

Adaptive Gradient Online Control

Deepan Muthirayan, Jianjun Yuan, Pramod P. Khargonekar

Abstract—In this work we consider the online control of a known linear dynamic system with adversarial disturbance and adversarial controller cost. The goal in online control is to minimize the regret, defined as the difference between cumulative cost over a period T and the cumulative cost for the best policy from a comparator class. For the setting we consider, we generalize the previously proposed online Disturbance Response Controller (DRC) to the adaptive gradient online Disturbance Response Controller. Using the modified controller, we present novel regret guarantees that improves the established regret guarantees for the same setting. We show that the proposed online learning controller is able to achieve intermediate intermediate regret rates between \sqrt{T} and $\log T$ for intermediate convex conditions, while it recovers the previously established regret results for general convex controller cost and strongly convex controller cost.

Index Terms—Online control, adversarial cost, regret, disturbance response controller, adaptive gradient descent

I. INTRODUCTION

Control of systems with uncertainties is a central challenge in control and is an extensively researched topic. There are various sub-fields in control such as stochastic control [6], [19], robust control [27] and adaptive control [17], [24] that address the challenge of controller synthesis for different types of uncertainties. In this work we are concerned with the problem of online control of systems with uncertainties such as disturbance and adversarial controller cost. The performance in online control is measured in terms of how the regret of performance, defined as the deviation of the performance from that of the best policy, scales with the duration T . The objective in online control is to design adaptive algorithms to disturbances and adversarial cost so that the regret scales sub-linearly in T , i.e., as T^α with $\alpha < 1$.

Classical adaptive control investigates the problem of control of systems with parametric, structural and parametrizable disturbance uncertainties [28]. The main focus in classical adaptive control is the stability of system and asymptotic tracking performance. Adaptive control has been studied for systems of all types such as linear, non-linear, and stochastic. There are many variants of adaptive control such as adaptive model predictive control [16], [20], adaptive learning control [22], [29], stochastic adaptive control [7] and robust adaptive control [17]. These variations address the design of adaptive controller for different variations of the basic adaptive control setting. Many papers and books have been written on adaptive control; see for example [7], [17], [24]. Thus, adaptive control

is a very rich and extensively studied topic. The key variation of the online control setting from the classical adaptive control is the regret objective and in some cases the general nature of the costs, where they could be adversarial and unknown a priori. Thus, the classical adaptive control approaches can be inadequate to analyse online control problems and are typically solved by merging tools from statistical learning, online learning and optimization, and control theory.

The field of online control has seen rising interest in the last few years. One of the first setting that was extensively explored is the Linear Quadratic Regulator (LQR) with the unknown system and stochastic disturbances. Abbasi & Czepesvari [1] were the first to study the online LQR problem with unknown system and stochastic disturbances. The authors proposed an adaptive algorithm that achieved \sqrt{T} regret w.r.t the best linear control policy, which is the optimal policy. After [1], several authors improved the algorithm of [1], which was an inefficient algorithm. Dean et al. [11] were the first to propose an efficient algorithm for the same problem. They showed that their algorithm achieved a regret of $\mathcal{O}(T^{2/3})$. Cohen et al. [10] and Mania et al. [21] improved on this result by providing an efficient algorithm with a regret guarantee of $\mathcal{O}(T^{1/2})$ for the same problem. Mania et al. [21] extended these results to the partial observation setting and established $\mathcal{O}(\sqrt{T})$ -regret for the partially observed Linear Quadratic Gaussian (LQG) setting. Cohen et al. [9] provided an $\mathcal{O}(\sqrt{T})$ algorithm for a variant of the online LQR, where the system is known and noise is stochastic but the controller cost function is an adversarially chosen quadratic function. Recently, Simchowitz et al. [25] showed that $\mathcal{O}(T^{1/2})$ is the optimal regret for the online LQR control problem.

While the above works focussed on online LQR, there are others who studied the control of much general systems: linear dynamic systems with adversarial disturbances and adversarial cost functions. Agarwal et al. [3] considered the control of a known linear dynamic system with additive adversarial disturbance and an adversarial convex controller cost function. They proposed an online learning algorithm that learnt a Disturbance Response Controller (DRC): a linear feedback of the portion of the output contributed by the disturbances upto certain history. They showed that their proposed controller achieves $\mathcal{O}(\sqrt{T})$ -regret with respect to the best DRC in hindsight. Agarwal et al. in a subsequent work [4] showed that a poly logarithmic regret is achievable for strongly convex controller cost and well conditioned stochastic disturbances. Hazan et al. [13] extended the setting of [3] to the case where the system is unknown. They showed that when the system is unknown, while $\mathcal{O}(\sqrt{T})$ -regret is not achievable, they can still achieve a sub-linear regret of $\mathcal{O}(T^{2/3})$ -regret. Recently, [26] generalized these results to provide similar regret guarantees for the same setting with partial observation for both known and unknown

This work is supported in part by the National Science Foundation under Grant ECCS-1839429. D. Muthirayan and P. P. Khargonekar are with the Department of Electrical Engineering and Computer Sciences, University of California Irvine, Irvine, CA (emails: deepan.m@uci.edu, pramod.khargonekar@uci.edu). Jianjun Yuan is with the Expedia Group (email: yuanx270@umn.edu).

systems.

In this work we study the online control setting of [26]: linear dynamic systems with additive disturbance and adversarial controller cost, where the system state is only partially observable. We assume that our system is known and our cost functions are general convex controller costs. Previous works in the online adversarial setting [3], [4], [13], [26], either assume the cost functions to be convex or strongly-convex. Reiterating the results of [26] for the known system case, what has been established is that $\mathcal{O}(\sqrt{T})$ regret is achievable when the cost function are convex, and $\mathcal{O}(\log T)$ regret is achievable when the cost functions are strongly convex. The question we address in this work is: *can we achieve intermediate regret guarantees for intermediate convex conditions?*

A. Our Contribution

The online control algorithm we propose is the adaptive gradient extension of the online learning disturbance response controller proposed in [3], [26]. Here the adaptive gradient refers to the adaptation of the gradient step size of the gradient learning algorithm used in [3], [26]. Thus, to the best of our knowledge, we present the *first adaptive gradient online learning control algorithm*. We show that the proposed learning algorithm *recovers the previously established regret guarantee of $\mathcal{O}(\sqrt{T})$ for general convex controller cost functions and $\mathcal{O}(\log T)$ for strongly-convex and smooth controller cost functions (see [26]), and simultaneously achieves an intermediate regret between $\mathcal{O}(\sqrt{T})$ and $\mathcal{O}(\log T)$ for intermediate convex conditions of the controller cost functions*. We prove our main result by establishing a new result for adaptive gradient online learning for the problem of Online Convex Optimization with Memory (OCO-M), which is the online convex optimization problem where the cost at a time step also depends on a certain history of past decisions.

B. Other Related Work

Online Convex Optimization (OCO): In the OCO framework, the learner encounters a sequence of convex loss functions which are unknown beforehand and may vary arbitrarily over time. The learner updates the estimate of the optimal solution at each time-step based on the previous losses and incurs a loss for its updated estimate as given by the loss function for this time step. At the end of each step, either the loss function may be revealed, a scenario referred to as full information feedback, or only the experienced loss is revealed, a scenario known as bandit feedback. The objective of the learner is to minimize the loss accumulated over time. Under the full information feedback setting, it has been established that the best possible regret scales as $\mathcal{O}(T^{1/2})$ (resp. $\mathcal{O}(\log T)$) for convex (resp. strongly convex) loss functions, where T is the number of time steps [2], [14], [32]. These results have also been extended to constrained online convex optimization where it has been shown that the best regret scales as $\mathcal{O}(T^{\max\{c, 1-c\}})$ for the cost and $\mathcal{O}(T^{1-c/2})$ for constraint violation, where c is a constant [18], [30]. When compared to OCO, the key difference in online control is the dependence of the decision on the state of the system, and thus in online

control what is to be learnt is a control policy instead of a single decision.

Policy Optimization: Fazel et al. [12] proved that the policy gradient based learning converges asymptotically to the optimal policy for the Linear-Quadratic Regulator (LQR) problem. Zhange et al. [31] extended this result to the $\mathcal{H}_2/\mathcal{H}_\infty$ control problem. Recently, [23] proved asymptotic convergence of a gradient based meta-learner for the LQR problem. All of these works provide asymptotic convergence guarantees.

Notation: We denote the transpose of a vector X by X^\top . We denote the expectation of a random variable X by $\mathbb{E}[X]$ and the expectation w.r.t a filtration \mathcal{F}_t by $\mathbb{E}[\cdot|\mathcal{F}_t]$. The minimum singular value of a matrix M is denoted by $\sigma_{\min}(M)$ and the minimum eigen value is denoted by $\lambda_{\min}(M)$. The function $\rho(\cdot)$ denotes the spectral radius of the input matrix. We define $\|\cdot\|$ to be 2-norm of the vector or the matrix as the case maybe. For a given variable X_t that is dependent on time t , $X_{t_1:t_2}$ is used to denote the sequence $(X_{t_1}, X_{t_1+1}, \dots, X_{t_2})$. By $\sum X_{t_1:t_2}$, we denote the sum of the elements in the sequence $X_{t_1:t_2}$. The big $\mathcal{O}(\cdot)$ is the standard order notation and $\tilde{\mathcal{O}}(\cdot)$ is the standard order notation that includes polylog factors.

II. PROBLEM PRELIMINARIES

The problem we consider is the online control of a linear dynamical system given by

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t + w_t, \\ y_t &= Cx_t + e_t, \end{aligned} \quad (1)$$

where $x_t \in \mathbb{R}^{d_x}$, is the state of the system, $u_t \in \mathbb{R}^{d_u}$, is the control input generated by the controller, w_t, e_t are bounded disturbances of appropriate dimensions and $y_t \in \mathbb{R}^{d_y}$, is the observed output. The objective is to regulate the response of this system so as to achieve sub-linear regret with respect to the best policy from a class of policies, also called the comparator policy.

The class of policies we consider for the comparator are *linear dynamic controllers*, denoted by Π . A linear dynamic controller $\pi \in \Pi$ is a linear dynamic system given by $(A_\pi, B_\pi, C_\pi, D_\pi)$ with the internal state $s_t^\pi \in \mathbb{R}^{d_\pi}$ and output being the control input at time t :

$$s_{t+1}^\pi = A_\pi s_t^\pi + B_\pi y_t, u_t^\pi = C_\pi s_t^\pi + D_\pi y_t \quad (2)$$

We denote the online controller for the system in Eq. (1) by \mathcal{C} . The controller at any point has only access to the following information at time t : (i) all prior cost functions $c_{1:t-1}$, (ii) all prior observations $y_{1:t-1}$, and (iii) all prior control inputs $u_{1:t-1}$. The controller, unlike the classical setting, does not have access to the future cost functions, which are adversarial. The controller has to choose a policy to compute the control action at time t based on this information.

The online control setting of ours is the following: The controller \mathcal{C} , on applying the control input u_t at time t , suffers the loss $l_t(y_t, u_t)$, an adversarially chosen convex function, which is a priori unknown. The controller can observe the loss function only after its decision at time step t . The controller can then use this information to update its control policy.

The performance of the online controller is measured by the regret which is the total cost incurred by the controller for a duration T minus the total cost incurred by the best controller in hindsight taken from the class of controllers Π . Denote the system output and the input corresponding to a controller $\pi \in \Pi$ by (y_t^π, u_t^π) . Let $J_T(\pi) = \sum_{t=1}^T l_t(y_t^\pi, u_t^\pi)$, $\pi \in \Pi$. Then, the regret for the controller \mathcal{C} is given by

$$R_T(\mathcal{C}) = \mathbb{E}[J_T(\mathcal{C})] - \min_{\pi \in \Pi} \mathbb{E}[J_T(\pi)]. \quad (3)$$

A. Assumptions

We state the assumptions we make below.

Assumption 1: The system is stable, i.e., $\rho(A) < 1$. The system matrices A, B are known.

The assumptions on the spectral radius (or the assumption that there is additional knowledge of a feedback rule to stabilize the system) are standard in online learning and control problems [1], [10], [11], [26]. We emphasize that analysis without stability or the knowledge of a stabilizing feedback law is still an hard and open challenge in online control. While there are works that investigate simultaneous safe exploration and control such as in Reinforcement Learning [8], these works do not study the finite performance objective such as regret.

Assumption 2: The noise w_t and e_t are bounded and stochastic i.i.d. Their distribution is known and $\mathbb{E}[w_t^s] = 0$, $\mathbb{E}[e_t^s] = 0$.

Assumption 3: The loss function l_t is convex and for $z^\top = [y_t^\top, u_t^\top]$, $(z')^\top = [(y')^\top, (u')^\top]$ such that $R = \max\{\|z\|, \|z'\|, 1\}$, $\|l_t(y_t, u_t) - l_t(y', u')\| \leq LR\|z - z'\|$.

The convexity assumption is standard in online learning and optimization and online control settings. Most of online control especially the setting with general adversarial cost functions and disturbances are built on tools from online convex analysis. This is because the tools for online optimization analysis have been well understood and developed for the convexity setting and such analysis for general non-convex cost setting are still non-existent. The second part of the assumption states that the loss functions are locally Lipschitz. We note that the assumptions stated here are exactly the assumptions in the state-of-the-art work in online control [26].

III. ONLINE CONTROL ALGORITHM

The online control algorithm we propose for the general controller \mathcal{C} is the *adaptive gradient* version of the online DRC (or DRC-GD) proposed in [26]. We call this the *disturbance response controller - adaptive gradient descent* (DRC-AGD). We briefly review the online DRC in [26], and then present the DRC-AGD algorithm.

A. Online Disturbance Response Controller

Let's define y_t^{nat} to be the natural output, the system output when the control inputs are zero, i.e.,

$$\begin{aligned} y_t^{nat} &= e_t + \sum_{s=0}^{t-1} CA^{t-s-1}w_s \\ &= y_t - \sum_{s=1}^{t-1} G^{[s]}u_{t-s}, \quad G^{[s]} = CA^{s-1}B. \end{aligned}$$

Since e_t, w_t are bounded for all t and $\rho(A) < 1$, y_t^{nat} is bounded for all t . We define R_{nat} to be the bound on y_t^{nat} . The DRC as defined in [26] is parameterized by a m -length sequence of matrices, denoted by $M = (M^{[i]})_{i=0}^{m-1}$. The DRC's control decision is given by

$$u_t = \sum_{s=0}^{m-1} M^{[s]}y_{t-s}^{nat}. \quad (4)$$

Let's define the following class of disturbance response controllers:

$$\mathcal{M}(m, R) = \left\{ M = (M^{[s]})_{s=0}^{m-1} : \|M\| = \sum_s \|M^{[s]}\| \leq R_M \right\} \quad (5)$$

The online learning algorithm or the DRC-GD proposed in [26] continuously updates the feedback gain M as the loss functions are revealed. It applies the control input as defined in Eq. (4) with the current value of the feedback gain M . The algorithm then updates the feedback gain M based on the revealed loss function, similar to how the decision is updated in OCO. Thus the disturbance feedback gain M is equivalent to the decision in OCO.

For the choice of regret as defined in Eq. (3), the disturbance response controller is a good choice given that the best disturbance response controller for the realized sequence of cost functions is approximately equal to the best linear dynamic controller. We will show this in the proof of our main result. Thus, by learning the disturbance response controller online the controller can get closer to the optimal linear dynamic controller. We pick the control structure as DRC instead of linear dynamic controller because the DRC control form has advantages from the point of view of online regret analysis. It enables the regret analysis to be approximated by the regret analysis of a limited memory problem, where memory refers to the number of past controller parameters the realized cost at a time t is dependent on. This will not be feasible with the linear dynamic control structure because the control input computed by a linear dynamic controller at any point of time is dependent on the entire history of control inputs unlike Eq. (4).

We introduce the following definitions for ease of presentation. Let $M^{[s]}(j)$ denote the j th row of the $M^{[s]}$ matrix. Let $z(i : j)$ denote the sub-vector of the vector z corresponding to the elements from i to j . Let P denote the vector given by $P(sq + (j-1)d_y + 1 : sq + jd_y) = (M^{[s]}(j))^\top$, where $q = d_y d_u$, $1 \leq j \leq d_u$. Essentially, this defines P to be the vector of the transposes of the rows of $M^{[s]}$ stacked one above the other. We introduce the following definitions that will be required for discussing the algorithms.

Definition 1: $u_t[M_t|y_{1:t}^{nat}] := \sum_{s=0}^{m-1} M_t^{[s]}y_{t-s}^{nat}$,
 $\tilde{y}_t[P_{t:t-h}|y_{1:t}^{nat}] := y_t^{nat} + \sum_{s=1}^h G^{[s]}u_{t-s}$,
 $F_t[P_{t:t-h}|y_{1:t}^{nat}] := l_t(\tilde{y}_t[P_{t:t-h}|y_{1:t}^{nat}], u_t[M_t|y_{1:t}^{nat}])$,
 $f_t(P|y_{1:t}^{nat}) := F_t[\{P, P, \dots, P\}|y_{1:t}^{nat}]$.

The term \tilde{y}_t is an approximate output that depends only on the past h control inputs. Consequently this approximate estimate is only a function of $P_{t:t-h}$ for a given $y_{1:t}^{nat}$. The function F_t is the loss l_t evaluated for this approximate output \tilde{y}_t and so

it is also only a function of $P_{t:t-h}$. The function f_t is the loss F_t when P_k , for all k s.t. $t \geq k \geq t-h$ is fixed to P , and so we term it as the memory-less loss.

Minimizing the regret (Eq. (3)) is an Online Convex Optimization problem with Memory (OCO-M) [5] because the loss function at a time step depends on the past control inputs, which is the case even with the approximated cost $F_t[P_{t:t-h}]$, a function of the truncated output \tilde{y}_t . Following the key idea in [5], the DRC-GD algorithm [26] uses the gradient of the memory-less function $f_t(\cdot)$ to update P . This, as can be expected, only minimizes the regret of $\sum f_t(\cdot)$ instead of the approximated cost $F_t[P_{t:t-h}]$. But as shown in [5], the memory-less regret closely approximates the regret of the approximated cost $F_t[P_{t:t-h}]$, which in turn, as we show later, is a good approximation of the regret of the actual realized cost.

Let $\mathcal{P}(m, R) = \left\{ P : \sum_{s=0}^{m-1} \|M^{[s]}\| \leq R_M \right\}$. The learning algorithm for the online DRC proposed in [26] initializes P to an element drawn from the set $\mathcal{P}(m, R)$. It then updates P along the gradient of the memory-less loss function $f_t(\cdot)$ as the loss functions (or cost) are revealed to continuously improve the feedback controller:

$$P \leftarrow \text{Proj}_{\mathcal{M}} \left(P - \eta_{t+1} \partial f_t \left(P | y_{1:t}^{nat} \right) \right). \quad (6)$$

In [26], the authors show that the disturbance response controller with the memory-less gradient update given by Eq. (6), where η_t is fixed to a particular value (see Theorem 2, [26]), achieves a regret of $\tilde{O}(\sqrt{T})$ when the cost functions are general convex functions and $\text{polylog}(T)$ when the cost functions are smooth and strongly convex. In this work, we extend this online DRC controller by using an adaptive step rate akin to [15] instead of a fixed step rate η . We discuss our extended algorithm in the next section.

B. Online Disturbance Response Controller: DRC-AGD

In this section, we present the DRC-AGD algorithm. First, we briefly review the adaptive gradient online learning algorithm [15] for the standard OCO problem and then present our new regret result for adaptive gradient learning for the OCO-M problem. We then introduce our DRC-AGD online control algorithm and use its result to analyse the regret of the DRC-AGD algorithm.

1) *Adaptive Gradient Online Learning*: Consider the standard online convex optimization (OCO) setting (see [15]). At time t , the player chooses an action u_t from some convex subset \mathcal{K} of \mathbb{R}^n , where $\max_{x \in \mathcal{K}} \|x\| \leq D$, and the adversary chooses a convex loss function $f_t(\cdot)$. The regret for the player over duration T is given by

$$R_T = \sum_{t=1}^T f_t(u_t) - \min_{u \in \mathcal{K}} \sum_{t=1}^T f_t(u) \quad (7)$$

Let f_t be H_t -strongly convex, i.e., let $f_t(u^*) \geq f_t(u) + \nabla f_t(u^* - u) + \frac{H_t}{2} \|u^* - u\|_2^2$ and $\|\nabla f_t\| \leq G_t$. Once the loss function is revealed at time t the algorithm can use the loss function to update its decision. The adaptive gradient online

learning algorithm proposed in [15] updates the decision u_t by the following gradient step:

$$u_{t+1} = \text{Proj}_{\mathcal{K}} \left(u_t - \eta_{t+1} \partial (f_t(u) + g_t(u)) \right) \\ \eta_{t+1} = \frac{1}{\sum H_{1:t} + \sum \lambda_{1:t}}, \quad (8)$$

where $\sum H_{1:t} = \sum_{k=1}^t H_k$, $\sum \lambda_{1:t} = \sum_{k=1}^t \lambda_k$, and λ_{t_s} are suitably defined parameters. Here, it is clear that the step rate at each time step is updated by the strong convexity H_t of the loss function at t as defined above. Thus the step rate is adapted and the algorithm is adaptive gradient online learning. The regret for this algorithm can be characterized as in the following Lemma.

Lemma 1: Consider the online update given by Eq. (8) with $g_t(u) = 1/2 \lambda_t \|u\|_2^2$. Then for any sequence of $\lambda_1, \lambda_2, \dots, \lambda_T$,

$$R_T \leq \frac{1}{2} D^2 \lambda_{1:T} + \frac{1}{2} \sum_{t=1}^T \frac{(G_t + \lambda_t D)^2}{\sum H_{1:t} + \sum \lambda_{1:t}}, \quad (9)$$

Please see Theorem 3.1. [15] for the proof. This is the basic result that the regret rate results in [15] are based on. Here, the parameters $\lambda_{1:T}$ can be suitably chosen based on the convex conditions to achieve intermediate regret rates for intermediate convex conditions of the sequence of loss functions; for example, conditions such as $H_t \propto t^{-\alpha}$. We direct the reader to [15] for a more detailed discussion of their results.

2) *Adaptive Gradient Online Learning for OCO-M*: In this section we discuss the extension of the adaptive gradient learning to the OCO-M problem. The difference in the OCO-M setting is that the cost function at a particular time t is also dependent on a certain history of the past decisions. More specifically, the cost functions f_t in OCO-M are a function of the decisions upto h time steps in the past, i.e., $u_{t:t-h}$, where h is a given number. Thus, the regret in the OCO-M problem is the following:

$$R_T = \sum_{t=1}^T f_t(u_{t:t-h}) - \min_{u \in \mathcal{K}} \sum_{t=1}^T f_t(u), \quad (10)$$

where we used $f_t(u)$ as a shorthand notation for the cost when $u_{t-k} = u$, for all k , where $0 \leq k \leq h$. In the next theorem we present the equivalent of Lemma 1 for the OCO-M problem, which we will use to analyse our main algorithm.

Theorem 1: For a sequence of $(h+1)$ -variate F_t define $f_t(u) = F_t(u, u, \dots, u)$. Let G_c be an upper bound on the coordinate wise Lipschitz constant of F_t , G_f be an upper bound on the Lipschitz constant of f_t , f_t be H_t -strongly convex, and D be an upper bound on the diameter of \mathcal{K} . Consider the online update given by Eq. (8), with $g_t(u) = 1/2 \lambda_t \|u\|_2^2$. Then for any sequence of $\lambda_1, \lambda_2, \dots, \lambda_T$, $\lambda_j \leq \lambda_i, j \geq i$,

$$R_T = \sum_{t=h+1}^T F_t(u_t, \dots, u_{t-h}) - \min_{u \in \mathcal{K}} \sum_{t=h+1}^T F_t(u, \dots, u) \\ \leq \frac{1}{2} D^2 \lambda_{1:T} + \frac{1}{2} \sum_{t=1}^T \frac{\tilde{G}_{f,t}^2}{\sum H_{1:t} + \sum \lambda_{1:t}},$$

where $\tilde{G}_{f,t} = \sqrt{(G_f + \lambda_t D)(G_f + \lambda_t D + 2G_c h^3/2)}$.

Please see Appendix for the proof.

3) *Adaptive Gradient Online Learning for Control*: Here, we extend the adaptive gradient descent learning idea to the online DRC. The gradient learning algorithm we propose, which we call as DRC-AGD, is the extension of Eq. (6) with an adaptive step rate similar to Eq. (8):

$$\begin{aligned} P_{t+1} &= \text{Proj}_{\mathcal{P}}(P_t - \eta_{t+1} \partial (\mathbb{E}[f_t[P_t|y_{1:t}^{nat}]] + g_t(P_t))) \\ g_t(P) &= \frac{1}{2} \lambda_t \|P\|_2^2, \quad \eta_{t+1} = \frac{1}{\sum_{s=1}^t H_{1:s} + \sum_{s=1}^t \lambda_{1:s}}, \end{aligned} \quad (11)$$

where the update is by the gradient of the memory-less cost $\mathbb{E}[f_t[P_t|y_{1:t}^{nat}]]$, with an adaptive step rate η_{t+1} , where H_t is the strong convexity of $\mathbb{E}[f_t[P_t|y_{1:t}^{nat}]]$ and λ_t s are suitably chosen parameters as before.

Algorithm 1 Disturbance Response Control - Adaptive Gradient Descent (DRC-AGD)

Input: Radius R_M , and the matrices $G^{[i]}$, h .

- 1 Initialize $P_1 \in \mathcal{P}$
 - 2 **for** $t = 1, \dots, T$ **do**
 - 3 Observe y_t and determine $y_t^{nat} = y_t - \sum_{i=1}^{t-1} G^{[i]} u_{t-i}$
 - 4 Choose $u_t = \sum_{s=0}^{m-1} M_t^{[s]} y_{t-s}^{nat}$
 - 5 Observe the loss function and suffer the loss $l_t(y_t, u_t)$
 - 6 Set $\eta_{t+1} = \frac{1}{\sum_{s=1}^t H_{1:s} + \sum_{s=1}^t \lambda_{1:s}}$
 - 7 $P_{t+1} = \text{Proj}_{\mathcal{P}}(P_t - \eta_{t+1} \partial (\mathbb{E}[f_t[P_t|y_{1:t}^{nat}]] + \frac{1}{2} \lambda_t \|P_t\|_2^2))$
 - 8 **end**
-

Algorithm 1 presents the full DRC-AGD algorithm.

4) *Main Results*: In DRC-AGD, the gradient of the memory-less cost $\mathbb{E}[f_t[P_t|y_{1:t}^{nat}]]$ is used. Hence, to apply Theorem 1 to the analysis of the DRC-AGD algorithm, we need to establish the strong convexity of $\mathbb{E}[f_t[P_t|y_{1:t}^{nat}]]$. We also need to establish that G_c and G_f exist for the memory-less cost $\mathbb{E}[f_t[P_t|y_{1:t}^{nat}]]$; we prove all of this as part of the main theorem. In the next lemma we characterize the strong convexity of $\mathbb{E}[f_t[P_t|y_{1:t}^{nat}]]$ in terms of the strong convexity H_t^l of l_t (recall how f_t is dependent on l_t in Definition 1).

Lemma 2: The function $\mathbb{E}[f_t[P_t|y_{1:t}^{nat}]]$ is H_t -strongly convex, where

$$H_t = H_t^l \left(\sigma_e^2 + \sigma_w^2 \left(\frac{\sigma_{\min}(C)}{1 + \|A\|_2^2} \right)^2 \right),$$

$$\nabla^2 l_t \geq H_t^l, \quad \mathbb{E}[w_t^s w_t^s] \geq \sigma_w^2, \quad \mathbb{E}[e_t^s e_t^s] \geq \sigma_e^2.$$

Please see Proposition 7.1, [26] for the proof. We introduce an additional definition before we discuss our main theorem.

Definition 2: $\psi(i) = \sum_{j \geq i} \|CA^{j-1}B\|_2, i > 0$. Since $\rho(A) < 1$, there exists $c > 0$ and $\rho \in (0, 1)$ such that $\psi(i) \leq C\rho^i$. $R_{G^*} = 1 + \psi(1)$.

In the next theorem we use Theorem 1 to characterize the regret for the DRC-AGD online control algorithm.

Theorem 2: Suppose Assumptions 1, 2, 3 hold. Suppose the algorithm 1 is run with $m, h \geq 1$ such that $\psi(m) \leq$

$R_{G^*}/T, \psi(h) \leq R_M/T$ then

$$\begin{aligned} R_T(\mathcal{C}) &\leq R_M^2 R_{G^*}^2 R_{nat}^2 (6L + 4(m+h)) \\ &+ \frac{1}{2} D^2 \lambda_{1:T} + \frac{1}{2} \sum_{t=1}^T \frac{(\tilde{G}_{f,t})^2}{\sum H_{1:t} + \sum \lambda_{1:t}}, \quad \text{where} \end{aligned}$$

$$G_f = G_C = L\sqrt{m}R_M R_{G^*} R_{nat}^2, \quad D = 2\sqrt{\min\{d_u, d_y\}}R_M,$$

$$\tilde{G}_{f,t} = \sqrt{(G_f + \lambda_t D)(G_f + \lambda_t D + 2G_c h^3/2)}.$$

Please see the Appendix for the proof. The proof proceeds by splitting the regret (Eq. (3)) into several terms; the burn-in loss, algorithm truncation error, f-policy error, comparator truncation error and the policy approximation error. This splitting follows the proof technique in [26]. The burn-in loss is just the realized cost corresponding to the first $m+h$ time steps. The burn-in loss can be trivially bounded (see for example Lemma 5.2. [26]). The algorithm truncation error is the difference between the realized cost for the remaining horizon and the cost that would be realized with the truncated output approximation \tilde{y}_t , i.e., $\sum F_t$. We recall that the output is truncated so that it depends only on the past h control inputs; see Definition 1 for the truncated output \tilde{y}_t and the corresponding loss F_t . This splitting is done because Theorem 1 can only be applied to fixed length memory while the actual realized cost is dependent on the entire history of control inputs. The f-policy error is the difference between the cost $\sum F_t$, which is the approximate cost by truncating the memory, and the same cost when $P_k = P \forall k$. Thus, the f-policy error is given by $\sum_{t=m+h+1}^T \mathbb{E}[F_t(P_{t:t-h}|y_{1:t}^{nat})] - \inf_P \sum_{t=m+h+1}^T \mathbb{E}[f_t(P|y_{1:t}^{nat})]$. Given the form of this regret term, we can apply Theorem 1 to bound the f-policy error. We note that the approximated cost with truncated memory under fixed P is different from the realized cost under a fixed disturbance response controller P . This introduces the comparator truncation error, the difference of the two costs, i.e., $\inf_P \sum_{t=m+h+1}^T \mathbb{E}[f_t(P|y_{1:t}^{nat})] - \inf_P \sum_{t=m+h+1}^T \mathbb{E}[l_t(y_t^P, u_t^P)]$. The policy approximation error is the difference between the realized cost for the best fixed P disturbance response controller and the cost for the best linear dynamic controller. The truncation errors and policy approximation error can also be bounded (see [26]). We give details of bounding the burn-in loss, truncation errors and the policy approximation error in the Appendix. Putting together the bounds of all these terms gives us the final result.

We note that the regret bound for DRC-AGD has terms similar to the regular adaptive gradient algorithm (see Lemma 1). Given this result, we can apply the analysis similar to [15] to establish regret scaling for various convex conditions. In the next corollary we discuss the specific scaling of the regret w.r.t T under various convex conditions and in particular show that the DRC-AGD algorithm interpolates between $T^{1/2}$ and $\log T$.

Corollary 1: Suppose Assumptions 1, 2, 3 hold. Suppose the algorithm 1 is run with $m, h \geq 1$ such that $\psi(m) \leq R_{G^}/T, \psi(h) \leq R_M/T, T \geq 4$ then*

- 1) for any sequence of convex loss functions l_t

$$R_T \leq \tilde{O}(\sqrt{T})$$

2) for any sequence of convex loss functions l_t with $H_t^l \geq H$

$$R_T \leq \tilde{\mathcal{O}}(\log T)$$

3) for $H_t^l = t^{-\alpha}$, and $0 < \alpha \leq 1/2$

$$R_T \leq \tilde{\mathcal{O}}(T^\alpha)$$

4) for $H_t^l = t^{-\alpha}$, and $\alpha > 1/2$

$$R_T \leq \tilde{\mathcal{O}}(\sqrt{T})$$

Please see the Appendix for the proof. We see that the DRC-AGD algorithm recovers the $\mathcal{O}(\sqrt{T})$ and $\mathcal{O}(\log T)$ result for strongly convex and general convex cost functions and at the same time achieves intermediate regret scaling for intermediate convex conditions. We emphasize that the regret scaling of $\tilde{\mathcal{O}}(T^\alpha)$ is valid for a more general condition such as $\sum H_{1:t} \geq t^{1-\alpha}$.

IV. CONCLUSION

In this work we considered the online control of a known linear dynamic system with adversarial disturbances and adversarial cost functions. Our objective is to improve regret rates established for this setting by prior works, which only considered either convex costs or strongly convex costs. Specifically, we addressed the question whether the regret rates can be improved when the convexity of controller cost functions are intermediate, i.e., between strongly convex and convex.

We proposed an adaptive gradient extension of the disturbance response controller proposed in prior works for the same problem we study. We proved that the proposed online learning controller recovers the previously established regret guarantee of $\mathcal{O}(\sqrt{T})$ for general convex controller cost functions and $\mathcal{O}(\log T)$ for strongly-convex and smooth controller cost functions (see [26]), and achieves an intermediate regret between $\mathcal{O}(\sqrt{T})$ and $\mathcal{O}(\log T)$ for intermediate convex conditions for the controller cost functions.

REFERENCES

- [1] Yasin Abbasi-Yadkori and Csaba Szepesvári. Regret bounds for the adaptive control of linear quadratic systems. In *Proceedings of the 24th Annual Conference on Learning Theory*, pages 1–26. JMLR Workshop and Conference Proceedings, 2011.
- [2] Jacob Abernethy, Alekh Agarwal, and Peter L Bartlett. A stochastic view of optimal regret through minimax duality. In *Proceedings of the 22nd Annual Conference on Learning Theory*, 2009.
- [3] Naman Agarwal, Brian Bullins, Elad Hazan, Sham Kakade, and Karan Singh. Online control with adversarial disturbances. *International Conference on Machine Learning*, pages 111–119, 2019.
- [4] Naman Agarwal, Elad Hazan, and Karan Singh. Logarithmic regret for online control. In *Advances in Neural Information Processing Systems*, pages 10175–10184, 2019.
- [5] Oren Anava, Elad Hazan, and Shie Mannor. Online learning for adversaries with memory: price of past mistakes. In *Advances in Neural Information Processing Systems*, pages 784–792. Citeseer, 2015.
- [6] Karl J Åström. *Introduction to stochastic control theory*. Courier Corporation, 2012.
- [7] Karl J Åström and Björn Wittenmark. *Adaptive control*. Courier Corporation, 2013.
- [8] Felix Berkenkamp, Matteo Turchetta, Angela P Schoellig, and Andreas Krause. Safe model-based reinforcement learning with stability guarantees. *arXiv preprint arXiv:1705.08551*, 2017.
- [9] Alon Cohen, Avinatan Hasidim, Tomer Koren, Nevena Lazic, Yishay Mansour, and Kunal Talwar. Online linear quadratic control. In *International Conference on Machine Learning*, pages 1029–1038. PMLR, 2018.
- [10] Alon Cohen, Tomer Koren, and Yishay Mansour. Learning linear-quadratic regulators efficiently with only \sqrt{T} regret. *International Conference on Machine Learning*, pages 1300–1309, 2019.
- [11] Sarah Dean, Horia Mania, Nikolai Matni, Benjamin Recht, and Stephen Tu. Regret bounds for robust adaptive control of the linear quadratic regulator. In *Advances in Neural Information Processing Systems*, pages 4188–4197, 2018.
- [12] Maryam Fazel, Rong Ge, Sham Kakade, and Mehran Mesbahi. Global convergence of policy gradient methods for the linear quadratic regulator. In *International Conference on Machine Learning*, pages 1467–1476, 2018.
- [13] Elad Hazan, Sham Kakade, and Karan Singh. The nonstochastic control problem. In *Algorithmic Learning Theory*, pages 408–421. PMLR, 2020.
- [14] Elad Hazan, Adam Kalai, Satyen Kale, and Amit Agarwal. Logarithmic regret algorithms for online convex optimization. In *International Conference on Computational Learning Theory*, pages 499–513. Springer, 2006.
- [15] Elad Hazan, Alexander Rakhlin, and Peter L Bartlett. Adaptive online gradient descent. *Advances in Neural Information Processing Systems*, pages 65–72, 2008.
- [16] Tor Aksel N Heirung, B Erik Ydstie, and Bjarne Foss. Dual adaptive model predictive control. *Automatica*, 80:340–348, 2017.
- [17] Petros A Ioannou and Jing Sun. *Robust adaptive control*. Courier Corporation, 2012.
- [18] Rodolphe Jenatton, Jim Huang, and Cédric Archambeau. Adaptive algorithms for online convex optimization with long-term constraints. In *International Conference on Machine Learning*, pages 402–411, 2016.
- [19] Panqanamala Ramana Kumar and Pravin Varaiya. *Stochastic systems: Estimation, identification, and adaptive control*. SIAM, 2015.
- [20] Matthias Lorenzen, Frank Allgöwer, and Mark Cannon. Adaptive model predictive control with robust constraint satisfaction. *IFAC-PapersOnLine*, 50(1):3313–3318, 2017.
- [21] Horia Mania, Stephen Tu, and Benjamin Recht. Certainty equivalent control of lqr is efficient. *arXiv preprint arXiv:1902.07826*, 2019.
- [22] Riccardo Marino, Patrizio Tomei, and Cristiano Maria Verrelli. Robust adaptive learning control for nonlinear systems with extended matching unstructured uncertainties. *International Journal of Robust and Nonlinear Control*, 22(6):645–675, 2012.
- [23] Igor Molybog and Javad Lavaei. Global convergence of MAML for LQR. *arXiv preprint arXiv:2006.00453*, 2020.
- [24] Shankar Sastry and Marc Bodson. *Adaptive control: stability, convergence and robustness*. Courier Corporation, 2011.
- [25] Max Simchowitz and Dylan Foster. Naive exploration is optimal for online lqr. In *International Conference on Machine Learning*, pages 8937–8948. PMLR, 2020.
- [26] Max Simchowitz, Karan Singh, and Elad Hazan. Improper learning for non-stochastic control. *arXiv preprint arXiv:2001.09254*, 2020.
- [27] Sigurd Skogestad and Ian Postlethwaite. *Multivariable feedback control: analysis and design*, volume 2. Citeseer, 2007.
- [28] Gang Tao. Multivariable adaptive control: A survey. *Automatica*, 50(11):2737–2764, 2014.
- [29] Miao Yu and Deqing Huang. Switching adaptive learning control for nonlinearly parameterized systems with disturbance of unknown periods. *International journal of robust and nonlinear control*, 25(9):1327–1337, 2015.
- [30] Jianjun Yuan and Andrew Lamperski. Online convex optimization for cumulative constraints. In *Advances in Neural Information Processing Systems*, pages 6137–6146, 2018.
- [31] Kaiqing Zhang, Bin Hu, and Tamer Basar. Policy optimization for \mathcal{H}_2 linear control with \mathcal{H}_∞ robustness guarantee: Implicit regularization and global convergence. *arXiv preprint arXiv:1910.09496*, 2019.
- [32] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *International Conference on Machine Learning*, pages 928–936, 2003.

APPENDIX A: PROOF OF THEOREM 1

The regret can be split as

$$\begin{aligned}
 R_T &= \sum_{t=h+1}^T F_t(u_t, \dots, u_{t-h}) - \sum_{t=h+1}^T F_t(u_t, \dots, u_t) \\
 &+ \sum_{t=h+1}^T F_t(u_t, \dots, u_t) - \min_{u \in \mathcal{K}} \sum_{t=h+1}^T F_t(u, \dots, u) \\
 &= \sum_{t=h+1}^T F_t(u_t, \dots, u_{t-h}) - \sum_{t=h+1}^T F_t(u_t, \dots, u_t) \\
 &+ \sum_{t=h+1}^T f_t(u_t) - \min_{u \in \mathcal{K}} \sum_{t=h+1}^T f_t(u).
 \end{aligned}$$

Lets call the second term as \tilde{R}_T , i.e.,

$$\tilde{R}_T = \sum_{t=h+1}^T f_t(u_t) - \min_{u \in \mathcal{K}} \sum_{t=h+1}^T f_t(u).$$

Given that

$$u_{t+1} = \text{Proj}_{\mathcal{K}}(u_t - \eta_{t+1} \partial(f_t(u_t) + g_t(u_t))), \quad (12)$$

Lemma 1 is applicable to \tilde{R}_T . Hence, we have that

$$\tilde{R}_T \leq \frac{1}{2} D^2 \sum \lambda_{1:T} + \frac{1}{2} \sum_{t=1}^T \frac{(G_f + \lambda_t D)^2}{\sum H_{1:t} + \sum \lambda_{1:t}}.$$

Next, we bound the first term. By the definition of G_c we have that

$$\begin{aligned}
 &\|F_t(u_t, \dots, u_{t-h}) - F_t(u_t, \dots, u_t)\|_2^2 \\
 &\leq G_c^2 \|[u_t^\top, \dots, u_{t-h}^\top]^\top - [u_t^\top, \dots, u_t^\top]^\top\|_2^2 \\
 &= G_c^2 \sum_{i=1}^h \|u_t - u_{t-i}\|_2^2 \\
 &\leq G_c^2 \sum_{i=1}^h \left(\sum_{j=1}^i \|u_{t-j+1} - u_{t-j}\|_2 \right)^2.
 \end{aligned}$$

Using Eq. (12) we have that

$$\|u_{t-j+1} - u_{t-j}\|_2 \leq \|\eta_{t-j+1} \partial(f_{t-j}(u_{t-j}) + g_{t-j}(u_{t-j}))\|_2.$$

Given that $\|\nabla f_{t-j}\| \leq G_c$ (this follows from the fact that L is a Lipschitz constant of f iff $\|\nabla f\|_2 \leq L$ for differentiable f), and $\|\nabla g_{t-j}(\cdot)\|_2 \leq \lambda_{t-j} D$, we have that

$$\|u_{t-j+1} - u_{t-j}\|_2 \leq \eta_{t-j+1} (G_f + \lambda_{t-j} D).$$

Using this observation we have that

$$\begin{aligned}
 &\|F_t(u_t, \dots, u_{t-h}) - F_t(u_t, \dots, u_t)\|_2^2 \\
 &\leq G_c^2 \sum_{i=1}^h \left(\sum_{j=1}^i \eta_{t-j+1} (G_f + \lambda_{t-j} D) \right)^2 \\
 &\leq G_c^2 \sum_{i=1}^h \left(\sum_{j=1}^i \eta_{t-h} (G_f + \lambda_{t-h} D) \right)^2
 \end{aligned}$$

$$\leq G_c^2 h^3 \eta_{t-h}^2 (G_f + \lambda_{t-h} D)^2.$$

That is

$$\begin{aligned}
 &\|F_t(u_t, \dots, u_{t-h}) - F_t(u_t, \dots, u_t)\|_2 \\
 &\leq G_c h^{3/2} \eta_{t-h} (G_f + \lambda_{t-h} D).
 \end{aligned}$$

Hence

$$\begin{aligned}
 &\sum_{t=h+1}^T F_t(u_t, \dots, u_{t-h}) - \sum_{t=h+1}^T F_t(u_t, \dots, u_t) \\
 &\leq G_c h^{3/2} \sum_{t=h+1}^T \eta_{t-h} (G_f + \lambda_{t-h} D) \\
 &= G_c h^{3/2} \sum_{t=h+1}^T \frac{(G_f + \lambda_{t-h} D)}{\sum H_{1:t-h} + \sum \lambda_{1:t-h}} \\
 &\leq G_c h^{3/2} \sum_{t=1}^T \frac{(G_f + \lambda_t D)}{\sum H_{1:t} + \sum \lambda_{1:t}}.
 \end{aligned}$$

Combining this with the bound on \tilde{R}_T we get that

$$R_T \leq \frac{1}{2} D^2 \sum \lambda_{1:T} + \frac{1}{2} \sum_{t=1}^T \frac{\tilde{G}_{f,t}^2}{\sum H_{1:t} + \sum \lambda_{1:t}},$$

where $\tilde{G}_{f,t} = \sqrt{(G_f + \lambda_t D)(G_f + \lambda_t D + 2G_c h^{3/2})}$. ■

APPENDIX C: PROOF OF THEOREM 2

For a policy $\pi^* \in \Pi$

$$J_T(\mathcal{C}) - J_T(\pi^*) = \sum_{t=1}^T l_t(y_t, u_t) - \sum_{t=1}^T l_t(y_t^{\pi^*}, u_t^{\pi^*})$$

We can split the regret as in [26]:

$$\begin{aligned}
 &\mathbb{E}[J_T(\mathcal{C})] - \mathbb{E}[J_T(\pi^*)] = \underbrace{\sum_{t=1}^{m+h} \mathbb{E}[l_t(y_t, u_t)]}_{\text{burn-in loss}} \\
 &+ \underbrace{\sum_{t=m+h+1}^T \mathbb{E}[l_t(y_t, u_t)] - \sum_{t=m+h+1}^T \mathbb{E}[F_t(P_{t:t-h}|y_{1:t}^{\text{nat}})]}_{\text{algorithm truncation error}} \\
 &+ \underbrace{\sum_{t=m+h+1}^T \mathbb{E}[F_t(P_{t:t-h}|y_{1:t}^{\text{nat}})] - \inf_P \sum_{t=m+h+1}^T \mathbb{E}[f_t(P|y_{1:t}^{\text{nat}})]}_{\text{f-policy error}} \\
 &+ \underbrace{\inf_P \sum_{t=m+h+1}^T \mathbb{E}[f_t(P|y_{1:t}^{\text{nat}})] - \inf_P \sum_{t=m+h+1}^T \mathbb{E}[l_t(y_t^P, u_t^P)]}_{\text{comparator truncation error}} \\
 &+ \underbrace{\inf_P \sum_{t=1}^T \mathbb{E}[l_t(y_t^P, u_t^P)] - \sum_{t=1}^T \mathbb{E}[l_t(y_t^{\pi^*}, u_t^{\pi^*})]}_{\text{policy approximation error}}.
 \end{aligned}$$

We leverage the results from [26] to bound the following terms: (i) burn-in loss, (ii) algorithm truncation error, (iii)

comparator truncation error, and (iv) policy approximation error. From Lemma 5.2, [26], we have that

$$\sum_{t=1}^{m+h} \mathbb{E}[l_t(y_t, u_t)] \leq 4R_{G^*}^2 R_{nat}^2 R_M^2 (m+h).$$

From Lemma 5.3, [26], we have that

$$\mathbb{E}[\text{Truncation errors}] \leq 4LTR_{G^*} R_{nat}^2 R_M^2 \psi(h+1)$$

Finally, from Theorem 1, [26], we have that

$$\mathbb{E}[\text{Policy app. error}] \leq 2LTR_M R_{G^*}^2 R_{nat}^2 \psi(m)$$

Next we bound the f-policy error term. Theorem 1 applies to this term. From Lemma 5.4, [26], we have that $f_t(\cdot|y_t^{nat})$ is G_f -Lipschitz, where $G_f = L\sqrt{m}R_M R_{G^*} R_{nat}^2$, $F_t(\cdot|y_t^{nat})$ is G_f -Lipschitz coordinate wise, i.e., $G_c = G_f$, and $D = 2\sqrt{\min\{d_u, d_y\}}R_M$. Then applying Theorem 1 to the f-policy error term we get that

$$\mathbb{E}[\text{f-policy error}] \leq \frac{1}{2}D^2 \sum \lambda_{1:T} + \frac{1}{2} \sum_{t=1}^T \frac{(\tilde{G}_{f,t})^2}{\sum H_{1:t} + \sum \lambda_{1:t}}.$$

This completes the proof. ■

APPENDIX D: PROOF OF COROLLARY 1

Consider the term

$$\hat{R}_T = \frac{1}{2}D^2 \sum \lambda_{1:T} + \frac{1}{2} \sum_{t=1}^T \frac{(\tilde{G}_{f,t})^2}{\sum H_{1:t} + \sum \lambda_{1:t}}.$$

We make the following observation.

$$\begin{aligned} (\tilde{G}_{f,t})^2 &= (G_f + \lambda_t D) \left(G_f + \lambda_t D + 2G_c h^{3/2} \right) \\ &\leq 2(G_f + \lambda_t D)^2 + 2G_c^2 h^3 \leq 4G_f^2 + 4\lambda_t^2 D^2 + 2G_c^2 h^3. \end{aligned}$$

Hence,

$$\begin{aligned} \hat{R}_T &\leq \frac{1}{2}D^2 \sum \lambda_{1:T} + \frac{1}{2} \sum_{t=1}^T \frac{4G_f^2 + 4\lambda_t^2 D^2 + 2G_c^2 h^3}{\sum H_{1:t} + \sum \lambda_{1:t}} \\ &\leq \frac{1}{2}D^2 \sum \lambda_{1:T} + 2 \sum_{t=1}^T \lambda_t D^2 + \sum_{t=1}^T \frac{2G_f^2 + G_c^2 h^3}{\sum H_{1:t} + \sum \lambda_{1:t}} \\ &\leq \frac{5}{2}D^2 \sum \lambda_{1:T} + \sum_{t=1}^T \frac{2G_f^2 + G_c^2 h^3}{\sum H_{1:t} + \sum \lambda_{1:t}}. \end{aligned} \quad (13)$$

Let $\hat{G}_f^2 := 2G_f^2 + G_c^2 h^3$. Next, we prove the main results case by case.

Case 1, any sequence of convex l_t : For this case set $\lambda_1 = \sqrt{T}$ and $\lambda_t = 0$, $t \geq 2$. Then from Theorem 2 and Eq. (13) we get that

$$\begin{aligned} R_T(\mathcal{C}) &\leq R_M^2 R_{G^*}^2 R_{nat}^2 (6L + 4(m+h)) \\ &\quad + \frac{5}{2}D^2 \sum \lambda_{1:T} + \sum_{t=1}^T \frac{\hat{G}_f^2}{\sum H_{1:t} + \sum \lambda_{1:t}} \\ &\leq R_M^2 R_{G^*}^2 R_{nat}^2 (6L + 4(m+h)) + \frac{5}{2}D^2 \sqrt{T} + \\ &\quad + \hat{G}_f^2 \sum_{t=1}^T \frac{1}{\sqrt{T}} = \mathcal{O}(\sqrt{T}). \end{aligned}$$

Case 2, any sequence of convex l_t such that $H_t^l \geq H$: In this case, from Lemma 2

$$\begin{aligned} H_t &\geq H_t^l \left(\sigma_e^2 + \sigma_w^2 \left(\frac{\sigma_{\min}(C)}{1 + \|A\|_2^2} \right)^2 \right) \\ &\geq H \left(\sigma_e^2 + \sigma_w^2 \left(\frac{\sigma_{\min}(C)}{1 + \|A\|_2^2} \right)^2 \right) = \tilde{H}. \end{aligned}$$

Set $\lambda_t = 0$, then from Theorem 2 and Eq. (13) we get that

$$\begin{aligned} R_T(\mathcal{C}) &\leq R_M^2 R_{G^*}^2 R_{nat}^2 (6L + 4(m+h)) \\ &\quad + \frac{5}{2}D^2 \sum \lambda_{1:T} + \sum_{t=1}^T \frac{\hat{G}_f^2}{\sum H_{1:t} + \sum \lambda_{1:t}} \\ &\leq R_M^2 R_{G^*}^2 R_{nat}^2 (6L + 4(m+h)) + \hat{G}_f^2 \sum_{t=1}^T \frac{1}{t\tilde{H}} \\ &= \mathcal{O}(\log T). \end{aligned}$$

Case 3, $H_t^l = Ht^{-\alpha}$, and $0 < \alpha \leq 1/2$: From Lemma 2 $H_t \geq \tilde{H}t^{-\alpha}$. Set $\lambda_1 = \tilde{H}T^\alpha$ and $\lambda_t = 0$, $t > 1$. Then from Theorem 2 and Eq. (13) we get that

$$\begin{aligned} R_T(\mathcal{C}) &\leq R_M^2 R_{G^*}^2 R_{nat}^2 (6L + 4(m+h)) \\ &\quad + \frac{5\tilde{H}}{2}D^2 T^\alpha + \frac{\hat{G}_f^2}{\tilde{H}} \sum_{t=1}^T \frac{1}{\left(\sum_{k=1}^t k^{-\alpha} + T^\alpha \right)}. \end{aligned}$$

Now $\sum_{k=1}^t k^{-\alpha} \geq \int_0^{t-1} (u+1)^{-\alpha} du = (1-\alpha)^{-1}(t^{1-\alpha} - 1)$. Using this fact we get that

$$\begin{aligned} R_T(\mathcal{C}) &\leq R_M^2 R_{G^*}^2 R_{nat}^2 (6L + 4(m+h)) \\ &\quad + \frac{5\tilde{H}}{2}D^2 T^\alpha + \frac{\hat{G}_f^2(1-\alpha)}{\tilde{H}} \sum_{t=1}^T t^{\alpha-1} \\ &\leq R_M^2 R_{G^*}^2 R_{nat}^2 (6L + 4(m+h)) \\ &\quad + \frac{5\tilde{H}}{2}D^2 T^\alpha + \frac{\hat{G}_f^2(1-\alpha)}{\tilde{H}\alpha} T^\alpha = \mathcal{O}(T^\alpha). \end{aligned}$$

Case 4, $H_t^l = Ht^{-\alpha}$, and $\alpha \geq 1/2$: In this case too $H_t \geq \tilde{H}t^{-\alpha}$. Set $\lambda_1 = \tilde{H}T^{1/2}$ and $\lambda_t = 0$, $t > 1$. Then from Theorem 2 and Eq. (13) we get that

$$\begin{aligned} R_T(\mathcal{C}) &\leq R_M^2 R_{G^*}^2 R_{nat}^2 (6L + 4(m+h)) \\ &\quad + \frac{5\tilde{H}}{2}D^2 T^{1/2} + \frac{\hat{G}_f^2}{\tilde{H}} \sum_{t=1}^T \frac{1}{T^{1/2}} = \mathcal{O}(\sqrt{T}) \blacksquare \end{aligned}$$