Lazy Lagrangians for Optimistic Learning With Budget Constraints

Anderson, Daron; Iosifidis, George; Leith, Douglas J.

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Lazy Lagrangians for Optimistic Learning With Budget Constraints

Daron Anderson, George Iosifidis, and Douglas J. Leith, *Senior Member, IEEE*

Abstract—**We consider the general problem of online convex optimization with time-varying budget constraints in the presence of predictions for the next cost and constraint functions, that arises in a plethora of network resource management problems. A novel saddle-point algorithm is designed by combining a Follow-The-Regularized-Leader iteration with prediction-adaptive dynamic steps. The algorithm achieves $\mathcal{O}(T^{(3-\beta)/4})$ regret and $\mathcal{O}(T^{(1+\beta)/2})$ constraint violation bounds that are tunable via parameter $\beta \in [1/2, 1)$ and have constant factors that shrink with the predictions quality, achieving eventually $\mathcal{O}(1)$ regret for perfect predictions. Our work extends the seminal FTRL framework for this new OCO setting and outperforms the respective state-of-the-art greedy-based solutions which naturally cannot benefit from predictions, without imposing conditions on the (unknown) quality of predictions, the cost functions or the geometry of constraints, beyond convexity.**

Index Terms—**Network control, network management, resource allocation, online convex optimization (OCO), online learning.**

I. INTRODUCTION

THE online convex optimization (OCO) framework introduced in [1] is employed to solve various learning problems in networks, ranging from spam filtering, to data caching [2], network routing [3] and flow control [4], among others. At each round t an algorithm selects an action x_t from a convex set $\mathcal{X} \subset \mathbb{R}^N$ and incurs cost $f_t(x_t)$, where the convex function $f_t : \mathcal{X} \mapsto \mathbb{R}$ is revealed after x_t is decided. The algorithm's performance is measured using the metric of *regret*:

$$\mathcal{R}_T = \sum_{t=1}^{T} \left(f_t(x_t) - f_t(x^\star) \right), \quad (1)$$

which quantifies the difference of the total cost from that of the best action selected with hindsight $x^\star \in \arg\min_{x \in \mathcal{X}} \sum_{t=1}^{T} f_t(x)$. The goal is to select actions $\{x_t\}$ that ensure sublinear regret, i.e., $\mathcal{R}_T = o(T)$.

A practical extension of this setting is the *constrained OCO* framework, where the actions must satisfy long-term constraints of time-varying functions:

$$g_t(x) = \left(g_t^{(1)}(x), g_t^{(2)}(x), \ldots, g_t^{(d)}(x) \right) \preceq 0,$$

which are unknown when x_t is decided. In this case we are additionally interested in achieving sublinear total constraint violation, $\mathcal{V}_T = o(T)$, where:

$$\mathcal{V}_T = \left\| \left[\sum_{t=1}^{T} g_t(x_t) \right]_+ \right\|. \quad (2)$$

Constrained OCO algorithms have applications in the control of capacitated communication systems; various network queuing problems [5]; and network management with multiple constraints and performance criteria [6]. Nevertheless, these problems are notoriously hard to tackle. In particular, [7] showed that no algorithm can achieve sublinear regret and constraint violation relative to the ideal benchmark:

$$x^\star \in \mathcal{X}_T^{\max} = \left\{ x \in \mathcal{X} \,\middle|\, \sum_{t=1}^{T} g_t(x) \preceq 0 \right\}.$$

Subsequent works considered instead benchmarks that respect the constraints for short time windows [8]; dynamic benchmarks $\{x_t^\star\}$ that satisfy separately each t-round constraint $g_t(x_t^\star) \preceq 0$ [6], [9]; or benchmarks [10], [11] restricted in:

$$\mathcal{X}_T = \left\{ x \in \mathcal{X} \,\middle|\, g_t(x) \preceq 0, \, \forall t \leq T \right\}.$$

Special cases of $\mathcal{X}_T$ are considered in [12], [13], [14], and [15] where $g_t(x) = g(x)$, $\forall t$; and in [16] which focuses on linearly-perturbed constraints $g_t(x) = g(x) + b_t$.

An aspect that has received less attention, however, is whether constrained OCO algorithms can be assisted by predictions for the next-round functions f_t and g_t. Such information can be provided by a pre-trained model that uses incomplete data and hence cannot be fully trusted – yet, can still assist the online algorithm. Leveraging predictions to improve learning algorithms is attracting increasing interest and has many practical applications, e.g., in data caching [17]; online rent-or-buy problems [18]; and in scheduling algorithms [19], among other areas. In this context, a key challenge is that the predictions might exhibit time-varying and unknown accuracy, which, furthermore, may vary across the cost and constraint functions. This confounds their incorporation in online learning algorithms and raises the question: *how much can predictions improve the performance of constrained OCO algorithms and how can we accrue these benefits in the presence of inaccurate, potentially even adversarial, predictions?*
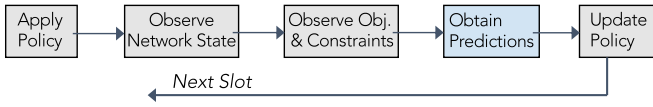
Fig. 1.   Block diagram of dynamic and adaptive network control solutions with prediction-assisted online learning.

Our goal is to tackle this question by developing a framework that brings together online learning [20] and data-driven (prediction-based) network management solutions, Fig. 1.

A. Background and Related Work

Early works studying the impact of predictions include [21] and [22] which considered linear costs $c_t = \nabla f_t(x_t)$ and predictions $\tilde{c}_t$ with guaranteed correlation $\tilde{c}_t^\top c_t \geq \alpha \|c_t\|^2$. These predictions improve the regret from $\mathcal{O}(\sqrt{T})$ to $\mathcal{O}(\log T)$. Reference [23] considered the case when at most B of the predictions fail the correlation condition and provided an $\mathcal{O}\big((1+\sqrt{B})/\alpha\big)\log(1+T-B)\big)$ regret algorithm, that was further extended to combine multiple predictors [24]. However, these prior models assume $\mathcal{X}$ is time-invariant. A different line of works use adaptive regularizers and define prediction errors $\varepsilon_t = \|c_t - \tilde{c}_t\|$ to obtain $\mathcal{O}\big(\sqrt{\sum_t \varepsilon_t^2}\big)$ regret bounds [25], [26]. We adapt these methods to time-varying constraints ($x_t \in \mathcal{X}_T$) where we incorporate predictions for the cost *and* constraint vectors.

From a network management perspective, the availability of predictions is common in wireless networks and (mobile) computing systems and several works use deep learning or other function approximators [27] to predict f_τ, g_τ for $\tau \geq t$ and select a better x_t [28], [29]. And predictions have been previously included in stochastic optimization algorithms [30], [31], which however assume the requests and system perturbations are stationary. Similarly, prior works using predictions in online learning [32], [33], [34], do not adapt to the predictions' quality nor consider budget or other time-varying constraints, i.e., operate w.r.t. a fixed set $\mathcal{X}$. If a learner has access to perfect predictions, then it might be possible to use them directly in order to select a better action.[1] Nonetheless, the problem becomes fundamentally different when the predictions have unknown and/or time-varying accuracy. This, more general setting requires the learner to adapt to predictions. To that end, we focus here on the richer constrained OCO problem where $\{x_t\}$ are also subject to *time-varying* budget constraints ($x \in \mathcal{X}_T$) for an *unknown* horizon T. We place *no assumptions on the predictions accuracy*, the geometry of set $\mathcal{X}_T$ or the functions $\{f_t, g_t\}$, beyond being convex.

Technically, our approach benefits from a novel "lazy" update that aggregates all previous cost and constraint vectors and uses *data-driven* steps that adapt to prediction errors. In particular, we build on FTRL, cf. [35], which we extend with time-varying accumulated constraints — a result of independent interest. Previous greedy-based algorithms for time-varying budget constraints and benchmarks in $\mathcal{X}_T$ include [8], [10] which achieve $\mathcal{R}_T = \mathcal{O}(\sqrt{T})$, $\mathcal{V}_T =$

TABLE I
$x^\star$ BELONGS IN $\mathcal{X}$ AND SATISFIES $g_t(x^\star) \preceq 0, \forall t$. FOR THE ALGORITHMS WITH TUNABLE BOUNDS WE PRESENT THE BEST ACHIEVABLE W.R.T. $\mathcal{R}_T$, WHILE [6] USES DYNAMIC REGRET AS [40], [41], AND [39]

Paper	Regret	Const.	Conditions
[9]	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(\sqrt{T})$	Slater; strongly cvx f_t, g_t
[11]	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(\sqrt{T})$	Slater; i.i.d. $\{g_t(x)\}_t$
[16]	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(\sqrt{T})$	Slater; $g_t(x) = g(x) + b_t$
[10]	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(T^{\frac{3}{4}})$	L_f, L_g, F, G, T known
[6]	$\mathcal{O}(T^{\frac{2}{3}})$	$\mathcal{O}(T^{\frac{2}{3}})$	$\|g_t(x) - g_{t-1}(x)\| \leq \epsilon$; T known
[8]	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(T^{\frac{3}{4}})$	Slater; T known
LLP	$\mathcal{O}(T^{\frac{3-\beta}{4}})$	$\mathcal{O}(T^{\frac{1+\beta}{2}})$	no predictions; $\beta \in [\frac{1}{2}, 1)$
	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(\sqrt{T})$	$-\mathcal{R}_T = \mathcal{O}(\sqrt{T})$
	$\mathcal{O}(1)$	$\mathcal{O}(T^{\frac{1+\beta}{2}})$	perfect predictions; $\beta \in [0, 1)$
	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(T^{\frac{5}{8}})$	no predict.; $g_t(x) = g(x) + b_t$

$\mathcal{O}(T^{3/4})$ assuming fixed and known horizon[2] T; [16] that offers $\mathcal{R}_T, \mathcal{V}_T = \mathcal{O}(\sqrt{T})$ but confines the constraints to be linearly-perturbed; [37] which studies a similar setup with specific switching-actions constraints; [11], [38] with $\mathcal{R}_T, \mathcal{V}_T = \mathcal{O}(\sqrt{T})$ that restrict the constraints to be i.i.d. stochastic or non-positive over a common subset space; and [6], [39], [40], [41] which achieve sublinear bounds w.r.t. t-slot benchmark $\{x_t^\star\}_t$ (*dynamic regret*) under additional assumptions on the variability of the functions, and by knowing several problem parameters. We note that dynamic regret bounds, although more refined, they cannot attain sublinear rates in the general case. We summarize how our work compares to prior works in Table I.

It is important to stress that none of those approaches can benefit from predictions, and hence their performance does not improve even if the costs and constraint functions are predictable. On the contrary, our approach ensures improved learning performance whenever good predictions are available.

B. Contributions

We study the general constrained OCO problem where in round t our algorithm, which we name LLP (*Lazy Lagrangians with Predictions*), has access to all prior cost gradients $\{\nabla f_i(x_i)\}_{i=1}^{t-1}$ and constraints $\{g_i(x)\}_{i=1}^{t-1}$, and receives predictions $\tilde{g}_t(\tilde{x}_t)$, $\nabla \tilde{f}_t(\tilde{x}_t)$ and $\tilde{g}_t(\cdot)$. After selecting x_t, LLP incurs cost $f_t(x_t)$ and violation $g_t(x_t)$, and the process repeats in the next round. Our first result, Theorem 1, presents the regret and constraint violation bounds and demonstrates how they benefit from predictions. Theorem 2 characterizes the (tunable) growth rates of the bounds and exhibits their dependency on the accumulated prediction errors. Theorem 3 and Lemma 3 present the respective bounds when LLP employs fully-linearized cost and constraint functions and non-proximal regularizers, cf. [35], in order to reduce its computation and memory requirements. For this linearized version, it suffices to have gradient predictions $\nabla \tilde{g}_t(\tilde{x}_t)$ instead of $\tilde{g}_t(\cdot)$. Indeed, in some practical problems it might be easier to obtain such

---

[1]E.g., one could use a simple iteration $x_t = \arg\min_{\tilde{g}_t(x) \leq 0} \tilde{f}_t(x)$, which however will yield arbitrarily bad performance for imperfect predictions.

[2]We note that the doubling trick cannot be trivially applied in OCO with budget constraints, as it deteriorates the bounds; see discussion in [36].

predictions (single vectors) compared to predictions for the entire constraint function; but we note that this is not always the case[3]; LLP can handle both scenarios. Finally, Lemma 4 presents LLP's performance for linearly-perturbed constraints, a special but important case that was studied in [16].

The performance of LLP is summarized in Table I. It achieves $\mathcal{R}_T = \mathcal{O}(T^{\frac{3-\beta}{4}})$ and $\mathcal{V}_T = \mathcal{O}(T^{\frac{1+\beta}{2}})$ for worst-case (or, no) predictions, which are tunable through parameter $\beta \in [1/2, 1)$. For instance, with $\beta = 1/2$, we obtain $\mathcal{R}_T = \mathcal{O}(T^{5/8})$ and $\mathcal{V}_T = \mathcal{O}(T^{3/4})$, that are further reduced to $\mathcal{R}_T, \mathcal{V}_T = \mathcal{O}(\sqrt{T})$ when $-\mathcal{R}_T = \mathcal{O}(\sqrt{T})$, i.e., when $\{x_t\}$ does not outperform $x^\star$ by more than that. With perfect predictions, LLP achieves $\mathcal{R}_T = \mathcal{O}(1)$, $\mathcal{V}_T = \mathcal{O}(T^{\frac{1+\beta}{2}})$ which are tunable via $\beta \in [0, 1)$; while for linearly-perturbed constraints (as in [16]) LLP ensures $\mathcal{R}_T = \mathcal{O}(\sqrt{T})$ and $\mathcal{V}_T = \mathcal{O}(T^{5/8})$. These results improve previously-known bounds for the general OCO problem with time-varying budget constraints, i.e., without imposing assumptions such as strong convexity of functions and domains, or knowing the horizon T. And they include as special cases the benchmarks with static or stationary constraints of [11], [12], [13], [14], and [36]. Importantly, unlike *all prior* constrained-OCO algorithms, the constant factors of $\mathcal{R}_T$ and $\mathcal{V}_T$ shrink proportionally to the predictions' accuracy. This is, probably, the most important advantage of LLP compared to all prior works which, even though in some special cases attain better theoretical bounds, they cannot benefit from predictions.

We believe these results pave the road for extending the seminal network utility maximization (NUM) framework [42], [43] with robust learning techniques which seamlessly encompass any available predictions of unknown quality.

C. Assumptions and Notation

We write $\{x_t\}$ for a sequence of vectors and use subscripts to index them; $\| \cdot \|$ denotes the Euclidean (ℓ_2) norm and $[x]_\mathcal{X}$, $[x]_+$ the ℓ_2-projection of x on sets $\mathcal{X}$ and $\mathbb{R}_+^N$. We use the index function $\mathbf{I}_\mathcal{X}(x) = 0$ if $x \in \mathcal{X}$ and $\mathbf{I}_\mathcal{X} = \infty$, otherwise. Vector c_t denotes the gradient $\nabla f_t(x_t)$ of f_t or an element of its subdifferential $\vartheta f_t(x_t)$ if it is non-differentiable; and $\nabla g_t(x)$ denotes the Jacobian of the vector-valued constraint. We use the shorthand notation $c_{1:t}$ for $\sum_{i=1}^t c_i$, and $\tilde{a}_t$ for the prediction of some vector (or, function) a_t.

The analysis requires the following basic assumptions.
A1. The set $\mathcal{X} \subset \mathbb{R}^N$ is convex and compact, and it holds $\|x\| \leq D$, $\forall x \in \mathcal{X}$.
A2. Functions $f_t, g_t^{(j)} : \mathcal{X} \mapsto \mathbb{R}$, $\forall t, j \leq d$, are convex and Lipschitz with constants $L_{f_t} \leq L_f$, $L_{g_t^j} \leq L_g$. Since $\mathcal{X}$ is compact it holds $|f_t(x)| \leq F$, $\|g_t(x)\| \leq G$, $\forall t, x \in \mathcal{X}$.
A3. Predictions $\tilde{c}_t, \tilde{g}_t(\cdot)$ and $\tilde{g}_t(\tilde{x}_t)$ are known at t.
A4. The prediction errors $\varepsilon_t \triangleq c_t - \tilde{c}_t$, $\delta_t \triangleq \nabla g_t(x_t) - \nabla \tilde{g}_t(x_t)$ are bounded: $\|\varepsilon_t\| \leq E_m$, $\|\delta_t\| \leq \Delta_m$, $\forall x_t \in \mathcal{X}$; and it holds $\|\tilde{c}_t\| \leq L_f$, $\|\tilde{g}_t(x)\| \leq G$, $\forall t, x \in \mathcal{X}$.

---

[3]As an example, consider the problem where $g_t(x)$ is the energy spent due to wireless transmission at rate x. Then, predicting $\tilde{g}_t(\cdot)$ requires simply to know the channel state, while predicting $\nabla \tilde{g}_t(\tilde{x}_t)$ requires *also* to know the actual transmitted rate in the next slot $\tilde{x}_t$.
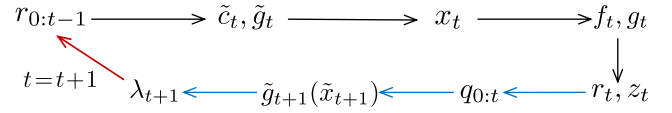

Fig. 2.   Key steps, timing and predictions of LLP.

D. Paper Organization

Section II introduces the LLP algorithm and the regret and constraint violation bounds. Section III presents the adaptive multi-step and characterizes the convergence rate of LLP, with special focus to the case of perfect predictions and worst-case (or no) predictions. Section IV modifies LLP for linearized constraints and non-proximal updates, and Sec. V derives the performance bounds for the special case of linearly-perturbed constraints. We conclude in Sec. VI. The paper is accompanied by an appendix, Sec. III, that includes the remaining proofs, explanatory figures, and numerical examples.

II. THE LLP ALGORITHM

Our approach is inspired by saddle-point methods that perform min-max operations on a convex-concave Lagrangian. Starting from the t-round problem:

$$\min_{x \in \mathcal{X}} \ f_t(x) \quad \text{s.t.} \ \ g_t(x) \preceq 0,$$

we introduce the dual variables $\lambda \in \mathbb{R}_+^d$ by relaxing $g_t(x) \preceq 0$, and define the *regularized* Lagrangian:

$$\mathcal{L}_t(x, \lambda) = r_t(x) + c_t^\top x + \lambda^\top g_t(x) - q_t(\lambda), \quad (3)$$

where we linearized $f_t(x)$. Function $r_t : \mathcal{X} \mapsto \mathbb{R}$ is a proximal[4] primal regularizer and $q_t : \mathbb{R}_+^d \mapsto \mathbb{R}$ a non-proximal dual regularizer. We also set $\mathcal{L}_0(x, \lambda) = r_0(x) - q_0(\lambda)$.

We coin the term *Lazy Lagrangians with Predictions* (LLP) for our Algorithm, which proceeds as follows, Fig. 2. In each round t, LLP uses observations $\{c_i\}_{i=1}^{t-1}$, $\{g_i\}_{i=1}^{t-1}$, dual variables $\{\lambda_i\}_{i=1}^t$, and predictions $\tilde{c}_t, \tilde{g}_t(\cdot)$ to perform an *optimistic* FTRL update:

$$x_t = \arg\min_{x \in \mathcal{X}} \left\{ \sum_{i=0}^{t-1} \mathcal{L}_i(x, \lambda_i) + \tilde{c}_t^\top x + \lambda_t^\top \tilde{g}_t(x) \right\}, \quad (4)$$

which induces cost $f_t(x_t)$ and constraint violation $g_t(x_t)$. After the t-round information c_t and $g_t(\cdot)$ is revealed, LLP calculates the *prescient* action:

$$z_t = \arg\min_{x \in \mathcal{X}} \left\{ \sum_{i=0}^t \mathcal{L}_i(x, \lambda_i) \right\}, \quad (5)$$

and uses prediction $\tilde{g}_{t+1}(\tilde{x}_{t+1})$ to update the duals:

$$\lambda_{t+1} = \arg\max_{\lambda \in \mathbb{R}_+^d} \left\{ \sum_{i=0}^t \mathcal{L}_i(z_i, \lambda) + \lambda^\top \tilde{g}_{t+1}(\tilde{x}_{t+1}) \right\} \quad (6)$$

where note the use of $\{z_t\}$ instead of $\{x_t\}$. The process then repeats in the next round.

LLP has key differences from previous constrained OCO algorithms. These stem from the usage of *lazy* as opposed to *greedy* updates in the primal and dual iteration, where instead of using x_{t-1} and λ_t to decide x_t and λ_{t+1} respectively,

---

[4]Regularizer $r_t(x)$ is called *proximal* with reference to an algorithm that yields $\{x_t\}$ if $x_t \in \arg\min_{x \in \mathcal{X}} r_t(x)$; and *non-proximal* otherwise [35].

Algorithm 1 Lazy Lagrangians With Predictions (LLP)

Input: $x_0 \in \mathcal{X}$, $\lambda_1 = 0$, $r_t(x)$, $q_t(\lambda)$ with (7), (8).
for $t = 1, 2, \ldots$ **do**
   ○ Calculate $r_{0:t-1}$ and decide x_t using (4)
   ● Pay cost $f_t(x_t)$ and violation $g_t(x_t)$
   ○ Calculate $r_{0:t}$ and decide z_t using (5)
   ● Receive predictions $\tilde{c}_{t+1}$, $\tilde{g}_{t+1}(\cdot)$, $\tilde{g}_{t+1}(\tilde{x}_{t+1})$
   ○ Calculate $q_{0:t}$ and decide λ_{t+1} using (6)
end for

---

we aggregate in a projection-free fashion all prior cost gradients and constraints. This approach can be traced back to lazy algorithms discussed in [1]; to *fictitious* (as opposed to *best response*) strategies in game theory [44]; and to FTRL algorithms [35] for problems with fixed constraints. However, to the best of our knowledge, this is the first time such lazy updates are used with time-varying budget constraints.

Performance. The regret and constraint violation are quantified using (1) and (2), respectively, with benchmark $x^\star \in \arg\min_{x \in \mathcal{X}_T} \sum_{t=1}^{T} f_t(x)$. The performance of LLP is shaped by the regularizers which adapt to predictions. In particular, we use the primal regularizers:

$$r_0(x) = \mathbf{I}_{\mathcal{X}}(x) \text{ and } r_t(x) = \sigma_t \|x - x_t\|^2 / 2, \ t \geq 1 \text{ with:}$$
$$\sigma_t = \sigma \left(\sqrt{h_{1:t}} - \sqrt{h_{1:t-1}} \right), \ h_t = \|\varepsilon_t + \lambda_t^\top \delta_t\|, \ \sigma > 0,$$
$$(7)$$

where $r_0(x)$ ensures that $x \in \mathcal{X}$; x_t is given by (4); and the regularization parameter σ_t accounts for the cost and constraint prediction errors, where the latter are modulated by the dual variables. The intuition for (7) is that we add regularization commensurate to the prediction errors; and the rationale for selecting this particular σ_t will be made clear below. On the other hand, we use the general dual regularizer:

$$q_0(\lambda) = \mathbf{I}_{\mathbb{R}_+^d}(\lambda) \text{ and } q_t(\lambda) = \phi_t \|\lambda\|^2 / 2, \ t \geq 1 \text{ with:}$$
$$\phi_t = (1/a_t) - (1/a_{t-1}), \ a_{t-1} \geq a_t > 0, \ \phi_0 = 1/a_0$$
$$(8)$$

where again $q_0(\lambda)$ ensures $\lambda \in \mathbb{R}_+^d$, and $\{a_t\}$ is the dual learning rate which, for the first Theorem, suffices to be non-increasing. Our first main result is the following.

Theorem 1: Under Assumptions (**A1**)-(**A4**) and with $\{r_t\}$ and $\{q_t\}$ satisfying (7) and (8), LLP ensures $\forall x^\star \in \mathcal{X}_T$:

(**a**): $\mathcal{R}_T \leq B_T \triangleq 2 \left(\sigma D^2 + \dfrac{L_f}{\sigma} \right) \sqrt{h_{1:T}} + \displaystyle\sum_{t=1}^{T} \dfrac{\xi_t^2}{\phi_{0:t-1}}$

(**b**): $\mathcal{V}_T \leq \sqrt{2\phi_{0:T-1}(B_T - \mathcal{R}_T)} + \dfrac{2L_g}{\sigma} \sqrt{h_{1:T}}$

where $\xi_t = \|g_t(z_t) - \tilde{g}_t(\tilde{x}_t)\|$, $h_t = \|\varepsilon_t + \lambda_t^\top \delta_t\|$, $\varepsilon_t = c_t - \tilde{c}_t$, $\delta_t = \nabla g_t(x_t) - \nabla \tilde{g}_t(x_t)$.

Discussion. We observe from Theorem 1 the effect of predictions on the bounds of $\mathcal{R}_T$ and $\mathcal{V}_T$, which diminish

proportionally to their accuracy. The bound of $\mathcal{R}_T$ is further reduced if we set $\sigma = \sqrt{L_f/D}$ (when these parameters are known), and settles to zero for perfect predictions, i.e. when $x_t = z_t, \forall t$, and:

$$\varepsilon_t = 0, \quad \delta_t = 0, \quad \xi_t = 0, \ \forall t \leq T,$$

while the same is not true for $\mathcal{V}_T$. Moreover, this theorem reveals the *tension* between $\mathcal{R}_T$ and $\mathcal{V}_T$. Indeed, observe that $-\mathcal{R}_T$ appears in the bound of $\mathcal{V}_T$ which means that when $\{x_t\}$ outperforms $x^\star$, we might incur higher constraint violation.

Observing the steps of the algorithm, we can see that LLP requires predictions for the next gradient $\tilde{c}_t = \nabla \tilde{f}_t(\tilde{x}_t)$, next cost function $\tilde{g}_t(\cdot)$ and next constraint point $\tilde{g}_t(\tilde{x}_t)$. It is important to note the timing of these predictions. Updating x_t requires $\tilde{c}_t$ and $\tilde{g}_t(\cdot)$ and access to regularizers $r_{0:t-1}$ which are calculated using the prediction errors up to slot $t - 1$. Knowledge of $\tilde{c}_t$ is the standard prediction that all prior works employ, e.g., see [22], [24] and references therein. On the other hand, since we have not linearized the constraint function, the respective predictions involve function $\tilde{g}_t(\cdot)$ and its next-round value $\tilde{g}_t(\tilde{x}_t)$. In Section IV we present a version of LLP where we linearize the constraints and hence use only gradient predictions for the constraints – similarly to cost functions.

The complexity of LLP is comparable to its greedy-based counterparts — sans the additional prescient update (5) — i.e., it requires the solution of strongly convex problems and a closed-form iteration for the dual update. Finally, it is worth emphasizing that the impossibility result of [7] holds even if $\{f_t, g_t\}$ are revealed before $\{x_t\}$ is selected, as stated in the next Lemma that is proved in the Appendix.

Lemma 1: No online algorithm can achieve concurrently sublinear regret and constraint violation, $\mathcal{R}_T = o(T)$, $\mathcal{V}_T = o(T)$, w.r.t. $x^\star \in \mathcal{X}_T^{\max}$, even if the algorithm selects $\{x_t\}$ with knowledge of $\{f_t, g_t\}$.

This result exhibits the challenges in tackling constrained OCO problems with time-varying budget constraints, when using the ideal set of benchmarks $\mathcal{X}_T^{max}$; and reveals that using predictions, even if they are known to be perfect, does not suffice to escape this limitation.

A. Regret Bound

Our strategy is to derive a regret bound w.r.t. prescient actions $\{z_t\}$, and then use the distance of $\{z_t\}$ from $\{x_t\}$ to prove Theorem 1(**a**). We will use the following Lemma that is proved in the Appendix.

Lemma 2: For the actions $\{x_t\}$ and $\{z_t\}$ obtained by (4) and (5), respectively, it holds:

$$\|x_t - z_t\| \leq \frac{\|\varepsilon_t + \lambda_t^\top \delta_t\|}{\sigma_{1:t}},$$

where $\epsilon_t = c_t - \tilde{c}_t$, $\delta_t = \nabla g_t(x_t) - \nabla \tilde{g}_t(x_t)$, σ_t from (7).

Now, to prove Theorem 1(**a**) we apply [26, Theorem 2] to the dual update formula[5] (6) with functions $\zeta(\lambda) =$

---

[5]Update (6) runs over the unbounded set $\mathbb{R}_+^d$, unlike the compact set in [26]. However, that result still holds here and suffices as we set $\lambda = 0$ to get $q_{0:T-1}(0) = 0$; see discussion and Lemma 5 in Appendix.

$-\lambda^\top g_t(z_t), \forall t$ and gradient predictions $-\tilde{g}_{t+1}(\tilde{x}_{t+1})$, to get:

$$-\sum_{t=1}^{T} \lambda_t^\top g_t(z_t) + \sum_{t=1}^{T} \lambda^\top g_t(z_t)$$

$$\leq q_{0:T-1}(\lambda) + \sum_{t=1}^{T} \|g_t(z_t) - \tilde{g}_t(\tilde{x}_t)\|_{(t-1),\star}^2 \quad (a)$$

$$= q_{0:T-1}(\lambda) + \sum_{t=1}^{T} \frac{\|g_t(z_t) - \tilde{g}_t(\tilde{x}_t)\|^2}{\phi_{0:t-1}}, \quad \forall \lambda \in \mathbb{R}_+^d, \quad (9)$$

where (a) holds since the dual regularizer is 1-strongly-convex w.r.t. $\|\lambda\|_{(t-1)} = \sqrt{\phi_{0:t-1}}\|\lambda\|$ which has dual norm $\|\lambda\|_{(t-1),\star} = \|\lambda\|/\sqrt{\phi_{0:t-1}}$.

Adding $\sum_{t=1}^{T}\left[r_t(z_t) + c_t^\top z_t\right]$ to both sides of (9):

$$\sum_{t=1}^{T} \left[r_t(z_t) + c_t^\top z_t - \lambda_t^\top g_t(z_t) + \lambda^\top g_t(z_t)\right]$$

$$\leq q_{0:T-1}(\lambda) + \sum_{t=1}^{T} \left[r_t(z_t) + c_t^\top z_t + \frac{\xi_t^2}{\phi_{0:t-1}}\right], \quad (10)$$

and dropping the first non-negative sum in the LHS, setting $\lambda = 0$ to get $q_{0:T-1}(0) = 0$, and adding/subtracting $\phi_t\|\lambda_t\|^2/2$ in the RHS so as to build $\mathcal{L}_t(z_t, \lambda_t), \forall t \leq T$, we arrive at:

$$\sum_{t=1}^{T} c_t^\top z_t \leq \sum_{t=1}^{T}\left(\mathcal{L}_t(z_t, \lambda_t) + \frac{\phi_t\|\lambda_t\|^2}{2} + \frac{\xi_t^2}{\phi_{0:t-1}}\right)$$

$$\overset{(a)}{\leq} \sum_{t=1}^{T}\left(\mathcal{L}_t(x^\star, \lambda_t) + \frac{\phi_t\|\lambda_t\|^2}{2} + \frac{\xi_t^2}{\phi_{0:t-1}}\right)$$

$$\overset{(b)}{\leq} 2D^2\sigma_{1:T} + \sum_{t=1}^{T}\left(c_t^\top x^\star + \frac{\xi_t^2}{\phi_{0:t-1}}\right),$$

where (a) stems from the Be-the-Leader (BTL) Lemma [45, Lemma 3.1] applied with $x^\star \in \mathcal{X}_T \subseteq \mathcal{X}$ to (5); and (b) from expanding $\mathcal{L}_t(x^\star, \lambda_t)$, using $g_t(x^\star) \preceq 0$ and $\|x^\star - x_t\|^2 \leq 4D^2, \forall t$. Adding $\sum_{t=1}^{T} c_t^\top x_t$ to both sides and rearranging:

$$\sum_{t=1}^{T} c_t^\top(x_t - x^\star) \leq 2D^2\sigma_{1:T} + \sum_{t=1}^{T} \frac{\xi_t^2}{\phi_{0:t-1}} + c_t^\top(x_t - z_t) \quad (11)$$

The last term can be upper-bounded using the Cauchy-Schwarz inequality and Lemma 2, i.e.:

$$\sum_{t=1}^{T} c_t^\top(x_t - z_t) \leq \sum_{t=1}^{T} \frac{\|c_t\|\|\varepsilon_t + \lambda_t^\top \delta_t\|}{\sigma_{1:t}} \overset{(\text{Assump. } \mathbf{A2})}{\leq}$$

$$\frac{L_f}{\sigma} \sum_{t=1}^{T} \frac{\|\varepsilon_t + \lambda_t^\top \delta_t\|}{\sqrt{\sum_{i=1}^{t} \|\varepsilon_i + \lambda_i^\top \delta_i\|}} \overset{(a)}{\leq} \frac{2L_f}{\sigma} \sqrt{\sum_{t=1}^{T} \|\varepsilon_t + \lambda_t^\top \delta_t\|}$$

where in (a) we used [46, Lemma 3.5], and this was made possible due to the specific formula of the regularization parameter σ_t. Replacing in (11) and using $\mathcal{R}_T \leq c_t^\top(x_t - x^\star)$, we eventually get:

$$\mathcal{R}_T \leq 2D^2\sigma_{1:T} + \frac{2L_f}{\sigma}\sqrt{\sum_{t=1}^{T} \|\varepsilon_t + \lambda_t^\top \delta_t\|} + \sum_{t=1}^{T} \frac{\xi_t^2}{\phi_{0:t-1}} \Rightarrow$$

$$\mathcal{R}_T = 2\left(\sigma D^2 + \frac{L_f}{\sigma}\right)\sqrt{h_{1:T}} + \sum_{t=1}^{T} \frac{\xi_t^2}{\phi_{0:t-1}},$$

which concludes the proof for Theorem 1**(a)**.

B. Constraint Violation Bound

To prove Theorem 1**(b)**, we start from (10) where we drop again the non-negative $\sum_{t=1}^{T} r_t(z_t)$ in the LHS, add and subtract the term $\sum_{t=1}^{T} \phi_t\|\lambda_t\|^2/2, \forall t$, and rearrange to get:

$$\lambda^\top \sum_{t=1}^{T} g_t(z_t) - q_{0:T-1}(\lambda)$$

$$\leq \sum_{t=1}^{T}\left(\mathcal{L}_t(z_t, \lambda_t) - c_t^\top z_t + \frac{\phi_t\|\lambda_t\|^2}{2} + \frac{\xi_t^2}{\phi_{0:t-1}}\right)$$

$$\leq \sum_{t=1}^{T}\left(\mathcal{L}_t(x^\star, \lambda_t) - c_t^\top z_t + \frac{\phi_t\|\lambda_t\|^2}{2} + \frac{\xi_t^2}{\phi_{0:t-1}}\right)$$

where again we applied BTL to $\mathcal{L}_t(z_t, \lambda_t)$. Next, expanding $\mathcal{L}_t(x^\star, \lambda_t)$, using $q_{0:T-1}(\lambda) = \phi_{0:T-1}\|\lambda\|^2/2$, $g_t(x^\star) \preceq 0, \forall t$ and $r_{0:T}(x^\star) \leq 4D^2\sigma_{1:T}$, we get:

$$\lambda^\top \sum_{t=1}^{T} g_t(z_t) - \frac{\phi_{0:T-1}}{2}\|\lambda\|^2$$

$$\leq 2D^2\sigma_{1:T} + \sum_{t=1}^{T}\left(c_t^\top(x^\star - z_t) + \frac{\xi_t^2}{\phi_{0:t-1}}\right)$$

$$\leq 2D^2\sigma_{1:T} + \sum_{t=1}^{T}\left(c_t^\top(x^\star - x_t) + c_t^\top(x_t - z_t) + \frac{\xi_t^2}{\phi_{0:t-1}}\right)$$

$$\leq 2D^2\sigma_{1:T} + \frac{L_f}{\sigma}\sqrt{\sum_{t=1}^{T} \|\varepsilon_t + \lambda_t^\top \delta_t\|} + \sum_{t=1}^{T} \frac{\xi_t^2}{\phi_{0:t-1}} - R_T$$

$$\Rightarrow \lambda^\top \sum_{t=1}^{T} g_t(z_t) - \frac{\phi_{0:T-1}}{2}\|\lambda\|^2 \leq B_T - \mathcal{R}_T, \quad \forall \lambda \in \mathbb{R}_+^d. \quad (12)$$

For the LHS of (12), we can use the result:

$$\frac{\left\|\left[\sum_{t=1}^{T} g_t(z_t)\right]_+\right\|^2}{2\phi_{0:T-1}} = \sup_{\lambda \in \mathbb{R}_+}\left[\lambda^\top \sum_{t=1}^{T} g_t(z_t) - \frac{\phi_{0:T-1}}{2}\|\lambda\|^2\right]$$

and if we denote with V_T^z the LHS norm and replace this in (12), we obtain:

$$V_T^z \leq \sqrt{2\phi_{0:T-1}(B_T - \mathcal{R}_T)} = \sqrt{\frac{2(B_T - \mathcal{R}_T)}{a_{T-1}}}. \quad (13)$$

Lastly, we define $w_t = \nabla g_t(x_t)^\top(z_t - x_t)$ and write:

$$\mathcal{V}_T = \left\|\left[\sum_{t=1}^{T} g_t(x_t) + w_t - w_t\right]_+\right\|$$

$$\overset{(a)}{\leq} \left\|\left[\sum_{t=1}^{T} g_t(x_t) + w_t\right]_+\right\| + \left\|[-w_{1:T}]_+\right\|$$

$$\overset{(b)}{\leq} \left\|\left[\sum_{t=1}^{T} g_t(z_t)\right]_+\right\| + \|-w_{1:T}\| \overset{(c)}{\leq} V_T^z + \frac{2L_g}{\sigma}\sqrt{h_{1:T}}$$

$$(14)$$

where (a) uses the identity $\|[\chi + \upsilon]_+\| \le \|[\chi]_+\| + \|[\upsilon]_+\|$; (b) the convexity of g_t; and (c) the Cauchy-Schwarz inequality, $\|\nabla g_t(x_t)\| \le G$, and Lemma 2. This concludes the proof.

The next section characterizes the convergence rates of the bounds, focusing on two special cases: when the predictions are perfect and when we have worst-case (or no) predictions.

III. CONVERGENCE RATES

We start by specifying the dual learning rate $\{a_t\}$. The rationale for selecting the primal regularizer was made clear in the proof of Theorem 1; here, we refine (8) in a way that ensures the desirable sublinear regret and constraint violation growth rates. In detail, we will be using:

$$a_t = \frac{a}{\max\left\{\sqrt{4G^2 + \sum_{i=1}^t \xi_i^2}, \ t^\beta\right\}}, \quad \beta \in [0,1). \quad (15)$$

This *multi-step* combines the typical time-adaptive step appearing in online gradient-descent algorithms [1] with a data-adaptive step that accounts for the prediction errors. This ensures that a_t will induce enough regularization when the predictions' quality are not satisfactory, but will keep diminishing even in the case of perfect predictions – a condition that is necessary in order to tame the growth rate of the dual vector. Finally, note that the term $4G^2 \ge \xi_t^2, \forall t$ corrects the off-by-one regularizer of the non-proximal dual update.

Before we analyze the convergence for the cases of perfect and worst predictions, it is important to emphasize that in each round t, LLP has at its disposal all the necessary information to calculate a_t. In particular, a_t is used to update the dual vector λ_{t+1} *after* the cost and constraint functions, f_t and g_t, have been revealed, and the prescient vector z_t is calculated. Hence, we know ξ_t before performing update (6). Furthermore, we stress that the analysis below does not make any assumptions on the apparatus which creates the predictions; in fact, our framework is orthogonal to how the predictions are being created and oblivious to their quality: if the predictions are accurate, LLP learns to trust them, and if they are inaccurate, LLP gradually discards them.

A. Perfect Predictions

The next Corollary to Theorem 1 describes the regret and constraint violation bounds for perfect predictions.

Corollary 1: [Perfect Predictions] When $\epsilon_t = \delta_t = \xi_t = 0, \forall t$, $\beta \in [0,1)$, Algorithm LLP ensures:

$$\mathcal{R}_T = \mathcal{O}(1), \qquad \mathcal{V}_T = \mathcal{O}(T^{\frac{1+\beta}{2}})$$

Indeed, for perfect predictions holds $h_{1:T} = 0$ independently of the value of $\{\lambda_t\}$, while the second term in the bound of $\mathcal{R}_T$ can be written (detailed derivation in Sec. III-C:

$$\sum_{t=1}^T \frac{\xi_t^2}{\phi_{0:t-1}} = \sum_{t=1}^T a_{t-1}\xi_t^2 = \min\left\{2a\sqrt{\sum_{t=1}^T \xi_t^2}, \frac{4aG^2}{1-\beta}T^{1-\beta}\right\}$$

which diminishes to zero when $\xi_t = 0, \forall t$. This manifests the advantage of this doubly-adaptive dual step which creates

a bound similar to those in OCO problems without budget constraints, see [26]. Furthermore, the step is simplified to $a_t = a/t^\beta$ which remains bounded. Hence $\mathcal{R}_T = B_T = \mathcal{O}(1)$, and if we substitute B_T in $\mathcal{V}_T$, we get:

$$\mathcal{V}_T \le \sqrt{\frac{-2\mathcal{R}_T}{a_{T-1}}} \le \sqrt{\frac{-2\mathcal{R}_T T^\beta}{a}} = \mathcal{O}\left(T^{\frac{1+\beta}{2}}\right) \quad (16)$$

Here, we can set $\beta = 0$ to get $\mathcal{V}_T = \mathcal{O}(\sqrt{T})$, and reduce the constant factor of $\mathcal{V}_T$ further by increasing[6] a.

Furthermore, note that when the regret is non-negative, i.e., when the sequence $\{x_t\}_t$ does not outperform $x^\star$, then we get $\mathcal{V}_T \le 0$ for any value of β. And, more generally, if there is a non-trivial bound for the negative regret, i.e., $-\mathcal{R}_T = \mathcal{O}(T^b)$ with $b < 1$, then the bound of the constraint violation is improved in a commensurate amount, namely $\mathcal{V}_T = \mathcal{O}(T^{\frac{b+\beta}{2}})$. Finally, it is worth stressing that, even when the predictions are perfect, the algorithm still needs to learn to trust them gradually, as their quality is a priori unknown.

B. Worst-Case Predictions

On the other hand, when we do not have any predictions at our disposal, or when these are as far as possible from the actual data, then the dual multi-step induces more regularization using the observed prediction errors. The performance of LLP in this scenario is captured by the following theorem.

> *Theorem 2:* Under Assumptions (**A1**)-(**A4**), with regularizers $\{r_t\}$, $\{q_t\}$ satisfying (7), (8), (15), and for worst-case predictions $\epsilon_t = E_m, \delta_t = \Delta_m, \xi_t = 2G, \forall t$, LLP ensures:
>
> $$\mathcal{R}_T = \mathcal{O}(T^{\frac{5}{8}}), \qquad \mathcal{V}_T = \mathcal{O}(T^{\frac{3}{4}}) \quad \text{when} \quad \beta < 1/2$$
> $$\mathcal{R}_T = \mathcal{O}(T^{\frac{3-\beta}{4}}), \qquad \mathcal{V}_T = \mathcal{O}(T^{\frac{1+\beta}{2}}) \quad \text{when} \quad \beta \ge 1/2$$

We see that the growth rates are tunable by parameter $\beta \in [0,1)$. For example, by setting $\beta = 2/3$ we obtain $\mathcal{R}_T = \mathcal{O}(T^{7/12})$ and $\mathcal{V}_T = \mathcal{O}(T^{5/6})$; while with $\beta = 3/5$ it is $\mathcal{R}_T = \mathcal{O}(T^{3/5})$ and $\mathcal{V}_T = \mathcal{O}(T^{4/5})$. These bounds improve the best-known results for the *general* constrained-OCO problem, while being comparable with results that consider special cases, such as knowing the (a priori fixed) time horizon T [10] or having only linearly-perturbed constraints [16].

What prevents the LLP bounds from improving further is the term $-\mathcal{R}_T$ that appears in $\mathcal{V}_T$. While we have used in the analysis the worst-case $-\mathcal{R}_T = \mathcal{O}(T)$, it is important to note that when $-\mathcal{R}_T = \mathcal{O}(\sqrt{T})$, i.e., when LLP does not outperform the benchmark by more than $\sqrt{T}$, then for perfect predictions we achieve $\mathcal{R}_T = \mathcal{O}(1)$, $\mathcal{V}_T = \mathcal{O}(T^{1/4})$ (setting $\beta = 0$), and for worst-case predictions it is $\mathcal{R}_T, \mathcal{V}_T = \mathcal{O}(\sqrt{T})$.

---

[6]Had we known the horizon T, as assumed in [10], we can set $a = T$ to obtain $\mathcal{V}_T = \mathcal{O}(1)$. However, this selection endangers increasing $\mathcal{R}_T$ when predictions are imperfect.

On the other hand, if LLP does outperform the benchmark consistently ($\forall T$) by at least $\Omega(\sqrt{T})$, we achieve negative regret and $\mathcal{V}_T = O(T^{2/3})$. The general case, for which the above two Corollaries hold, is when the sample path is such that LLP bounces above and below the performance of $\{x^\star\}$. A schematic description of these cases is included in the Appendix and the different achieved rates by LLP are summarized in Table I.

Concluding, it is worth discussing the inherent difficulties of the problem at hand. Namely, one might argue that we could directly apply the optimistic FTRL result of [26] (or our equivalent Lemma 5) both to the primal and to the dual update, and combine the results to bound $\mathcal{R}_T$ and $\mathcal{V}_T$. The interested reader, however, can verify that this straightforward strategy leads to much worse bounds. Our approach instead is to apply the optimistic FTRL only to the dual update and carefully reconstruct tighter bounds for the Lagrangian, while using a fixed-point iteration to find the exact (minimum) growth rate of $\|\lambda_T\|$. Moreover, unlike prior works such as [16] or [13], we update $\{x_t\}$ using $\{\lambda_t\}$ (instead of λ_{t-1}); and then update $\{\lambda_{t+1}\}$ using the newly calculated $\{x_t\}$. This strategy facilitates the inclusion of predictions as we only need to predict x_t and not λ_t. Besides, it is exactly this circular relation between the primal and dual variables which renders the inclusion of predictions in OCO with budget constraints fundamentally different from the respective OCO problem without time-varying constraints.

C. Proof of Theorem 2

Our strategy is to bound the growth rate of the dual vector norm and use it to bound $\mathcal{R}_T$ and $\mathcal{V}_T$. First, we define its minimum growth rate $k = \min\{\varphi : \|\lambda_t\| = \mathcal{O}(t^\varphi)\}$, and introduce $\Lambda_t = \max\{\|\lambda_i\| : i \leq t\}$, where $\Lambda_t \leq \Lambda_T = \mathcal{O}(T^k)$, $\forall t \leq T$. Using the closed-form solution[7] of (6), we can write:

$$\|\lambda_{t+1}\| = \left\| \left[a_t \left(\sum_{i=1}^t g_i(z_i) + \tilde{g}_{t+1}(\tilde{x}_{t+1}) \right) \right]_+ \right\|$$
$$\leq a_t \left(V_t^z + G \right) \quad (17)$$

where we used that $\|\tilde{g}_t(z)\|, \|g_t(z)\| \leq G$, $\forall t$, and the triangle inequality. Also, we can write:

$$h_{1:T} = \sum_{t=1}^T \left\| \epsilon_t + \lambda_t^\top \delta_t \right\|$$
$$\leq T(E_m + \Delta_m \Lambda_T) = \mathcal{O}(T^{k+1}) \quad (18)$$

We will use this bound and the definition of λ_t, which ties it to V_t^z, to find a smaller growth rate than the one we would get by directly bounding V_t^z in (17). Indeed, starting from (17) and using (13), we have:

$$\|\lambda_{T+1}\| \leq a_T \sqrt{2(B_T - R_T)/a_{T-1}} + a_T G \overset{a_{T-1} \geq a_T}{\Longrightarrow}$$
$$\|\lambda_{T+1}\| \leq \sqrt{2a_T(B_T - R_T)} + a_T G. \quad (19)$$

[7]Eq. (6) simplifies to $\lambda_{t+1} = \arg\min_{\lambda \in \mathbb{R}_+^d} \left\{ \frac{\|\lambda\|^2}{2a_t} - \lambda^\top v \right\}$ with $v = \tilde{g}_{t+1}(\tilde{x}_{t+1}) + \sum_{i=1}^t g_i(z_i)$.

Next, note that based on the definition of a_t and the fact that we consider worst-case predictions (which increase linearly with T), the following inequalities hold:

$$a_T \leq \frac{a}{\sqrt{4G^2 + \sum_{t=1}^T \xi_t^2}} = \mathcal{O}(T^{-1/2})$$
$$a_T \leq \frac{a}{T^\beta} = \mathcal{O}(T^{-\beta}).$$

Thus, it follows that:

$$a_T = \mathcal{O}(T^\theta), \quad \theta = \min\left\{ -\beta, -\frac{1}{2} \right\} \leq 0. \quad (20)$$

Similarly, the following inequalities hold:

$$\sum_{t=1}^T a_{t-1}\xi_t^2 \leq \sum_{t=1}^T \frac{a\xi_t^2}{\sqrt{\sum_{i=1}^t \xi_i^2}} \leq 2a\sqrt{\sum_{t=1}^T \xi_t^2} = \mathcal{O}(\sqrt{T})$$
$$\sum_{t=1}^T a_{t-1}\xi_t^2 \leq \sum_{t=1}^T a\frac{\xi_t^2}{t^\beta} \leq \frac{4aG^2}{1-\beta}T^{1-\beta} = \mathcal{O}(T^{1-\beta}),$$

where we used [46, Lemma 3.5], the identity $\sum_{t=1}^T t^{-\beta} \leq T^{1-\beta}/(1-\beta)$ (Lemma 6 in Appendix) and $\xi_t \leq 2G$, $\forall t$. Hence we obtain:

$$\sum_{t=1}^T a_{t-1}\xi_t^2 = \mathcal{O}(T^n), \quad n = \min\left\{ \frac{1}{2}, 1-\beta \right\} \leq 1. \quad (21)$$

Finally, replacing B_T in (19) with its definition from Theorem 1, we arrive at:

$$\|\lambda_T\| = \mathcal{O}\left(\max\left\{ T^{\frac{k+1+2\theta}{4}}, T^{\frac{n+\theta}{2}}, T^{\frac{1+\theta}{2}}, T^\theta \right\} \right), \quad (22)$$

where we used the worst case bound $-\mathcal{R}_T = \mathcal{O}(T)$ and $a_T G = \mathcal{O}(T^\theta)$. To find the dominant term, note that, since $\beta \in [0,1)$, it is $n \leq 1$, and hence $(1+\theta)/2 \geq (n+\theta)/2$, thus we omit the second term. Also, $\theta \leq 0$ and hence we can omit the last term; and finally, from (17) we observe that $k \leq 1+\theta \leq 1$, thus, the third term is larger than the first, and we conclude with $\|\lambda_T\| = \mathcal{O}(T^k) = \mathcal{O}(T^{(1+\theta)/2})$.

Having found the growth rate of $\|\lambda_T\|$ to be $k = (1+\theta)/2$, we use (18) and (21) to refine the bounds:

$$h_{1:T} = \mathcal{O}\left(T^{\frac{3+\theta}{2}} \right), \quad B_T = \mathcal{O}\left(\max\left\{ T^{\frac{3+\theta}{4}}, T^n \right\} \right) \quad (23)$$

and these conclude the proof of Theorem 2(**a**). For the constraint violation $\mathcal{V}_T$, observe first that $(1/a_T) =$

$$\frac{\max\left\{ \sqrt{G^2 + \sum_{t=1}^T \xi_t^2}, T^\beta \right\}}{a} = \mathcal{O}(T^\nu), \quad \nu = \max\left\{ \beta, \frac{1}{2} \right\}$$

and hence holds $\nu = -\theta$. Using this bound along with (23) and $-\mathcal{R}_T = \mathcal{O}(T)$, we get from Theorem 1(**b**):

$$\mathcal{V}_T = \mathcal{O}\left(\max\left\{ T^{\frac{3(1-\theta)}{8}}, T^{\frac{n-\theta}{2}}, T^{\frac{1-\theta}{2}}, T^{\frac{3+\theta}{4}} \right\} \right).$$

We conclude by noticing that:

$$\frac{1-\theta}{2} > \frac{3(1-\theta)}{8} \quad \text{and} \quad \frac{1-\theta}{2} \geq \frac{n-\theta}{2},$$

and observing that conditioning on the value of β, we get the bounds in Theorem 2.

IV. LESS COMPUTATIONS AND PREDICTIONS

We discuss next how to reduce the computation and memory requirements of LLP by using non-proximal primal regularizers; and the impact of linearizing the constraint functions on the required predictions.

A. LLP With Non-Proximal Regularizers

This new algorithm, LLP2, uses the same dual regularizer (8) and update (6), but the general primal regularizer:

$$r_0(x) = \mathbf{I}_{\mathcal{X}}(x) \text{ and } r_t(x) = \sigma_t\|x\|^2/2, \ \forall t \geq 1, \text{ with:}$$
$$\sigma_t = \sigma\left(\sqrt{h_{1:t} + \mu_{t+1}} - \sqrt{h_{1:t-1} + \mu_t}\right) \quad (24)$$

where $\mu_t \triangleq E_m + a_{t-1}Gt\Delta_m$. The new updates are:

$$x_t = \arg\min_{x\in\mathbb{R}^N}\left\{\sum_{i=0}^{t-1}\mathcal{L}_i(x,\lambda_i) + \tilde{c}_t^\top x + \lambda_t^\top \tilde{g}_t(x)\right\}, \quad (25)$$
$$z_t = \arg\min_{x\in\mathbb{R}^N}\left\{\sum_{i=0}^{t-1}\mathcal{L}_i(x,\lambda_i) + c_t^\top x + \lambda_t^\top g_t(x)\right\}, \quad (26)$$

where $\mathcal{L}_t(x,\lambda)$ is defined using $r_t(x)$ from (24), and note the off-by-one regularizer of (26) compared to (5).

These non-proximal regularizers facilitate solving for $\{x_t\}$ and $\{z_t\}$ since $r_{1:t}(x) = \sigma_{1:t}\|x\|^2/2$ involves only one quadratic term and can be represented in constant memory space, unlike $r_{1:t}(x) = \sum_{i=1}^{t}\sigma_i\|x - x_i\|^2/2$ that expands with time. On the other hand, non-proximal updates yield looser bounds, cf. [35], and require a new saddle-point analysis. Interestingly, they do not worsen the growth of LLP2, but do prevent it from achieving $\mathcal{R}_T = \mathcal{O}(1)$ for perfect predictions.

Theorem 3: Under (**A1**)-(**A4**) and with $\{r_t\}$ and $\{q_t\}$ satisfying (24) and (8), LLP2 ensures for every $x^\star \in \mathcal{X}_T$:

$$\mathcal{R}_T \leq \widehat{B}_T \triangleq 2\left(\sigma D^2 + \frac{L_f}{\sigma}\right)\sqrt{h_{1:T} + \mu_{T+1}} + \sum_{t=1}^{T}\frac{\xi_t^2}{\phi_{0:t-1}}$$

$$\mathcal{V}_T \leq \sqrt{2\phi_{0:T-1}\left(\widehat{B}_T - \mathcal{R}_T\right)} + \frac{2L_g}{\sigma}\sqrt{h_{1:T} + \mu_{T+1}}$$

where $\mu_T = E_m + a_{T-1}GT\Delta_m$.

The proof of Theorem 3 can be found in the Appendix.

Corollary 2: LLP2 achieves the same bounds as in Theorem 2 in the general case, and under perfect predictions it ensures:

$$\mathcal{R}_T = \mathcal{O}\left(T^{\frac{1-\beta}{2}}\right), \quad \mathcal{V}_T = \mathcal{O}\left(T^{\frac{1+\beta}{2}}\right), \quad \beta \in [0, 1).$$

For example, LLP2 with perfect predictions and $\beta = 0$ achieves $\mathcal{R}_T = \mathcal{O}(\sqrt{T})$ and $\mathcal{V}_T = \mathcal{O}(\sqrt{T})$; and with $\beta = 1/3$ yields $\mathcal{R}_T = \mathcal{O}(T^{1/3})$ and $\mathcal{V}_T = \mathcal{O}(T^{2/3})$. Hence, it outperforms the state-of-the-art constrained-OCO algorithms with no predictions, but does not perform as well as LLP for perfect predictions.

B. LLP With Linearized Constraints

Another way to reduce the computation load of LLP is to linearize the constraint function. This, however, is not trivial since we cannot recover $\mathcal{V}_T$ by simply using the convexity of $\{g_t\}$, as we do with $\mathcal{R}_T$ and the linearization of $\{f_t\}$. Hence, we use linear proxies for the constraints and their predictions:

$$g_t^\ell(x) = g_t(x_t) + \nabla g_t(x_t)^\top(x - x_t), \quad (27)$$
$$\tilde{g}_t^\ell(x) = \tilde{g}_t(\tilde{x}_t) + \nabla\tilde{g}_t(\tilde{x}_t)^\top(x - \tilde{x}_t). \quad (28)$$

LLP with linearized constraints, which we call LLP3, runs similarly to Algorithm 1, but uses predictions $\tilde{c}_t$, $\nabla\tilde{g}_t(\tilde{x}_t)$ and $\tilde{g}_{t+1}(\tilde{x}_{t+1})$, i.e., does not need to predict the entire constraint function — nor $\tilde{x}_t$, despite appearing in (28). And this does not affect its performance.

Lemma 3: LLP3 achieves the $\mathcal{R}_T$, $\mathcal{V}_T$ bounds and convergence rates in Theorems 1 and 2, respectively.

The proof of the Lemma and the details of the linearized LLP can be found in the Appendix. Now, whether it is more difficult to predict the next constraint gradient or the entire next constraint function, is a question pertaining to the problem at hand, and practitioners can select the respective version of LLP that suit their needs. For instance, as explained in Section I, for single-parameter functions it is easier to predict the entire next-slot function by guessing the parameter, as opposed to predicting the next gradient and next value, which requires a guess for the system's operation point as well. In other cases predicting the structure of an entire function (e.g., if g_{t+1} is totally different thatn g_t) might be more challenging.

V. LINEARLY-PERTURBED CONSTRAINTS

In this section we consider the special type of constraints that are linearly-perturbed, which was studied first in [16]. In detail, the constraints and their predictions are:

$$g_t(x) = g(x) + b_t \text{ and } \tilde{g}_t(x) = g(x) + \tilde{b}_t. \quad (29)$$

where $b_t, \tilde{b}_t \in \mathbb{R}^d$ are the unknown per-slot perturbations that are added to the fixed (and known) function component $g(x)$. For example, consider a network routing problem where the network graph and capacities are fixed, $g_t = g, \forall t$, but the incoming flow varies in an arbitrary fashion according to vectors $\{b_t\}_t$, [42]. This simplification has important ramifications for the analysis and, eventually, improves the bounds of LLP as follows:

Lemma 4: Under the conditions of Theorem 1, with constraints and predictions given by (29), LLP ensures:

$$\mathcal{R}_T \leq 0, \mathcal{V}_T = \mathcal{O}\left(T^{(1+\beta)/2}\right), \text{ perfect predictions, } \beta\in[0, 1)$$
$$\mathcal{R}_T = \mathcal{O}(\sqrt{T}), \mathcal{V}_T = \mathcal{O}(T^{5/8}), \text{ worst predictions, } \beta = 1/2.$$

This result improves the bounds for the case of general constraints of the previous section, and yields only $1/8$ worse constraint violation than the bounds in [16] (and the same regret bounds). However, [16] cannot leverage predictions. That is, even if we have perfect predictions at our disposal, [16] will still offer the same regret and constraint violation bounds.

On the contrary, we see that LLP, due to its prediction-adaptive steps, yields no regret for perfect predictions and the constant factors of the bounds shrink commensurately with the predictions' accuracy. This becomes clear if we express the bounds as follows:

$$\mathcal{R}_T \le B_T \triangleq A_1 \sqrt{h_{1:T}} + \min \left\{ 2a \sqrt{\sum_{t=1}^T \xi_t^2}, \; A_2 T^{1-\beta} \right\}$$
(30)

$$\mathcal{V}_T \le \left[A_3 \max \left\{ K_T, T^\beta \right\} \left(B_T - R_T \right) \right]^{\frac{1}{2}} + A_4 \sqrt{h_{1:T}} \quad (31)$$

where we have defined the parameters:

$$A_1 = 2\sigma D^2 + (2L_f/\sigma), A_2 = 4aG^2/(1-\beta), A_3 = 2/a$$

$$A_4 = 2L_g/\sigma, h_{1:T} = \sum_{t=1}^T \|\epsilon_t\|, K_T = \left[G^2 + \sum_{t=1}^T \xi_t^2 \right]^{1/2}.$$

Note that in this case the quantity $h_{1:T}$ does not depend on the dual vectors. This is due to the fact that the perturbations do not affect the primal step (see details Sec. VI-H of the Appendix), which disentangles – to some extent – the primal and dual iterations. The proof of the lemma and the details for deriving (30) and (31) can be found in the Appendix.

VI. CONCLUSION

LLP differs from related algorithms since the primal and dual updates are *lazy*. This allows an FTRL-based design, which is widely used in fixed-constraints OCO algorithms ($x \in \mathcal{X}$) but is new in the context of time-varying constrains ($x \in \mathcal{X}_T$). The LLP order bounds, even with worst-case or no predictions, are competitive with existing algorithms while dropping several impractical assumptions these are using. Indeed, prior algorithms with time-varying constraints require strongly convex cost and constraint functions or linearly-perturbed fixed constraints, and rely on the Slater condition. Other proposals assume time-invariant constraints, still rely on a fixed and known time horizon T (which cannot be remedied by the doubling trick); and need access to all Lipschitz constants and constraint bounds. Hence, LLP can be applied in a wider range of network management problems.

The most important advantage of LLP is that it encompasses predictions of unknown quality. This is the first work that proposes and tackles this problem in the context of OCO with time-varying budget constraints. Clearly, despite the good performance of prior works, none of them can benefit from the availability of (potentially inaccurate) predictions. LLP, instead, gains directly from them as the constant factors of the regret and constraint violation shrink in proportion to predictions' quality; this is oftentimes even more beneficial than having a slightly tighter learning rate w.r.t. T, especially in problems with large dimension (D is comparable to T). And when the predictions are perfect, LLP achieves $\mathcal{R}_T = \mathcal{O}(1)$ and $\mathcal{V}_T = \mathcal{O}(\sqrt{T})$. Last but not least, this framework is unified as it and can run without predictions (setting them zero) since we impose no assumptions on their quality, and can be applied to problems with time-invariant constraints. Hence, it opens the road for employing network datasets and measurements to make predictions (e.g., using Deep Learning for predicting capacity or demand), without the concern of their accuracy, that indeed might vary widely in different cases.

Concluding, there are several interesting directions for future work, such as extending this framework to dynamic regret benchmarks by imposing additional restrictions on the variability of cost and constraint functions, or exploring whether the achieved bounds can be further reduced.

APPENDIX

The Appendix includes the missing proofs from the main document; the supporting lemmas and their proofs; and additional discussion for the main results.

A. Performance Cases of LLP

We start by discussing the different cases regarding the performance of LLP in order to facilitate the reader understanding the consequences of Theorems 1 and 2. Figure 3 summarizes the three possible scenarios. Case (i) is realized when $-\mathcal{R}_T = \mathcal{O}(\sqrt{T})$, i.e., when LLP does not outperform $x^\star$ by more than this growth rate, and here LLP achieves competitive rates, $\mathcal{R}_T, \mathcal{V}_T = \mathcal{O}(\sqrt{T})$ and zero regret for perfect predictions. Case (ii) arises when the condition $-\mathcal{R}_T = \mathcal{O}(\sqrt{T})$ is consistently violated and in fact yields even better performance in terms of regret, while maintaining the general $\mathcal{V}_T = \mathcal{O}(T^{2/3})$. And finally, Case (iii) arises when the above condition might be violated during some time intervals and sample paths, but not consistently; and for this scenario the general bounds $\mathcal{R}_T, \mathcal{V}_T = \mathcal{O}(T^{2/3})$ hold. In all cases, the constant factors of the regret diminish as the predictions' quality improves.

B. Proof of Lemma 1

Lemma 1: No online algorithm can achieve concurrently sublinear regret and constraint violation, $\mathcal{R}_T = o(T)$, $\mathcal{V}_T = o(T)$, w.r.t. $x^\star \in X_T^{\max}$, even if the algorithm selects x_t with knowledge of f_t, g_t.

Proof: We provide an opponent strategy that ensures there is an increasing sequence $t(1), t(2), \ldots$ of rounds with either $\mathcal{R}_{t(i)} \ge t(i)/8$ or $\mathcal{V}_{t(i)} \ge t(i)/8$. Our opponent will select f_{t+1}, g_{t+1} based only on $x_1, \ldots, x_t$. Hence the impossibility result holds even if the player knows f_{t+1}, g_{t+1} on round $t+1$.

Consider the domain $\mathcal{X} = [0, 1]$. The cost functions are linear and the pair (f_t, g_t) is always one of $p = (-x, -1)$ or $q = (-2x, 2x-1)$. Before giving the opponent strategy we make some general observations. To derive the set $\sum_{t=1}^T g_i(x) \le 0$ suppose the opponent plays p exactly n times and q exactly $T - n$ times. Then we have:

$$\sum_{t=1}^T g_t(x) = -n + (T-n)(2x-1) = 2(T-n)x - T$$

Hence the constraint set $G_T = \left\{ x \in \mathcal{X} : \sum_{t=1}^T g_i(x) \le 0 \right\}$ is the part of $[0, 1]$ with

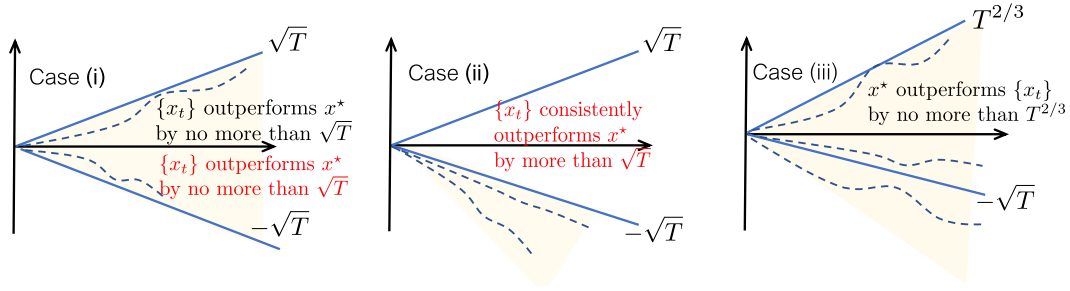$$x \le \frac{T}{2(T-n)} = \frac{1}{2} + \frac{n}{2(T-n)}.$$

Fig. 3. Performance region and cases for the Regret of LLP, under worst-case (or, no available) predictions.

In particular for $n \geq T/2$ the second term is at least $1/2$ and $G_T = [0,1]$. Since f_t are negative linear, the regret is with respect to $x^\star = 1$. For $f_t(x) = -x$ we have $f_t(x_t) - f_t(x^\star) = 1 - x_t$ and for $f_t(x) = -2x$ we have $f_t(x_t) - f_t(x^\star) = 2(1 - x_t)$. In each case $f_t(x_t) - f_t(x^\star) \geq 1 - x_t$, since $2(1 - x_t) \geq 1 - x_t$ when $x_t \in [0,1]$, and so $\mathcal{R}_T \geq \sum_{t=1}^{T}(1 - x_t)$.

Now suppose the rounds are broken into blocks $\{1, 2, \ldots\} = I_1 \cup J_1 \cup I_2 \cup J_2 \cup \ldots$ with each $|I_n| = |J_n|$. Define each $I = \bigcup_n I_n$ and $J = \bigcup_n J_n$ and $\overline{x}_{t-1} = \frac{1}{t-1}\sum_{i=1}^{t-1} x_i$. Assume the opponent has the strategy: leftmargin=5mm

1) Each $(f_t, g_t) = q \iff t \in I$.
2) Each $(f_t, g_t) = p \iff t \in J$.
3) For each $n \in I$ we have $\overline{x}_{t-1} \geq 3/4$.
4) For each m and each $t = \min J_m$ we have $\overline{x}_{t-1} < 3/4$.

There are two cases to consider. First assume (a) there are infinitely many I_m, J_m. Since each $|I_m| = |J_m|$ we see on the final turn $t(m)$ of each J_m the opponent has played p exactly $n = T/2$ times. Hence the above says $G_{t(m)} = [0,1]$ and $x^\star = 1$ and the regret is $R_{t(m)} \geq \sum_{i=1}^{t(m)}(1 - x_i)$. On turn $s(m) = \max I_m$ the regret is at most:

$$\sum_{i=1}^{s(m)}(1 - x_i) = s(m) - \sum_{i=1}^{s(m)} x_i = s(m) - s(m)\overline{x}_{s(m)}$$

$$= s(m)(1 - \overline{x}_{s(m)}) \geq \frac{s(m)}{4} \geq \frac{t(m)}{8}.$$

where the first inequality uses (4) and the second uses $t(m) = s(m) + 1 \geq t(m)/2$. Since there are infinitely many $t(1), t(2), \ldots$ there are infinitely many turns t with $R_t \geq t/8$. Hence the regret is $\Omega(T)$.

Now assume (b) there are only finitely many I_m, J_m. Since each $|I_m| = |J_m|$ the blocks are $I_1, J_1, \ldots, J_{m-1}, I_m$ with $|I_m| = \infty$. To see the violation is $\Omega(T)$ observe (3) gives $\overline{x}_s \geq 3/4$ for $s = \max J_{m-1}$. By (1) the opponent plays $q = (2x, -2x + 1)$ on turns $s+1, s+2, \ldots$ Thus for $T \geq 4s$ the constraint violation is:

$$\sum_{t=1}^{T} g_t(x_t) = \sum_{t=1}^{s} g_t(x_t) + \sum_{t>s}^{T} g_t(x_t) \geq -s + \sum_{t>s}^{T}(2x_t - 1)$$

$$= -T + 2\sum_{t>s}^{T} x_t = -T - \sum_{t=1}^{s} x_t + 2\sum_{t=1}^{T} x_t$$

$$\geq -(T + s) + 2\overline{x}_T \geq -(T + s) + \frac{3}{2}T$$

where the last inequality uses (3). Since $T \geq 4s$ the RHS is at most $T/2 - s \geq T/2 - T/4 = T/8$. Hence we can take $t(i) = 4s + i$.

To complete the proof we give an opponent strategy that satisfies these four conditions. Reference [7] suggest the following approach:

(i) Play q on turns $\min I_n, \min I_n + 1, \ldots, m$ where m is the first turn with $\overline{x}_m < 3/4$.
(ii) End I_n and begin J_n.
(iii) Play p over the next $|I_n|$ turns.
(iv) End J_n and begin I_{n+1}.

The decision on turn $m + 1$ to end I_n depends only on the average $\overline{x}_m$ of $x_1, \ldots, x_m$. Hence the opponent does not need to see the player's current move to implement the strategy. Equivalently the player is allowed to see the opponent's next move. This concludes the proof. $\qquad \square$

C. Proof of Lemma 2

Using the property of the proximal regularizers, $x_t = \arg\min_x r_t(x), \forall t$, we can expand (4) and write:

$$x_t = \arg\min_{x \in \mathbb{R}^N} \left\{ r_{0:t-1}(x) + c_{1:t-1}^\top x + \sum_{i=1}^{t-1} \lambda_i^\top g_i(x) \right.$$

$$\left. + \tilde{c}_t^\top x + \lambda_t^\top \tilde{g}_t(x) \right\} \overset{\text{add } r_t(x)}{\Longrightarrow}$$

$$x_t = \arg\min_{x \in \mathbb{R}^N} \left\{ r_{0:t}(x) + c_{1:t-1}^\top x + \sum_{i=1}^{t-1} \lambda_i^\top g_i(x) \right.$$

$$\left. + \tilde{c}_t^\top x + \lambda_t^\top \tilde{g}_t(x) \right\},$$

where adding the t-round regularizer $r_t(x)$ does not change the minimizer of the RHS argument – and it is easy to see this using a contradiction argument.

Now, recall that the prescient action is:

$$z_t = \arg\min_{x \in \mathbb{R}^N} \left\{ r_{0:t}(x) + c_{1:t}^\top x + \sum_{i=1}^{t} \lambda_i^\top g_i(x) \right\}.$$

Applying Lemma 7 from [35], with:

$$\phi_1(x) = r_{0:t}(x) + c_{1:t-1}^\top x + \sum_{i=1}^{t-1} \lambda_i^\top g_i(x) + \tilde{c}_t^\top x + \lambda_t^\top \tilde{g}_t(x)$$

$$\phi_2(x) = r_{0:t}(x) + c_{1:t}^\top x + \sum_{i=1}^{t} \lambda_i^\top g_i(x) \quad \text{and}$$

$$\psi(x) = (c_t - \tilde{c}_t)^\top x + \lambda_t^\top \left(g_t(x) - \tilde{g}_t(x) \right)$$

and recalling that the regularizer $r_{0:t}(x)$ is 1-strongly-convex w.r.t. norm $\|x\|_{(t)} = \sqrt{\sigma_{1:t}}\|x\|$ that has dual norm

$\|x\|_{(t),\star} = \|x\|/\sqrt{\sigma_{1:t}}$, we can write:

$$\|x_t - z_t\|_{(t)} \leq \|c_t - \tilde{c}_t + \lambda_t^\top (\nabla g_t(x_t) - \nabla \tilde{g}_t(x_t))\|_{(t),\star} \Rightarrow$$

$$\sqrt{\sigma_{1:t}}\|x_t - z_t\| \leq \frac{\|c_t - \tilde{c}_t + \lambda_t^\top (\nabla g_t(x_t) - \nabla \tilde{g}_t(x_t))\|}{\sqrt{\sigma_{1:t}}} \Rightarrow$$

$$\|x_t - z_t\| \leq = \frac{\|\varepsilon_t + \lambda_t^\top \delta_t\|}{\sigma_{1:t}}$$

D. Proof of Lemma 5 (Optimistic FTRL)

In the proof of Theorem 1 we relied on [26, Theorem 2]. However, that work considers a learning problem over a compact convex set, while the dual update to which we apply this result has an unbounded decision space $\lambda \in \mathbb{R}_+^d$. This indeed does not pose a problem for our analysis. Firstly, one can see that for the standard FTRL analysis it suffices to have closed sets, see for example [35]. Secondly, the boundedness of the set is useful when we need to upper-bound the term $q_{0:T-1}(\lambda)$ that appears in the RHS of (9). In our analysis, this is not necessary as we cancel this term by setting $\lambda = 0$.

To complete this discussion, we provide here an alternative proof for [26, Theorem 2] that makes clear it is valid even if the decision set is not bounded. We note that this is presented in terms of the primal variables and functions $\{f_t(x_t)\}$ to streamline the presentation, but the application to the dual variables and dual updates is straightforward.

Lemma 5: Let $\{r_t\}$ be a sequence of non-negative regularizers, and let $\tilde{c}_t$ be the learner's estimate of $c_t = \nabla f_t(x_t)$. Assume also that the function $\kappa_{0:t-1} : x \mapsto c_{1:t-1}^\top x + \tilde{c}_t^\top x + r_{0:t-1}(x)$ is 1-strongly convex w.r.t. norm $\|\cdot\|_{(t-1)}$, and consider the update $x_t = \arg\min_{x \in \mathcal{X}} \kappa_{0:t-1}(x)$, where $r_0(x) = \mathbf{I}_\mathcal{X}(x)$ and $\tilde{c}_1 = 0$. Then the following regret bound holds:

$$\mathcal{R}_T \leq r_{0:T-1}(x) + \sum_{t=1}^T \|c_t - \tilde{c}_t\|_{(t-1),\star}^2, \forall x \in \mathcal{X}$$

Proof: Using the auxiliary functions $m_t(x) = r_{t-1}(x) + \tilde{c}_t^\top x$ and $n_t(x) = c_t^\top x - \tilde{c}_t^\top x$, we can write $x_t = \arg\min_{x \in \mathcal{X}}\{m_{1:t}(x) + n_{1:t-1}(x)\}$ and $z_t = \arg\min_{x \in \mathcal{X}}\{m_{1:t}(x) + n_{1:t}(x)\}$ which correspond to the primal and prescient updates for the non-proximal version of the optimistic FTRL. We first prove the following relation using induction:

$$\sum_{t=1}^T m_t(x_t) + n_t(z_t) \leq m_{1:T}(x^\star) + n_{1:T}(x^\star), \ \forall x^\star \in \mathcal{X}. \tag{32}$$

For $t = 1$, it is:

$$m_1(x_1) + n_1(z_1) = r_0(x_1) + \tilde{c}_1^\top x_1 + c_1^\top z_1 - \tilde{c}_1^\top z_1$$
$$\leq r_0(x^\star) + \tilde{c}_1^\top x^\star + c_1^\top x^\star - \tilde{c}_1^\top x^\star$$
$$= m_1(x^\star) + n_1(x^\star)$$

which holds since $r_0(x_1) = r_0(x^\star) = 0$, $\tilde{c}_1^\top x_1 = 0$, and $z_1 = \arg\min_{x \in \mathcal{X}} \{m_1(x) + n_1(x)\}$. Assume it holds for $t = \tau$ and add to both sides $m_{\tau+1}(x^\star) + n_{\tau+1}(x')$ for some $x^\star, x' \in \mathcal{X}$:

$$\sum_{t=1}^\tau m_t(x_t) + m_{\tau+1}(x^\star) + \sum_{t=1}^\tau n_t(z_t) + n_{\tau+1}(x')$$
$$\leq m_{1:\tau+1}(x^\star) + n_{1:\tau}(x^\star) + n_{\tau+1}(x')$$

set $x^\star = x_{t+1}$, $x' = z_{t+1}$, to get:

$$\sum_{t=1}^{\tau+1} m_t(x_t) + n_t(z_t) \leq m_{1:\tau+1}(x_{t+1}) + n_{1:\tau}(x_{t+1}) + n_{\tau+1}(z_{t+1})$$
$$\leq m_{1:\tau+1}(z_{t+1}) + n_{1:\tau+1}(z_{t+1}) \text{ (by definition of } x_{t+1})$$
$$\leq m_{1:\tau+1}(x^\star) + n_{1:\tau+1}(x^\star) \text{ (by definition of } z_{t+1})$$

which concludes our induction step. Hence, we proved (32).

Replacing $m_t(x)$ and $n_t(x)$, dropping the non-negative term $\sum_{t=1}^{T-1} r_t(x_t)$ in the LHS, adding $\sum_{t=1}^T c_t^\top x_t$ to both sides and rearranging, we eventually get $\mathcal{R}_T \leq$

$$\leq r_{0:T-1}(x^\star) + \sum_{t=1}^T \left(c_t^\top (x_t - z_t) \right) + \sum_{t=1}^T \left(\tilde{c}_t^\top (z_t - x_t) \right)$$
$$\leq r_{0:T-1}(x^\star) + \sum_{t=1}^T \left(c_t - \tilde{c}_t \right)^\top (x_t - z_t)$$
$$\leq r_{0:T-1}(x^\star) + \sum_{t=1}^T \|c_t - \tilde{c}_t\|_{(t-1),\star} \|x_t - z_t\|_{(t-1)} \tag{33}$$

where in the last step we used the Cauchy-Schwarz inequality. For the term $\|x_t - z_t\|_{(t-1)}$ we can apply [35, Lemma 7] with:

$$\phi_1(x) = r_{0:t-1}(x) + c_{1:t-1}^\top x + \tilde{c}_t^\top x$$
$$\phi_2(x) = r_{0:t-1}(x) + c_{1:t-1}^\top x + \tilde{c}_t^\top x + (c_t^\top x - \tilde{c}_t^\top x)$$
$$\psi(x) = c_t^\top x - \tilde{c}_t^\top x$$

to get the bound $\|x_t - z_t\|_{(t-1)} \leq \|c_t - \tilde{c}_t\|_{(t-1),\star}$. Replacing in (33) we conclude the proof noting that we did not use boundedness of $\mathcal{X}$ in any step of the proof. $\square$

E. Proof of Lemma 6

Lemma 6: For $d \in (0, 1)$ we have

$$\sum_{t=1}^T t^{-d} \leq \frac{T^{1-d}}{1-d}$$

Proof: Let $F(x) = \lfloor x + 1 \rfloor^{-d}$ be defined over $[0, \infty)$. Clearly $F(x) = (n+1)^{-c}$ for each $x \in [n, n+1)$ and $n = 0, 1, 2, \ldots$, and so the sum on the left equals $\int_0^T F(x)dx$. Since $\lfloor x + 1 \rfloor \geq x$ and $d > 0$ we have $F(x) = \lfloor x + 1 \rfloor^{-d} \leq x^{-d}$ and $\int_0^T F(x)dx \leq \int_0^T x^{-d}dx = \frac{T^{1-d}}{1-d}$. $\square$

F. Proof of Theorem 3

First, note we can directly obtain the bound:

$$\|\lambda_T\| \leq \left\| \left[a_{T-1} \left(\sum_{i=1}^{T-1} g_i(z_i) + \tilde{g}_T(\tilde{z}_T) \right) \right]_+ \right\| \leq a_{T-1} T G$$

where the RHS term appears in μ_T. Hence, the strong convexity of LLP2 can be lower-bounded as:

$$\sigma_{1:t} = \sigma\sqrt{h_{1:t} + \mu_{t+1}} \geq \sigma\sqrt{h_{1:t+1}}. \tag{34}$$

Next, we replace Lemma 2 with an updated bound.

Lemma 7: For the actions x_t and z_t obtained by (25) and (26), respectively, it holds:

$$\|x_t - z_t\| \leq \frac{\|\epsilon_t + \lambda_t^\top \delta_t\|}{\sigma_{1:t-1}} \overset{(34)}{\leq} \frac{\|\epsilon_t + \lambda_t^\top \delta_t\|}{\sqrt{\sum_{i=1}^t \|\epsilon_i + \lambda_i^\top \delta_i\|}} \tag{35}$$

It is easy to see that since the dual update has not changed, equation (9) holds as is, and we can readily obtain (11), and eventually write:

$$\mathcal{R}_T \leq 2D^2\sigma_{1:T} + \frac{2L_f}{\sigma}\sqrt{h_{1:T}} + \sum_{t=1}^T \frac{\xi_t^2}{\phi_{0:t-1}}$$
$$\leq 2\left(\sigma D^2 + \frac{L_f}{\sigma}\right)\sqrt{h_{1:T} + \mu_{T+1}} + \sum_{t=1}^T a_{t-1}\xi_t^2 \triangleq \widehat{B}_T.$$

Therefore similarly to (19), we can write:

$$\|\lambda_{T+1}\| \leq a_T\sqrt{2(\widehat{B}_T - \mathcal{R}_T)/a_T} + a_1 G.$$

And finally, observe that since the dual regularizer $q_t(\lambda)$ and learning rate a_t are given by (8), (15), the bounds (20), (21) hold for LLP2 as well. Putting these together, we arrive at:

$$\|\lambda_T\| = \mathcal{O}\left(\max\left\{T^{\frac{k+1+2\theta}{4}}, T^{\frac{1+3\theta}{4}}, T^{\frac{n+\theta}{2}}, T^{\frac{1+\theta}{2}}\right\}\right).$$

That is, the growth rate of $\|\lambda_T\|$ remains $\mathcal{O}(T^{(1+\theta)/2})$ and is not affected by the non-proximal primal regularizers. Similarly, the growth rate of $\widehat{B}_T$ is the same as that of B_T, see (23), since $h_{1:T} = \mathcal{O}(T^{(3+\theta)/2})$ dominates $\mu_{T+1} = \mathcal{O}(T^{1+\theta})$. Therefore, the growth rate of $\mathcal{R}_T$ and of $\mathcal{V}_T$ are exactly as those of LLP.

On the other hand, the $\mathcal{R}_T$ bound of LLP2 is not zeroed for perfect predictions. Indeed, when $\epsilon_t = \delta_t = \xi_t = 0, \forall t$, and $h_{1:T} = 0$, we get:

$$\mathcal{R}_T \leq \sqrt{\mu_{T+1}} \quad \text{and} \quad \mathcal{V}_T \leq \sqrt{-2\mathcal{R}_T/a_{T-1}} + \sqrt{\mu_{T+1}},$$

where $\mu_{T+1} = E_m + a_T(T+1)G\Delta_m > 0$ and $a_T = aT^{-\beta}$. Hence, $\mathcal{R}_T = \mathcal{O}(T^{(1-\beta)/2})$ and $\mathcal{V}_T = \mathcal{O}(T^{(1+\beta)/2})$ which shows that we cannot achieve zero regret, and even more so, that as we reduce the bound of $\mathcal{R}_T$ by increasing β, we deteriorate in a commensurate amount the bound of $\mathcal{V}_T$.

G. Proof of Lemma 3

If we define $\mathcal{L}_t(x, \lambda)$ as in (3) but replace $g_t(x)$ with $g_t^\ell(x)$ and also use the linearized prediction $\tilde{g}_t^\ell(x)$, with:

$$g_t^\ell(x) = g_t(x_t) + \nabla g_t(x_t)^\top (x - x_t),$$
$$\tilde{g}_t^\ell(x) = \tilde{g}_t(\tilde{x}_t) + \nabla\tilde{g}_t(\tilde{x}_t)^\top (x - \tilde{x}_t), \tag{36}$$

then, the updates that we use in LLP2 are:

$$x_t = \arg\min_{x\in\mathbb{R}^N}\left\{\sum_{i=0}^{t-1}\mathcal{L}_i(x,\lambda_i) + \tilde{c}_t^\top x + \lambda_t^\top \tilde{g}_t^\ell(x)\right\} \Rightarrow$$
$$x_t = \arg\min_{x\in\mathbb{R}^N}\left\{r_{0:t-1}(x) + c_{0:t-1}^\top x + \sum_{i=0}^{t-1}\lambda_i^\top g_i^\ell(x)\right.$$

$$\left. + \tilde{c}_t^\top x + \lambda_t^\top\left(\nabla\tilde{g}_t(\tilde{x}_t)^\top x\right)\right\},$$

$$z_t = \arg\min_{x\in\mathbb{R}^N}\left\{\sum_{i=0}^t \mathcal{L}_i(x,\lambda_i)\right\}$$
$$= \arg\min_{x\in\mathbb{R}^N}\left\{r_{0:t}(x) + c_{1:t}^\top x + \sum_{i=0}^t \lambda_i^\top g_i^\ell(x)\right\},$$

$$\lambda_{t+1} = \arg\max_{\lambda\in\mathbb{R}^d}\left\{\sum_{i=0}^t \mathcal{L}_i(z_i,\lambda_i) + \lambda^\top \tilde{g}_{t+1}^\ell(\tilde{x}_{t+1})\right\}$$
$$= \arg\max_{\lambda\in\mathbb{R}^d}\left\{\lambda^\top\left(\sum_{i=1}^t g_i^\ell(z_i)\right.\right.$$
$$\left.\left. + \tilde{g}_{t+1}(\tilde{x}_{t+1})\right) - \phi_{0:t}\|\lambda\|^2/2\right\} \tag{37}$$

where the primal and dual regularizers are given again by (7), (8) using the modified error parameters, $\forall t$:

$$\delta_t = \nabla g_t(x_t) - \nabla\tilde{g}_t(\tilde{x}_t), \quad \xi_t = \|g_t^\ell(z_t) - \tilde{g}_t^\ell(\tilde{x}_t)\|, \quad \text{and}$$
$$\epsilon_t = c_t - \tilde{c}_t \quad \text{as before.}$$

Note that in case of perfect predictions, i.e., when:

$$\nabla\tilde{g}_t(\tilde{x}_t) = \nabla g_t(x_t), \quad c_t = \tilde{c}_t, \quad \tilde{g}_t(\tilde{x}_t) = g_t(x_t), \quad \forall t,$$

then $\varepsilon_t = 0$, $\delta_t = 0$, and

$$\xi_t = \left\|g_t(x_t) + \nabla g_t(x_t)^\top (z_t - x_t) - \tilde{g}_t(\tilde{x}_t)\right.$$
$$\left. - \nabla\tilde{g}_t(\tilde{x}_t)^\top (\tilde{x}_t - \tilde{x}_t)\right\| = \left\|\nabla g_t(x_t)^\top (z_t - x_t)\right\| = 0,$$

where the last step follows as for perfect predictions, clearly, it holds $z_t = x_t$.

Next, it is easy to see that Lemma 2 holds and yields the same bound $\|x_t - z_t\| \leq \|\epsilon_t + \lambda_t^\top \delta_t\|/\sigma_{1:t}$ with the redefined $\{\delta_t\}$. Applying [26, Theorem 2] to (37), we get:

$$-\sum_{t=1}^T \lambda_t^\top g_t^\ell(z_t) + \sum_{t=1}^T \lambda^\top g_t^\ell(z_t)$$
$$\leq q_{0:T-1}(\lambda) + \sum_{t=1}^T \|g_t^\ell(z_t) - \tilde{g}_t(\tilde{x}_t)\|_{(t-1),\star}^2 =$$
$$= q_{0:T-1}(\lambda) + \sum_{t=1}^T \frac{\|g_t^\ell(z_t) - \tilde{g}_t(\tilde{x}_t)\|^2}{\phi_{0:t-1}}, \quad \forall\lambda\in\mathbb{R}_+^d. \tag{38}$$

Then, we can repeat the analysis in Sec. II-A, noting:

$$g_t^\ell(x^\star) = g_t(x_t) + \nabla g_t(x_t)^\top (x^\star - x_t) \preceq g_t(x^\star) \preceq 0, \quad \forall t$$

due to convexity of $g_t(\cdot)$ and the property of $x^\star$, to arrive at the same bound B_T for the regret $\mathcal{R}_T$, sans the redefined $\{\delta_t\}$ and $\{\xi_t\}$ parameters.

Similarly, repeating the analysis of Sec. II-B we get:

$$\lambda^\top\left(\sum_{t=1}^T g_t^\ell(z_t)\right) - \frac{\phi_{0:T-1}}{2}\|\lambda\|^2 \leq B_T - \mathcal{R}_T, \quad \forall\lambda\in\mathbb{R}_+^d. \tag{39}$$

Minimizing the LHS, similarly to Sec. II-B, we obtain:

$$V_T^{z,\ell} \leq \sqrt{2(B_T - \mathcal{R}_T)/a_{T-1}} \quad \text{where} \quad \mathcal{V}_T^{z,\ell} \triangleq \left\|\left[\sum_{t=1}^T g_t^\ell(z_t)\right]_+\right\|.$$
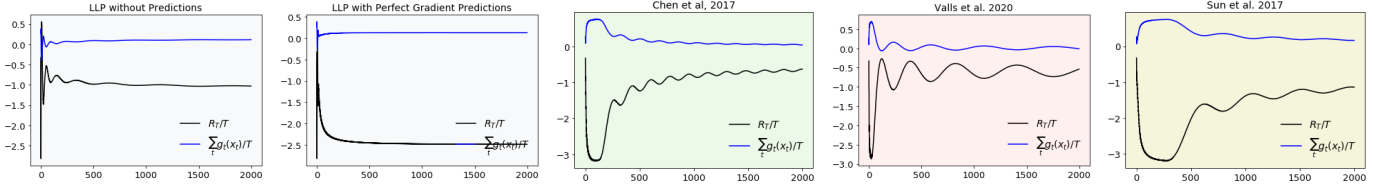
Fig. 4. Benchmark comparisons for: $f_t(x) = -4x$ if $t\%2 = 0$, $f_t(x) = -x$ otherwise; $g_t(x) = 0.79x + 0.26$ if $t\%2 = 0$, $g_t(x) = 0.64x - 0.135$ otherwise; $\mathcal{X} = [-1, 1]$. The 1st LLP plot does not use any predictions, while the 2nd LLP plot uses perfect predictions for $\nabla f_t(x_t)$ and $\nabla g_t(x_t)$ but no predictions for $g_{t+1}(x_{t+1})$. For the experiments presented here we have used $\beta = 0.2$.

Finally, note that using the definition of $g_t^\ell(x)$, we can write:

$$g_t(x_t) = g_t^\ell(z_t) - \nabla g_t(x_t)^\top (z_t - x_t) \Rightarrow$$
$$\sum_{t=1}^T g_t(x_t) = \sum_{t=1}^T g_t^\ell(z_t) + \sum_{t=1}^T \nabla g_t(x_t)^\top (x_t - z_t).$$

Hence, we obtain the bound:

$$\left\| \left[\sum_{t=1}^T g_t(x_t) \right]_+ \right\|$$
$$= \left\| \left[\sum_{t=1}^T g_t^\ell(z_t) + \nabla g_t(x_t)^\top (x_t - z_t) \right]_+ \right\|$$
$$\le \left\| \left[\sum_{t=1}^T g_t^\ell(z_t) \right]_+ \right\| + \left\| \left[\sum_{t=1}^T \nabla g_t(x_t)^\top (x_t - z_t) \right]_+ \right\|,$$

where we used the identity $\|[\chi + \upsilon]_+\| \le \|[\chi]_+\| + \|[\upsilon]_+\|$. Therefore we arrived at the same result as with the non-linearized constraints:

$$\mathcal{V}_T \le V_T^{z,\ell} + \frac{2L_g}{\sigma} \sqrt{h_{1:T}} = \sqrt{\frac{2(B_T - \mathcal{R}_T)}{a_{T-1}}} + \frac{2L_g}{\sigma} \sqrt{h_{1:T}},$$

by using a slightly different proof.

Finally, it follows directly from the proof of Theorem 2 that the convergence rates are not affected by this linearization of the constraints. We conclude by stressing that in this version of LLP we only required predictions $\tilde{c}_t$, $\nabla \tilde{g}_t(\tilde{x}_t)$ and $\tilde{g}_{t+1}(\tilde{x}_{t+1})$; while with non-linearized constraints LLP was using predictions $\tilde{c}_t$, $\tilde{g}_t(\cdot)$ and $\tilde{g}_{t+1}(\tilde{x}_{t+1})$.

H. Proof of Lemma 4

For this specific type of constraints, the primal and prescient updates are as follows:

$$x_t = \arg\min_{x \in \mathcal{X}} \left\{ r_{0:t-1}(x) + c_{1:t-1}^\top x + \tilde{c}_t^\top x + \lambda_{1:t} g(x) \right\}$$
$$z_t = \arg\min_{x \in \mathcal{X}} \left\{ r_{0:t}(x) + c_{1:t}^\top x + \lambda_{1:t} g(x) \right\},$$

while the dual update remains the same:

$$\lambda_{t+1} = \arg\max_{\lambda \in \mathbb{R}_+^d} \left\{ -q_{0:t}(\lambda) + \lambda^\top \left(\tilde{g}_{t+1}(\tilde{x}_{t+1}) + \sum_{i=0}^t g_t(z_i) \right) \right\}.$$

With this modification, Lemma 2 yields the bound:

$$\|x_t - z_t\| \le \frac{\|\epsilon_t\|}{\sigma_{1:t}}, \quad \text{with } \epsilon_t = c_t - \tilde{c}_t, \ \sigma_t = \sigma \sqrt{\sum_{i=1}^t \|\epsilon_i\|}$$

and therefore:

$$\sum_{t=1}^T c_t^\top (x_t - z_t) \le \sum_{t=1}^T \frac{\|c_t\| \|\epsilon_t\|}{\sigma_{1:t}} \le 2\frac{L_f}{\sigma} \sqrt{\sum_{t=1}^T \|\epsilon_t\|}.$$

Similarly, we can redefine B_T and re-derive (12) as follows:

$$\mathcal{R}_T + \lambda^\top \left(\sum_{t=1}^T g_t(z_t) - \frac{\|\lambda\|^2}{2a_{T-1}} \right) \le B_T$$

where $B_T \triangleq 2\left(\sigma D^2 + \frac{L_f}{\sigma} \right) \sqrt{\sum_{t=1}^T \|\epsilon_t\|} + \sum_{t=1}^T a_{t-1} \xi_t^2$

and $\xi_t = \|g_t(z_t) - \tilde{g}_t(\tilde{x}_t)\|$. Selecting the λ that maximizes the second term in the LHS as before, we arrive at:

$$\mathcal{R}_T + \frac{a_{T-1}}{2} (V_T^z)^2 \le B_T \text{ and } \mathcal{V}_T \le V_T^z + \frac{2L_g}{\sigma} \sqrt{\sum_{t=1}^T \|\epsilon_t\|} \tag{40}$$

From this result, we can see directly that when $\epsilon_t = \xi_t = 0, \forall t$, we get $\mathcal{R}_T \le 0$ since $B_T = 0$ and a_{T-1} is positive. And for the constraints, it holds:

$$V_T \le \sqrt{\frac{-2\mathcal{R}_T}{a_{T-1}}} = \mathcal{O}\left(T^{\frac{1+\beta}{2}} \right).$$

For worst-case predictions, we can drop the positive term $a_{T-1}(V_T^z)^2$ and write:

$$\mathcal{R}_T \le B_T = \mathcal{O}\left(\max\left\{ \sqrt{T}, T^{1-\beta} \right\} \right),$$

and replace $-\mathcal{R}_T \le FT = \mathcal{O}(T)$ in (40) and rearrange to obtain the bound for V_T^z and then using the relation of $\mathcal{V}_T$ to V_T (see (40)) to bound the former. Finally, it is interesting to observe that the derivation of the bounds did not require explicitly bounding the dual variables, and this stems from the fact that x_t is independent of the constraint perturbations.

I. Numerical Tests

Finally, we conclude by providing some simple, yet illuminating, numerical results comparing LLP with three competitor algorithms: the MOSP algorithm by Chen et al. [6]; our previous work Valls et al. [16]; and Sun et al. [10]. Figure 4 presents the first set of results. The algorithms run on the following cost and constraint functions:

$$f_t(x) = \begin{cases} -4x & \text{if } t \mod 2 = 0 \\ -x & \text{otherwise} \end{cases}.$$
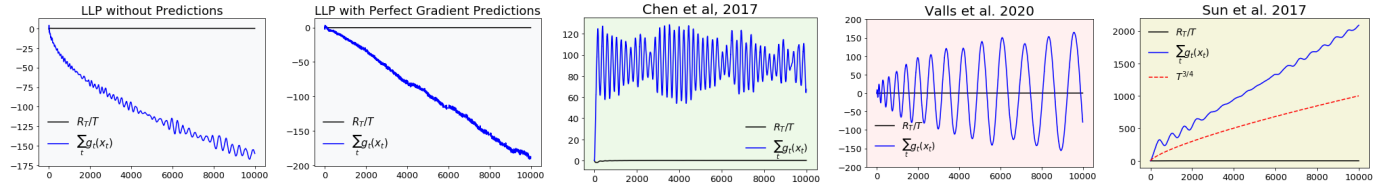
Fig. 5. Benchmark comparisons for: $f_t(x) = -2x$; $g_t(x) = x$ with probability $0.1/(t+1)^{0.05}$ and $g_t(x) = -0.01$ otherwise; $\mathcal{X} = [-1, 1]$. For the experiments presented here we have used $\beta = 0.2$.

and

$$g_t(x) = \begin{cases} 0.79x + 0.26 & \text{if } t \bmod 2 = 0 \\ 0.64x - 0.135 & \text{otherwise} \end{cases}.$$

where $x \in \mathcal{X} = [-1, 1]$. The code for this experiment can be found in *https://github.com/giosifid/LLP*, with a detailed description of the various involved parameters.

For the first LLP run we do not use any predictions, so as to demonstrate the efficacy of the algorithm even when no predictions are available. The second LLP plot runs the linearized version of the algorithm and uses perfect gradient predictions for the cost and constraint functions, but no predictions for the next constraint value. The three competitors have been optimized for $\mathcal{R}_T$, by using the steps and tuning parameters that are suggested in their respective references. We observe that LLP achieves lower regret from all competitors, and it reaches that point faster.

In the second experiment, we run the algorithms on the time-invariant cost function $f_t(x) = -2x$, $\forall t$, and constraint:

$$g_t(x) = \begin{cases} x & \text{with probability } \frac{0.1}{(t+1)^{0.05}} \\ -0.01 & \text{otherwise} \end{cases}.$$

where, again, $x \in \mathcal{X} = [-1, 1]$. Note that in this example we plot the total, not the average, constraint violation so as to shed light on the actual operation of each algorithm. We observe that LLP satisfies continuously the constraints in each t, while the competitors oscillate or fail to converge, despite that the cost function is constant.

The above results demonstrate that LLP performs quite well in practice, where even in these simple examples (one dimension space, time-invariant cost functions, etc.) it has clear advantages over the state-of-art competitors. For example, we see that it achieves fast lower regret points (first experiment); and is able to handle the probabilistic constraints in the second example – which is not surprising given that it uses a lazy dual update scheme which turns to be robust in such variations.

ACKNOWLEDGMENT

REFERENCES

[1] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proc. ICML*, 2003, pp. 928–936.

[2] G. S. Paschos, A. Destounis, and G. Iosifidis, "Online convex optimization for caching networks," *IEEE/ACM Trans. Netw.*, vol. 28, no. 2, pp. 625–638, Apr. 2020.

[3] B. Awerbuch and R. Kleinberg, "Online linear optimization and adaptive routing," *J. Comput. Syst. Sci.*, vol. 74, no. 1, pp. 97–114, Feb. 2008.

[4] L. Vigneri, G. Paschos, and P. Mertikopoulos, "Large-scale network utility maximization: Countering exponential growth with exponentiated gradients," in *Proc. IEEE INFOCOM*, Apr. 2019, pp. 1630–1638.

[5] M. J. Neely, "Stochastic network optimization with application to communication and queuing systems," *Synth. Lectures Commun. Netw.*, vol. 3, no. 1, p. 211, 2010.

[6] T. Chen, Q. Ling, and G. B. Giannakis, "An online convex optimization approach to proactive network resource allocation," *IEEE Trans. Signal Process.*, vol. 65, no. 24, pp. 6350–6364, Dec. 2017.

[7] S. Mannor, J. N. Tsitsiklis, and J. Y. Yu, "Online learning with sample path constraints," *J. Mach. Learn. Res.*, vol. 10, pp. 569–590, Mar. 2009.

[8] N. Liakopoulos, A. Destounis, G. Paschos, T. Spyropoulos, and P. Mertikopoulos, "Cautious regret minimization: Online optimization with long-term budget constraints," in *Proc. ICML*, 2019, pp. 3944–3952.

[9] X. Yi, X. Li, T. Yang, L. Xie, T. Chai, and K. H. Johansson, "Distributed bandit online convex optimization with time-varying coupled inequality constraints," *IEEE Trans. Autom. Control*, vol. 66, no. 10, pp. 4620–4635, Oct. 2021.

[10] W. Sun, D. Dey, and A. Kapoor, "Safety-aware algorithms for adversarial contextual bandit," in *Proc. ICML*, 2017, pp. 3280–3288.

[11] H. Yu, M. J. Neely, and X. Wei, "Online convex optimization with stochastic constraints," in *Proc. NIPS*, 2017, pp. 1–11.

[12] M. Mahdavi, R. Jin, and T. Yang, "Trading regret for efficiency: Online convex optimization with long term constraints," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 2503–2528, 2012.

[13] R. Jenatton, J. C. Huang, and C. Archambeau, "Adaptive algorithms for online convex optimization with long-term constraints," in *Proc. ICML*, 2016, pp. 402–411.

[14] J. Yuan and A. Lamperski, "Online convex optimization for cumulative constraints," in *Proc. NeurIPS*, 2018, pp. 1–10.

[15] N. Immorlica, K. A. Sankararaman, R. Schapire, and A. Slivkins, "Adversarial bandits with knapsacks," in *Proc. IEEE FOCS*, Nov. 2019, pp. 202–219.

[16] V. Valls, G. Iosifidis, D. Leith, and L. Tassiulas, "Online convex optimization with perturbed constraints: Optimal rates against stronger benchmarks," in *Proc. AISTATS*, 2020, pp. 2885–2895.

[17] T. Lykouris and S. Vassilvitskii, "Competitive caching with machine learned advice," in *Proc. ICML*, 2018, pp. 1–28.

[18] R. Kumar, M. Purohit, and Z. Svitkina, "Improving online algorithms via ML predictions," in *Proc. NeurIPS*, 2018, pp. 1–10.

[19] S. Lattanzi, T. Lavastida, B. Moseley, and S. Vassilvitskii, "Online scheduling via learned weights," in *Proc. ACM SODA*, 2020, pp. 1859–1877.

[20] E. Hazan, "Introduction to online convex optimization," *Found. Trends Optim.*, vol. 2, nos. 3–4, pp. 157–325, 2016.

[21] E. Hazan and N. Megiddo, "Online learning with prior knowledge," in *Proc. COLT*, 2007, pp. 499–513.

[22] O. Dekel, A. Flajolet, N. Haghtalab, and P. Jaillet, "Online learning with a hint," in *Proc. NIPS*, 2017, pp. 1–10.

[23] A. Bhaskara, A. Cutkosky, R. Kumar, and M. Purohit, "Online learning with imperfect hints," in *Proc. ICML*, 2020, pp. 822–831.

[24] A. Bhaskara, A. Cutkosky, R. Kumar, and M. Purohit, "Online linear optimization with many hints," in *Proc. NeurIPS*, 2020, pp. 9530–9539.

[25] S. Rakhlin and K. Sridharan, "Optimization, learning, and games with predictable sequences," in *Proc. NIPS*, 2013, pp. 1–9.

[26] M. Mohri and S. Yang, "Accelerating online convex optimization via adaptive prediction," in *Proc. AISTATS*, 2016, pp. 848–856.

[27] D. Raca et al., "On leveraging machine and deep learning for throughput prediction in cellular networks: Design, performance, and challenges," *IEEE Commun. Mag.*, vol. 58, no. 3, pp. 11–17, Mar. 2020.

[28] Y. Shen, Y. Shi, J. Zhang, and K. B. Letaief, "LORM: Learning to optimize for resource management in wireless networks with few training samples," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 665–679, Jan. 2020.

[29] F. Jiang, K. Wang, L. Dong, C. Pan, W. Xu, and K. Yang, "Deep-learning-based joint resource scheduling algorithms for hybrid MEC networks," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6252–6265, Jul. 2020.

[30] K. Chen and L. Huang, "Timely-throughput optimal scheduling with prediction," *IEEE/ACM Trans. Netw.*, vol. 26, no. 6, pp. 2457–2470, Dec. 2018.

[31] X. Huang et al., "Online VNF chaining and predictive scheduling: Optimality and trade-offs," *IEEE/ACM Trans. Netw.*, vol. 29, no. 4, pp. 1867–1880, Aug. 2021.

[32] N. Chen, A. Agarwal, A. Wierman, S. Barman, and L. L. H. Andrew, "Online convex optimization using predictions," in *Proc. ACM SIGMETRICS*, Jun. 2015, pp. 191–204.

[33] N. Chen, J. Comden, Z. Liu, A. Gandhi, and A. Wierman, "Using predictions in online optimization: Looking forward with an eye on the past," in *Proc. ACM SIGMETRICS*, Jun. 2016, pp. 193–206.

[34] Y. Lin, G. Goel, and A. Wierman, "Online optimization with predictions and non-convex losses," *Proc. ACM Meas. Anal. Comput. Syst.*, vol. 4, no. 1, pp. 1–32, Mar. 2020, Art. no. 18.

[35] H. B. McMahan, "A survey of algorithms and analysis for adaptive online learning," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 3117–3166, 2017.

[36] H. Yu and M. J. Neely, "A low complexity algorithm with $O(\sqrt{T})$ regret and $O(1)$ constraint violations for online convex optimization with long term constraints," *J. Mach. Learn.*, vol. 21, no. 1, pp. 1–24, 2020.

[37] G. Wang, Y. Wan, T. Yang, and L. Zhang, "Online convex optimization with continuous switching constraint," in *Proc. NeurIPS*, 2021, pp. 1–12.

[38] M. J. Neely and H. Yu, "Online convex optimization with time-varying constraints," 2017, *arXiv:1702.04783*.

[39] T. Chen, Q. Ling, Y. Shen, and G. B. Giannakis, "Heterogeneous online learning for 'thing-adaptive' fog computing in IoT," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4328–4341, Dec. 2018.

[40] X. Cao and K. R. Liu, "Online convex optimization with time-varying constraints and bandit feedback," *IEEE Trans. Autom. Control*, vol. 64, no. 7, pp. 2665–2680, Jul. 2018.

[41] Q. Liu, W. Wu, L. Huang, and Z. Fang, "Simultaneously achieving sublinear regret and constraint violations for online convex optimization with time-varying constraints," *Perform. Eval.*, vol. 152, Dec. 2021, Art. no. 102240.

[42] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proc. IEEE*, vol. 95, no. 1, pp. 255–312, Jan. 2007.

[43] L. Georgiadis, M. J. Neely, and L. Tassiulas, "Resource allocation and cross-layer control in wireless networks," *Found. Trends Netw.*, vol. 1, no. 1, pp. 1–144, 2006.

[44] D. Fudenberg and D. K. Levine, "Consistency and cautious fictitious play," *J. Econ. Dyn. Control*, vol. 19, nos. 5–7, pp. 1065–1089, Jul. 1995.

[45] N. Cesa-Bianchi and G. Lugosi, *Prediction, Learning, and Games*. Cambridge, U.K.: Cambridge Univ. Press, 2006, pp. 1–394.

[46] P. Auer, N. Cesa-Bianchi, and C. Gentile, "Adaptive and self-confident on-line learning algorithms," *J. Comput. Syst. Sci.*, vol. 64, no. 1, pp. 48–75, Feb. 2002.

Daron Anderson received the degree from the Department of Mathematics, Trinity College Dublin (TCD), Dublin, Ireland, in 2014, and the Ph.D. degree from the National University of Ireland, Galway, Ireland, in 2019. He is currently a Research Fellow with TCD, working on online learning algorithms. He received a scholarship from the Irish Research Council for his Ph.D. degree.

George Iosifidis received the Diploma degree in electronics and telecommunications engineering from the Greek Air Force Academy, Athens, in 2000, and the Ph.D. degree from the University of Thessaly, Volos, Greece, in 2012. From 2016 to 2020, he was an Assistant Professor with Trinity College Dublin, Dublin, Ireland. He is currently an Assistant Professor with the Delft University of Technology (TU Delft), The Netherlands. His research interests include network optimization and economics. More information can be found at: www.FutureNetworksLab.net

Douglas J. Leith (Senior Member, IEEE) received the master's and Ph.D. degrees from the University of Glasgow, Glasgow, Scotland, in 1986 and 1989, respectively. He moved to the National University of Ireland, Maynooth, Ireland, in 2001, to establish the Hamilton Institute (www.hamilton.ie), where he was the Founding Director from 2001 to 2014. Toward the end of 2014, he moved to the School of Computer Science and Statistics, Trinity College Dublin, to take up the Chair in Computer Systems. His current research interests include wireless networks, congestion control, optimization, and data privacy.