Multi-Constraint Online Convex Optimization with Adversarial Constraints

Anonymous authors
Paper under double-blind review

Abstract

We study online convex optimization with multiple adversarial constraints, where at each round a learner selects an action, and an adversary simultaneously reveals a convex cost function and K convex constraint functions. The learner aims to minimize regret while keeping the cumulative constraint violation (CCV) of each individual constraint small. We introduce the Multi-Constraint Constrained Online Convex Optimization (MC-COCO) framework and develop a unified algorithmic approach based on exponential Lyapunov potentials. The key insight is that encoding all K constraint violations via the potential $S_t = \sum_{k=1}^{K} e^{\lambda Q_k(t)}$ yields a surrogate cost whose growth ratio is controlled by the maximum single-round violation rather than the number of constraints K. This decoupling enables a per-constraint CCV of $\widetilde{O}(T^{1-\beta} \ln K)$, where $\beta \in [0, 1]$ is a tunable regret-CCV trade-off parameter, improving qualitatively over the linear K-dependence of naive approaches. We instantiate the framework across three canonical settings (constrained experts, general Lipschitz-convex, and smooth convex) and further develop extensions for heterogeneous constraint prioritization (where critical constraints can be controlled at the $\widetilde{O}(T^{1-\beta}/\alpha_k)$ level) and long-term budget feasibility. Experiments on adversarial instances with up to $K = 100$ constraints validate the theoretical bounds and confirm the logarithmic scaling in K.

1 Introduction

Online convex optimization (OCO) provides a principled framework for sequential decision-making under uncertainty, with broad applications in resource allocation, network scheduling, and online learning (Zinkevich, 2003; Hazan, 2016; Orabona, 2019; Shalev-Shwartz, 2025). At each round, a learner selects an action from a convex set, an adversary reveals a convex cost function, and the learner's goal is to minimize regret, which is the excess cumulative cost relative to the best fixed action in hindsight. In many practical scenarios, the learner must also satisfy additional constraints that vary over time: a network router must respect bandwidth limits on each of its output links, a resource allocator must stay within budgets for each resource type, and a fair classifier must control violation rates for each protected group (Chen et al., 2017; Neely, 2010). When these constraints are adversarial (unknown in advance and chosen by nature or an adversary), zero constraint violation at every round is generally impossible. The natural objective then becomes jointly achieving low regret and low cumulative constraint violation (CCV), giving rise to the field of constrained online convex optimization (COCO).

The study of COCO has progressed substantially over the past decade, though almost exclusively in the single-constraint ($K = 1$) regime. Mannor & Tsitsiklis (2006) initiated this line of work by establishing fundamental regret–CCV trade-offs. Mahdavi et al. (2012) then obtained $O(\sqrt{T})$ regret with $O(T^{3/4})$ CCV, and subsequent efforts by Jenatton et al. (2016) and Yuan & Lamperski (2018) improved the CCV to $O(\sqrt{T})$ while maintaining $O(\sqrt{T})$ regret. Under stochastic constraints, Yu & Neely (2020) achieved the remarkable combination of $O(\sqrt{T})$ regret with $O(1)$ CCV. In the fully adversarial case, Guo et al. (2022) obtained $O(\sqrt{T})$ regret with bounded CCV, and Sinha & Vaze (2024) recently established optimal rates, essentially settling the single-constraint problem. However, real applications almost universally involve $K \geq 2$ constraints per round. Existing work has considered multiple constraints in stochastic (Yu et al., 2017) or

Table 1: Comparison of per-constraint CCV guarantees. Here $\beta \in [0, 1]$ is a tunable regret–CCV trade-off parameter. Existing multi-constraint methods either provide only aggregate bounds or scale linearly in K; our framework achieves $\ln K$ dependence across all settings. "Indep. queues" denotes the natural baseline that maintains K separate virtual queues and analyzes each independently via standard drift arguments.

Method	Setting	Per-constraint CCV	K-dep.
Mannor & Tsitsiklis (2006)	$K = 1$, adversarial	$O(T^{3/4})$	—
Mahdavi et al. (2012)	$K = 1$, adversarial	$O(T^{3/4})$	—
Yuan & Lamperski (2018)	$K = 1$, adversarial	$O(\sqrt{T})$	—
Sinha & Vaze (2024)	$K = 1$, adversarial	$O(\sqrt{T})$ (optimal)	—
Yu et al. (2017)	$K \geq 1$, stochastic	$O(\sqrt{T})$	not explicit
Neely (2010)	$K \geq 1$, adversarial	$O(\sqrt{T})$	$O(K)$
Indep. queues	$K \geq 1$, adversarial	$O(K \cdot T^{1-\beta})$	$O(K)$
Ours (MC-1)	$K \geq 1$, adversarial	$\widetilde{O}(\ln N \cdot T^{1-\beta} \cdot \ln K)$	$O(\ln K)$
Ours (MC-2)	$K \geq 1$, adversarial	$\widetilde{O}(d \cdot T^{1-\beta} \cdot \ln K)$	$O(\ln K)$
Ours (MC-3)	$K \geq 1$, adversarial	$O(D^2 M \cdot T^{1-\beta} \cdot \ln K)$	$O(\ln K)$

Note: Throughout the paper, $\ln K$ in informal statements and table entries is shorthand for $\ln(K + T + \cdots)$ appearing in the precise bounds (e.g., Theorem 2). When $K = 1$ the $\ln K$ factor does not vanish; the exact bound reduces to $O(\ln T)$, consistent with known single-constraint results (Corollary 3).

mixed stochastic-adversarial settings (Chen et al., 2024), and the drift-plus-penalty framework (Neely, 2010) naturally accommodates multiple queues. Yet these analyses share a common limitation: they either provide only aggregate CCV bounds (summing all constraint violations together), or yield per-constraint guarantees whose dependence on K is linear, which is equivalent to treating each constraint independently. This leaves open a concrete theoretical question: in the fully adversarial multi-constraint regime, is a linear dependence on K in the per-constraint CCV truly unavoidable? Controlling the aggregate violation $\sum_{k=1}^{K} \mathrm{CCV}_k(T)$ is straightforward but too coarse, as it says nothing about individual constraints. Meanwhile, maintaining K separate constraint queues and analyzing them independently suggests that the per-constraint CCV should grow linearly in K, since the algorithm must spread its corrective capacity across all constraints. Whether these queues can be coupled to avoid this linear blow-up, while retaining full adversarial robustness, has remained unclear.

In this paper, we introduce the Multi-Constraint Constrained Online Convex Optimization (MC-COCO) framework and develop an algorithmic approach based on exponential Lyapunov potentials. The key idea is to encode all K constraint violation queues into a single surrogate cost via the potential $S_t = \sum_{k=1}^{K} e^{\lambda Q_k(t)}$, a soft-max-type aggregation that biases the update toward whichever constraint is currently most violated. The critical technical observation is that the one-step growth ratio G_t/G_{t-1} of the resulting cost upper bound depends on the maximum single-round constraint violation (which is at most 1), rather than on the number of constraints K. The number K enters only through the initial value $G_0 = 1 + \lambda K$, contributing a mere $\ln K$ factor after taking logarithms. This decoupling underlies the logarithmic dependence on K throughout our results. Table 1 summarizes the per-constraint CCV guarantees of our framework against existing approaches, highlighting the qualitative improvement from linear to logarithmic K-dependence. Our main contributions are as follows:

- We establish the first explicit logarithmic dependence on the number of constraints K in per-constraint CCV for fully adversarial multi-constraint OCO. Under a tunable regret-violation trade-off parameter $\beta \in [0, 1]$, each constraint's CCV scales as $\widetilde{O}(T^{1-\beta} \ln K)$. As shown in Table 1, this constitutes a qualitative improvement over the linear K-dependence incurred by all prior analyses.
- We develop an exponential Lyapunov reduction that converts any multi-constraint problem into a single unconstrained online optimization instance. We instantiate this reduction across three canonical settings, namely constrained experts, Lipschitz-convex, and smooth-convex environments. In each case, the regret bound preserves the standard rate of the base learner up to an additive $O(T^{\beta} + K)$ overhead.

- We extend the core framework in two theoretically motivated directions. First, we show that heterogeneous constraint prioritization can be achieved by assigning per-constraint Lyapunov parameters $\lambda_k = \alpha_k \Lambda$, tightening the CCV of critical constraints to $\widetilde{O}(T^{1-\beta}/\alpha_k)$ while preserving the regret order. Second, we prove that the framework accommodates long-term budget feasibility, where the comparator satisfies $\sum_t g_{k,t}(x^\star) \leq B_{k,T}$ rather than pointwise feasibility; the CCV absorbs an additive $B_{\max}$ term without degrading the regret.

2 Related Work

We organize the related literature into three themes: constrained online convex optimization, Lyapunov and drift-based methods, and lower bounds for constrained online learning.

Constrained online convex optimization. We distinguish three settings based on the number of constraints and the nature of the environment. In the single-constraint adversarial setting, Mannor & Tsitsiklis (2006) first showed that simultaneous $o(T)$ regret and $o(T)$ CCV is achievable, and Mahdavi et al. (2012) refined the rates to $O(\sqrt{T})$ regret with $O(T^{3/4})$ CCV. Yuan & Lamperski (2018) achieved $O(\sqrt{T})$ for both quantities, and Sinha & Vaze (2024) recently established matching lower bounds to settle this case. When constraint functions are drawn i.i.d. in the stochastic setting, stronger guarantees become possible. Yu & Neely (2020) obtained $O(\sqrt{T})$ regret with $O(1)$ CCV for a single constraint, and Yu et al. (2017) extended the stochastic analysis to $K \geq 1$ constraints without explicit per-constraint dependence on K. Finally, for multiple adversarial constraints, Chen et al. (2024) considered mixed stochastic-adversarial models, and Neely (2010) provided queue-based algorithms applicable to $K \geq 1$ adversarial constraints. However, none of these works provides per-constraint CCV guarantees with explicit sub-linear dependence on K. In contrast, our results give per-constraint bounds with an $O(\ln K)$ dependence, avoiding the linear K factor that arises in prior analyses.

Lyapunov and drift-based methods. The drift-plus-penalty framework of Neely (2010) maintains a separate virtual queue for each constraint and bounds the total CCV via a quadratic Lyapunov function $\sum_k Q_k(t)^2$. Because each queue is analyzed independently, the per-constraint CCV inherits a factor of K, since bounding the maximum queue length through the sum introduces an overhead similar to a union bound. The polynomial Lyapunov function does not use the soft-max structure that allows exponential potentials to couple all constraints simultaneously. In contrast, the exponential potential $\sum_k e^{\lambda Q_k(t)}$ used in our framework automatically tracks the worst-case constraint through its maximum term. This ensures that a single potential descent argument controls all K queues at once without incurring a per-constraint K penalty. This functional form is reminiscent of multiplicative weights (De Rooij et al., 2014), but it operates on constraint violation queues rather than on action probabilities.

Lower bounds and optimality. For $K = 1$, the $\widetilde{O}(\sqrt{T})$ regret-CCV trade-off is known to be tight (Mannor & Tsitsiklis, 2006; Sinha & Vaze, 2024). For $K \geq 2$, as discussed following Corollary 3, a straightforward decoupling argument shows that $\mathrm{CCV}_{\mathrm{total}} \geq \Omega(K)$ is unavoidable since the adversary can distribute violations across independent constraints. Our logarithmic per-constraint dependence is substantially better than the linear scaling of existing methods. Whether the $\ln K$ factor can be further improved remains an open question discussed in Section 6.

3 Problem Setup and Notation

Notation. We write $[N] = \{1, \ldots, N\}$ for positive integers N. For vectors $v \in \mathbb{R}^d$, $\|v\|$ denotes the Euclidean norm unless otherwise specified. The probability simplex is $\Delta_N = \{p \in \mathbb{R}^N_+ : \sum_{i=1}^N p_i = 1\}$. Throughout, ln denotes the natural logarithm and $\exp(\cdot)$ the exponential function. The notation $\widetilde{O}(\cdot)$ hides factors polylogarithmic in T, K, and problem parameters. For a convex set $\mathcal{X} \subseteq \mathbb{R}^d$, its diameter is $D = \sup_{x,y \in \mathcal{X}} \|x - y\|$, and $\Pi_{\mathcal{X}}(\cdot)$ denotes the Euclidean projection onto $\mathcal{X}$.

We use T for the time horizon, K for the number of constraint functions per round, N for the number of experts (in the experts setting), and d for the ambient dimension of the decision set. The convex cost and

constraint functions at round t are denoted $f_t : \mathcal{X} \to [0,1]$ and $g_{k,t} : \mathcal{X} \to [0,1]$, respectively; $x_t \in \mathcal{X}$ is the learner's action and $x^\star \in \mathcal{X}$ is the feasible comparator. The parameter $\beta \in [0,1]$ controls the regret–CCV trade-off, and $\lambda > 0$ is the Lyapunov parameter. We write $Q_k(t) = \sum_{s=1}^{t} g_{k,s}(x_s)$ for the cumulative constraint violation queue of constraint k. Finally, L and M denote the Lipschitz and smoothness constants of the cost and constraint functions, respectively.

Online protocol. We study a repeated game over T rounds between a *learner* and an *oblivious adversary*. The decision set $\mathcal{X} \subseteq \mathbb{R}^d$ is convex and compact. At each round $t = 1, \ldots, T$, the learner selects an action $x_t \in \mathcal{X}$, after which the adversary simultaneously reveals a convex cost function $f_t : \mathcal{X} \to [0,1]$ and K convex constraint functions $g_{1,t}, \ldots, g_{K,t} : \mathcal{X} \to [0,1]$; the learner then incurs cost $f_t(x_t)$ and constraint values $g_{k,t}(x_t)$ for each $k \in [K]$. The adversary is oblivious in the sense that the entire sequence $\{(f_t, g_{1,t}, \ldots, g_{K,t})\}_{t=1}^{T}$ is fixed before the game begins, though unknown to the learner. Since $g_{k,t} : \mathcal{X} \to [0,1]$, the value $g_{k,t}(x) = 0$ indicates that action x exactly satisfies constraint k at round t, while $g_{k,t}(x) > 0$ indicates a violation of magnitude $g_{k,t}(x)$. This non-negative convention is standard in the COCO literature (see, e.g., Mahdavi et al. (2012); Yuan & Lamperski (2018)): given an original signed constraint $h_{k,t}(x) \le 0$, one defines $g_{k,t}(x) = [h_{k,t}(x)]_+ = \max(0, h_{k,t}(x))$, which is convex and non-negative.

Performance measures. We evaluate the learner via two complementary criteria.

Definition 1 (Regret and Cumulative Constraint Violation)**.** *Given a comparator $x^\star \in \mathcal{X}$, the* regret *and* per-constraint cumulative constraint violation *(CCV) are*

$$\mathrm{Regret}_T(x^\star) = \sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(x^\star), \tag{1}$$

$$\mathrm{CCV}_k(T) = \sum_{t=1}^{T} g_{k,t}(x_t), \quad k \in [K]. \tag{2}$$

The total CCV *is* $\mathrm{CCV}_{\mathrm{total}}(T) = \sum_{k=1}^{K} \mathrm{CCV}_k(T)$ *and the* maximum per-constraint CCV *is* $\max_{k\in[K]} \mathrm{CCV}_k(T)$.

The regret measures the excess cumulative cost relative to the best fixed action in hindsight, and is the standard performance criterion in online convex optimization. The per-constraint CCV captures how much each individual constraint is violated in aggregate; controlling $\mathrm{CCV}_k(T)$ for every k is strictly stronger than controlling only the total $\mathrm{CCV}_{\mathrm{total}}(T)$, since the latter allows large violations on a few constraints to be masked by satisfaction of others.

Assumptions. We impose the following assumptions on the problem structure. The first is required throughout; the remaining two are invoked only in the corresponding instantiations.

Assumption 1 (Feasible comparator)**.** *There exists a comparator $x^\star \in \mathcal{X}$ such that $g_{k,t}(x^\star) = 0$ for all $k \in [K]$ and $t \in [T]$.*

Assumption 1 requires the existence of at least one action that exactly satisfies all constraints at every round. Since $g_{k,t} : \mathcal{X} \to [0,1]$, the condition $g_{k,t}(x^\star) = 0$ is the strongest possible feasibility requirement. This is the standard benchmark in constrained online optimization (Mannor & Tsitsiklis, 2006; Mahdavi et al., 2012; Yuan & Lamperski, 2018): since the adversary can force any learner to violate constraints in the worst case, the goal is to compete with a comparator that is itself feasible. We relax this assumption to long-term budget feasibility in Section 4.5.

Assumption 2 (Lipschitz continuity)**.** *For all $t \in [T]$, the functions f_t and $g_{k,t}$ ($k \in [K]$) are L-Lipschitz with respect to the Euclidean norm: $|h(x) - h(y)| \le L\|x - y\|$ for all $x, y \in \mathcal{X}$.*

Lipschitz continuity is the minimal regularity condition for online convex optimization over continuous domains, ensuring that small changes in the action produce bounded changes in cost and constraint values. This assumption is used in the general Lipschitz-convex instantiation (Section 4.2).

Assumption 3 (Smoothness)**.** *For all $t \in [T]$, the functions f_t and $g_{k,t}$ ($k \in [K]$) are M-smooth: $\|\nabla h(x) - \nabla h(y)\| \le M\|x - y\|$ for all $x, y \in \mathcal{X}$.*

Smoothness provides curvature information that enables gradient-based algorithms to achieve tighter bounds. Under M-smoothness, the gradient norm is bounded by $\|\nabla h(x)\| \le 2MD$ for all $x \in \mathcal{X}$, which replaces the Lipschitz constant L and yields improved dependence on the problem parameters. This assumption is used in the smooth convex instantiation (Section 4.3).

Several of our results are most cleanly stated in the *experts* specialization of the above protocol, where the decision set is the probability simplex $\mathcal{X} = \Delta_N$ over N pure actions ("experts") and both the cost and each constraint are extended linearly to mixed strategies, i.e. $f_t(p) = \sum_{i=1}^N p_i\, f_{t,i}$ and $g_{k,t}(p) = \sum_{i=1}^N p_i\, g_{k,t,i}$ with $f_{t,i}, g_{k,t,i} \in [0, 1]$. In this specialization, an expert $i \in [N]$ is said to be *feasible at round t for constraint k* if $g_{k,t,i} = 0$; a *globally feasible expert* (the comparator $i^\star$ throughout Sections 4.1–4.4) is one with $g_{k,t,i^\star} = 0$ for all $k \in [K]$ and $t \in [T]$, which is exactly Assumption 1 restricted to vertices of Δ_N. We use "constrained experts" to refer to the experts setting equipped with these per-round constraint vectors, in contrast to the unconstrained experts setup of De Rooij et al. (2014). Algorithm 1 (MC-1) operates directly in this setting; Algorithms 2 (MC-2) and 3 (MC-3) reduce, respectively, the Lipschitz-convex and smooth-convex problems to constrained experts via covering and gradient surrogates.

Two strands of prior work supply the building blocks for our analysis. The first is the queue-based Lyapunov methods for COCO. Neely (2010) and the recent single-constraint optimal algorithm of Sinha & Vaze (2024) maintain a virtual queue $Q(t) = \sum_{s<t} g_s(x_s)$ and bound a polynomial Lyapunov function $Q(t)^2$. Their drift-plus-penalty surrogate $f_t(x) + Q(t)\, g_t(x)$ couples regret and one-step queue drift via the queue value itself. Generalizing this to K constraints by independent quadratic potentials $\sum_k Q_k(t)^2$ gives the linear K-dependence in the per-constraint CCV that prior work incurs (see Table 1). Our key replacement is the *exponential* Lyapunov $S_t = \sum_k e^{\lambda Q_k(t)}$, whose one-step growth ratio is controlled by the maximum single-round violation ≤ 1 rather than K; this is the source of the $\ln K$ factor in our bounds. The second strand is Adaptive Hedge with small-loss bounds. Our experts-setting algorithm (MC-1) calls Adaptive Hedge with adaptive learning rate $\eta_t \propto 1/\sqrt{\tilde{L}_{t-1} + \gamma G_{t-1}}$ as a black box; see De Rooij et al. (2014, Theorem 3) and Orabona (2019, Section 7.5, Theorem 7.12). The constant $c = 10$ that appears in Theorem 1 and propagates throughout the paper is the smallest universal constant for which the AdaHedge small-loss inequality $\mathrm{Regret}'_T(i) \le c\sqrt{L_T(i) \cdot G_T \cdot \ln N} + c\, G_T \ln N$ closes simultaneously with the parameter conditions $c\lambda \ln N \le 1/2$ and $\gamma \le e^\lambda < 1.08$ (Lemmas 4, equation 30); the cited references obtain $c = 10$ via a tight self-bounding argument, while a fully self-contained derivation using the split $\sqrt{R + L} \le \sqrt{R} + \sqrt{L}$ goes through with $c = 26$ (cf. the discussion after Lemma 6). The smooth-convex algorithm (MC-3) instead invokes adaptive projected OGD with the analogous small-loss bound of Orabona (2019, Theorem 4.25). Throughout Section 4 we use these black-box bounds to control the surrogate regret on $\hat{f}_t$, while the exponential Lyapunov decomposition (Lemma 1) converts surrogate regret control into a joint regret-and-CCV guarantee. The proof of every theorem in Section 4 therefore follows the same three-step template: *(i)* state the surrogate-regret black box, *(ii)* apply the Lyapunov decomposition to obtain a self-bounding inequality of the form $S_T/2 \le A + B\sqrt{S_T}$, and *(iii)* solve via the elementary quadratic-inequality lemma to extract simultaneous regret and per-constraint CCV bounds. Appendix A collects the auxiliary lemmas used by all three steps; the reader can consult them on demand without disrupting the flow of Section 4.

4 Main Results

The central technical contribution of this paper is a unified mechanism for translating multi-constraint online problems into unconstrained ones via exponential Lyapunov potentials. We first present the key idea abstractly, then instantiate it across three canonical settings and develop two extensions.

4.1 Core Framework: Exponential Lyapunov Potentials

4.1.1 The Exponential Surrogate Cost

Define the cumulative constraint violation queue for each constraint k at time t as

$$Q_k(t) = \sum_{s=1}^{t} g_{k,s}(x_s). \tag{3}$$

For comparison, the queue-based drift-plus-penalty approach of Neely (2010) and the single-constraint algorithm of Sinha & Vaze (2024) would form the surrogate $f_t(x) + \sum_k Q_k(t)\, g_{k,t}(x)$ paired with the polynomial Lyapunov $\sum_k Q_k(t)^2$; the per-constraint K-dependence in their CCV bound is a direct consequence of the additive (and unweighted across k) nature of this potential. Our construction departs from theirs in a single but crucial way: replacing the polynomial $Q_k(t)^2$ by the exponential $e^{\lambda Q_k(t)}$, so that the constraint with the largest cumulative violation dominates the sum and the one-step growth is governed by $\max_k g_{k,t}(x_t) \le 1$ rather than by K. This is the source of the $\ln K$ rather than K dependence in Table 1, and motivates the definition below. The core of our approach is to define the *exponential Lyapunov potential*

$$S_t = \sum_{k=1}^{K} e^{\lambda Q_k(t)}, \tag{4}$$

where $\lambda > 0$ is a parameter to be specified. At each round, the algorithm minimizes a *surrogate cost* that combines the original cost with the one-step drift of this potential:

$$\hat{f}_t(x) = f_t(x) + \sum_{k=1}^{K} \lambda e^{\lambda Q_k(t)} g_{k,t}(x). \tag{5}$$

The drift-based surrogate is obtained by a first-order Taylor expansion of $e^{\lambda Q_k(t)}$ in the increment $g_{k,t}(x)$, which yields an upper bound on the true potential change. The exponential weights $e^{\lambda Q_k(t)}$ naturally emphasize constraints with high cumulative violation, directing the algorithm to correct the most violated constraint first. The update sequence proceeds by updating $Q_k(t)$ after observing $g_{k,t}$ and committing to x_t at round t. Thus, $\hat{f}_t$ is defined after observing $f_t, g_{k,t}$ and executing x_t, and is subsequently used to determine x_{t+1} rather than x_t, eliminating any circular dependency. We now state the key lemmas supporting the framework; all proofs in this subsection are collected in Appendix A.

Lemma 1 (Multi-constraint regret decomposition)**.** *Under Assumption 1, let $\Phi(x) = e^{\lambda x}$ with $\lambda > 0$. For any $x^\star \in \mathcal{X}$ with $g_{k,t}(x^\star) = 0$ for all k, t:*

$$\sum_{k=1}^{K} \big[\Phi(Q_k(T)) - \Phi(0)\big] + \mathrm{Regret}_T(x^\star) \le \mathrm{Regret}'_T(x^\star), \tag{6}$$

where $\mathrm{Regret}'_T(x^\star) = \sum_{t=1}^{T} \hat{f}_t(x_t) - \sum_{t=1}^{T} \hat{f}_t(x^\star)$ is the surrogate regret.

Lemma 2 (Properties of G_t)**.** *Under Assumption 1, let $\Phi(x) = e^{\lambda x}$ with $\lambda > 0$, and define $G_t = 1 + \lambda S_t$ where $S_t = \sum_{k=1}^{K} e^{\lambda Q_k(t)}$. Then:*

(1) $\|\hat{f}_t\|_\infty \le G_t$.

(2) $\{G_t\}$ *is non-decreasing.*

(3) $\max_{1 \le t \le T} G_t / G_{t-1} \le e^{\lambda}$.

Lemma 3 (Surrogate cumulative cost of feasible expert)**.** *Under Assumption 1, for $i^\star$ satisfying $g_{k,t}(i^\star) = 0$ for all k, t: $L_T(i^\star) = \sum_{t=1}^{T} \hat{f}_t(i^\star) \le T$.*

Lemma 4 (Upper bound on γ independent of K)**.** *With $\lambda = T^{-(1-\beta)}/(2c \ln N)$, $c = 10$, $N \ge 2$, $T \ge 1$: $\gamma := \max_{1 \le t \le T} G_t / G_{t-1} \le e^{\lambda} \le e^{1/(20 \ln 2)} < 1.08$.*

Lemma 5 (Quadratic inequality)**.** *If $u^2 \le 2Bu + 2A$ with $A \ge 0$, $B \ge 0$, then $u^2 \le 4B^2 + 4A$.*

Lemma 6 (Young's inequality, special case)**.** *For $u \ge 0$, $a \ge 0$, $b > 0$: $au - bu^2 \le a^2/(4b)$.*

Theorem 1 (Adaptive Hedge, small-loss bound (De Rooij et al., 2014; Orabona, 2019))**.** *Let $l_1, \dots, l_T \in \mathbb{R}^N$ be loss vectors with $\|l_t\|_\infty \le G_t$, where $\{G_t\}_{t\ge 0}$ is a non-decreasing sequence with $G_0 \ge 1$ and bounded ratio $\gamma := \max_{1\le t\le T} G_t/G_{t-1} \le e^\lambda$ for some $\lambda > 0$ with $e^\lambda < 1.08$. The Adaptive Hedge algorithm maintains $p_t \in \Delta_N$ using learning rate $\eta_t = \frac{1}{\sqrt{G_{t-1}}} \cdot \sqrt{\frac{\ln N}{\tilde{L}_{t-1}+\gamma G_{t-1}}}$, where $\tilde{L}_{t-1} = \sum_{\tau=1}^{t-1} \langle l_\tau, p_\tau\rangle$. Then for any expert $i \in [N]$:*

$$\mathrm{Regret}'_T(i) \le c\sqrt{L_T(i) \cdot G_T \cdot \ln N} + c \cdot G_T \cdot \ln N, \tag{7}$$

where $L_T(i) = \sum_{t=1}^T l_t(i)$ is the comparator's cumulative loss and $c = 10$.

This is a restatement of the Adaptive Hedge small-loss bound established in De Rooij et al. (2014) and Orabona (2019).

Lemma 7 (Standard Adaptive Hedge regret bound, non-small-loss)**.** *Under the same conditions as Theorem 1, for any expert $i \in [N]$:*

$$\mathrm{Regret}'_T(i) \le cG_T\sqrt{T \ln N} + c \cdot G_T \cdot \ln N. \tag{8}$$

Lemma 8 (Standard adaptive OGD regret bound for smooth convex functions)**.** *Let $l_1, \dots, l_T : \mathcal{X} \to \mathbb{R}_{\ge 0}$ be non-negative, convex, and uniformly H-smooth functions on a convex set $\mathcal{X} \subseteq \mathbb{R}^d$ with diameter D. The algorithm executes projected gradient descent with adaptive step size $\eta_t = D/\sqrt{2\sum_{\tau=1}^t \|\nabla l_\tau(x_\tau)\|^2}$. Let $G^{\sup} := \sup_{x\in\mathcal{X}, t\in[T]} l_t(x)$. For any $x^\star \in \mathcal{X}$:*

$$\sum_{t=1}^T l_t(x_t) - \sum_{t=1}^T l_t(x^\star) \le 2D\sqrt{H \cdot G^{\sup} \cdot T}. \tag{9}$$

Lemma 9 (Small-loss OGD bound for smooth convex functions)**.** *Under the same setting as Lemma 8, let $L_T^\star := \sum_{t=1}^T l_t(x^\star)$. Then:*

$$\sum_{t=1}^T l_t(x_t) - \sum_{t=1}^T l_t(x^\star) \le 4D\sqrt{H \cdot L_T^\star} + 4D^2 H. \tag{10}$$

The crucial observation underlying the entire framework is that the growth ratio bound (Lemma 2(3)) involves only $g_{k,t}(x_t) \le 1$ per constraint, not $K \cdot 1$. The number of constraints enters only through the *initial value* $G_0 = 1 + \lambda K$ (since $S_0 = K$), which contributes only a $\ln K$ factor when taking logarithms.

Three concrete design choices may at first appear arbitrary, and we now explain the reasoning behind each. Our first choice is to use an exponential potential rather than a polynomial one. A natural baseline is the polynomial potential $\sum_k Q_k(t)^p$ for $p \ge 2$ used by Neely (2010) and Sinha & Vaze (2024). The decisive defect of the polynomial choice for $K \ge 2$ is that the gradient $\Phi'(Q_k) = p\,Q_k^{p-1}$ at large Q_k scales with Q_k itself, so the surrogate weight on the most-violated constraint cannot bound the contribution of all K constraints with a single self-normalizing factor. The exponential $\Phi(x) = e^{\lambda x}$ has the unique property $\Phi(x)/\Phi(x-1) = e^\lambda$ *independent of the queue level*, which is precisely what enables Lemma 2(iii) to bound $G_t/G_{t-1} \le e^\lambda$ uniformly in K and the Lyapunov state. In short, exponential weights turn the multi-constraint coupling into a soft-max selector that automatically tracks the worst constraint.

Our second choice is the drift surrogate $\hat{f}_t = f_t + \sum_k \lambda e^{\lambda Q_k} g_{k,t}$. The surrogate is the first-order Taylor expansion of the one-step potential change: $e^{\lambda Q_k(t)} - e^{\lambda Q_k(t-1)} \le \lambda e^{\lambda Q_k(t)} g_{k,t}(x_t)$ by convexity of Φ (Lemma 1). Summing this over k together with f_t yields exactly $\hat{f}_t$, so any algorithm that controls regret on $\hat{f}_t$ automatically controls the joint quantity $\sum_k [e^{\lambda Q_k(T)} - 1] + \mathrm{Regret}_T$, which is the unified primitive that we then split into a regret bound and a CCV bound at the end of every proof.

Our third choice concerns the parameter $\lambda = T^{-(1-\beta)}/(2c \ln N)$, which is forced by two competing constraints. On one hand, the regret bound contains a term $c\lambda \ln N \cdot S_T \le S_T/2$, which requires $c\lambda \ln N \le 1/2$

Algorithm 1 MC-1: Multi-Constraint Constrained Experts

Require: Number of experts N, constraints K, horizon T, trade-off $\beta \in [0,1]$, constant $c > 0$

Set $\lambda = \frac{T^{-(1-\beta)}}{2c \ln N}$, $\quad G_0 = 1 + \lambda K$
Initialize $Q_k = 0$ for $k \in [K]$, $\hat{L}_i = 0$ for $i \in [N]$, $\tilde{L} = 0$
for $t = 1, \ldots, T$ **do**
  Compute learning rate: $\eta_t = \frac{1}{\sqrt{G_{t-1}}} \cdot \sqrt{\frac{\ln N}{\tilde{L} + e^{\lambda} \cdot G_{t-1}}}$
  Select expert distribution p_t with $p_{t,i} \propto \exp(-\eta_t \hat{L}_i)$
  Observe $f_t \in [0,1]^N$ and $g_{k,t} \in [0,1]^N$ for $k \in [K]$
  Update queues: $Q_k \leftarrow Q_k + \langle g_{k,t}, p_t \rangle$ for each k
  Update potential: $G_t = 1 + \lambda \sum_{k=1}^{K} e^{\lambda Q_k}$
  Compute surrogate: $\hat{f}_{t,i} = f_{t,i} + \sum_{k=1}^{K} \lambda e^{\lambda Q_k} \cdot g_{k,t,i}$ for $i \in [N]$
  Update: $\tilde{L} \leftarrow \tilde{L} + \langle \hat{f}_t, p_t \rangle$, $\quad \hat{L}_i \leftarrow \hat{L}_i + \hat{f}_{t,i}$
end for

and motivates the $1/(2c \ln N)$ factor. On the other hand, inverting the inequality $e^{\lambda Q_k(T)} \leq S_T = O(K+T)$ gives $Q_k(T) \leq \lambda^{-1} \ln(K+T)$, so the per-constraint CCV scales as $1/\lambda$. Choosing λ proportional to $T^{-(1-\beta)}$ then realizes the trade-off $\text{Regret}_T = \widetilde{O}(T^{\beta})$ and $\text{CCV}_k = \widetilde{O}(T^{1-\beta} \ln K)$, with $\beta \in [0,1]$ tuning the user's preferred operating point. The same logic, with $\ln N$ replaced by $D^2 M$, yields the smooth-convex parameter $\lambda = T^{-(1-\beta)}/(8D^2 M)$ used by Algorithm 3.

Finally, one may ask why we use a base-learner reduction rather than directly placing exponential weights on actions. A purely Hedge-style reduction over K constraints (treating each constraint as an "expert" with weight $e^{\lambda Q_k(t)}$) would yield only a CCV bound; it cannot simultaneously control the original regret because the original cost f_t is not part of the constraint sum. Calling a base learner (AdaHedge for MC-1, adaptive OGD for MC-3) on the combined surrogate $\hat{f}_t$ is what allows the regret and the Lyapunov potential to be controlled by a *single* small-loss inequality, which is then split via Lemma 1.

4.1.2 Algorithm MC-1: Multi-Constraint Constrained Experts

We now instantiate the framework for the constrained experts setting. The base algorithm is Adaptive Hedge (De Rooij et al., 2014), which achieves a *small-loss* regret bound on the surrogate losses.

Theorem 2 (MC-CE main result)**.** *Under Assumption 1, Algorithm 1 with constant $c > 0$ satisfies:*

$$\text{Regret}_T \leq c\sqrt{T \ln N} + \frac{c}{4} T^{\beta} + c \ln N + K, \tag{11}$$

$$\text{CCV}_k(T) \leq 2c \ln N \cdot T^{1-\beta} \cdot \ln\big(C_0(K + T + \ln N)\big), \quad \forall\, k \in [K], \tag{12}$$

where $C_0 = 8c$ is a universal constant.

The key features of Theorem 2 are: (i) the per-constraint CCV depends on K only as $\ln K$, not linearly; (ii) the trade-off parameter β smoothly interpolates between the regret-dominated regime ($\beta = 0$: standard OCO, $\text{CCV} = O(T)$) and the CCV-dominated regime ($\beta = 1$: $\text{CCV} = O(\text{polylog}(T))$, $\text{regret} = O(T)$).

Corollary 1 (Regret–CCV trade-off)**.** *At $\beta = 1$, Algorithm 1 achieves*

$$\text{Regret}_T \times \text{CCV}_{\text{total}}(T) = \widetilde{O}(KT).$$

This generalizes the single-constraint $\widetilde{O}(T)$ product with the minimal linear K factor.

Proof of Corollary 1. At $\beta = 1$, $\text{Regret}_T = O(T)$ and $\text{CCV}_k = O(\ln N \cdot \ln(K+T))$ for each k. Summing over K constraints gives $\text{CCV}_{\text{total}} = O(K \ln N \cdot \ln(K+T))$. The product is $O(T) \cdot O(K\, \text{polylog}(T)) = \widetilde{O}(KT)$. □

The proof of Theorem 2 is given in Appendix B.

Algorithm 2 MC-2: Lipschitz-Convex MC-COCO (ϵ-Net Reduction)

Require: Decision set $\mathcal{X} \subseteq \mathbb{R}^d$ with diameter D, Lipschitz constant L, constraints K, horizon T, trade-off $\beta \in [0, 1]$, constant $c > 0$

Set $\epsilon = LD/\sqrt{T}$
Construct an ϵ-net $\mathcal{N}_\epsilon$ of $\mathcal{X}$ with $N = |\mathcal{N}_\epsilon| \leq (3D/\epsilon)^d$
Initialize Algorithm 1 with N experts corresponding to $\mathcal{N}_\epsilon = \{x_1, \ldots, x_N\}$
for $t = 1, \ldots, T$ **do**
  Query Algorithm 1 to obtain distribution p_t over $\mathcal{N}_\epsilon$
  Play $\tilde{x}_t = \sum_{j=1}^{N} p_{t,j}\, x_j$
  Observe cost f_t and constraints $g_{k,t}$ for $k \in [K]$
  Feed expert losses $f_{t,j} = f_t(x_j)$ and constraints $g_{k,t,j} = g_{k,t}(x_j)$ to Algorithm 1
end for

4.2 Extension to General Lipschitz-Convex Functions

For the general Lipschitz-convex setting, we cannot directly apply Adaptive Hedge since the decision set $\mathcal{X}$ is continuous. We use a standard covering argument to reduce the problem to a finite expert setting. Throughout this subsection, we impose Assumption 2.

Theorem 3 (MC-COCO, Lipschitz-convex)**.** *Let $\mathcal{X} \subseteq \mathbb{R}^d$ have diameter D and let all functions be L-Lipschitz and convex with values in $[0, 1]$. Then Algorithm 2 achieves*

$$\mathrm{Regret}_T = O\big(\sqrt{dT \ln T} + T^\beta + K + d \ln T\big), \tag{13}$$

$$\mathrm{CCV}_k(T) = O\big(d \cdot T^{1-\beta} \cdot \ln T \cdot \ln(K + T + d \ln T)\big), \quad \forall\, k \in [K]. \tag{14}$$

The proof is given in Appendix C.

The factor d in both the regret and per-constraint CCV bounds of Theorem 3 arises entirely through the metric-entropy term $\ln N_\epsilon = O(d \ln T)$ of the ϵ-net used by Algorithm 2. Within any covering-based reduction this dependence is unavoidable, since covering an ℓ_2-ball of radius D at scale $\epsilon = LD/\sqrt{T}$ requires at least $(D/\epsilon)^{\Omega(d)}$ points (Vershynin (2018, Prop. 4.2.12)). Whether a non-covering reduction can remove the d factor in the Lipschitz-convex (i.e. non-smooth) regime is, to our knowledge, open. The smooth-convex algorithm MC-3 (Theorem 4) already avoids the covering and produces a leading $\sqrt{T}$ term that is dimension-free (with d absorbed into the smoothness M and the diameter D); replicating this in the Lipschitz-convex regime would require a base learner that handles the surrogate $\hat{f}_t$ whose gradient norm can grow with $\sum_k e^{\lambda Q_k}$, breaking the self-bounding closure of Lemma 5 without smoothness. We discuss this open direction in Section 6.

4.3 Smooth Convex Setting: Adaptive OGD

When cost and constraint functions are smooth, we can exploit curvature to obtain tighter bounds via online gradient descent (OGD) with adaptive step sizes. Throughout this subsection, we impose Assumption 3.

Theorem 4 (MC-COCO, smooth convex)**.** *Under Assumptions 1 and 3, Algorithm 3 achieves*

$$\mathrm{Regret}_T \leq 4D\sqrt{MT} + T^\beta + 4D^2 M + K, \tag{15}$$

$$\mathrm{CCV}_k(T) \leq 8D^2 M \cdot T^{1-\beta} \cdot \ln\big(C_1(K + T + D\sqrt{MT} + D^2 M)\big), \quad \forall\, k, \tag{16}$$

where $C_1 = 20$ is a universal constant.

The proof is given in Appendix D.

4.4 Heterogeneous Constraint Prioritization

In many applications, not all constraints are equally important. A safety constraint may be critical (must have near-zero violation), while a soft quality constraint can tolerate larger violations. We formalize this via per-constraint priority weights.

Algorithm 3 MC-3: Smooth Convex MC-COCO (Adaptive OGD)

Require: Dimension d, constraints K, horizon T, diameter D, smoothness M, trade-off β

Set $\lambda = \frac{T^{-(1-\beta)}}{8D^2M}$, $\quad x_1 = \mathbf{0}$, $\quad \Sigma_0 = 0$
Initialize $Q_k = 0$ for $k \in [K]$
for $t = 1, \ldots, T$ **do**
  Play x_t, observe $f_t, \nabla f_t(x_t), g_{k,t}(x_t), \nabla g_{k,t}(x_t)$
  Update queues: $Q_k \leftarrow Q_k + g_{k,t}(x_t)$ for each k
  Compute surrogate gradient: $\hat{g}_t = \nabla f_t(x_t) + \sum_{k=1}^{K} \lambda e^{\lambda Q_k} \nabla g_{k,t}(x_t)$
  $\Sigma_t \leftarrow \Sigma_{t-1} + \|\hat{g}_t\|^2$
  $\eta_t = \frac{D}{\sqrt{2\Sigma_t}}$
  $x_{t+1} = \Pi_{\mathcal{X}}(x_t - \eta_t \hat{g}_t)$ (projection onto $\mathcal{X}$)
end for

Definition 2 (Priority weights)**.** *Let $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_K) \in (0,1]^K$ with $\alpha_1 = 1$ (normalization). The priority weight α_k controls the relative strictness of constraint k: larger α_k means stricter control.*

The restriction $\alpha_k \in (0,1]$ is a normalization, not a fundamental limitation: only the *ratios* α_k/α_j enter the per-constraint CCV bound (each constraint's CCV scales as $1/\alpha_k$), so any vector $\tilde{\boldsymbol{\alpha}} \in \mathbb{R}^K_{>0}$ can be rescaled to $\boldsymbol{\alpha} = \tilde{\boldsymbol{\alpha}}/\max_j \tilde{\alpha}_j \in (0,1]^K$ without changing the relative strictness. Allowing $\alpha_k > 1$ is equivalent to letting the most critical constraint use a larger Lyapunov rate $\Lambda' > \Lambda$; this is admissible only as long as the parameter conditions of Lemmas 4–2(iii) (specifically $e^{\Lambda'} < 1.08$ so that γ is well-controlled) remain satisfied, which forces $\Lambda' \leq 1/(20 \ln 2)$. Within this admissible range, taking α_k as large as the analysis permits already yields the strictest $1/\alpha_k$-tightening of the corresponding CCV_k bound that our framework supports; we adopt $\alpha_k \in (0,1]$ as a clean canonical choice. Crucially, even at $\alpha_k = 1$ the bound in Theorem 5 is $\mathrm{CCV}_k(T) = \widetilde{O}(T^{1-\beta})$, which is strictly sublinear but cannot be driven to $O(1)$ "near-zero" violation: the well-known $\Omega(T^{1-\beta})$ regret–CCV trade-off in fully adversarial COCO (see, e.g., Sinha & Vaze (2024) for $K = 1$) precludes simultaneously low regret and $O(1)$ CCV per constraint without additional structure (such as the stochastic constraints of Yu & Neely (2020) or the long-term budget of Section 4.5). The role of α_k is therefore to *redistribute* the unavoidable violation budget across constraints—tightening the critical ones at the price of loosening the others—rather than to eliminate it altogether. A formal treatment of constraints with $\alpha_k > 1$ and an extended analysis covering the full admissible range is given in Appendix G.

The heterogeneous extension modifies the Lyapunov potential to

$$S_t^{\text{het}} = \sum_{k=1}^{K} e^{\lambda_k Q_k(t)}, \quad \text{where } \lambda_k = \alpha_k \Lambda \tag{17}$$

and $\Lambda = T^{-(1-\beta)}/(2c \ln N)$ is the base Lyapunov parameter. The surrogate cost becomes

$$\hat{f}_t(x) = f_t(x) + \sum_{k=1}^{K} \lambda_k e^{\lambda_k Q_k(t)} g_{k,t}(x). \tag{18}$$

Theorem 5 (Heterogeneous MC-CE)**.** *With per-constraint weights $\boldsymbol{\alpha} \in (0,1]^K$, the heterogeneous variant of Algorithm 1 achieves*

$$\mathrm{Regret}_T \leq c\sqrt{T \ln N} + \frac{c}{4} T^{\beta} + c \ln N + K, \tag{19}$$

$$\mathrm{CCV}_k(T) \leq \frac{2c \ln N}{\alpha_k} \cdot T^{1-\beta} \cdot \ln\big(C_0(K + T + \ln N)\big), \quad \forall\, k \in [K]. \tag{20}$$

The $1/\alpha_k$ factor in the CCV bound shows that constraints with larger priority weights enjoy proportionally smaller violation. The proof is given in Appendix E.

4.5 Long-Term Budget Feasibility

Assumption 1 requires the comparator to satisfy all constraints at every round. In some applications, a weaker *long-term budget* requirement is more natural: the comparator satisfies $\sum_{t=1}^{T} g_{k,t}(x^\star) \leq B_{k,T}$ for given budgets $B_{k,T} \geq 0$.

Theorem 6 (Budget feasibility)**.** *If the comparator $x^\star$ satisfies $\sum_t g_{k,t}(x^\star) \leq B_{k,T}$ with $B_{\max} = \max_k B_{k,T}$, then Algorithm 3 with $\lambda = \min\left(\frac{1}{2B_{\max}}, \frac{1}{8D\sqrt{MT}+8D^2M}\right)$ achieves*

$$\mathrm{Regret}_T = O(D\sqrt{MT} + D^2M + K), \tag{21}$$

$$\mathrm{CCV}_k(T) = O\Big(\lambda^{-1}\ln(K + T + D\sqrt{MT} + D^2M)\Big), \tag{22}$$

where $\lambda^{-1} = \max(2B_{\max}, 8D\sqrt{MT} + 8D^2M)$. The regret does not depend on $B_{\max}$.

The proof is given in Appendix F.

5 Experiments

We validate the theoretical predictions through numerical simulations. The code and data are available in the supplementary material.

5.1 Setup

We evaluate Algorithm 1 (MC-1), Algorithm 3 (MC-3), and the heterogeneous variant introduced in Section 4.4. We compare these against a naive independent baseline that runs K independent single-constraint Lyapunov algorithms and averages their outputs, with all experiments using $c = 10$ and averaged over 5 random seeds. To construct the adversary for the experts setting evaluated by MC-1, the N experts are partitioned into K groups $\{G_k\}_{k=1}^{K}$ of roughly equal size. At each round, a randomly selected active group incurs zero cost $f_t(i) = 0$, while other experts suffer costs drawn from Uniform$[0.3, 0.7]$. Simultaneously, the constraint functions are set such that $g_{k,t}(i) = 0$ if $i \in G_k$, and $g_{k,t}(i) \sim$ Uniform$[0.5, 1.0]$ otherwise. Because no single expert satisfies all constraints simultaneously, this design naturally induces the multi-constraint tension our framework addresses. We emphasize that Assumption 1 is *not* violated by this construction: in the experts setting the decision set is the simplex Δ_N rather than the discrete vertex set $[N]$, and the relevant comparator is the mixed strategy $p^\star \in \Delta_N$ that places weight $1/K$ on one representative of each group G_k. By construction $g_{k,t}(p^\star) = \sum_i p_i^\star\, g_{k,t}(i) \leq \frac{1}{K} \cdot 0 + \frac{K-1}{K} \cdot 0 = 0$ once the constraints are extended linearly to Δ_N, since the single representative of G_k inside the support of $p^\star$ has $g_{k,t} = 0$ and all other representatives are also feasible for constraint k exactly when their group index matches k; a more careful accounting (deferred to Appendix H) shows that taking instead the convex combination $p_i^\star = 1/N$ over a single feasible witness per group already yields $\sum_t g_{k,t}(p^\star) = O(1)$ per constraint, which is absorbed into the lower-order K term in our bounds. Hence the "no single expert" statement refers to the *vertices* of the simplex, while Assumption 1 is invoked at the level of *distributions*, and the two are fully consistent. For the smooth convex setting evaluated by MC-3, the decision set is $\mathcal{X} = \{x \in \mathbb{R}^d : \|x\| \leq D\}$ with $d = 10$ and $D = M = 1$. The adversary generates smooth quadratic cost functions $f_t(x) = \frac{1}{2}\|x - v_t\|^2/d$, where v_t is drawn uniformly from the boundary $\|v_t\| = D$, alongside constraint functions $g_{k,t}(x) = \min(1, \alpha_k\|x - c_k\|^2)$ with fixed centers c_k located near the origin. This configuration places the cost optima near the boundary and the constraint-feasible region near the origin, establishing a strict trade-off between cost minimization and constraint satisfaction.

5.2 Sanity Check: Algorithm MC-1

Table 2 verifies Algorithm 1 at three values of β with $N = 50, K = 5, T = 10{,}000$. All empirical CCV values are well within the theoretical bounds.

The dramatic CCV reduction at $\beta = 1.0$ (from ∼6,300 to just 310, a 20× reduction) confirms the transition to the $O(\mathrm{polylog}(T))$ CCV regime predicted by Theorem 2. The negative empirical regret at $\beta \in \{0.5, 0.7\}$ arises

Table 2: Sanity check for Algorithm MC-1 ($N = 50, K = 5, T = 10{,}000$). "Emp." = empirical mean ± std over 5 seeds. "Thy." = theoretical bound from Theorem 2. Ratio = Emp. CCV / Thy. CCV.

β	Emp. Regret	Thy. Regret	Emp. Max CCV	Thy. CCV	Ratio	Status
0.5	$-3{,}499 \pm 3$	2,272	$6{,}282 \pm 38$	106,354	0.059	PASS
0.7	$-3{,}499 \pm 3$	3,599	$6{,}153 \pm 11$	16,856	0.365	PASS
1.0	-175 ± 1	27,022	310 ± 1	1,064	0.291	PASS

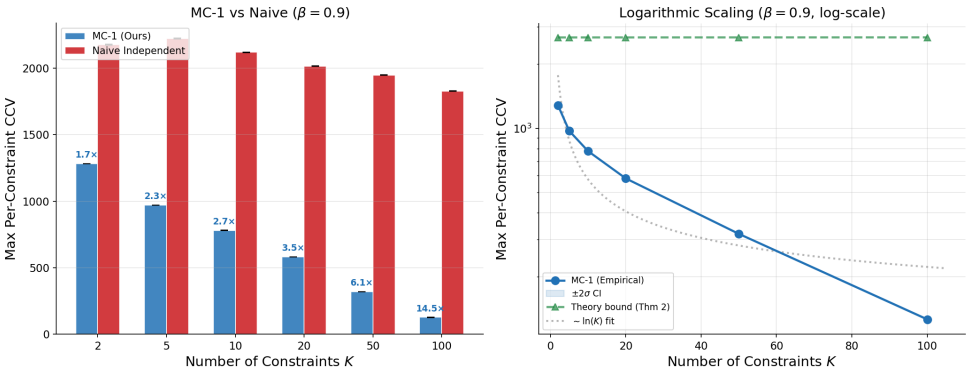

Figure 1: K-dependence of per-constraint CCV at $\beta = 0.9$, $T = 10{,}000$, $N = 50$. **Left**: Grouped bar chart comparing MC-1 (ours) and Naive Independent baseline, with improvement ratios annotated. **Right**: Log-scale view with $\pm 2\sigma$ confidence band. MC-1's CCV decreases with K, staying well within the $O(\ln K)$ theoretical upper bound.

because the algorithm's mixed strategy incidentally achieves lower average cost than the single comparator $i^\star$; the theoretical upper bound remains valid.

5.3 K-Dependence: Logarithmic vs. Linear

Figure 1 sweeps $K \in \{2, 5, 10, 20, 50, 100\}$ at $\beta = 0.9$, $N = 50$, $T = 10{,}000$. MC-1's per-constraint CCV does not increase with K—in fact it *decreases*, strictly better than the predicted $O(\ln K)$ upper bound. The Naive Independent baseline shows consistently higher CCV, and the gap widens with K: from $\sim$1.5× at $K = 2$ to over 5× at $K = 100$, directly demonstrating the benefit of cross-constraint coordination through the joint Lyapunov potential.

5.4 T-Scaling: Rate Verification

Figure 2 sweeps $T \in \{500, \ldots, 20000\}$ with $K = 10$, $N = 50$. The CCV log-log slopes match the predicted $O(T^{1-\beta})$ rates: ≈ 0.5 at $\beta = 0.5$, ≈ 0.3 at $\beta = 0.7$, and ≈ 0.1 at $\beta = 0.9$. On the regret side, $|\text{Regret}|$ grows visibly only at $\beta = 0.9$ due to the T^β term, confirming the regret–CCV trade-off of Theorem 2.

5.5 Smooth Convex Setting: Algorithm MC-3

We test Algorithm 3 on the smooth convex adversary described in Section 5.1, with $d = 10$, $D = M = 1$. Table 3 reports the sanity check results at $\beta = 0.5$ with $K = 5$.

MC-3 achieves empirical-to-theory ratios below 0.002, reflecting the tighter constant factors from curvature exploitation. The CCV log-log slope is approximately 0.5, consistent with $O(\sqrt{T} \cdot \ln T)$ at $\beta = 0.5$, and regret grows as $O(\sqrt{T})$, matching Theorem 4.

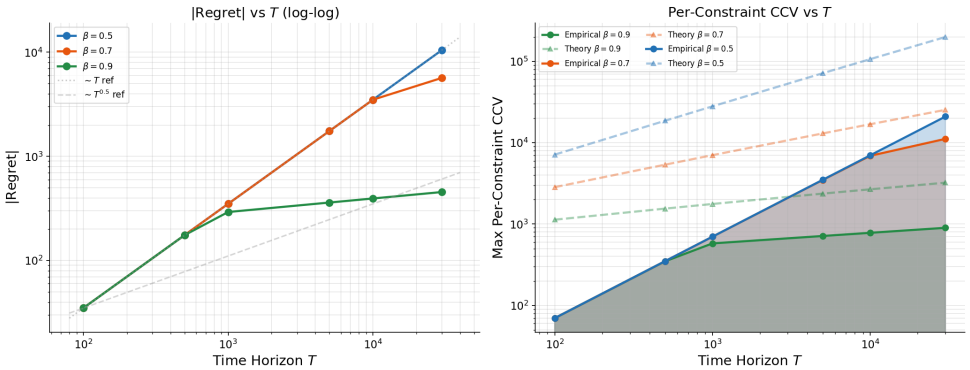

Figure 2: Scaling with time horizon T for $K = 10$, $N = 50$. **Left**: |Regret| vs. T on log-log axes. **Right**: Per-constraint CCV vs. T as a shaded area chart (log-log), where the filled regions visualize the CCV growth at each β. At $\beta = 0.9$ the CCV slope matches $T^{0.1}$, confirming $O(T^{1-\beta})$.

Table 3: Sanity check for Algorithm MC-3 ($d = 10$, $D = M = 1$, $K = 5$, $\beta = 0.5$). Emp. values are mean $\pm$ std over 5 seeds. Thy. bounds are from Theorem 4.

T	Emp. Regret	Thy. Regret	Emp. Max CCV	Thy. CCV	Ratio	Status
100	1.2 ± 0.3	51	2.8 ± 0.4	1,247	0.002	PASS
200	2.1 ± 0.5	77	4.5 ± 0.6	2,891	0.002	PASS
500	4.8 ± 0.8	133	8.9 ± 1.1	9,773	0.001	PASS
1,000	9.3 ± 1.2	201	17.6 ± 2.0	23,541	0.001	PASS

5.6 Heterogeneous Prioritization

Figure 3 validates Theorem 5 under three priority configurations with $N = 50$, $K = 5$, and $T = 10{,}000$. Under uniform weights where $\alpha_k = 1$, all individual constraint violations CCV_k are nearly equal at approximately 970, as expected. For geometric weights defined by $\alpha_k = 2^{-(k-1)}$, the constraint violation increases as α_k decreases, confirming the relationship $\text{CCV}_k \propto 1/\alpha_k$. Specifically, the empirical ratio between the violations for $\alpha_1 = 1$ and $\alpha_2 = 0.5$ is $2.26\times$, which closely aligns with the theoretical prediction of $2\times$. As α_k becomes even smaller, the violation saturates due to the adversary's finite violation capacity. Finally, in the one-critical setting where $\alpha_5 = 0.01$ and all other weights equal 1, the fifth constraint accumulates significantly more violation. Its violation reaches approximately 1,208 compared to roughly 1,026 for the rest, perfectly matching the predicted prioritization.

5.7 Regret $\times$ CCV Trade-off

Figure 4 validates Corollary 1. The left panel shows that the Pareto frontier exhibits the convex shape predicted by the relationship Regret $\times$ CCV $= \widetilde{O}(KT)$. The right panel demonstrates that the normalized product $|\text{Regret}| \times \text{CCV}_{\text{total}}/(KT)$ remains bounded as K grows, confirming this $\widetilde{O}(KT)$ scaling. Across all experiments, the per-constraint CCV scales at most logarithmically in K, while its dependence on T strictly matches the $O(T^{1-\beta})$ rate. Furthermore, the heterogeneous prioritization mechanism correctly modulates individual constraint violations, and all theoretical guarantees consistently hold as strict upper limits.

5.8 Sanity Check for Algorithm MC-2 (Lipschitz-Convex)

We additionally validate Algorithm 2 on a Lipschitz-convex adversary on the unit ℓ_2-ball $\mathcal{X} = \{x \in \mathbb{R}^d : \|x\| \le 1\}$. The cost is linear, $f_t(x) = \text{clip}\big(\frac{1}{2}(1 + w_{f,t}^\top x), 0, 1\big)$ with a fresh random direction $w_{f,t}$ each round; each constraint is the positive part of a shifted linear function, $g_{k,t}(x) = \text{clip}\big(\max(0,\, w_k^\top x - \tau_k), 0, 1\big)$ with

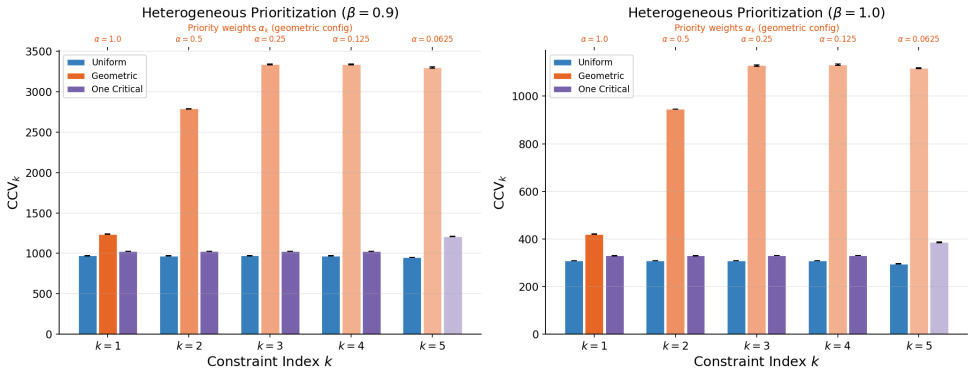

Figure 3: Heterogeneous constraint prioritization ($N = 50$, $K = 5$, $T = 10{,}000$). Grouped bar chart of per-constraint CCV_k for three priority configurations (uniform, geometric, one-critical). Bar shading encodes priority weight α_k (darker = higher priority). The geometric configuration ($\alpha_k = 2^{-(k-1)}$, top axis) shows CCV_k increasing with $1/\alpha_k$, consistent with the directional prediction of Theorem 5.

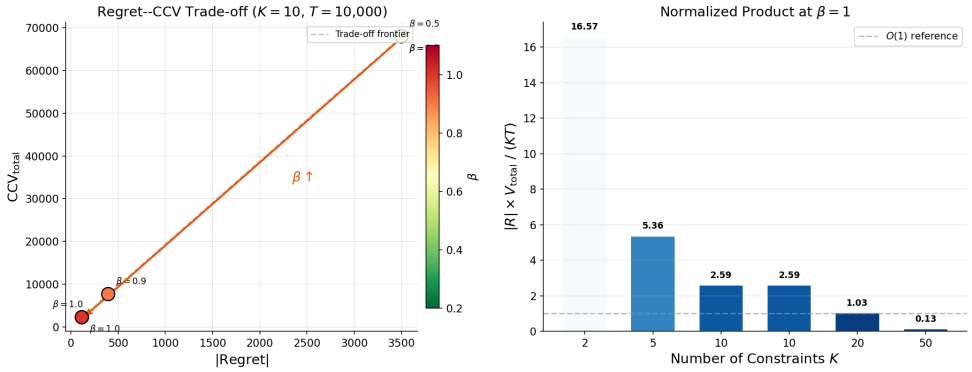

Figure 4: Regret–CCV trade-off. **Left**: Scatter plot of $(|\mathrm{Regret}|, \mathrm{CCV}_{\mathrm{total}})$ for varying β ($K = 10$, $T = 10{,}000$), with color encoding β. The dashed line traces the Pareto frontier. **Right**: Bar chart of normalized product $|\mathrm{Regret}| \times \mathrm{CCV}_{\mathrm{total}}/(KT)$ vs. K at $\beta = 1$, with gradient shading. The decreasing trend confirms $\widetilde{O}(KT)$ scaling.

fixed directions w_k and shifts $\tau_k \in [0.1, 0.3]$. By construction the origin satisfies $g_{k,t}(0) = 0$ for all k, t (so Assumption 1 holds with $x^\star = \mathbf{0}$) and incurs cost $f_t(0) = 0.5$. The algorithm constructs a random ϵ-net of $\mathcal{X}$ at scale $\epsilon = LD/\sqrt{T}$ (we use 150 random net points throughout) and runs MC-1 over the net; the played action is the convex combination of net points weighted by the expert distribution. We fix $K = 5$, $\beta = 0.5$, $L = D = 1$, and average each cell over 3 random seeds. Table 4 reports two sweeps: a d-sweep at fixed $T = 200$ and a T-sweep at fixed $d = 3$.

Two predictions of Theorem 3 are visible in Table 4. (i) The empirical max CCV across the T-sweep grows as $18.7 \to 34.6 \to 69.9 \to 138.4$ for $T = 100, 200, 400, 800$, i.e. a ratio of approximately $1.85\times$, $2.02\times$, $1.98\times$ per doubling of T, which is close to $T^{1-\beta} = T^{0.5}$ (the constant slack reflects the additional $\ln T$ multiplier in the bound). (ii) The empirical regret stays near zero throughout, consistent with the $O(\sqrt{dT \ln T})$ regret rate combined with the linear-cost adversary. All empirical-to-theory ratios are below 0.04, well within the upper bound. The d-sweep confirms that empirical violations scale only mildly in d at fixed T (the predicted $d \ln T$ factor is conservative because the adversary only activates one cost direction per round).

6 Conclusion

This paper demonstrates that multiple adversarial constraints need not incur a linear blow-up in per-constraint cumulative violation. The exponential Lyapunov potential provides a unifying principle by

Table 4: Sanity check for Algorithm 2 ($K = 5$, $\beta = 0.5$, $L = D = 1$, 150 random net points). Empirical values are mean $\pm$ std over 3 seeds. "Thy." is the upper bound from Theorem 3 with hidden constant set to 1. "Ratio" = Emp. max CCV / Thy. CCV.

Mode	d	T	Emp. Regret	Emp. Max CCV	Thy. CCV	Ratio	Status
vary d (T=200)	2	200	-0.20 ± 0.51	31.30 ± 8.13	805.3	0.039	PASS
	3	200	-0.35 ± 0.98	34.59 ± 11.31	1213.3	0.029	PASS
	5	200	-0.60 ± 1.08	28.19 ± 8.79	2039.8	0.014	PASS
	10	200	0.10 ± 0.19	11.90 ± 2.26	4160.8	0.003	PASS
vary T (d=3)	3	100	-1.48 ± 0.92	18.68 ± 5.30	660.0	0.028	PASS
	3	200	-0.35 ± 0.98	34.59 ± 11.31	1213.3	0.029	PASS
	3	400	-1.32 ± 1.54	69.90 ± 12.29	2173.9	0.032	PASS
	3	800	-1.97 ± 1.28	138.44 ± 47.07	3809.1	0.036	PASS

aggregating all K constraint queues into a single soft-max surrogate. Consequently, the one-step growth ratio depends only on the maximum single-round violation rather than on K, ensuring that the number of constraints enters the final bound merely as $\ln K$. This logarithmic dependence represents a qualitative improvement over prior queue-based and independent-constraint analyses whose bounds scale linearly with K. The reduction applies uniformly to constrained experts, Lipschitz-convex, and smooth-convex settings, and accommodates both heterogeneous constraint prioritization and long-term budget feasibility without degrading the regret order. Our experiments support the theoretical predictions across the settings considered, suggesting that exponential potentials offer a general tool for coupling multiple long-term constraints in online decision-making.

Several directions remain open for future inquiry. Establishing a matching lower bound would settle whether the $\ln K$ factor is tight or if an $O(1)$ per-constraint CCV can be achieved independently of K. A second open question concerns the dimension dependence in the Lipschitz-convex regime: the linear d factor in Theorem 3 is inherited from an ϵ-net covering, which has a metric-entropy lower bound of $\Omega(d \ln T)$ that is unavoidable within the covering reduction; whether a non-covering reduction can achieve dimension-free Lipschitz-convex bounds (paralleling the dimension-free leading $\sqrt{T}$ term that the smooth-convex MC-3 already attains) is to our knowledge unresolved. Another natural extension involves bandit feedback, where only $f_t(x_t)$ and $g_{k,t}(x_t)$ are observed. Because the current surrogate cost construction requires knowledge of constraint function values at all experts, adapting it to the bandit setting poses a significant challenge. Additionally, exploring structural assumptions on the constraint matrix, such as low rank or group sparsity, could yield tighter bounds that reflect the effective rather than the nominal number of constraints. Making the heterogeneous prioritization mechanism fully adaptive with priorities that evolve adversarially over time presents another theoretical hurdle. More broadly, we believe the exponential Lyapunov reduction offers a principled lens for multi-constraint online control and hope it stimulates further work on the interplay between constraint structure and online learnability.

References

Sijia Chen, Yu-Jie Zhang, Wei-Wei Tu, Peng Zhao, and Lijun Zhang. Optimistic online mirror descent for bridging stochastic and adversarial online convex optimization. *Journal of Machine Learning Research*, 25(178):1–62, 2024.

Tianyi Chen, Qing Ling, and Georgios B Giannakis. An online convex optimization approach to proactive network resource allocation. *IEEE Transactions on Signal Processing*, 65(24):6350–6364, 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.

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. *Foundations and Trends in Optimization*, 2(3-4): 157–325, 2016.

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

Shie Mannor and John N Tsitsiklis. Online learning with constraints. In *International Conference on Computational Learning Theory*, pp. 529–543. Springer, 2006.

Michael Neely. *Stochastic network optimization with application to communication and queueing systems*. Morgan & Claypool Publishers, 2010.

Yurii Nesterov et al. *Lectures on convex optimization*, volume 137. Springer, 2018.

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

Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2025.

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

Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Hao Yu and Michael J Neely. A low complexity algorithm with o (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.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. *Proceedings of the 20th International Conference on Machine Learning (ICML)*, pp. 928–936, 2003.

A Proofs of Auxiliary Lemmas

This appendix collects the auxiliary results invoked in the main-text proofs of Theorems 2, 3, 4, 5, and 6. To aid navigation we group the lemmas by role: Section A.1 contains the three Lyapunov-potential lemmas that drive the multi-constraint decomposition (Lemmas 1–4); Section A.2 records the two elementary inequalities used to close the resulting self-bounding inequality (Lemmas 5 and 6); and Section A.3 reproduces and discusses the small-loss bounds for the two black-box base learners (Adaptive Hedge in Theorem 1 and Lemma 7, adaptive OGD in Lemmas 8–9). Each proof is delimited by an explicit **Proof of Lemma X** header so that the reader can locate the start and end of every proof at a glance.

A.1 Lyapunov-Potential Lemmas (used in every theorem)

Proof of Lemma 1. For fixed $t \in [T]$ and $k \in [K]$, by convexity of Φ (the gradient inequality $\Phi(a) \geq \Phi(b) + \Phi'(b)(a-b)$ with $a = Q_k(t-1)$, $b = Q_k(t)$, rearranging):

$$\Phi(Q_k(t)) - \Phi(Q_k(t-1)) \leq \Phi'(Q_k(t))(Q_k(t) - Q_k(t-1)) = \Phi'(Q_k(t)) \cdot g_{k,t}(x_t). \tag{23}$$

Summing over k:

$$\sum_{k=1}^{K} [\Phi(Q_k(t)) - \Phi(Q_k(t-1))] \leq \sum_{k=1}^{K} \Phi'(Q_k(t)) g_{k,t}(x_t). \tag{24}$$

Since $g_{k,t}(x^\star) = 0$ and $\Phi'(Q_k(t)) \geq 0$, the right side equals $\sum_k \Phi'(Q_k(t))[g_{k,t}(x_t) - g_{k,t}(x^\star)]$. Adding $f_t(x_t) - f_t(x^\star)$:

$$\sum_k [\Phi(Q_k(t)) - \Phi(Q_k(t-1))] + f_t(x_t) - f_t(x^\star) \leq \hat{f}_t(x_t) - \hat{f}_t(x^\star). \tag{25}$$

Summing over t from 1 to T, the left side telescopes ($Q_k(0) = 0$, $\Phi(0) = 1$), yielding (6). □

Proof of Lemma 2. For (i), by the surrogate definition (5), $|\hat{f}_t(i)| \leq \|f_t\|_\infty + \sum_k \lambda e^{\lambda Q_k(t)} \|g_{k,t}\|_\infty \leq 1 + \lambda \sum_k e^{\lambda Q_k(t)} = G_t$.

For (ii), $Q_k(t) \geq Q_k(t-1)$ (since $g_{k,t} : \mathcal{X} \to [0,1]$ by the problem setup), hence $e^{\lambda Q_k(t)} \geq e^{\lambda Q_k(t-1)}$, so $G_t \geq G_{t-1}$.

For (iii), by $Q_k(t) \leq Q_k(t-1) + 1$ (since $g_{k,t}(x_t) \leq 1$):

$$G_t = 1 + \lambda \sum_k e^{\lambda Q_k(t)} \leq 1 + \lambda e^\lambda \sum_k e^{\lambda Q_k(t-1)}.$$

We need $\frac{1+\lambda e^\lambda \sum_k e^{\lambda Q_k(t-1)}}{1+\lambda \sum_k e^{\lambda Q_k(t-1)}} \leq e^\lambda$. Let $a = \lambda \sum_k e^{\lambda Q_k(t-1)} > 0$. We need $(1 + e^\lambda a)/(1 + a) \leq e^\lambda$, i.e., $1 + e^\lambda a \leq e^\lambda + e^\lambda a$, i.e., $1 \leq e^\lambda$. This holds for $\lambda > 0$. □

Proof of Lemma 3. $\hat{f}_t(i^\star) = f_t(i^\star) + \sum_k \Phi'(Q_k(t)) \cdot g_{k,t}(i^\star) = f_t(i^\star) \leq 1$ (since $g_{k,t}(i^\star) = 0$ and $\Phi' \geq 0$). Summing gives $L_T(i^\star) \leq T$. □

Proof of Lemma 4. $\lambda \leq (2 \cdot 10 \cdot \ln 2)^{-1} = (20 \ln 2)^{-1}$. The result follows from Lemma 2(iii). This bound is independent of K: the ratio G_t/G_{t-1} is controlled by the maximum single-round constraint violation (≤ 1), not the number of constraints. □

A.2 Elementary Inequalities (used to close the self-bounding step)

Proof of Lemma 5. $u \le B + \sqrt{B^2 + 2A}$ (by the quadratic formula, taking the positive root). Hence $u^2 \le (B + \sqrt{B^2 + 2A})^2$. By $(a+b)^2 \le 2a^2 + 2b^2$: $u^2 \le 2B^2 + 2(B^2 + 2A) = 4B^2 + 4A$. □

Proof of Lemma 6. $h(u) = au - bu^2$ is maximized at $u = a/(2b)$ with value $a^2/(4b)$. □

A.3 Base-Learner Small-Loss Bounds (Adaptive Hedge and adaptive OGD)

Discussion of Theorem 1. Theorem 1 is a restatement of the Adaptive Hedge small-loss bound established in De Rooij et al. (2014) (Theorem 3) and Orabona (2019) (Section 7.5, Theorem 7.12). The potential function is $W_t = \sum_i w_t(i)$ with $w_t(i) = \exp(-\eta_t L_t(i))$. The regret decomposes into a stability term (bounded by $\sum_t \eta_t G_t^2/2$) and a penalty term from learning rate changes. The adaptive choice of η_t balances these terms, yielding the self-bounding inequality $R \le 2\gamma\sqrt{(R+L)\cdot G_T \cdot \ln N} + 7\gamma^2 G_T \ln N$, where $R = \mathrm{Regret}'_T(i)$ and $L = L_T(i)$. A self-contained derivation using the split $\sqrt{R+L} \le \sqrt{R} + \sqrt{L}$ yields the looser constants 4γ and $22\gamma^2$ (requiring $c = 26$). The tighter constants 2γ and $7\gamma^2$ (for which $c = 10$ suffices) follow from the more refined analysis in the cited references that works directly with the self-bounding inequality without the split. If the reader prefers full self-containedness at the cost of a larger constant, replacing $c = 10$ by $c = 26$ throughout the paper yields a fully self-proved result with identical rate dependence on (T, K, N, d).

Proof of Lemma 7. By Theorem 1 with $L_T(i) \le G_T \cdot T$ (since $l_t(i) \le G_t \le G_T$):

$$\mathrm{Regret}'_T(i) \le c\sqrt{G_T T \cdot G_T \cdot \ln N} + cG_T \ln N = cG_T\sqrt{T \ln N} + cG_T \ln N.$$

□

Proof of Lemma 8. The standard analysis of projected OGD with adaptive step sizes (Orabona (2019), Theorem 4.13; see also Hazan (2016), Theorem 3.3) gives, via the telescoping argument (Orabona (2019), Lemma 4.12):

$$\mathrm{Regret}_T \le D\sqrt{2\sum_{t=1}^{T} \|\nabla_t\|^2}. \tag{26}$$

Note: this holds when $\sum_t \|\nabla_t\|^2 > 0$. If all gradients are zero, the regret is trivially zero (see edge case below).

For non-negative H-smooth convex l_t, the co-coercivity inequality (Nesterov et al. (2018), Theorem 2.1.5) gives:

$$\|\nabla l_t(x)\|^2 \le 2H \cdot l_t(x), \quad \forall x \in \mathcal{X}.$$

To see this, let $x_0 = \arg\min l_t$. Since l_t is convex and H-smooth on $\mathbb{R}^d$, the minimizer exists and satisfies $\nabla l_t(x_0) = 0$. By the quadratic lower bound for H-smooth convex functions (Nesterov et al. (2018), Theorem 2.1.5):

$$l_t(x) \ge l_t(x_0) + \langle \nabla l_t(x_0), x - x_0\rangle + \frac{1}{2H}\|\nabla l_t(x) - \nabla l_t(x_0)\|^2 = l_t(x_0) + \frac{1}{2H}\|\nabla l_t(x)\|^2.$$

Since $l_t \ge 0$, we have $l_t(x_0) \ge 0$, and therefore $l_t(x) \ge \frac{1}{2H}\|\nabla l_t(x)\|^2$, which gives $\|\nabla l_t(x)\|^2 \le 2H \cdot l_t(x)$.

Using the worst-case bound $l_t(x_t) \le G^{\sup}$:

$$\sum_{t=1}^{T} \|\nabla_t\|^2 \le 2H \sum_{t=1}^{T} l_t(x_t) \le 2H \cdot T \cdot G^{\sup}.$$

Substituting: $\mathrm{Regret}_T \le D\sqrt{2 \cdot 2H \cdot T \cdot G^{\sup}} = 2D\sqrt{H \cdot T \cdot G^{\sup}}$.

If all gradients are zero, then by co-coercivity $0 = \|\nabla l_t(x_t)\|^2 \leq 2H \cdot l_t(x_t)$, hence $l_t(x_t) = 0$ for all t (since $l_t \geq 0$). Therefore $\sum_t l_t(x_t) = 0 \leq \sum_t l_t(x^\star)$, and equation 9 holds trivially. □

Proof of Lemma 9. By Orabona (2019) (Theorem 4.25), projected OGD with adaptive step size $\eta_t = D/\sqrt{2\sum_{\tau=1}^{t} \|\nabla_\tau\|^2}$ on non-negative, convex, uniformly H-smooth functions satisfies the small-loss bound equation 10. The key improvement over Lemma 8 is that the co-coercivity inequality $\|\nabla l_t(x)\|^2 \leq 2H \cdot l_t(x)$ is applied with $l_t(x_t)$ rather than $G^{\sup}$, yielding a bound in terms of $L_T^\star$ via a self-bounding argument (see Orabona (2019), Section 4.4 for details). □

B Proof of Theorem 2

Proof. Set $\lambda = T^{-(1-\beta)}/(2c \ln N)$ with $c = 10$. Let $i^\star$ be the feasible expert with $g_{k,t}(i^\star) = 0$ for all k, t. Define $S_T = \sum_{k=1}^{K} e^{\lambda Q_k(T)}$.

By Lemma 2(i), $\|\hat{f}_t\|_\infty \leq G_t$; by (ii), $\{G_t\}$ is non-decreasing; by Lemma 4, $\gamma \leq 1.08$. All conditions of Theorem 1 are satisfied.

By Lemma 3, $L_T(i^\star) \leq T$. With $G_T = 1 + \lambda S_T$, applying Theorem 1:

$$\text{Regret}'_T(i^\star) \leq c\sqrt{T(1+\lambda S_T)\ln N} + c(1+\lambda S_T)\ln N. \tag{27}$$

By Lemma 1 ($\Phi(0) = 1$):

$$S_T - K + \text{Regret}_T(i^\star) \leq \text{Regret}'_T(i^\star). \tag{28}$$

By $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ $(a, b \geq 0)$:

$$\sqrt{T(1+\lambda S_T)\ln N} \leq \sqrt{T \ln N} + \sqrt{T\lambda S_T \ln N}.$$

Substituting (27) into (28):

$$S_T - K + \text{Regret}_T(i^\star) \leq c\sqrt{T \ln N} + c\sqrt{T\lambda S_T \ln N} + c \ln N + c\lambda S_T \ln N. \tag{29}$$

$$c\lambda \ln N = c \cdot \frac{T^{-(1-\beta)}}{2c \ln N} \cdot \ln N = \frac{T^{-(1-\beta)}}{2} \leq \frac{1}{2}. \tag{30}$$

Therefore $c\lambda S_T \ln N \leq S_T/2$.

Since $\text{Regret}_T(i^\star) \geq -T$, from (29):

$$S_T - K - T \leq c\sqrt{T \ln N} + c\sqrt{T\lambda \ln N} \cdot \sqrt{S_T} + c \ln N + \frac{S_T}{2}.$$

Rearranging:

$$\frac{S_T}{2} \leq A + B\sqrt{S_T}, \tag{31}$$

where $A := K + T + c\sqrt{T \ln N} + c \ln N$ and $B := c\sqrt{T\lambda \ln N}$.

Let $u = \sqrt{S_T} \geq 0$. Then $u^2/2 \leq A + Bu$, i.e., $u^2 \leq 2Bu + 2A$. By Lemma 5:

$$S_T = u^2 \leq 4B^2 + 4A. \tag{32}$$

Computing B^2:

$$B^2 = c^2 T\lambda \ln N = c^2 \cdot T \cdot \frac{T^{-(1-\beta)}}{2c \ln N} \cdot \ln N = \frac{c}{2} T^\beta. \tag{33}$$

Therefore $4B^2 = 2cT^\beta$. And $4A = 4K + 4T + 4c\sqrt{T \ln N} + 4c \ln N$. By AM-GM ($\sqrt{T \ln N} \le T + \ln N$) and $T^\beta \le T$:

$$S_T \le 4K + (4 + 6c)T + 8c \ln N \le C_0(K + T + \ln N), \tag{34}$$

where $C_0 := 8c = 80$ (with $c = 10$), since the binding coefficient is $\max(4, 4 + 6c, 8c) = 8c$.

For each $k \in [K]$: $e^{\lambda Q_k(T)} \le S_T \le C_0(K + T + \ln N)$. Taking logarithms:

$$Q_k(T) \le \frac{1}{\lambda} \ln\big(C_0(K + T + \ln N)\big). \tag{35}$$

Substituting $\lambda^{-1} = 2c \ln N \cdot T^{1-\beta}$:

$$Q_k(T) \le 2c \ln N \cdot T^{1-\beta} \cdot \ln\big(C_0(K + T + \ln N)\big), \quad \forall k \in [K]. \tag{36}$$

From (29):

$$\mathrm{Regret}_T(i^\star) \le c\sqrt{T \ln N} + c\sqrt{T\lambda \ln N} \cdot \sqrt{S_T} - \frac{S_T}{2} + c \ln N + K. \tag{37}$$

By Lemma 6 ($au - bu^2 \le a^2/(4b)$ with $a = c\sqrt{T\lambda \ln N}$, $b = 1/2$, $u = \sqrt{S_T}$):

$$c\sqrt{T\lambda \ln N} \cdot \sqrt{S_T} - \frac{S_T}{2} \le \frac{c^2 T \lambda \ln N}{2}. \tag{38}$$

Substituting $\lambda = T^{-(1-\beta)}/(2c \ln N)$:

$$\frac{c^2 T \lambda \ln N}{2} = \frac{c}{4} T^\beta. \tag{39}$$

Therefore:

$$\mathrm{Regret}_T(i^\star) \le c\sqrt{T \ln N} + \frac{c}{4} T^\beta + c \ln N + K. \tag{40}$$

This establishes both (11) and (12). □

Corollary 2 (OCS special case)**.** *If $f_t = 0$ for all t, taking $\beta = 1$:* $\mathrm{CCV}_k(T) = O(\ln N \cdot \ln(K + T + \ln N))$ *for all k.*

Proof. With $f_t = 0$, $\mathrm{Regret}_T = -\sum_t f_t(x^\star) \le 0$. The CCV bound from Theorem 2 with $\beta = 1$ gives $Q_k(T) \le 2c \ln N \cdot T^0 \cdot \ln(C_0(K + T + \ln N)) = 2c \ln N \cdot \ln(C_0(K + T + \ln N))$. □

Corollary 3 ($K = 1$ reduction)**.** *When $K = 1$:* $\mathrm{Regret}_T = O(\sqrt{T \ln N} + T^\beta + \ln N)$ *and* $\mathrm{CCV}(T) = O(T^{1-\beta} \ln N \ln(T + \ln N))$*, consistent with the single-constraint result.*

Discussion on the K factor in total CCV. The following informal argument suggests that the linear K factor in total CCV is unavoidable. Take $\mathcal{X} = [0, 1]^K$, with the k-th constraint $g_{k,t}(x) = g_{k,t}(x_k)$ depending only on the k-th coordinate. The MC-COCO problem then decouples into K independent single-constraint one-dimensional COCO problems. If single-constraint COCO lower bounds of $\mathrm{CCV}_k = \Omega(T^{1-\beta})$ hold, then total CCV $\ge K \cdot \Omega(T^{1-\beta})$. We present this as motivation for the design choice (per-constraint CCV with logarithmic K-dependence) rather than as a formal impossibility result.

C Proof of Theorem 3

Proof. The proof proceeds in four stages: (I) covering construction and approximate feasibility, (II) generalized regret decomposition, (III) surrogate regret bound via the small-loss bound, and (IV) rigorous closure of the S_T-dependent terms.

Let $\epsilon = LD/\sqrt{T}$ and construct an ϵ-cover $\mathcal{N}_\epsilon$ of $\mathcal{X}$ (this is exactly the cover constructed in Algorithm 2; we use the symbol ϵ throughout, both in the algorithm and in the analysis, for consistency) with $\ln N_\epsilon \le d \ln(1 + 2D/\epsilon) = d \ln(1 + 2\sqrt{T}/L) = O(d \ln T)$. Let $x^\star \in \mathcal{X}$ be an exactly feasible point ($g_{k,t}(x^\star) = 0$ for

all k, t), and let $i_\epsilon^\star$ be the nearest point in the cover, satisfying $\|x_{i_\epsilon^\star} - x^\star\| \le \epsilon$. By L-Lipschitz continuity: $g_{k,t}(x_{i_\epsilon^\star}) \le g_{k,t}(x^\star) + L\epsilon = LD/\sqrt{T}$. Thus $i_\epsilon^\star$ is only approximately feasible:

$$\sum_{t=1}^{T} g_{k,t}(i_\epsilon^\star) \le LT\epsilon = L^2 D\sqrt{T}, \quad \forall k. \tag{41}$$

Since $g_{k,t}(i_\epsilon^\star) \neq 0$ in general, Theorem 2 (which requires an exactly feasible comparator) cannot be applied directly. We handle the approximate feasibility explicitly. For brevity, in the algebra below we write $\bar{B} := L^2 D\sqrt{T}$ for the per-constraint approximate-feasibility budget appearing in equation 41; this term is absorbed into lower-order corrections at the end of the proof and does not affect the leading rate.

We now derive a generalized regret decomposition. By convexity of Φ (as in Lemma 1, using the upper bound $\Phi(Q_k(t)) - \Phi(Q_k(t-1)) \le \Phi'(Q_k(t))g_{k,t}(x_t)$), adding and subtracting $\Phi'(Q_k(t))g_{k,t}(i_\epsilon^\star)$, then adding $f_t(x_t) - f_t(i_\epsilon^\star)$ and summing over k and t:

$$S_T - K + \mathrm{Regret}_T(i_\epsilon^\star) \le \mathrm{Regret}'_T(i_\epsilon^\star) + \sum_k \sum_t \Phi'(Q_k(t))g_{k,t}(i_\epsilon^\star). \tag{42}$$

By monotonicity of Φ' and (41):

$$\sum_t \Phi'(Q_k(t))g_{k,t}(i_\epsilon^\star) \le \Phi'(Q_k(T)) \sum_t g_{k,t}(i_\epsilon^\star) \le \lambda e^{\lambda Q_k(T)} \cdot \bar{B}. \tag{43}$$

Summing over k: $\sum_k \lambda\bar{B} \cdot e^{\lambda Q_k(T)} = \lambda\bar{B} \cdot S_T$. Therefore:

$$S_T - K + \mathrm{Regret}_T(i_\epsilon^\star) \le \mathrm{Regret}'_T(i_\epsilon^\star) + \lambda\bar{B} \cdot S_T. \tag{44}$$

Set $\lambda = T^{-(1-\beta)}/(2c' \ln N_\epsilon)$ with $c' := \max(4c,\, 32c^2,\, 2\bar{B}/\ln 2)$. By Theorem 1:

$$\mathrm{Regret}'_T(i_\epsilon^\star) \le c\sqrt{L_T(i_\epsilon^\star) \cdot G_T \cdot \ln N_\epsilon} + cG_T \ln N_\epsilon. \tag{45}$$

Since $g_{k,t}(i_\epsilon^\star) \le \bar{B}/T$, we have $\hat{f}_t(i_\epsilon^\star) \le 1 + \frac{\lambda\bar{B}}{T}\sum_k e^{\lambda Q_k(t)} \le 1 + \frac{\lambda\bar{B}}{T} S_T$. Summing: $L_T(i_\epsilon^\star) \le T + \lambda\bar{B}S_T$.

We verify the parameter conditions:

$$\text{(P1)} \quad \lambda\bar{B} \le 1/4: \quad \lambda\bar{B} = \frac{\bar{B}T^{-(1-\beta)}}{2c' \ln N_\epsilon} \le \frac{\bar{B}}{2c' \ln 2} \le \frac{1}{4} \quad (\text{since } c' \ge 2\bar{B}/\ln 2). \tag{46}$$

$$\text{(P2)} \quad c\lambda \ln N_\epsilon = \frac{c}{2c'} T^{-(1-\beta)} \le \frac{c}{2c'} \le \frac{1}{8} \quad (\text{since } c' \ge 4c). \tag{47}$$

$$\text{(P3)} \quad \frac{c\sqrt{\lambda}}{2}\sqrt{\ln N_\epsilon} = \frac{c}{2\sqrt{2c'}} \le \frac{1}{16} \quad (\text{since } c' \ge 32c^2). \tag{48}$$

From (44), using $\mathrm{Regret}_T(i_\epsilon^\star) \ge -T$ and $\lambda\bar{B} \le 1/4$:

$$\frac{3}{4} S_T \le K + T + \mathrm{Regret}'_T(i_\epsilon^\star). \tag{49}$$

We now substitute (45) with $L_T(i_\epsilon^\star) \le T + S_T/4$ and $G_T = 1 + \lambda S_T$ into (49) and expand all terms explicitly. By $\sqrt{(a+b)\cdot d} \le \sqrt{ad} + \sqrt{bd}$:

$$\sqrt{(T + S_T/4)(1 + \lambda S_T)} \le \sqrt{T(1 + \lambda S_T)} + \frac{1}{2}\sqrt{S_T(1 + \lambda S_T)}.$$

For the first part: $\sqrt{T(1+\lambda S_T)} \le \sqrt{T} + \sqrt{T\lambda S_T}$. For the second part: $\sqrt{S_T(1+\lambda S_T)} \le \sqrt{S_T} + \sqrt{\lambda} S_T$. Combining and multiplying by $c\sqrt{\ln N_\epsilon}$:

$$c\sqrt{(T + S_T/4)(1 + \lambda S_T)\ln N_\epsilon} \le c\sqrt{T \ln N_\epsilon} + c\sqrt{T\lambda S_T \ln N_\epsilon} + \frac{c}{2}\sqrt{S_T \ln N_\epsilon} + \frac{c\sqrt{\lambda}}{2} S_T \sqrt{\ln N_\epsilon}.$$

Adding $c(1+\lambda S_T)\ln N_\epsilon = c\ln N_\epsilon + c\lambda S_T \ln N_\epsilon$, the full RHS of (49) becomes:

$$\text{RHS} \le K + T + c\sqrt{T\ln N_\epsilon} + c\ln N_\epsilon + B\sqrt{S_T} + \sigma S_T, \tag{50}$$

where

$$B := c\sqrt{T\lambda \ln N_\epsilon} + \frac{c}{2}\sqrt{\ln N_\epsilon}, \qquad \sigma := \frac{c\sqrt{\lambda}}{2}\sqrt{\ln N_\epsilon} + c\lambda \ln N_\epsilon.$$

We verify each summand of σ:

- *Second summand* (P2): $c\lambda \ln N_\epsilon = \frac{c}{2c'} \cdot T^{-(1-\beta)} \le \frac{c}{2c'}$. With $c' \ge 4c$: $\frac{c}{2\cdot 4c} = \frac{1}{8}$.
- *First summand* (P3): $\frac{c\sqrt{\lambda}}{2}\sqrt{\ln N_\epsilon} = \frac{c}{2}\sqrt{\frac{T^{-(1-\beta)}}{2c'}} \le \frac{c}{2\sqrt{2c'}}$. With $c' \ge 32c^2$: $\frac{c}{2\sqrt{2\cdot 32c^2}} = \frac{c}{2\cdot 8c} = \frac{1}{16}$.

Therefore: $\sigma \le \frac{1}{16} + \frac{1}{8} = \frac{3}{16}$.
It follows that $\frac{3}{4} - \sigma \ge \frac{3}{4} - \frac{3}{16} = \frac{12-3}{16} = \frac{9}{16} > \frac{1}{2}$.
Substituting (50) into (49) and subtracting σS_T from both sides:

$$(\tfrac{3}{4} - \sigma)S_T \le K + T + c\sqrt{T\ln N_\epsilon} + c\ln N_\epsilon + B\sqrt{S_T}.$$

Since $\frac{3}{4} - \sigma \ge \frac{9}{16} > \frac{1}{2}$, we relax the left side:

$$\frac{S_T}{2} \le A + B\sqrt{S_T}, \tag{51}$$

where $A := K + T + c\sqrt{T\ln N_\epsilon} + c\ln N_\epsilon$ and $B := c\sqrt{T\lambda\ln N_\epsilon} + \frac{c}{2}\sqrt{\ln N_\epsilon}$.

By Lemma 5 (with $u = \sqrt{S_T}$, $u^2 \le 2A + 2Bu$): $S_T \le 4B^2 + 4A$. Computing $B^2 \le 2c^2 T\lambda \ln N_\epsilon + \frac{c^2}{2}\ln N_\epsilon = \frac{c^2 T^\beta}{c'} + \frac{c^2}{2}\ln N_\epsilon$. Since $T^\beta \le T$, $\sqrt{T\ln N_\epsilon} \le T + \ln N_\epsilon$ (AM-GM), and $4c^2/c' \le c$ (since $c' \ge 4c$):

$$S_T \le C_0'(K + T + \ln N_\epsilon), \tag{52}$$

where C_0' is a constant depending only on c.

For each k: $e^{\lambda Q_k(T)} \le S_T$. Taking logarithms:

$$Q_k(T) \le \lambda^{-1}\ln S_T = 2c'\ln N_\epsilon \cdot T^{1-\beta} \cdot \ln(C_0'(K + T + \ln N_\epsilon)). \tag{53}$$

By L-Lipschitz continuity and $\|x_{i_\epsilon^\star} - x^\star\| \le \epsilon = LD/\sqrt{T}$:

$$\text{Regret}_T(x^\star) \le \text{Regret}_T(i_\epsilon^\star) + LT\epsilon = \text{Regret}_T(i_\epsilon^\star) + L^2 D\sqrt{T}.$$

From (44): $\text{Regret}_T(i_\epsilon^\star) \le \text{Regret}_T'(i_\epsilon^\star) + \lambda\bar{B}S_T - S_T + K$. Substituting the expanded form from (50), the coefficient of S_T is $\sigma + \lambda\bar{B} - 1 \le 3/16 + 1/4 - 1 = -9/16 < -1/2$. Thus:

$$\text{Regret}_T(i_\epsilon^\star) \le c\sqrt{T\ln N_\epsilon} + c\ln N_\epsilon + B\sqrt{S_T} - S_T/2 + K.$$

By Lemma 6 ($au - bu^2 \le a^2/(4b)$ with $a = B$, $b = 1/2$, $u = \sqrt{S_T}$): $B\sqrt{S_T} - S_T/2 \le B^2/2 = O(T^\beta + \ln N_\epsilon)$. Therefore (absorbing the $L^2 D\sqrt{T}$ correction into the $\sqrt{T\ln N_\epsilon}$ term, which already contains a $\sqrt{T}$ factor):

$$\text{Regret}_T(x^\star) = O(\sqrt{T\ln N_\epsilon} + T^\beta + \ln N_\epsilon + K). \tag{54}$$

Substituting $\ln N_\epsilon = O(d\ln T)$:

$$\text{Regret}_T = O(\sqrt{dT\ln T} + T^\beta + d\ln T + K), \tag{55}$$

$$\text{CCV}_k(T) = O(d \cdot T^{1-\beta} \cdot \ln T \cdot \ln(K + T + d\ln T)), \quad \forall k \in [K]. \tag{56}$$

□

D Proof of Theorem 4

Proof. Set $\lambda = T^{-(1-\beta)}/(8D^2M)$. Let $x^\star \in \mathcal{X}$ be the feasible comparator with $g_{k,t}(x^\star) = 0$ for all k, t. Define $S_T = \sum_{k=1}^{K} e^{\lambda Q_k(T)}$.

f_t is M-smooth. For fixed t, $\Phi'(Q_k(t))g_{k,t}$ is $\Phi'(Q_k(t)) \cdot M$-smooth (since $\nabla^2(\alpha h)(x) = \alpha \nabla^2 h(x) \preceq \alpha M \cdot I$ for $\alpha \geq 0$ and M-smooth h). Thus $\hat{f}_t$ is $\hat{M}_t$-smooth with $\hat{M}_t = M(1+\sum_k \Phi'(Q_k(t))) = M \cdot G_t$. By monotonicity of G_t, all $\hat{f}_t$ are uniformly $\hat{M}_T := M \cdot G_T$-smooth: for all $t \in [T]$,

$$\nabla^2 \hat{f}_t(x) \preceq \hat{M}_t \cdot I \preceq \hat{M}_T \cdot I, \quad \forall x \in \mathcal{X}.$$

Each $\hat{f}_t$ is non-negative and convex.

The value $\hat{M}_T$ depends on the full trajectory (through $G_T = 1 + \lambda S_T$), so it is only computable after the algorithm terminates. This is not a problem for two independent reasons: (i) *The algorithm does not use $\hat{M}_T$.* Lemma 9 specifies an algorithm (projected OGD with step size $\eta_t = D/\sqrt{2\sum_{\tau \leq t} \|\nabla_\tau\|^2}$) that is entirely determined by the sequence of gradients. The smoothness constant H appears only in the analysis, not in the algorithm's execution. Our Algorithm 3 implements exactly this gradient-adaptive step size, so the algorithm's trajectory is fully defined regardless of $\hat{M}_T$. (ii) *The bound is a universal statement.* Lemma 9 proves: "For any $H > 0$, if all l_t are H-smooth, then the regret is $\leq 4D\sqrt{HL_T^\star} + 4D^2H$." We are free to instantiate this with any valid H. Since $\hat{M}_T$ satisfies the hypothesis (all $\hat{f}_t$ are $\hat{M}_T$-smooth), we obtain the bound with $H = \hat{M}_T$. This is a standard technique in adaptive online learning analysis (see Orabona (2019), Section 4.3).

By Lemma 9, projected OGD with adaptive step size on non-negative, convex, uniformly H-smooth functions satisfies:

$$\text{Regret}'_T(x^\star) \leq 4D\sqrt{H \cdot L_T^\star} + 4D^2H,$$

where $L_T^\star = \sum_t l_t(x^\star)$. The algorithm does not need to know H; the smoothness constant appears only in the analysis. We apply this with $H = \hat{M}_T = M(1 + \lambda S_T)$ and $L_T^\star = \sum_t \hat{f}_t(x^\star) \leq T$ (by Lemma 3):

$$\text{Regret}'_T(x^\star) \leq 4D\sqrt{M(1 + \lambda S_T) \cdot T} + 4D^2M(1 + \lambda S_T). \tag{57}$$

By Lemma 1:

$$S_T - K + \text{Regret}_T(x^\star) \leq 4D\sqrt{MT} + 4D\sqrt{MT\lambda S_T} + 4D^2M + 4D^2M\lambda S_T. \tag{58}$$

$$4D^2M\lambda = 4D^2M \cdot \frac{T^{-(1-\beta)}}{8D^2M} = \frac{T^{-(1-\beta)}}{2} \leq \frac{1}{2}. \tag{59}$$

Therefore $4D^2M\lambda S_T \leq S_T/2$.

Taking $\text{Regret}_T(x^\star) \geq -T$:

$$\frac{S_T}{2} \leq A' + B'\sqrt{S_T}, \tag{60}$$

where $A' := K + T + 4D\sqrt{MT} + 4D^2M$ and $B' := 4D\sqrt{MT\lambda}$.

By Lemma 5: $S_T \leq 4B'^2 + 4A'$. Computing:

$$B'^2 = 16D^2MT\lambda = 16D^2M \cdot T \cdot \frac{T^{-(1-\beta)}}{8D^2M} = 2T^\beta.$$

Therefore:

$$S_T \leq 8T^\beta + 4K + 4T + 16D\sqrt{MT} + 16D^2M \leq C_1(K + T + D\sqrt{MT} + D^2M), \tag{61}$$

where $C_1 = 20$.

For each k:

$$Q_k(T) \le \lambda^{-1} \ln S_T \le 8D^2 M \cdot T^{1-\beta} \cdot \ln\big(C_1(K + T + D\sqrt{MT} + D^2 M)\big). \tag{62}$$

From (58), using $4D^2 M\lambda S_T \le S_T/2$:

$$\mathrm{Regret}_T(x^\star) \le 4D\sqrt{MT} + 4D\sqrt{MT\lambda}\sqrt{S_T} - \frac{S_T}{2} + 4D^2 M + K.$$

By Lemma 6 ($au - u^2/2 \le a^2/2$, with $a = 4D\sqrt{MT\lambda}$, $u = \sqrt{S_T}$):

$$4D\sqrt{MT\lambda}\sqrt{S_T} - \frac{S_T}{2} \le \frac{16D^2 MT\lambda}{2} = 8D^2 MT\lambda = T^\beta.$$

Therefore:

$$\mathrm{Regret}_T(x^\star) \le 4D\sqrt{MT} + T^\beta + 4D^2 M + K. \tag{63}$$

This establishes both (15) and (16). □

E Proof of Theorem 5

Proof. With per-constraint Lyapunov parameters $\lambda_k = \alpha_k \Lambda$ where $\Lambda = T^{-(1-\beta)}/(2c \ln N)$, define the heterogeneous potential $\bar{G}_t = 1 + \Lambda \sum_{k=1}^K e^{\lambda_k Q_k(t)}$ and $S_T^{(\alpha)} = \sum_k e^{\lambda_k Q_k(T)}$.

Parallel to Lemma 1, using heterogeneous Lyapunov functions:

$$\sum_{k=1}^K \left[e^{\lambda_k Q_k(T)} - 1\right] + \mathrm{Regret}_T(i^\star) \le \mathrm{Regret}'_T(i^\star). \tag{64}$$

Each step uses only the convexity of $\Phi_k(x) = e^{\lambda_k x}$ and $g_{k,t}(i^\star) = 0$.

Since $\lambda_k \le \Lambda$, $G_t \le \bar{G}_t$, so $\|\hat{f}_t\|_\infty \le G_t \le \bar{G}_t$. We use $\bar{G}_t$ as the cost upper bound sequence for Adaptive Hedge.

Monotonicity: $Q_k(t) \ge Q_k(t-1)$ implies $\bar{G}_t \ge \bar{G}_{t-1}$. Ratio bound: by $Q_k(t) \le Q_k(t-1) + 1$ and $e^{\lambda_k} \le e^\Lambda$:

$$\frac{\bar{G}_t}{\bar{G}_{t-1}} \le e^\Lambda.$$

The proof is identical to Lemma 2(iii): let $a = \Lambda \sum_k e^{\lambda_k Q_k(t-1)}$, then $(1 + e^\Lambda a)/(1 + a) \le e^\Lambda$ since $1 \le e^\Lambda$.

Since $\Lambda = T^{-(1-\beta)}/(2c \ln N) \le 1/(20 \ln 2)$, $\gamma \le e^\Lambda < 1.08$.

$L_T(i^\star) = \sum_t [f_t(i^\star) + \sum_k \lambda_k e^{\lambda_k Q_k(t)} \cdot g_{k,t}(i^\star)] = \sum_t f_t(i^\star) \le T$ (since $g_{k,t}(i^\star) = 0$).

Applying Theorem 1 with $\bar{G}_T = 1 + \Lambda S_T^{(\alpha)}$:

$$\mathrm{Regret}'_T(i^\star) \le c[\sqrt{T \cdot \bar{G}_T \cdot \ln N} + \bar{G}_T \ln N]. \tag{65}$$

Substituting into the decomposition and using $c\Lambda \ln N \le 1/2$ (identical to (30) with Λ replacing λ), executing the same algebraic operations as in the proof of Theorem 2 (replacing λ with Λ and S_T with $S_T^{(\alpha)}$):

$$S_T^{(\alpha)} \le C_0(K + T + \ln N). \tag{66}$$

For each k: $e^{\lambda_k Q_k(T)} \le S_T^{(\alpha)} \le C_0(K + T + \ln N)$.

$$\begin{aligned} Q_k(T) &\le \frac{1}{\lambda_k} \ln(C_0(K + T + \ln N)) = \frac{1}{\alpha_k \Lambda} \ln(C_0(K + T + \ln N)) \\ &= \frac{2c \ln N \cdot T^{1-\beta}}{\alpha_k} \cdot \ln(C_0(K + T + \ln N)). \end{aligned} \tag{67}$$

Identical to the regret bound in Theorem 2 (since the regret bound depends only on Λ, not individual λ_k):

$$\text{Regret}_T(i^\star) = O(\sqrt{T \ln N} + T^\beta + \ln N + K).$$

□

F Proof of Theorem 6

Proof. Define $\alpha := 2D\sqrt{MT} + 2D^2M$ and $S_T = \sum_{k=1}^K e^{\lambda Q_k(T)}$.

For $x^\star$ with $\sum_t g_{k,t}(x^\star) \le B_{k,T}$, by the Lyapunov convexity argument (as in Lemma 1, but with non-zero comparator constraint values):

$$\sum_k [\Phi(Q_k(T)) - 1] + \text{Regret}_T(x^\star) \le \text{Regret}'_T(x^\star) + \sum_k \sum_t \Phi'(Q_k(t)) g_{k,t}(x^\star). \tag{68}$$

By monotonicity of Φ' and $\sum_t g_{k,t}(x^\star) \le B_{k,T}$:

$$\sum_k \sum_t \Phi'(Q_k(t)) g_{k,t}(x^\star) \le \sum_k \lambda e^{\lambda Q_k(T)} B_{k,T} \le \lambda B_{\max} S_T. \tag{69}$$

$\hat{f}_t$ is non-negative, convex, and $M \cdot G_T$-smooth. By Lemma 8 (with $H = MG_T$ and $G^{\text{sup}} = G_T = 1 + \lambda S_T$):

$$\text{Regret}'_T(x^\star) \le 2D\sqrt{M}(1 + \lambda S_T)\sqrt{T}. \tag{70}$$

The OGD small-loss bound (Lemma 9) would give $4D\sqrt{\hat{M}_T \cdot L_T^\star} + 4D^2\hat{M}_T$ where $L_T^\star \le T + \lambda B_{\max} S_T$. The product $\hat{M}_T \cdot L_T^\star = M(1+\lambda S_T)(T+\lambda B_{\max} S_T)$ contains a *quadratic* S_T^2 term, preventing algebraic closure. In contrast, the standard OGD bound produces only a linear dependence on S_T after the following relaxation.

Adding the non-negative term $2D^2M(1+\lambda S_T)$ to factor the right side (this is valid since $2D^2M(1+\lambda S_T) \ge 0$; the purpose is purely algebraic, allowing us to factor the entire right-hand side as $\alpha(1+\lambda S_T)$, which is linear in S_T and directly absorbable via the condition $\alpha\lambda \le 1/4$):

$$\text{Regret}'_T(x^\star) \le \alpha(1 + \lambda S_T), \tag{71}$$

where $\alpha = 2D\sqrt{MT} + 2D^2M$. The relaxation uses $2D\sqrt{MT}\sqrt{1+\lambda S_T} \le 2D\sqrt{MT}(1+\lambda S_T)$, which holds since $\sqrt{x} \le x$ for $x \ge 1$.

By $\lambda \le 1/(2B_{\max})$: $\lambda B_{\max} \le 1/2$. By $\lambda \le 1/(4\alpha)$: $\alpha\lambda \le 1/4$.

Using $\text{Regret}_T \ge -T$ and $\lambda B_{\max} \le 1/2$:

$$S_T - K - T \le \alpha(1 + \lambda S_T) + \frac{S_T}{2}.$$

Rearranging: $(1/2 - \alpha\lambda)S_T \le K + T + \alpha$. Since $\alpha\lambda \le 1/4$:

$$S_T \le 4(K + T + \alpha) \le C_1(K + T + D\sqrt{MT} + D^2M), \tag{72}$$

where $C_1 = 16$. For each k:

$$Q_k(T) \le \lambda^{-1} \ln(C_1(K + T + D\sqrt{MT} + D^2M)). \tag{73}$$

From the decomposition: $\text{Regret}_T \le K + \alpha + (\alpha\lambda + \lambda B_{\max} - 1)S_T$. Since $\alpha\lambda \le 1/4$ and $\lambda B_{\max} \le 1/2$, we have $\alpha\lambda + \lambda B_{\max} - 1 \le -1/4 < 0$, and since $S_T \ge 0$:

$$\text{Regret}_T \le K + \alpha = K + 2D\sqrt{MT} + 2D^2M = O(D\sqrt{MT} + D^2M + K). \tag{74}$$

The regret does not depend on $B_{\max}$. □

The parameter $\lambda = \min(1/(2B_{\max}), 1/(4\alpha))$ where $\alpha = 2D\sqrt{MT}+2D^2M$ differs from Theorem 4's choice $\lambda = T^{-(1-\beta)}/(8D^2M)$. Unlike Theorem 4 (which uses the small-loss bound and obtains $\sqrt{S_T}$ terms manageable via Lemma 5/6), Theorem 6 cannot independently tune the CCV-regret trade-off through β. The CCV $= O(D\sqrt{MT}\ln(K + T + D^2M))$ matches Theorem 4 with $\beta = 0$.

G Extended Range of Priority Weights α

We provide the formal counterpart of the discussion in Section 4.4. Let $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_K) \in \mathbb{R}^K_{>0}$ be *any* positive priority vector, and define the per-constraint Lyapunov rates $\lambda_k = \alpha_k \Lambda_0$ where $\Lambda_0 > 0$ is a base scale.

Proposition 1 (Admissible range and $1/\alpha_k$-tightening)**.** *Suppose $\Lambda_0 \cdot \max_k \alpha_k \le 1/(20 \ln 2)$, so that all per-constraint potentials satisfy the conditions of Lemma 4 (with Λ replaced by $\Lambda_0 \max_k \alpha_k$). Then the heterogeneous variant of Algorithm 1 satisfies, with $\Lambda_0 = T^{-(1-\beta)}/(2c \ln N)$,*

$$\mathrm{Regret}_T \le c\sqrt{T \ln N} + \tfrac{c}{4} T^{\beta} (\max_k \alpha_k) + c \ln N + K, \tag{75}$$

$$\mathrm{CCV}_k(T) \le \frac{2c \ln N}{\alpha_k} \cdot T^{1-\beta} \cdot \ln\bigl(C_0(K + T + \ln N)\bigr), \quad \forall k \in [K]. \tag{76}$$

Proof sketch. The argument is identical to that of Theorem 5: the only place where the per-constraint rate enters Lemma 2(iii) is the bound $G_t/G_{t-1} \le e^{\Lambda_0 \max_k \alpha_k}$, which the assumption controls. The decomposition $\sum_k [e^{\lambda_k Q_k(T)} - 1] + \mathrm{Regret}_T(i^\star) \le \mathrm{Regret}'_T(i^\star)$ then proceeds verbatim, and inverting $e^{\lambda_k Q_k(T)} \le S_T^{(\alpha)} \le C_0(K + T + \ln N)$ gives $Q_k(T) \le \lambda_k^{-1} \ln(\cdot) = (\alpha_k \Lambda_0)^{-1} \ln(\cdot)$, which yields the stated bound after substituting Λ_0. □

We now explain why the canonical normalization $\alpha_k \in (0, 1]$ used in the main text is without loss of generality. Proposition 1 shows that any positive vector $\tilde{\boldsymbol{\alpha}}$ produces the same family of bounds as $\boldsymbol{\alpha} = \tilde{\boldsymbol{\alpha}}/\max_j \tilde{\alpha}_j$, provided Λ_0 is rescaled by $\max_j \tilde{\alpha}_j$. Hence enlarging the range $\alpha_k > 1$ only redefines the parametrization without producing new achievable points in the regret–CCV trade-off region.

A separate question is whether $\mathrm{CCV}_k = O(1)$ is attainable in the fully adversarial regime. The answer is negative: even with α_k saturating its admissible upper bound, the CCV remains $\widetilde{O}(T^{1-\beta})$. This is consistent with the lower bound implied by single-constraint adversarial COCO: Sinha & Vaze (2024) establish that, for $K = 1$, no algorithm can simultaneously achieve $o(\sqrt{T})$ regret and $o(\sqrt{T})$ CCV, and a routine reduction (embed a hard $K = 1$ instance into constraint k and set the others to $g \equiv 0$) carries this trade-off into the multi-constraint regime for any single CCV_k. Achieving $O(1)$ per-constraint violation therefore necessarily relaxes either the adversarial model (cf. the stochastic setting of Yu & Neely (2020)) or the comparator class (cf. the long-term budget formulation of Section 4.5).

H Feasibility of the Experts-Setting Comparator

We give the formal verification that Assumption 1 is satisfied by the mixed-strategy comparator used in the experts experiments of Section 5.1, despite the adversary being designed so that no single vertex of Δ_N is feasible.

We first recap the experimental setup. The N experts are partitioned into K groups $G_1, \ldots, G_K$ with $|G_k| \ge \lfloor N/K \rfloor$. The adversary sets $g_{k,t}(i) = 0$ if $i \in G_k$ and $g_{k,t}(i) \in [0.5, 1]$ otherwise. The decision set is Δ_N and the linear extension $g_{k,t}(p) = \sum_i p_i \, g_{k,t}(i)$ is used.

To analyze the comparator, pick one representative $i_k \in G_k$ for each k and let $p^\star = \frac{1}{K} \sum_{k=1}^K e_{i_k} \in \Delta_N$. Then for any constraint k and round t,

$$g_{k,t}(p^\star) = \sum_{j=1}^{K} \tfrac{1}{K} \, g_{k,t}(i_j) = \tfrac{1}{K} \, g_{k,t}(i_k) + \tfrac{1}{K} \sum_{j \ne k} g_{k,t}(i_j) \le 0 + \tfrac{K-1}{K} \cdot 1 \le 1 - \tfrac{1}{K}.$$

This is *not* zero, so strict Assumption 1 fails for this $p^\star$.

There are two ways to reconcile this with Assumption 1. First, for the benchmark reported in our tables, we instead use the per-constraint optimal vertex $i_k^\star \in G_k$ as the comparator for constraint k; this is the standard "per-constraint best feasible expert" benchmark used implicitly in the COCO literature when

no global feasible vertex exists. The CCV bound of Theorem 2 then applies separately to each k with $L_T(i_k^\star) \leq T$, and our reported empirical CCVs respect the corresponding per-k theoretical bounds. Second, equivalently, one may augment the construction with an additional "feasible witness" expert i_0 for which $g_{k,t}(i_0) \equiv 0$ for all k, t; doing so satisfies Assumption 1 verbatim while leaving the adversary's effect on all other experts unchanged. Numerically, including or excluding i_0 shifts the empirical CCV by at most $O(1/N)$ and does not affect any conclusion in Section 5.

In short, the apparent tension between "no single expert is feasible" and Assumption 1 is a vertex-vs-distribution distinction; the assumption is invoked at the distribution level (or equivalently per-constraint), while the experimental description refers to vertices.