||^2}{\eta_t} \\ &= \frac{||x_1^\star - x_1||^2}{\eta_1} - \frac{||x_T^\star - x_{T+1}||^2}{\eta_T} + \underbrace{\sum_{t=1}^{T-1} \frac{\eta_t ||x_{t+1} - x_{t+1}^\star||^2 - \eta_{t+1} ||x_{t+1} - x_t^\star||^2}{\eta_t \eta_{t+1}}}_{\text{(B)}}. \end{aligned} \tag{27}$$

We next upper bound term (B) in (27). We have

$$
\begin{aligned}
(B) &= \sum_{t=1}^{T-1} \frac{||\sqrt{\eta_t}x_{t+1} - \sqrt{\eta_t}x^\star_{t+1}||^2 - ||\sqrt{\eta_{t+1}}x_{t+1} - \sqrt{\eta_{t+1}}x^\star_t||^2}{\eta_t\eta_{t+1}} \\
&= \sum_{t=1}^{T-1} \frac{\langle(\sqrt{\eta_t} + \sqrt{\eta_{t+1}})x_{t+1} - \sqrt{\eta_t}x^\star_{t+1} - \sqrt{\eta_{t+1}}x^\star_t, (\sqrt{\eta_t} - \sqrt{\eta_{t+1}})x_{t+1} - \sqrt{\eta_t}x^\star_{t+1} + \sqrt{\eta_{t+1}}x^\star_t\rangle}{\eta_t\eta_{t+1}} \\
&\leq \sum_{t=1}^{T-1} \frac{||(\sqrt{\eta_t} + \sqrt{\eta_{t+1}})x_{t+1} - \sqrt{\eta_t}x^\star_{t+1} - \sqrt{\eta_{t+1}}x^\star_t||\;||(\sqrt{\eta_t} - \sqrt{\eta_{t+1}})x_{t+1} - \sqrt{\eta_t}x^\star_{t+1} + \sqrt{\eta_{t+1}}x^\star_t||}{\eta_t\eta_{t+1}},
\end{aligned}
$$

where the last step follows from Cauchy-Schwarz inequality. Note that the first term in the numerator can be bounded as:

$$
||(\sqrt{\eta_t}+\sqrt{\eta_{t+1}})x_{t+1}-\sqrt{\eta_t}x^\star_{t+1}-\sqrt{\eta_{t+1}}x^\star_t|| \leq \sqrt{\eta_t}||x_{t+1}-x^\star_{t+1}||+\sqrt{\eta_{t+1}}||x_{t+1}-x^\star_t|| \leq (\sqrt{\eta_t}+\sqrt{\eta_{t+1}})D,
$$

where we have used the triangle inequality. Using this, we have

$$
\begin{aligned}
(B) &\leq D\sum_{t=1}^{T-1} \frac{(\sqrt{\eta_t} + \sqrt{\eta_{t+1}})\;||(\sqrt{\eta_t} - \sqrt{\eta_{t+1}})x_{t+1} - \sqrt{\eta_t}x^\star_{t+1} + \sqrt{\eta_{t+1}}x^\star_t||}{\eta_t\eta_{t+1}} \\
&= D\sum_{t=1}^{T-1} \frac{(\sqrt{\eta_t} + \sqrt{\eta_{t+1}})||(\sqrt{\eta_t} - \sqrt{\eta_{t+1}})(x_{t+1} - x^\star_{t+1}) + \sqrt{\eta_{t+1}}(x^\star_t - x^\star_{t+1})||}{\eta_t\eta_{t+1}} \\
&\overset{(a)}{\leq} D\sum_{t=1}^{T-1} \frac{(\sqrt{\eta_t} + \sqrt{\eta_{t+1}})\;[(\sqrt{\eta_t} - \sqrt{\eta_{t+1}})D + \sqrt{\eta_{t+1}}||x^\star_t - x^\star_{t+1}||]}{\eta_t\eta_{t+1}} \\
&\overset{(b)}{\leq} D\sum_{t=1}^{T-1} \frac{(\eta_t - \eta_{t+1})D\; + 2\eta_t||x^\star_{t+1} - x^\star_t||}{\eta_t\eta_{t+1}} \\
&= D^2(\frac{1}{\eta_T} - \frac{1}{\eta_1}) + 2D\sum_{t=1}^{T-1} \frac{||x^\star_{t+1} - x^\star_t||}{\eta_{t+1}} \\
&\overset{(c)}{\leq} D^2(\frac{1}{\eta_T} - \frac{1}{\eta_1}) + \frac{2D\mathcal{P}_T(x^\star_{1:T})}{\eta_T},
\end{aligned}
$$

where in step (a), we have used the triangle inequality, and in step (c), we have used the definition of the path length of the comparator. The non-increasing property of the step sizes, *i.e.,* $\eta_t \geq \eta_{t+1}, \forall t \geq 1$ was used in steps (a), (b), and (c). Finally, combining the above bound with Eqns. (26) and (27), we conclude

$$
\begin{aligned}
&2(\sum_{t=1}^{T} \hat{f}_t(x_t) - \hat{f}_t(x^\star_t)) \\
&\leq \frac{||x^\star_1 - x_1||^2}{\eta_1} - \frac{||x^\star_T - x_{T+1}||^2}{\eta_T} + D^2(\frac{1}{\eta_T} - \frac{1}{\eta_1}) + \frac{2D\mathcal{P}_T(x^\star_{1:T})}{\eta_T} + \sum_{t=1}^{T}\eta_t||\nabla_t||^2 \\
&\leq \frac{D^2 + 2D\mathcal{P}_T(x^\star_{1:T})}{\eta_T} + \sum_{t=1}^{T}\eta_t||\nabla_t||^2.
\end{aligned}
$$

Hence, the universal dynamic regret (3) of Algorithm 3 can be upper bounded as

$$\text{UD-REGRET}(\hat{f}_{1:T}; x^\star_{1:T}) \leq \frac{\max(D^2, 2D)(1 + \mathcal{P}_T(x^\star_{1:T}))}{2\eta_T} + \underbrace{\frac{1}{2}\sum_{t=1}^{T} \eta_t ||\nabla_t||^2}_{(C)}. \quad (28)$$

The dynamic regret bound in Eqn. (28) holds for Algorithm 3 with any non-increasing step sizes. Assuming the path-length is known to be bounded as $\mathcal{P}_T(x^\star_{1:T}) \leq \mathcal{P}_T$, using the specific choice of the step size sequence $\eta_t = \frac{(D+1)\sqrt{1+\mathcal{P}_T}}{\sqrt{2\sum_{\tau=1}^{t}||\nabla_\tau||^2}}, t \geq 1$, we can upper bound term (C) as follows

$$\begin{aligned}
\frac{1}{2}\sum_{t=1}^{T} \eta_t ||\nabla_t||^2 &= \frac{(D+1)\sqrt{1+\mathcal{P}_T}}{2\sqrt{2}} \sum_{t=1}^{T} \frac{||\nabla_t||^2}{\sqrt{\sum_{\tau=1}^{t} ||\nabla_\tau||^2}} \\
&\leq \frac{(D+1)\sqrt{1+\mathcal{P}_T}}{2\sqrt{2}} \int_0^{\sum_{t=1}^{T}||\nabla_t||^2} \frac{dz}{\sqrt{z}} \\
&= \frac{(D+1)\sqrt{1+\mathcal{P}_T}}{\sqrt{2}} \sqrt{\sum_{t=1}^{T} ||\nabla_t||^2}.
\end{aligned}$$

Hence, from (28), the dynamic regret for Algorithm 3 can be bounded as

$$\text{D-REGRET}(\hat{f}_{1:T}; \mathcal{P}_T) \leq (D+1)\sqrt{2(1+\mathcal{P}_T)}\sqrt{\sum_{t=1}^{T} ||\nabla_t||^2}. \quad (29)$$

In a similar fashion, plugging in the path-length independent step size sequence $\eta_t = \frac{(D+1)}{\sqrt{2\sum_{\tau=1}^{t}||\nabla_\tau||^2}}$, $t \geq 1$, into (28), we obtain the universal dynamic regret bound (13).

7.4. Proof of Theorem 19

Since $\text{diam}(\mathcal{X}) \leq D$, the path length $\mathcal{P}_T(u_{1:T})$ of any comparator sequence can be trivially bounded as $0 \leq \mathcal{P}_T(u_{1:T}) \leq DT$. Algorithm 4 maintains $N := \lceil \frac{1}{2}\log_2(1+DT) \rceil + 1$ experts where the i^{th} expert estimates the path length to be $\rho_i := 2^{i-1},\ i \in [N]$. This ensures that for each candidate path length $\mathcal{P}_T(u_{1:T})$, there is an estimate ρ_i which approximates the true path length within a constant factor, *i.e.,*

$$\forall \mathcal{P}_T(u_{1:T}),\ \exists \rho_i \quad \text{s.t.} \quad \frac{1}{2}\rho_i \leq \sqrt{1+\mathcal{P}_T(u_{1:T})} \leq \rho_i. \quad (30)$$

At round t, the i^{th} expert predicts $x^i_t \in \mathcal{X}$ using AdaGrad (Algorithm 3). Let us define X_t to be a $d \times N$ matrix with x^i_t being its i^{th} column. The expert-tracking algorithm AdaHedge gives a probability distribution $w_t \in \Delta_{N-1}$. The final action at round t is chosen to be a convex combination of all experts' output, *i.e.,* $x_t = X_t w_t$.

Now for a comparator sequence $u_{1:T}$, assume that Eqn. (30) holds for some $i \in [N]$. Then we have

$$\begin{aligned}
&\text{UD-REGRET}(f_{1:T}; u_{1:T}) \\
&= \sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(u_t) \\
&\overset{(a)}{=} \sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(x_t^i) + \sum_{t=1}^{T} f_t(x_t^i) - \sum_{t=1}^{T} f_t(u_t) \\
&\overset{(b)}{\leq} \underbrace{\sum_{t=1}^{T} \langle X_t^T \nabla f_t(X_t w_t)\,,\, w_t - e_i \rangle}_{\text{term (I)}} + \underbrace{\sum_{t=1}^{T} f_t(x_t^i) - \sum_{t=1}^{T} f_t(u_t)}_{\text{term (II)}},
\end{aligned} \tag{31}$$

where inequality (b) follows from the convexity of the cost function f_t. Next, we bound term (I) and term (II) separately.

Bounding term (I): To bound term (I), we invoke the following result from Orabona (2019, Section 7.6) concerning the AdaHedge algorithm. The setup for the AdaHedge algorithm is the same as the classic Prediction with Experts' Advice (PEA) problem with N experts (Cesa-Bianchi and Lugosi, 2006). The main distinguishing feature of AdaHedge compared to Hedge is that AdaHedge is adaptive to the range of the loss vectors, and hence, no uniform upper bound to the loss vectors is necessary to tune its learning rate.

Theorem 22 (Static Regret bound of AdaHedge) *For the Prediction with Expert Advice problem, the* AdaHedge *algorithm, when run sequentially for T rounds with the loss vectors $l_1, \ldots, l_T \in \mathbb{R}^N$, achieves the following static regret bound with respect to any fixed expert $i \in [N]$:*

$$\textit{Regret}_T(i) \leq 2\sqrt{(4 + \ln N) \sum_{t=1}^{T} ||l_t||_\infty^2}.$$

We now apply the above result to the linear loss functions $\langle l_t, w \rangle = \langle X_t^T \nabla f_t(X_t w_t), w \rangle, t \geq 1$. We can bound the ℓ_∞-norm of the t^{th} loss vector as follows:

$$||X_t^T \nabla f_t(X_t w_t)||_\infty = \max_{1 \leq i \leq N} |\langle x_t^i, \nabla f_t(X_t w_t) \rangle| \overset{(a)}{\leq} ||\nabla f_t(X_t w_t)||_2\, ||x_t^i||_2 \overset{(b)}{\leq} \frac{D}{2} ||\nabla_t||_2, \tag{32}$$

where in (a) we have used the Cauchy–Schwarz inequality and (b) follows from the boundedness of the decision set and the assumption that the decision set lies in an Euclidean ball of radius D. Finally, this yields the following upper bound for term (I):

$$\text{Term (I)} \leq D\sqrt{4 + \ln N} \sqrt{\sum_{t=1}^{T} ||\nabla_t||_2^2}. \tag{33}$$

Bounding term (II): Recall that the expert i is defined to be such that its ρ_i satisfies (30). Furthermore, expert i runs AdaGrad with the learning rate

$$\eta_t^i = \frac{(D+1)\rho_i}{\sqrt{2\sum_{\tau=1}^{t} ||\nabla f(x_\tau^i)||^2}}, t \geq 1. \tag{34}$$

Using Eqn. (12) from Theorem 18, the universal dynamic regret of the i^{th} expert can be bounded as:

$$\text{Term (II)} \leq \text{D-Regret}_T(\mathcal{P}_T(u_{1:T})) \leq 2(D+1)\sqrt{2(1+\mathcal{P}_T(u_{1:T}))}\sqrt{\sum_{t=1}^{T} ||\nabla_t||^2}. \tag{35}$$

Combining the bounds for terms (I) and (II), we arrive at the desired result.