Adaptive Regret for Control of Time-Varying Dynamics

Paula Gradu PGRADU@BERKLEY.EDU
UC Berkeley

Elad Hazan EHAZAN@PRINCETON.EDU
Princeton University, Google AI Princeton

Edgar Minasyan MINASYAN@PRINCETON.EDU
Princeton University, Google AI Princeton

Editors: N. Matni, M. Morari, G. J. Pappas

Abstract

We consider the problem of online control of systems with time-varying linear dynamics. To state meaningful guarantees over changing environments, we introduce the metric of *adaptive regret* to the field of control. This metric, originally studied in online learning, measures performance in terms of regret against the best policy in hindsight on *any interval in time*, and thus captures the adaptation of the controller to changing dynamics. Our main contribution is a novel efficient meta-algorithm: it converts a controller with sublinear regret bounds into one with sublinear *adaptive regret* bounds in the setting of time-varying linear dynamical systems. The underlying technical innovation is the first adaptive regret bound for the more general framework of online convex optimization with memory. Furthermore, we give a lower bound showing that our attained adaptive regret bound is nearly tight for this general framework.

Keywords: time-varying dynamics, online control, adaptive regret, online learning

1. Introduction

Reinforcement learning and control have essentially identical objectives: to maximize long-term reward in a Markov decision process. The focus in control theory is many times on dynamical systems that arise in the real world, motivated by physical applications such as robotics and autonomous vehicles. In these applications the dynamics have succinct descriptions coming from physics equations. These are seldom linear! Even for simple physical systems such as the inverted pendulum, the dynamics are nonlinear. Furthermore, dynamics in the real world often change with time. For example, the dynamics of an UAV flying from source to target may change due to the volatility of the weather conditions (wind, rain, and so forth).

In terms of **provable methods**, the theory of optimal and robust control has focused on efficient algorithms for linear time invariant (LTI) systems. Nonlinear systems are significantly harder, and in fact NP-hard to control in general (Blondel and Tsitsiklis, 2000). There are several different approaches to deal with nonlinear dynamics, that we detail in the related work section. In this paper we consider the approach of iterative linearization by using the first order approximation, that was popularized by planning methods such as iLQR, iLC and iLQG. This allows one to model nonlinear dynamics as **linear time-varying** (LTV) dynamical systems. However, instead of the standard approach of applying planning methods for linear dynamical systems, we build on recent regret minimization algorithms for online control.

To present our approach to the problem, we first describe the recent literature on online control that differs from classical techniques by measuring performance in terms of regret. We then proceed to show how to borrow concepts from online learning in changing environments to attain meaningful guarantees for control of time-varying systems.

1.1. Online control of linear dynamical systems

A recent advancement in machine learning literature studies the control of dynamical systems in the online learning, or regret minimization, framework. In the nonstochastic control setting, the controller faces a dynamical system given by

$$x_{t+1} = A_t x_t + B_t u_t + w_t \ . \tag{1}$$

Here A_t, B_t describe the system dynamics, x_t is the state, u_t is the control and w_t is the potentially adversarial (nonstochastic) perturbation. Prior literature considers solely the LTI case, where $A_t \equiv A, B_t \equiv B$. The controller chooses the control signal u_t, and incurs loss $c_t(x_t, u_t)$, for an adversarially chosen convex cost function c_t. Since the perturbations and cost functions are arbitrary or chosen adversarially, the best policy is ill-defined a priori. Thus, the performance metric in this model is worst-case regret w.r.t. the best policy in hindsight from a certain policy class Π. Formally,

$$\text{Regret}_T = \sum_{t=1}^{T} c_t(x_t, u_t) - \min_{\pi \in \Pi} \sum_{t=1}^{T} c_t(x_t^\pi, u_t^\pi) \ . \tag{2}$$

Several benchmark policy classes have been considered in the recent control literature. The simplest to describe is the class of linear state feedback policies, i.e. policies that choose the control as a linear function of the state, $u_t = K x_t$. These policies are known to be optimal for the $\mathcal{H}_2$ control and $\mathcal{H}_\infty$ control formulations for LTI systems.

From this starting point we would like to extend nonstochastic control: **can we prove regret bounds for LTV systems?** How would such bounds even look like? To address this question we first consider the field of online learning, where the metric of regret is well studied, and investigate its extension to changing environments.

1.2. Adaptive regret for online convex optimization

In the problem of online convex optimization (OCO), a learner iteratively chooses a point in a convex decision set, i.e. $z_t \in \mathcal{K} \subseteq \mathbb{R}^d$. An adversary then chooses a loss function $f_t : \mathcal{K} \mapsto \mathbb{R}$. The goal of the learner is to minimize regret, or loss compared to the best fixed decision in hindsight, given as

$$\sum_{t=1}^{T} f_t(z_t) - \min_{z^\star \in \mathcal{K}} \sum_{t=1}^{T} f_t(z^\star) \ .$$

The theory of OCO gives rise to efficient online algorithms with sublinear regret, e.g. $O(\sqrt{T})$ over T iterations, implying that on average the algorithm competes with the best fixed decision in hindsight. However, the standard regret metric is not suitable for changing environments, where the fixed optimal solution in hindsight is *poor*. For example, consider a scenario with $f_t = f$ for the first $T/2$ iterations, and $f_t = g$ for the last $T/2$ iterations. Here, the standard regret metric ensures convergence to the minimum of $f + g$, i.e. the best fixed decision in hindsight which potentially incurs *linear* loss.[1] Yet, the optimal solution for this scenario is to shift between the minimum of f to that of g midway!

For this reason, the metric of adaptive regret was developed by Hazan and Seshadhri (2009). It captures the supremum over all local regrets in any contiguous interval I, defined as

$$\sup_{I=[r,s]\subseteq[1,T]} \left[\sum_{t=r}^{s} f_t(z_t) - \min_{z_I^\star \in \mathcal{K}} \sum_{t=r}^{s} f_t(z_I^\star) \right] . \tag{3}$$

The strength of this definition is that it does *not* try to model the changes in the environment. Instead, the responsibility is on the learner to try to compete with the best local predictor $z_I^\star$ at all times. For the example above, if an algorithm does not converge to either the optimum of f in $[1, T/2]$ or the optimum of g in $[1, T/2]$, it would suffer *linear* adaptive regret. In general, algorithms that minimize adaptive regret by definition minimize regret as well and additionally are capable of quickly switching between local optima. This metric is thus more appropriate for our investigation of *LTV dynamical systems*.

1.3. Contributions

In the setting of online control over LTV systems as in (1), the adaptive regret metric implies the following: an algorithm that minimizes adaptive regret is capable of competing against different policies from the class Π throughout the time horizon T. Formally, adaptive regret against a policy class Π is given as

$$\text{AdRegret}_T = \sup_{I=[r,s]\subseteq[1,T]} \left\{ \sum_{t=r}^{s} c_t(x_t, u_t) - \min_{\pi_I^\star \in \Pi} \sum_{t=r}^{s} c_t(x_t^{\pi_I^\star}, u_t^{\pi_I^\star}) \right\} . \tag{4}$$

Just like we surveyed for OCO, the main change from standard regret for control is the supremum over all intervals, and the fact that the minimum over the policies is local to the particular interval. This ensures that an algorithm with sublinear adaptive regret guarantees enjoys low regret against the best-in-hindsight policy $\pi_I^\star$ on any interval I. Hence, the algorithm captures any changes in the dynamics of the LTV system by implicitly tracking the local optimal (in Π) policy.

The main challenge in applying existing adaptive regret methods from online learning to control and reinforcement learning is the long-term effect that actions have. We first overcome this challenge in the setting of online convex optimization *with memory* (Anava et al., 2015). This setting allows one to transfer learning in stateful environments to online learning, following the methodology proposed in Agarwal et al. (2019a). The main contributions of our paper can be summarized as (i) adaptive regret results for online control over LTV systems and (ii) technical contributions in the OCO with memory setting possibly of independent interest.

Adaptive Regret over LTV systems. We propose an efficient meta-algorithm MARC, Algorithm 1, that converts a base controller with standard regret bounds to a control algorithm with provable sublinear adaptive regret guarantees in the setting of linear, time-varying systems. We apply MARC over recent algorithmic results in nonstochastic control, and obtain an efficient algorithm that attains $\tilde{O}(\sqrt{\mathrm{OPT}})$ adaptive regret against the class of disturbance response control (DRC) policies, where OPT is the cost of the best policy in hindsight over the entire horizon.

[1] To see this, take $f(z) = \|z - z_f^\star\|^2$ and $g(z) = \|z - z_g^\star\|^2$, then the optimum of $f + g$ is the midpoint of $z_f^\star$ and $z_g^\star$ and over the whole interval suffers cumulative loss linear in T.

Adaptive Regret in OCO with memory. Our derivation goes through the framework of OCO with memory, for which we give an efficient adaptive regret algorithm. Specifically, our algorithm guarantees $\tilde{O}(\sqrt{\mathrm{OPT}})$ adaptive regret for strongly convex functions with memory, where OPT is the best loss in hindsight. This is the first such guarantee for a fundamental setting in prediction. The aforementioned challenge of obtaining adaptive regret in the setting of OCO with memory is that an online algorithm needs to change its decision slowly to cope with the memory constraint. On the other hand, an online algorithm needs to be agile to quickly adjust to environment changes and minimize adaptive regret. These two requirements are contradictory. We formalize this intuition by proving the following lower bound.

Theorem 1 (Informal Theorem) *Any algorithm for OCO with memory has AdRegret*$_T = \Omega(\sqrt{T})$*, even over strongly convex loss functions.*

The lower bound above essentially shows our results to be tight for nonstochastic control algorithms that are based on OCO with memory given that $\mathrm{OPT} = O(T)$. However, we note that despite the general lower bound it is still possible to attain $o(\sqrt{T})$ adaptive regret in certain favorable settings. In fact, the positive results in this work, given as first-order adaptive regret bounds, suggest exactly this: Algorithm 1 suffers adaptive regret much smaller than $\tilde{O}(\sqrt{T})$ when the cost of the best policy in hindsight $\mathrm{OPT} = o(T)$ is sublinear.

Paper Organization. In subsection 1.4 we discuss related work. We provide important background and formalize the problem at hand in section 2. Section 3 describes the main meta-algorithm and its performance guarantee for online control over changing dynamics. Our experimental results are presented in section 4. The arXiv version includes details, formal theorems, proofs and all else skipped in the main body of the paper for clarity of exposition.

1.4. Related work

The field of optimal and adaptive control is vast and spans decades of research, see e.g. Stengel (1994); Zhou et al. (1996) for survey. In terms of nonlinear control, we can divide the literature into several main approaches. The iterative linearization approach takes the local linear approximation via the gradient of the nonlinear dynamics. One can apply techniques from optimal control to solve the resulting changing linear system. Iterative planning methods such as iLQR (Tassa et al., 2012), iLC (Moore, 2012) and iLQG (Todorov and Li, 2005) fall into this category. Our approach also takes this route.

Another approach is using convex relaxations of the nonlinear dynamics to cope with the hardness of the underlying non-convex optimization. These methods are applied for both $\mathcal{H}_2$ control (see e.g. Majumdar et al. (2020)), and $\mathcal{H}_\infty$ control (see Bansal et al. (2017)) formulations. They are highly effective in some cases, but do not scale well to high dimensional problems. Finally, the nonlinear system can also be linearized via the Koopman operator as detailed in Budisic et al. (2012); Rowley and Dawson (2017).

In this work we restrict our discussion to online control of changing linear dynamical systems with low *adaptive regret*. To the best of our knowledge, this is the first work with adaptive regret bounds shown for time-varying dynamics.

Online convex optimization and adaptive regret. We make extensive use of techniques from the field of online learning and regret minimization in games (Cesa-Bianchi and Lugosi, 2006; Hazan, 2016). Most relevant to our work is the literature on adapting to changing environments in online learning, which starts from the works of Herbster and Warmuth (1998); Bousquet and Warmuth (2002). The notion of adaptive regret was introduced in Hazan and Seshadhri (2009), and significantly studied since as a metric for adaptive learning in OCO (Adamskiy et al., 2016; Zhang et al., 2019). An alternative metric for changing systems studied in online learning is called dynamic regret Zinkevich (2003). It has been estabilished that dynamic regret is a weaker notion than *strongly* adaptive regret Daniely et al. (2015), in the sense that a sublinear bound on the former implies sublinear dynamic regret, and the reverse is not true Zhang et al. (2018).

Regret minimization for online control. In classical control theory, the disturbances are assumed to be i.i.d. Gaussian and the cost functions are known ahead of time. In the online LQR setting (Abbasi-Yadkori and Szepesvári, 2011; Dean et al., 2018; Mania et al., 2019; Cohen et al., 2018), a fully-observed time-invariant linear dynamic system is driven by i.i.d. Gaussian noise and the learner incurs a cost which is (potentially changing) quadratic in state and input. When the costs are fixed, the optimal policy for this setting is known to be linear $u_t = Kx_t$, where K is the solution to the algebraic Ricatti equation. Several online methods (Mania et al., 2019; Cohen et al., 2019, 2018) attain $\sqrt{T}$ regret for this setting, and are able to cope with changing loss functions. Regret bounds for partially observed systems were studied in Lale et al. (2020a,b,c), with the most general and recent bounds in Simchowitz et al. (2020).

Agarwal et al. (2019a) consider a significantly more general and challenging setting, called nonstochastic control, in which the disturbances and cost functions are adversarially chosen, and the cost functions are arbitrary convex costs. In this setting they give an efficient algorithm that attains $\sqrt{T}$ regret. This result was extended to *unknown* LTI systems in Hazan et al. (2019), and the partial observability setting in Simchowitz et al. (2020). Logarithmic regret for the nonstochastic perturbation setting was obtained in Simchowitz (2020). For a survey of recent techniques and results see Hazan and Singh (2021).

A roughly concurrent line of work considers minimizing (dynamic) regret against the optimal open-loop control sequence in both LTI and LTV systems. Li et al. (2019) achieve this by leveraging a finite lookahead window while Goel and Hassibi (2021) reduce the regret minimization problem to $\mathcal{H}_\infty$ control. Zhang et al. (2021) follow up our work to devise methods with *strongly* adaptive regret guarantees however these regret bounds, as opposed to ours, are not first-order.

2. Problem Setting and Preliminaries

Notation. Throughout this work we use $[n] = [1, 2, ..., n]$ as a shorthand, $\|\cdot\|$ is used for Euclidean and spectral norms, $O(\cdot)$ hides absolute constants, $\tilde{O}(\cdot)$ hides terms poly-logarithmic in T.

Online LTV Control A time-varying linear (LTV) dynamical system is given by the following dynamics equation,

$$\forall t \in [T], \quad x_{t+1} = A_t x_t + B_t u_t + w_t,$$

where $x_t \in \mathbb{R}^{d_x}$ is the (observable) system state, $u_t \in \mathbb{R}^{d_u}$ is the control, and (A_t, B_t) are the system matrices with $A_t \in \mathbb{R}^{d_x \times d_x}$, $B_t \in \mathbb{R}^{d_x \times d_u}$, $w_t \in \mathbb{R}^{d_x}$ is the disturbance. In our work, we allow w_t to be adversarially chosen. This is the key assumption in the *nonstochastic* control literature. The additional generality of adversarial perturbations allows the disturbance to model slight deviations from linearity along with inherent noise.

We consider the setting of *known* systems, i.e. after taking an action u_t the controller observes the next state x_{t+1} as well as the current system matrices (A_t, B_t). This allows the controller to compute the disturbance $w_t = x_{t+1} - A_t x_t - B_t u_t$, so the knowledge of x_{t+1} and w_t is interchangeable. A control algorithm $\mathcal{C}$ chooses an action $u_t = \mathcal{C}(x_1, \ldots, x_t)$ based on previous information. It then suffers loss $c_t(x_t, u_t)$ and observes the cost function c_t. We remark that an adaptive adversary chooses all $(A_t, B_t), w_t, c_t$. We make the following basic assumptions common in the nonstochastic control literature.

Assumption 2.1 *The disturbances are bounded in norm,* $\max_t \|w_t\| \le W$.

Assumption 2.2 *There exist* $C, C_B \ge 1$ *and* $\rho \in (0, 1)$ *such that for all* t *and* $H \in [1, t)$*,*

$$\Phi_t^H = \prod_{i=t}^{t-H+1} A_i, \; \|\Phi_t^H\|_{op} \le C \cdot \rho^H, \; \|B_t\|_{op} \le C_B \, .$$

Assumption 2.3 *The cost functions* $c_t : \mathbb{R}^{d_x} \times \mathbb{R}^{d_u} \to \mathbb{R}$ *are general convex functions that satisfy the conditions* $0 \le c_t(x, y) \le 1$ *and* $\|c_t(x, y)\| \le L_c \max\{1, \|x\| + \|u\|\}$ *for some* $L_c > 0$.

The standard performance metric of controller $\mathcal{C}$ over horizon T is regret with respect to a class of policies Π as defined in (2) denoted $\text{Regret}_T(\mathcal{C})$. We instead minimize for the *adaptive regret* metric of $\mathcal{C}$ with respect to Π as defined in (4), and denote it $\text{AdRegret}_T(\mathcal{C})$. In case the control algorithm is randomized, we take the expectation of the metric over the randomness in the algorithm.

The choice of the policy class Π is essential for the performance of a control algorithm. One target class of policies we compare against in this paper is disturbance response controllers (DRC), whose control is a linear function of the states the controller would have reached in the absence of exogenous control input, for some history-length parameter H. This comparator class is known to approximate to arbitrarily high precision the state-of-the-art in LTI control: linear dynamical controllers (LDC). For more in depth discussion of this and other policy classes, please refer to the arXiv version.

This choice is a consequence of recent advances in convex relaxation for control (Agarwal et al., 2019a,b; Hazan et al., 2019; Simchowitz et al., 2020) via a reduction of online control to the setting of online convex optimization (OCO) with memory. The intuition behind this approach is that even though actions have long-term effect in control, their effect is dissipating geometrically fast in time. Thus, actions and states that occurred far in the past have only marginal effect on the dynamics as a whole. The formal statement for this intuition is given in Definition 2, a generalization of Definition 2.1 from Agarwal et al. (2020).

Before stating the formal definition, we first describe the notion of an *action set sequence*. For a fixed horizon T, let $\mathcal{U}_t \subset \mathbb{R}^{d_u}$ be the constraint set for action u_t for each $t \in [T]$. Denote the action set sequence $\mathcal{U}_{1:T} = \{\mathcal{U}_1, \ldots, \mathcal{U}_T\}$ and use $u_{1:T} \in \mathcal{U}_{1:T}$ to indicate $u_t \in \mathcal{U}_t$ for all $t \in [T]$. We remark that $\mathcal{U}_t$ potentially depends on the system dynamics up to time t and the action set sequence $\mathcal{U}_{1:T}$ depends on the family of control algorithms used, but *not* on the particular individual instance.

Definition 2 *The action set sequence $\mathcal{U}_{1:T}$ is said to have (H, ε)-bounded memory if for all fixed arbitrary $u_{1:T} \in \mathcal{U}_{1:T}$ and all $t \in [T]$,*

$$|c_t(x_t, u_t) - c_t(\hat{x}_t, u_t)| \;\leq\; \varepsilon,$$

where we define $\hat{x}_t$ to be the proxy state with memory H for the sequence of actions $u_{1:T}$: for $t > H$, $\hat{x}_t$ is the state reached by the system if we artificially set $x_{t-H} = 0$ and simulate the dynamics (1) with the actions $u_{t-H}, \ldots, u_{t-1}$.

Suppose an action set sequence $\mathcal{U}_{1:T}$ has (H, ε)-bounded memory. Then, given that $\hat{x}_t = \hat{x}_t(u_{t-H}, \ldots, u_{t-1})$, the performance guarantees of a proxy cost function $f_t(u_{t-H:t}) = c_t(\hat{x}_t, u_t)$ imply guarantees for the control setting. Furthermore, regret minimization of $f_t(u_{t-H:t})$ can be done in the setting of OCO with memory. Finally, we state the necessary properties for a controller $\mathcal{C}$ to be considered a base control algorithm.

Definition 3 *A control algorithm $\mathcal{C}$ with an action set sequence $\mathcal{U}_{1:T}$ is called a base controller if:*

(i) $\mathcal{U}_{1:T}$ has (H, ε)-bounded memory.

(ii) for all $t \in [T]$, the proxy loss $f_t(u_{t-H:t}) = c_t(\hat{x}_t, u_t)$ is coordinate-wise L-Lipschitz.

Note that the properties of a base controller concern the control algorithm setup not the controller instance itself. The definition of a base controller serves simply as an abstraction: in the arXiv version we show it is not vacuous given that all previous control algorithms in the nonstochastic control literature satisfy the conditions (Agarwal et al., 2019a; Simchowitz, 2020).

Adaptive Regret for Online Convex Optimization with Memory In the setting of online convex optimization (OCO) with memory the adversary reveals the loss function $f_t : \mathcal{K}^{H+1} \mapsto \mathbb{R}$ that applies to the past $H + 1$ decisions of the player, and the player suffers loss $f_t(z_{t-H:t})$ where $z_{i:j} = \{z_i, \ldots, z_j\}$ with $i < j$. Define the surrogate loss $\tilde{f}_t : \mathcal{K} \mapsto \mathbb{R}$ to be the function with all $H + 1$ arguments equal, i.e. $\tilde{f}_t(z) = f_t(z, \ldots, z)$ (this reduces to the standard setting with $H = 0$). The regret in this setting is defined with respect to the best surrogate loss in hindsight as follows,

$$\text{Regret}_T = \sum_{t=1}^{T} f_t(z_{t-H:t}) - \min_{z \in \mathcal{K}} \sum_{t=1}^{T} \tilde{f}_t(z) \; .$$

The notion of adaptive regret in OCO with memory is defined analogously to (3), i.e. the supremum of the local standard regret over all contiguous intervals,

$$\text{AdRegret}_T(\mathcal{A}) = \sup_{I=[r,s]\subseteq[T]} \left[\sum_{t=r}^{s} f_t(z_{t-H:t}) - \min_{z_I^\star \in \mathcal{K}} \sum_{t=r}^{s} \tilde{f}_t(z_I^\star) \right] \; .$$

The OCO setting with memory, as outlined in Anava et al. (2015), reduces to the standard setting by assuming a Lipschitz condition on f and relating f to $\tilde{f}$. To quantify this relation, it is crucial to keep track of the movement between the consecutive actions by the learner. Henceforth, we define the notion of *action shift*, a metric for the stability of the algorithm, as the overall shifting in distance of consecutive actions by $\mathcal{A}$,

$$\mathcal{S}_T(\mathcal{A}) = \sup_{z_1, \ldots, z_T \leftarrow \mathcal{A}} \left[\sum_{t=1}^{T-1} \|z_{t+1} - z_t\| \right] \; . \tag{5}$$

Action shift is necessary to quantify an algorithm's prediction stability: the predictions are stable if the action shift is small and this is crucial in the setting with memory. On the other hand, adaptive regret encourages an algorithm to move quickly to adapt to environment changes, thus compromises the prediction stability by driving the action shift to be large.

3. Online Control of Time-Varying Dynamics

For the setting of online control described in Section 2, we devise MARC (Algorithm 1): a meta-algorithm that takes a base controller $\mathcal{C}$ and "transforms" its standard regret guarantees into adaptive regret bounds. It does so by maintaining $N = T$ copies of the base controller $(\mathcal{C}_1, \dots, \mathcal{C}_N)$, with the restriction that each $\mathcal{C}_i$ plays $u_t^i = 0$ for $t < i$ and only starts running $\mathcal{C}$ at round $t = i$. At each round t, MARC chooses the action $u_t = u_t^i$ given by $\mathcal{C}_i$ with some probability that reflects $\mathcal{C}_i$'s performance so far. As long as $\mathcal{C}$ is a base controller according to Definition 3, we can transfer our general results on adaptive regret for online convex optimization with memory to the control setting, yielding Theorem 4 given below.

Black-box use of $\mathcal{C}$. Since the system is known to the meta-controller, each controller $\mathcal{C}_i$ can construct a simulated environment with its own actions u_t^i and identical system matrices and disturbances. In particular, once the meta-controller observes the new state x_{t+1}, it computes the corresponding disturbance $w_t = x_{t+1} - A_t x_t - B_t u_t$ and feeds it to the base controllers along with the system matrices (A_t, B_t). Afterwards, each base controller $\mathcal{C}_i$ simulates the system environment with its own action, i.e. $x_{t+1}^i = A_t x_t^i + B_t u_t^i + w_t$. Such behavior allows for black-box use of results for the base controllers since each acts separately in response to the same dynamics.

Algorithm 1 Meta Adaptive Regret Controller (MARC)

```
Input: horizon $T$, action set sequence $\mathcal{U}_{1:T}$, $N = T$ controllers $\mathcal{C}_1, \dots, \mathcal{C}_N$, parameters $\eta, \sigma$
Setup: assign $w_1^i = 1$ and feedback $\mathcal{F}_1^i = \{x_1^i = 0\}, \forall i \in [N]$, denote $W_t = \sum_{i=1}^N w_t^i$
for $t = 1, ..., T$ do
    compute each action $u_t^i$ by $\mathcal{C}_i$ given $\mathcal{F}_t^i$
    if $t = 1$ then
        choose $i_t = i$ w.p. $p_t^i = w_t^i / W_t$ for all $i \in [N]$
    else
        keep $i_t = i_{t-1}$ w.p. $w_t^{i_{t-1}} / w_{t-1}^{i_{t-1}}$, o.w. choose $i_t = i$ w.p. $p_t^i = w_t^i / W_t$ for all $i \in [N]$
    end
    choose action $u_t = u_t^{i_t}$, observe $c_t(\cdot, \cdot)$, suffer cost $c_t(x_t, u_t)$
    observe new state $x_{t+1}$, compute $w_t$ disturbance, obtain $x_{t+1}^i$ given $u_t^i, w_t$ for all $i \in [N]$
    let $f_t(u_{t-H:t}) = c_t(\hat{x}_t, u_t)$ be proxy cost, $\tilde{f}_t(u) = f_t(u, \dots, u)$ be surrogate proxy cost
    compute $\overline{w}_{t+1}^i = w_t^i e^{-\eta \tilde{f}_t(u_t^i)}$ and $w_{t+1}^i = (1 - \sigma)\overline{w}_{t+1}^i + \sigma \overline{W}_{t+1}/N$ for all $i \in [N]$
    update $\mathcal{F}_{t+1}^i = \mathcal{F}_t^i \cup \{x_{t+1}^i, u_t^i, c_t\}$ for all $i \in [N]$
end
```

Efficient implementation. We remark that Algorithm 1 is not computationally efficient relative to a base controller $\mathcal{C}$: it has the computational complexity of T such controllers. Yet, our algorithm can be implemented in an efficient manner by keeping track of and updating only $O(\log T)$ active controllers. The inactive ones are represented by the stationary $u_t = 0$ controller. This efficient version incurs only a $O(\log T)$ extra multiplicative adaptive regret factor relative to Theorem 4 and only $O(\log T)$ computational overhead relative to the base controller $\mathcal{C}$. For the sake of clarity and brevity, we present Algorithm 1 without this component and present the efficient implementation formally in the arXiv version.

Theorem 4 *Let $\mathcal{C}$ be a base control algorithm by Definition 3 with $\epsilon = T^{-1}$ and denote $\mathrm{OPT} = \min_{\pi \in \Pi} \sum_{t=1}^{T} c_t(x_t^{\pi}, u_t^{\pi})$. With the parameter choices of $\eta = \tilde{\mathcal{O}}(\mathcal{S}_T(\mathcal{C})\sqrt{\mathrm{OPT}})^{-1}$ and $\sigma = T^{-1}$, Algorithm 1 (MARC) achieves the following adaptive regret against the class of policies Π:*

$$\mathbb{E}\left[\textit{AdRegret}_T(\textit{MARC})\right] \ \leq\ \tilde{\mathcal{O}}^2\left(\textit{Regret}_T(\mathcal{C}) + \mathcal{S}_T(\mathcal{C})\sqrt{\mathrm{OPT}}\right), \tag{6}$$

where $\textit{Regret}_T(\mathcal{C})$ is the regret $\mathcal{C}$ attains w.r.t. Π, and $\mathcal{S}_T(\mathcal{C})$ is its action shift.

Theorem 4 guarantee shows that Algorithm 1 converts a standard regret bound of a base controller into an adaptive one with an extra additive term as in (6). Hence, of utmost interest are algorithms from the literature that achieve logarithmic regret (and action shift). We showcase the use of our meta-algorithm MARC on the DRC-ONS (Simchowitz, 2020) algorithm in Corollary 5.

Corollary 5 (MARC-DRC-ONS) *Let $\mathcal{C}$ be the DRC-ONS algorithm from (Simchowitz, 2020). Assume the cost functions c_t are strongly convex. The MARC-DRC-ONS algorithm that performs MARC over $\mathcal{C}$ enjoys the following adaptive regret guarantee w.r.t. the DRC policy class Π_{drc} (see arXiv version for formal definition and discussion),*

$$\mathbb{E}\left[\textit{AdRegret}_T(\textit{MARC-DRC-ONS})\right] \ \leq\ \tilde{\mathcal{O}}\left(\sqrt{\mathrm{OPT}}\right). \tag{7}$$

The result above holds for any LTV system that satisfies Assumptions 2.1, 2.2, 2.3. Note that the adaptive regret bound $\tilde{\mathcal{O}}(\sqrt{\mathrm{OPT}})$ can be considerably smaller than $\sqrt{T}$ if the LTV system (along with the disturbances and costs) is favorable. More importantly, the bound, and more specifically OPT, depends directly on the policy benchmark we are competing against, Π_{drc} in this case. Hence, algorithms that enjoy low regret against larger policy classes automatically enjoy better adaptive regret bounds via MARC. The formal statement and proof of Corollary 5 and application to other control algorithms can be found in the arXiv version.

4. Experimental Results

To show the applicability of our algorithm in more realistic (and harder) scenarios, we consider the control of a nonlinear system via iterative linearization as detailed in the introduction. We experiment on the inverted pendulum environment, a commonly used benchmark consisting of a nonlinear and unstable system, popularized by OpenAI Gym (Brockman et al., 2016). We experiment with both the original noiseless system, and a modified version in which we introduce a sinusoidal shock in the middle of the run, i.e. for timesteps $t \in [T/3, 2T/3]$ an adversarial perturbation of $0.3 * \sin(t/2\pi)$ is added to the state.

[2] We hide factors of H, L since they have been shown to be poly-logarithmic in T in the nonstochastic control literature.

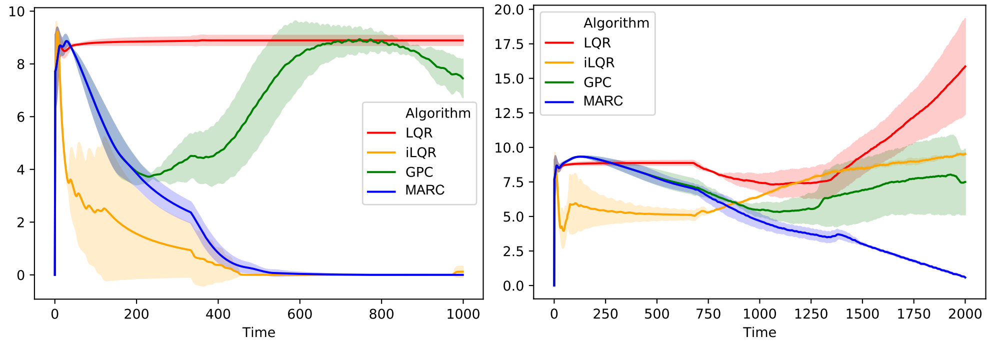

Figure 1: Comparison of $T/3$-window averaged costs on a noiseless pendulum environment (left) and a pendulum environment experiencing a midway sinusoidal shock (right).

For MARC, we implement the efficient version of Algorithm 1 using the GPC algorithm from Agarwal et al. (2019b) as the base controller. As sanity checks, we compare our performance to GPC and to the linear controller LQR which acts according to the algebraic Ricatti equation computed at the start of the experiment. More relevantly, we also compare against iLQR, a planning method for non-linear control via iterative linearization, implemented as in Tassa et al. (2012).

In the left plot of Figure 1, we see that our method enables a controller originally developed for linear systems (GPC) to be used to solve this harder, nonlinear task, only slightly slower than the iLQR baseline. In the right plot, we see that iLQR is unable to adapt to an unanticipated shock due to its static and environment-agnostic design. Yet, our controller MARC demonstrates its robustness to the adversarial noise, and succeeds at this new harder task. More generally, we see that our algorithm works well in the setting of nonlinear control via iteratize linearization. These results confirm that the proposed approach is highly promising even from a practical standpoint, and provides a viable alternative to the classic planning approach.

5. Conclusion

We considered the control of time-varying linear dynamical systems from the perspective of online learning. Using tools from the theory of adaptive regret, we devise new efficient algorithms with provable guarantees in both online control and online prediction: they attain near-optimal *first-order* regret bounds on any interval in time.

In terms of future directions and open problems, it is interesting to extend our results to *strongly* adaptive regret: in particular, it is interesting to answer the question whether strongly adaptive first-order regret, i.e. depending on the optimal cost per interval, can be achieved. This cannot be done trivially by the approach of Daniely et al. (2015) and answering this question would resolve optimality in this setting given our lower bound. The provided expected regret results can be stated with high probability using standard techniques with an additional $\sqrt{T}$ term in the regret. However, obtaining high probability bounds without impeding the first-order regret bound is quite more challenging and of independent interest to attain.

Finally, our guarantees hold with respect to adaptive, rather than oblivious adversaries, which is crucial for nonlinear control. It is interesting to map out which properties of nonlinear dynamics allow effective control via the LTV approximation.

References

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, 2011.

Dmitry Adamskiy, Wouter M Koolen, Alexey Chernov, and Vladimir Vovk. A closer look at adaptive regret. *The Journal of Machine Learning Research*, 17(1):706–726, 2016.

Naman Agarwal, Brian Bullins, Elad Hazan, Sham Kakade, and Karan Singh. Online control with adversarial disturbances. In *International Conference on Machine Learning*, pages 111–119, 2019a.

Naman Agarwal, Elad Hazan, and Karan Singh. Logarithmic regret for online control. In *Advances in Neural Information Processing Systems*, pages 10175–10184, 2019b.

Naman Agarwal, Nataly Brukhim, Elad Hazan, and Zhou Lu. Boosting for control of dynamical systems, 2020.

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, 2015.

Somil Bansal, Mo Chen, Sylvia Herbert, and Claire J Tomlin. Hamilton-jacobi reachability: A brief overview and recent advances. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 2242–2253. IEEE, 2017.

Vincent D Blondel and John N Tsitsiklis. A survey of computational complexity results in systems and control. *Automatica*, 36(9):1249–1274, 2000.

Olivier Bousquet and Manfred K Warmuth. Tracking a small set of experts by mixing past posteriors. *Journal of Machine Learning Research*, 3(Nov):363–396, 2002.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym, 2016.

Marko Budisic, Ryan Mohr, and Igor Mezic. Applied koopmanism. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 22(4), Dec 2012.

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

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, 2018.

Alon Cohen, Tomer Koren, and Yishay Mansour. Learning linear-quadratic regulators efficiently with only $\sqrt{T}$ regret. In *International Conference on Machine Learning*, pages 1300–1309, 2019.

Amit Daniely, Alon Gonen, and Shai Shalev-Shwartz. Strongly adaptive online learning. In *International Conference on Machine Learning*, pages 1405–1411. PMLR, 2015.

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.

Gautam Goel and Babak Hassibi. Regret-optimal control in dynamic environments, 2021.

Elad Hazan. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

Elad Hazan and Comandur Seshadhri. Efficient learning algorithms for changing environments. In *Proceedings of the 26th annual international conference on machine learning*, pages 393–400. ACM, 2009.

Elad Hazan and Karan Singh. Tutorial: online and non-stochastic control, July 2021.

Elad Hazan, Sham M. Kakade, and Karan Singh. The nonstochastic control problem, 2019.

Mark Herbster and Manfred K Warmuth. Tracking the best expert. *Machine learning*, 32(2):151–178, 1998.

Sahin Lale, Kamyar Azizzadenesheli, Babak Hassibi, and Anima Anandkumar. Regret bound of adaptive control in linear quadratic gaussian (lqg) systems, 2020a.

Sahin Lale, Kamyar Azizzadenesheli, Babak Hassibi, and Anima Anandkumar. Logarithmic regret bound in partially observable linear dynamical systems, 2020b.

Sahin Lale, Kamyar Azizzadenesheli, Babak Hassibi, and Anima Anandkumar. Regret minimization in partially observable linear quadratic control, 2020c.

Yingying Li, Xin Chen, and Na Li. Online optimal control with linear dynamics and predictions: Algorithms and regret analysis. *Advances in Neural Information Processing Systems*, 32:14887–14899, 2019.

Anirudha Majumdar, Georgina Hall, and Amir Ali Ahmadi. Recent scalability improvements for semidefinite programming with applications in machine learning, control, and robotics. *Annual Review of Control, Robotics, and Autonomous Systems*, 3:331–360, 2020.

Horia Mania, Stephen Tu, and Benjamin Recht. Certainty equivalence is efficient for linear quadratic control. In *Advances in Neural Information Processing Systems*, pages 10154–10164, 2019.

Kevin L Moore. *Iterative learning control for deterministic systems*. Springer Science & Business Media, 2012.

Clarence W. Rowley and Scott T.M. Dawson. Model reduction for flow analysis and control. *Annual Review of Fluid Mechanics*, 49(1):387–417, 2017.

Max Simchowitz. Making non-stochastic control (almost) as easy as stochastic, 2020.

Max Simchowitz, Karan Singh, and Elad Hazan. Improper learning for non-stochastic control, 2020.

Robert F Stengel. *Optimal control and estimation*. Courier Corporation, 1994.

Y. Tassa, T. Erez, and E. Todorov. Synthesis and stabilization of complex behaviors through online trajectory optimization. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4906–4913, 2012.

Emanuel Todorov and Weiwei Li. A generalized iterative lqg method for locally-optimal feedback control of constrained nonlinear stochastic systems. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 300–306. IEEE, 2005.

Lijun Zhang, Tianbao Yang, Zhi-Hua Zhou, et al. Dynamic regret of strongly adaptive methods. In *International conference on machine learning*, pages 5882–5891. PMLR, 2018.

Lijun Zhang, Tie-Yan Liu, and Zhi-Hua Zhou. Adaptive regret of convex and smooth functions. *arXiv preprint arXiv:1904.11681*, 2019.

Zhiyu Zhang, Ashok Cutkosky, and Ioannis Ch Paschalidis. Strongly adaptive oco with memory. *arXiv preprint arXiv:2102.01623*, 2021.

Kemin Zhou, John C. Doyle, and Keith Glover. *Robust and Optimal Control*. Prentice-Hall, Inc., USA, 1996. ISBN 0134565673.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML'03, page 928–935. AAAI Press, 2003. ISBN 1577351894.