Regret Analysis of Learning-Based MPC With Partially Unknown Cost Function

Ilgin Dogan, Zuo-Jun Max Shen, Member, IEEE, and Anil Aswani, Member, IEEE

Abstract—The exploration–exploitation tradeoff is an inherent challenge in data-driven adaptive control. Though this tradeoff has been studied for multiarmed bandits (MABs) and reinforcement learning for linear systems, it is less well studied for learning-based control of nonlinear systems. A significant theoretical challenge in the nonlinear setting is that there is no explicit characterization of an optimal controller for a given set of cost and system parameters. We propose the use of a finite-horizon oracle controller with full knowledge of parameters as a reasonable surrogate to an optimal controller. This allows us to develop policies in the context of learning-based model-predictive control (MPC) and conduct a control-theoretic analysis using techniques from MPC and optimization theory to show that these policies achieve low regret with respect to this finite-horizon oracle. Our simulations exhibit the low regret of our policy on a heating, ventilation, and air-conditioning model with partially unknown cost function.

Index Terms—Learning-based control, model-predictive control (MPC), nonmyopic exploitation, restless bandits.

I. INTRODUCTION

Reinforcement learning (RL) research [1], [2], [3] focuses on regret analysis for primarily unconstrained linear systems. On the other hand, adaptive model-predictive control (MPC), including learning-based model-predictive control (LB-MPC), seeks to ensure constraint satisfaction in the presence of models that are updated as more data become available [4], [5], [6], [7]. The relationship between MPC and RL has not yet been fully explored.

Our article aims to better connect these two areas. We make two main contributions. First, we discuss how comparing finite-horizon policies with different horizon lengths leads to ambiguous regret notions in the evaluation of learning-based control policies. Thus, we propose a regret notion that compares a finite-horizon learning-based policy with a finite-horizon oracle controller as the benchmark. Second, we bound this regret notion for a class of learning-based control policies for which we prove constraint satisfaction. An important aspect of our regret analysis is that we have to consider the stability of our policy when bounding the regret. In this sense, our analysis draws a connection between stability of the nonlinear control system and regret performance of the learning policy.

A. Partially Unknown Cost Function

MPC usually assumes that the system dynamics and a cost function are exactly known. However, these may be partially unknown in real-world systems that motivate our setup.

1) Heating, Ventilation, Air-Conditioning (HVAC) Systems: Since HVAC uses a large part of total building energy, improving HVAC energy efficiency using MPC has been studied [8], [9], [10], [11]. However, past works typically assume perfect knowledge of a cost function that characterizes the tradeoff between energy efficiency and occupant comfort. In practice, the quantity of tradeoff is different for each occupant and is a priori unknown to the controller. It makes sense to learn an ideal tradeoff from occupant-reported data [12] and then adapt the MPC operation in response, which is an example of MPC with a partially unknown cost function.

2) Clinical-Inventory Management: Inventory management in hospitals involves periodically restocking drugs and medical supplies, and MPC for inventory management [13], [14], [15], [16] is powerful as it naturally captures the dynamics of consuming and purchasing drugs and supplies. Although past work typically assumes that consumption dynamics are completely characterized, it is not realistic for the demand in hospitals due to unforeseeable medical emergencies. It then makes sense from a practical standpoint to learn about the demand from such events and then adapt the MPC operation in response, which is an example of MPC with learning for the dynamics.

B. Exploration–Exploitation Tradeoff

A challenge in LB-MPC is to jointly optimize the control to minimize a cost function and to steer the system to get more information about unknown system or cost parameters [17]. This exploration–exploitation tradeoff has been formally studied in the setting of multiarmed bandits (MABs) [18], [19], [20] and RL for finite Markov chains [21], [22], [23] and for linear systems [24], [25], [26], [27].

Most work on MABs assumes (weak) stationarity because computing the optimal policy with nonstationary is PSPACE-hard [28]. In RL of control systems, past work on nonlinear systems is limited [29], [30], [31], [32], [33], [34] because the optimal controller for linear systems with a quadratic cost is completely characterized by the algebraic Riccati equation: This allows one to convert the RL problem into simply a parameter estimation problem. However, extending these ideas to nonlinear systems is nontrivial as there is no such simple characterization of the optimal controller, and therefore, alternative approaches are needed. We design a learning-based controller for nonlinear and nonstationary systems where the policy explores to improve the estimation methodology embedded in the learning mechanism.

C. Outline

The rest of this article is organized as follows. Section II covers preliminaries. Section III defines our setup and proves safety properties for a class of control policies. Section IV introduces $N$-step dynamic regret, and Sections V and VI present a finite sample analysis for the parameter estimation and regret analysis for the nonmyopic $\epsilon$-greedy algorithm. Numerical experiments are done in Section VII. Finally, Section VIII concludes this article.
II. PRELIMINARIES

A polytope \( P \) in \( \mathbb{R}^n \) can be represented as an intersection of a set of half-spaces [35]:
\[
P = \{ x : P_i x \leq q_i, \ i = 1, \ldots, d \}, \quad P_i \in \mathbb{R}^{n \times 1}, \ q_i \in \mathbb{R}^d.
\]
Let \( U \) and \( V \) be two sets. The linear transformation of \( U \) by a matrix \( \mathcal{R} \) is \( \mathcal{R}U = \{ Ru : u \in U \} \). Their Minkowski sum [36] is defined as \( U \oplus V = \{ u + v : u \in U; v \in V \} \) and Pontryagin set difference [37] is defined as \( U \ominus V = \{ u : u \in U \setminus V \} \). Note \( \mathcal{R}(U \oplus V) \subseteq \mathcal{R}U \ominus \mathcal{R}V \) and \( (U \ominus V) \oplus V \subseteq U \).

III. PROBLEM FORMULATION

Let \( x_t \in \mathbb{R}^n \) be states and \( u_t \in \mathbb{R}^m \) be inputs. We assume that \( x_t \in \mathcal{X} \) and \( u_t \in \mathcal{U} \) are constrained by (compact) polytopes \( \mathcal{X} \) and \( \mathcal{U} \). The true system dynamics are \( x_{t+1} = f(x_t, u_t, \theta_0) = Ax_t + Bu_t + g(x_t, u_t, \theta_0) \), where \( A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m} \), and \( \theta_0 \in \Theta \) for some compact set \( \Theta \subseteq \mathbb{R}^{q} \), and the nonlinear function \( g(x_t, u_t, \theta_0) : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n \) is parametrically unknown \( (\theta_0 \text{ is unknown}) \). This setup can handle stochastic costs \( c_t \) (as opposed to rewards) by setting \( r_t = -c_t \). We standardize our notation for rewards.

The control problem is to sequentially choose inputs to maximize expected total reward at the end of a finite-time horizon \( T = \{0, \ldots, T\} \). At time \( t \), the controller has access to past rewards, inputs, and states. Hence, any policy \( u_t = \lambda_t(F_t) \) will be a sequence (with respect to \( t \)) of functions of \( F_t = \{r_0, \ldots, r_{t-1}, u_0, \ldots, u_{t-1}, x_0, \ldots, x_t\} \).

We distinguish between different policies by using superscripts for the sequence of functions \( \lambda_t \) characterizing the policy.

A. LMBPC Formulation

LMBPC uses two models: a learned model to enhance performance and a nominal model to provide robustness [5]. Because \( A \) and \( B \) are known in the setup, the controller uses as its nominal model \( x_{t+1} = Ax_t + Bu_t + g(x_t, \bar{u}_t, \theta_0) \), where \( \bar{u}_t \in \mathbb{R}^m \) is the system state of the nominal model. The "\( \bar{\theta} \)" notation denotes that the initial condition is taken to be \( \bar{x}_t = x_t \), where \( x_t \) is the true state at time \( t \). Because \( g(\cdot, \cdot, \bar{\theta}) \) is also known, the controller uses as its learned model \( \tilde{x}_{t+1} = Ax_t + Bu_t + g(\tilde{x}_{t+1}, \tilde{u}_t, \bar{\theta}_0) \), where \( \tilde{x}_t \) is the system state of the learned model and \( \bar{\theta}_0 \) is the controller’s estimator of \( \theta_0 \) at time \( t \). Here, LMBPC learns the true dynamics by updating its estimate of \( \bar{\theta}_0 \) as more state measurements become available.

We must first discuss the terminal set used for the MPC. Assuming that \((A, B)\) is stabilizable, there exists a constant state-feedback matrix \( K \in \mathbb{R}^{n \times m} \) such that \((A+BK)\) is Schur stable. We assume that \( \Omega \subseteq \mathcal{X} \) is a maximal output admissible disturbance invariant set [37], meaning that for some stabilizing \( K \) it satisfies: 1) \( \Omega \subseteq \{ \mathcal{P} : \mathcal{P} \in \mathcal{X} : K\mathcal{P} \in \mathcal{U} \} \) (constraint satisfaction) and 2) \((A+BK)\Omega \subseteq \mathcal{W} \subseteq \Omega \) (disturbance invariance). The intuition is that \( \Omega \) is a set of states satisfying the constraints \( \mathcal{X} \) for which there exists a feasible action keeping the true state within \( \Omega \) despite the uncertainty of the nominal model. Several algorithms [37], [38], [39], [40] can compute this set, and therefore, we assume that \( \Omega \) is available to the controller.

With the set \( \Omega \), we consider an (simplified) LMBPC variant that maximizes the expected \( N \)-step reward. Our results can be generalized straightforwardly to the full formulation [5], but we do not consider this as it adds substantial notational complexity that hinders showcasing the stochastic aspects of our setting. The LMBPC formulation of a finite horizon \( N \) is
\[
N(x_t, \theta, t) = \max_{k=0}^{N} h(x_t, \bar{u}_t, \bar{\theta}_0, \theta, k)
\]
\[
\text{s.t.} \quad \tilde{x}_{t+k+1} = A\tilde{x}_{t+k} + Bu_{t+k} + g(\tilde{x}_{t+k+1}, \bar{u}_t, \bar{\theta}_0), \quad k \in \{N-1\},
\]
\[
\bar{x}_{t+k} \in \mathcal{X}, \quad k \in [N],
\]
\[
\bar{u}_t \in \mathcal{U}, \quad k \in [N],
\]
\[
\tilde{x}_{t+1} \in \Omega \ominus \mathcal{W}, \quad \tilde{x}_{t} = \tilde{x}_{t+1} = x_t
\]
where \( \{k\} = \{0, \ldots, k\} \) and \( [k] = \{1, \ldots, k\} \). The difference between this simplified variant and the full formulation is that, here, we apply the invariant set \( \Omega \) at the first time step, an idea previously used in [41], whereas the full formulation uses a robust tube framework to apply \( \Omega \) at the \( \mathcal{N} \)th time point. Our results apply to the above LMBPC formulation and may generalize to the similar variants, but it is unclear if they would generalize to other LMBPC forms without further study.

B. Safety of LMBPC Variant

Because applying the invariant set to the first time point in an MPC formulation is nonstandard, we first formally prove that this LMBPC variant ensures recursive properties of robust constraint satisfaction and robust feasibility.

**Theorem 1:** Suppose that \( \{u_{t+1}, \ldots, u_{N+1}\} \) are feasible for \( V_N(x_t, \theta, t) \) for any \( \theta \). If \( \Omega \) is a maximal output admissible disturbance invariant set, then choosing \( u_t = u_{t+k} \) ensures that: 1) \( x_{t+1} \in \mathcal{X} \) (robust constraint satisfaction) and 2) there exists values \( \{u_{t+1}, \ldots, u_{N+1}\} \) that are feasible for \( V_N(x_{t+1}, \theta', t+1) \) for any \( \theta' \) (robust feasibility).

**Proof:** Since \( \{u_{t+1}, \ldots, u_{N+1}\} \) are feasible for \( V_N(x_t, \theta, t) \), then \( \Pi_{t+1} = A\Pi_t + Bu_t \in \mathcal{W} \subseteq \Omega \ominus \mathcal{W} \) by (2). By relating the true dynamics to the nominal model, the true next state is \( x_{t+1} = \Pi_{t+1} + w_t \) for some \( w_t \in \mathcal{W} \). This means \( x_{t+1} \in (\Omega \ominus \mathcal{W}) \ominus \mathcal{W} \subseteq \Omega \), the last set inclusion follows from the constraint satisfaction property in the definition of \( \Omega \). By the definition (2) of \( V_N(x_{t+1}, \theta', t+1) \), we have that \( x_{t+1} = x_{t+1} \). However, we just showed that \( x_{t+1} = x_{t+1} \). Hence, \( x_{t+1} = x_{t+1} \). Now, set \( u_{t+1} + Kx_{t+1} = Kx_{t+1} \), and note that the constraint satisfaction property of \( \Omega \) means \( u_{t+1} + Kx_{t+1} \in \mathcal{U} \). Since \( \Pi_{t+1} + Bu_{t+1} \in (A+BK)\Pi_t \ominus \mathcal{W} \subseteq (A+BK)\Pi_t \ominus \mathcal{W} \), we have \( \Pi_{t+1} + Bu_{t+1} \in (A+BK)\Omega \ominus (A+BK)\mathcal{W} \subseteq (A+BK)\Omega \ominus (A+BK)\mathcal{W} \), the last set inclusion follows by the disturbance invariance property of \( \Omega \). Therefore, \( x_{t+1} + Bu_{t+1} \in (A+BK)\Omega \ominus (A+BK)\mathcal{W} \subseteq \mathcal{W} \) for \( k \in [N-1] \). Thus, \( \{u_{t+1}, \ldots, u_{N+1}\} \) are feasible for \( V_N(x_{t+1}, \theta', t+1) \).

**Remark 1:** An important feature of the above result is that there is no required relationship between \( \theta \) and \( \theta' \). Since estimates of \( \theta \) are updated through learning, this shows that the safety properties of this LMBPC variant are decoupled from the design of the learning process.

C. Technical Assumptions

Our learning-based control problem is well posed under certain regularity assumptions described below.

**Assumption 1:** The rewards \( r_t \) are conditionally independent given \( \theta_0 \) and \( x_0 \), or equivalently, given \( \theta_0 \) and the complete sequence of \( \{x_0, \ldots, x_T\} \).
Similar to the independent rewards of the stationary MABs, we have independence of \( r_t \) and \( r_{t'} \). \( \forall t \neq t' \).

Assumption 2: The log-likelihood ratio \( \ell(x, u; t, \theta, \theta') = \log \frac{p(x_t | u_t \theta)}{p(x_t | u_t \theta')} \) is locally \( L_2 \)-Lipschitz continuous with respect to \( x \) on the compact set \( \mathcal{X} \) for \( \theta, \theta' \in \Theta \), \( u \in \mathcal{U} \).

This ensures continuity of the reward distribution with respect to the parameters. If two parameter sets are close to each other in value, then the resulting distributions will also be similar.

Assumption 3: The distribution \( \mathbb{P}_{x,u,\theta} \) for all \( x \in \mathcal{X} \), \( u \in \mathcal{U} \), and \( \theta \in \Theta \) is sub-Gaussian with parameter \( \sigma \), and either \( p(r|x, u, \theta) \) has a finite support or \( \ell(r|x, u, \theta, \theta') \) is locally \( L_{f_{\theta}} \)-Lipschitz with respect to \( r \).

This assumption ensures that sample averages are close to their means and is satisfied by many distributions (e.g., Gaussian with known variance). Our last condition ensures that the dynamics and the expectation function are well behaved.

Assumption 4: Repeated composition of the true dynamics with itself up to \( N - 1 \) times, \( f^{t+k}(x_t, u_{t+1}, \ldots, u_{t+k}, \theta) \), is Lipschitz continuous with respect to \( x_t \in \mathcal{X} \) and \( u_{t+k} \in \mathcal{U} \) with constants \( L_{f_{\theta}} \) and \( L_{f_{\theta}} \), respectively. Besides, the expectation \( h(x_t, u_t, \theta) \), for \( u_t = \Lambda_{f_1}(\mathcal{F}_1) \), is Lipschitz continuous with respect to \( x_t \in \mathcal{X} \) and \( u_t \in \mathcal{U} \) with constants \( L_{h_{\theta}} \) and \( L_{h_{\theta}} \), respectively, for all \( \theta \in \Theta \).

### IV. N-STEP DYNAMIC REGRET

Our interest is in evaluating the performance of an LBMP\( \text{C} \) exploitation policy for a given \( N \leq T \) that is \( \Lambda_{t,N}^G(\mathcal{F}_t) = u_{t,N}^G(\theta_t) \) for the corresponding value from the maximizer of \( V_N(x_t, \theta_t, t) \), where \( \theta_t \) is the control policy’s estimates of the unknown \( \theta_0 \). Data-driven policies are often evaluated by comparing performance to a benchmark policy, and it is typical to benchmark using the optimal policy [42, 43, 44]. In our setting, the optimal policy is a sequence of functions \( \Lambda_{t,F_1} \) for maximizing \( \sum_{t=0}^{T-1} h(x_t, u_t, \theta_0) \) subject to the control policy (which does not include \( \theta_0 \)). However, computing optimal policies for the problems we consider is PSPACE-hard [28]. Even their structure is not known for our setup, including for the specific case of linear dynamics and quadratic cost function with unknown coefficients.

An alternative benchmark is an oracle policy that has perfect knowledge of \( \theta_0 \). Specifically, we will use the LBMP\( \text{C} \) oracle policy that is \( \Lambda_{t,N}^G(\mathcal{F}_t) = u_{t,N}^G(\theta_t) \) for the corresponding value from the maximizer of \( V_N(x_t, \theta_0, t) \), as defined in (2). However, there are two subtleties that have to be discussed.

The first subtlety is that the horizon length of the LBMP\( \text{C} \) oracle policy could potentially be different than the horizon length of the LBMP\( \text{C} \) policy. However, using different control horizon lengths can lead to different sums of expected rewards over the entire control horizon \( T \). Though this behavior is well known within the MPC community, its implication on evaluating learning-based control policies has not been previously appreciated. The implication is that comparing policies with different horizon lengths leads to a poorly defined regret notion, and that we should compare oracle policies and learning-based policies with the same finite horizon.

The second subtlety is that the presence of nonlinear dynamics in our setup means that the state trajectory of a system always controlled by a benchmark policy can be very different than that of a system always controlled by a learning policy, even if the learning policy converges toward the benchmark policy. For this reason, we define a regret notion to compare a finite-horizon benchmark policy to a finite-horizon learning-based policy. We consider an \( \epsilon \)-greedy policy \( \Lambda_{t,N}^G \), respectively, and let \( x_t \) and \( u_t \) be the state and input for the system as controlled by the oracle policy \( \Lambda_{t,N}^G \), respectively, and let \( x_{t'} \) and \( u_{t'} \) be the state and input for the system as controlled by the \( \epsilon \)-greedy policy \( \Lambda_{t,N}^G \), respectively. Then, the expected \( N \)-step dynamic regret is defined as

\[
R_{N,T} = \sum_{t=0}^{T} h(x_t, \Lambda_{t,N}^G(\mathcal{F}_t), \theta_0) - h(x_t, \Lambda_{t,N}^G(\mathcal{F}_t), \theta_0)
\]

where \( \mathcal{F}_t \) is as defined in (1) and \( \mathcal{F}_t' \) is as defined in (1) with \( x' \) and \( u' \) replacing \( x \) and \( u \), respectively. This definition is closely related to the traditional dynamic regret [45, 46], and the novel aspect of ours is that it compares two \( N \)-step finite-horizon policies.

### V. PARAMETER ESTIMATION

Let the variables \( \{r_i\}_{i=0}^{T-1} \) be the actual observed values of the rewards up to time \( t \). Using Assumption 1, the joint likelihood \( p(r_{\{i\}_{i=0}^{T-1}} | x_0, \ldots, x_T, u_0, \ldots, u_{T-1}, \theta) \) can be expressed as \( \prod_{t=0}^{T-1} p(r_t | x_t, u_t, \theta) P(x_{t+1} | x_t, \theta) \). Here, the one-step transition likelihood \( P(x_{t+1} | x_t, \theta) \) is a degenerate distribution with all probability mass at \( x_t \), by perpetuation of the dynamics \( f(x_t, u_t, \theta) \) with initial conditions \( x_{t+1} \). Thus, the maximum likelihood estimator (MLE) for \( \theta \) is

\[
\hat{\theta}_T = \arg \min_{\theta \in \Theta} \sum_{t=0}^{T-1} \log p(r_t | x_t, u_t, \theta) s.t. x_{t+1} = f(x_t, u_t, \theta) \forall t \in \{0, \ldots, T-1\}.
\]

This MLE problem can be computed using optimization, dynamic programming, or various filtering techniques for different problem structures. The Kalman filter (KF) is a recursive estimator for linear–quadratic discrete-time systems. In more complex systems with non-Gaussian distributions and nonlinear dynamics, the extended KF and the particle filter are well-known estimators [47, 48, 49]. For practical purposes, these efficient approaches motivate the use of MLE in our policy. Furthermore, if the controller did not have perfect state measurements, we could use the noisy state data to estimate the dynamics in the constraints of (4) [50, 51], which would also alleviate any potential infeasibility issues of the MLE.

We further analyze the concentration properties of the solution to (4) and take an approach to the theoretical analysis that generalizes that of [20]. We begin by introducing the notion of trajectory Kullback–Leibler (KL) divergence. Since this problem includes the joint distribution of a trajectory of values, the concentration bound for the parameter estimates is computed with regard to the trajectory KL divergence.

**Definition 1:** The trajectory KL divergence between the parameter trajectories \( \theta, \theta' \in \Theta \) with the same input sequence \( \Pi_T = \{u_{t}\}_{t=0}^{T-1} \) is

\[
D_{KL}(\theta || \theta') = \sum_{t=0}^{T-1} D_{KL}(\mathbb{P}_{f_t(x_t, u_t, \theta, \theta')} || \mathbb{P}_{f_t(x_t, u_t, \theta', \theta')}),
\]

where \( \Pi_t \) is the given sequence of control inputs from time 0 to \( t \), \( f_t \) is the repeated composition of the dynamics \( f \) with itself \( t \) times subject to \( \Pi_t \), and \( D_{KL} \) is the standard KL divergence.

We have an observability assumption with the implication that the distance between two different parameters \( \theta, \theta' \in \Theta \) is bounded proportional to their trajectory KL divergence.

**Assumption 5:** For a given input sequence \( \Pi_T \) and parameters \( \theta \neq \theta' \), if \( D_{KL}(\theta || \theta') \leq \delta \), then \( ||\theta - \theta'|| \leq C\delta \) for \( C > 0 \).

We next reformulate the MLE problem (4) by removing the state dynamics constraints through repeated composition of \( f \), that is, \( \hat{\theta}_T = \arg \min_{\theta \in \Pi_T} \sum_{t=0}^{T-1} \log \frac{p(r_t | f_t(x_t, u_t, \theta), \theta)}{p(r_t | f_t(x_t, u_t, \theta', \theta))}. \) This reformulation is helpful for our theoretical analysis since for fixed \( \theta \), the expected value of the above objective function under \( \mathbb{P}_{f(x_t, u_t, \theta)} \) is simply \( \frac{1}{T} D_{KL}(\theta || \theta'). \) Hence, we can interpret the MLE problem as minimizing the trajectory KL divergence between the distribution of potential sets of parameters and that of the true parameter set. This interpretation is helpful for us to derive our concentration inequalities. For conciseness of our analysis in this article, we present the final...
Algorithm 1: Nonmyopic $\epsilon$-Greedy Algorithm.

1: Set $\epsilon > 0$ and $x_0 \in \mathcal{X}$
2: for $t \in T$ do
3: Set: $\epsilon_t = \min \{1, \frac{1}{t}\}$
4: Sample: $s_t \sim \text{Bernoulli}(\epsilon_t)$
5: if $s_t = 1$ then
6: Randomly select: $u_{t|t} \in \mathcal{U}(x_t)$
7: $\Lambda_{t|t}^{N}(\mathcal{F}_t) = u_{t|t}$
8: else
9: Compute: $\tilde{\theta}_t$ from (4)
10: Compute: $u_{t|t}^*(\tilde{\theta}_t)$ from $V_N(x_t, \tilde{\theta}_t, t)$ (2)
11: $\Lambda_{t|t}^{N}(\mathcal{F}_t) = u_{t|t}^*(\tilde{\theta}_t)$
12: end if
13: Observe: $r_t$ and $x_{t+1}$
14: end for

For clarity, we consider a randomization at the initial system state and then assume noise-free transients for the subsequent states, which is common in the line of RL for finite sample analysis [52, 53, 54, 55]. Our analysis here provides a strong ground for the generalization of our policy to the setting of imperfect state measurements as an important direction for future work. Note that the exploration probability $\epsilon_t$ decays over time. This reduces the cost of exploration by ensuring that the algorithm makes fewer unnecessary explorations as more data are collected and the estimates of our policy improve.

A. Lipschitzian Stability of Non-Myopic Exploitation

We prove Lipschitzian stability, with respect to perturbations of parameter values, of optimal solutions of the nonmyopic exploitation policy $\Lambda_N^*(\mathcal{F}_t) = u_{t|t}^*(\tilde{\theta}_t)$ by proving a second-order growth condition and Lipschitz continuity of the difference of the perturbed and unperturbed objective functions.

Lemma 1: Suppose that $U_{N,t} = \{u_{t|t}, \ldots, u_{t+N|t}\}$ is a feasible input sequence for $V_N(x_t, \tilde{\theta}_t, t)$. Let $J_N(x_t, U_{N,t}, \tilde{\theta}_t, t)$ be the estimated N-step reward of this input sequence at time $t$, i.e.,

$$J_N(x_t, U_{N,t}, \tilde{\theta}_t, t) = \sum_{k=0}^{N} h(\xi_{t+k|t}, u_{t+k|t}, \tilde{\theta}_t)$$

where $\xi_{t+k|t} = f(\xi_{t+k}, u_{t+k}, \tilde{\theta}_t)$ for $k \in \{N - 1\}$ as given in (2). Then, $J_N(x_t, U_{N,t}, \tilde{\theta}_t, t)$ is $(L_{f,u}, L_{h,u})$-Lipschitz continuous with respect to $U_{N,t}$ on the compact set $U_{N+1}$ for any feasible input sequence $U_{N,t} = \{u_{t|t}, \ldots, u_{t+N|t}\}$.

Proof: By Assumption 4, $f^{t+k}(x_t, x_{t+k|t}, \tilde{\theta}_t)$ is $L_{f,u}$-Lipschitz continuous and $h(\xi_{t+k|t}, u_{t+k|t}, \tilde{\theta}_t)$ is $L_{h,u}$-Lipschitz continuous with respect to $u_{t+k|t} \in \mathcal{U}$. Then, by preservation of Lipschitz continuity across functional compositions and addition, we have the desired condition.

Lemma 1 implies the second-order growth condition for $V_N(x_t, \tilde{\theta}_t, t)$ since it shows $J_N$ increases at least linearly over a compact set. We next present the second condition required for the Lipschitzian stability of the maximizer of $V_N(x_t, \tilde{\theta}_t, t)$.

Assumption 6: Let $U_{N,t}^*(\tilde{\theta}_t) = \{u_{t|t}^*(\tilde{\theta}_t), \ldots, u_{t+N|t}^*(\tilde{\theta}_t)\}$ be maximizers of $V_N(x_t, \tilde{\theta}_t, t)$ and $V_N(x_t, \tilde{\theta}_t, t)$. Then, for $\kappa \geq 0$, we have

$$\|J_N(x_t, U_{N,t}^*(\tilde{\theta}_t), \tilde{\theta}_t, t) - J_N(x_t, U_{N,t}(\tilde{\theta}_t), \tilde{\theta}_t, t) - J_N(x_t, U_{N,t}^*(\tilde{\theta}_t), \tilde{\theta}_t, t)\| \leq \kappa \|\tilde{\theta}_t - \theta\|$$

We now give a sufficient condition for Assumption 6.

Proposition 1: For any $\theta \in \Theta$ and real constant $L_\omega \geq 0$, if $\|\nabla_u J_N(x_t, U_{N,t}^*(\tilde{\theta}_t), \tilde{\theta}_t, t) - \nabla_u J_N(x_t, U_{N,t}(\tilde{\theta}_t), \tilde{\theta}_t, t)\| \leq L_\omega \|\tilde{\theta}_t - \theta\|$ holds, then Assumption 6 is satisfied.

Proof: Let $s(\tau) = U_{N,t}^*(\tilde{\theta}_t) + \tau \cdot (U_{N,t}(\tilde{\theta}_t) - U_{N,t}^*(\tilde{\theta}_t))$. This implies $s(0) = U_{N,t}^*(\tilde{\theta}_t)$ and $s(1) = U_{N,t}(\tilde{\theta}_t)$. Then

$$J_N(x_t, U_{N,t}^*(\tilde{\theta}_t), \tilde{\theta}_t, t) - J_N(x_t, U_{N,t}(\tilde{\theta}_t), \theta, t)$$

$$= J_N(x_t, U_{N,t}^*(\tilde{\theta}_t), \tilde{\theta}_t, t) - J_N(x_t, U_{N,t}(\tilde{\theta}_t), \tilde{\theta}_t, t)$$

$$= \int_0^1 \nabla_j J(x_t, s(\tau), \tilde{\theta}_t, t) (U_{N,t}^*(\tilde{\theta}_t) - U_{N,t}(\tilde{\theta}_t)) d\tau$$

$$- \int_0^1 \nabla_j J(x_t, s(\tau), \tilde{\theta}_t, t) (U_{N,t}^*(\tilde{\theta}_t) - U_{N,t}(\tilde{\theta}_t)) d\tau$$

(6)
where (8) follows by the H"{o}lder’s inequality, and (9) follows by the assumed property in Proposition 1. This gives us the desired result in Assumption 6 by setting \( \kappa = \sqrt{NL_J}. \)

**Lemma 2:** If the state dynamics \( f(x, u, \theta) \) and the expectation function \( h(x, u, \theta) \) are polynomial functions, then the sufficient condition given in Proposition 1 holds.

**Proof:** Since (5) is the average of compositions of two polynomials \( f \) and \( h \), it is polynomial. Then, \( \nabla_u J_x(x, U, \theta, t) \) is polynomial on the bounded domain \( \mathcal{X} \times \mathcal{U}^{N+1} \times \Theta \). Hence, by [56, Corollary 8.2], \( \nabla_u J_x(x, U, \theta, t) \) is locally Lipschitz with respect to \( \theta \in \Theta \) for any \( x \in \mathcal{X}, U \in \mathcal{U}^{N+1}, t \in \mathcal{T}. \)

A specific example where Lemma 2 holds is a discrete-time linear time-invariant system with \( f(x, u, \theta) = Ax + Bu \) and \( h(x, u, \theta) = x^TQx + u^TRu \), where \( \theta = \{Q, R, A, B\} \).

**Lemma 3:** If Assumption 6 and Lemma 1 hold, then the Lipschitzian stability property follows by [57, Proposition 43.2], i.e., \( ||U^*_{\mathcal{N}_t}(\tilde{\theta}) - U^*_{\mathcal{N}_t}(\tilde{\theta})|| \leq c_u^{-1}\kappa||\tilde{\theta} - \theta|| \) for \( c_u > 0 \).

Since \( ||w_{\mathcal{U}_t}(\tilde{\theta}) - u_{\mathcal{U}_t}(\tilde{\theta})|| \leq ||U^*_{\mathcal{N}_t}(\tilde{\theta}) - U^*_{\mathcal{N}_t}(\tilde{\theta})|| \), we conclude that the nonmyopic exploitation policy \( \Lambda^+_t(N, F_t) = u^*_{\mathcal{U}_t}(\tilde{\theta}) \) corresponding from the maximizer of \( V_N(x_t, \tilde{\theta}_t) \) is \( c_u^{-1}\kappa \)-Lipschitz continuous with respect to \( \tilde{\theta}_t \in \Theta \).

**B. Regret Analysis**

We next characterize the N-step dynamic regret \( R_{N,T} \) of Algorithm 1. By definition, \( R_{N,T} \) compares the LBMPC oracle policy \( \Lambda^{N,N}(F_t) \) for the system \( x_t, u_t \) as controlled by the learning policy to our nonmyopic \( \varepsilon \)-greedy policy \( \Lambda^{N,N}(F_t) \) for the system \( x'_t, u'_t \) as controlled by the learning policy that uses the LBMPC policy \( \Lambda^{N,N}(F_t) \) at greedy exploitation steps. We start by bounding a weaker notion that compares the actions chosen under the states \( x'_t \) achieved by \( \Lambda^{N,N}(F_t) \):

**Theorem 3:** The nonmyopic \( \varepsilon \)-greedy policy \( \Lambda^{N,N}(F_t) \) and the LBMPC oracle policy \( \Lambda^{N,N}(F_t) \) satisfy the following regret for the system states \( x'_t \) that are achieved by \( \Lambda^{N,N}(F_t) \):

\[
R_{1:T}(N) = \sum_{t=0}^{T} h(x'_t, \Lambda^{N,N}(F_t), \theta_0) - \sum_{t=0}^{T} h(x'_t, \Lambda^{N,N}(F_t), \theta_0) 
\leq \mathcal{M} \exp \left( \frac{\sigma^2}{\sqrt{c_u}} \right) \sqrt{T \log T} + \frac{\mathcal{M}c(1 - \log(c + 1) + \log T)}{c_u} \sqrt{3T \log T} + \frac{L_{h,u}K \mathcal{M}}{c_u} \sqrt{\frac{2 \log T}{L^2}}
\]

where \( C > 0, c_f(d_x, d_u) \) is the constant in Theorem 2, and \( C \) is a bound on the finite summation \( \sum_{t=1}^{T} (\log T)^2 \).

**Proof:** For notational convenience, let \( \mathcal{E}[M_t] = h(x_t, \Lambda^{N,N}(F_t), \theta_0) - h(x'_t, \Lambda^{N,N}(F_t), \theta_0) \). Let \( \mathcal{T}^N \subset \mathcal{T} \) and \( \mathcal{T}^N \subset \mathcal{T} \) be the set of random time points that Algorithm 1 performs exploration and exploitation, respectively. Noticing that the cardinalities \( |\mathbb{T}^N| \) and \( |\mathbb{T}^N| \) are random variables, we have:

\[
\mathcal{E}[M_t] = h(x_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0) - h(x'_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0) + \sum_{t \in \mathbb{T}^N} h(x'_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0) - h(x'_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0)
\]

We note that \( \mathcal{E}[M_t] \) is a bound value since \( \lambda, \theta_t, \) and \( \mathcal{U} \) are all compact sets and \( h(x, u, \theta) \) is a bounded continuous function on this domain. Then, assuming \( \mathcal{E}[M_t] \leq \mathcal{M} \), we obtain:

\[
\mathcal{E}[M_t] \leq \mathcal{M} \mathcal{E}[\mathbb{T}^N]
\]

We can rewrite each term inside the summation above as:

\[
h(x'_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0) - h(x'_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0)
\]

\[
= h(x'_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0) - h(x'_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0)
\]

\[
+ h(x'_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0) - h(x'_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0)
\]

\[
= (13, a) + (13, b).
\]

Then \( \mathcal{E}[\delta_t] = \mathcal{M} \mathcal{E}[\mathbb{T}^N \times\mathcal{U}] \), we obtain:

\[
\sum_{t \in \mathcal{T}^N} (13, a)
\]

\[
= h(x'_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0) - h(x'_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0)
\]

\[
+ h(x'_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0) - h(x'_t, u_{\mathcal{U}_t}(\tilde{\theta}), \theta_0)
\]

\[
= (13, a) + (13, b).
\]

Now, we have \( \mathcal{E}[\delta_t] \leq \mathcal{M} \mathcal{E}[\mathbb{T}^N \times\mathcal{U}] \), then, for \( \delta_t = \sqrt{4L^2_{\mathcal{U}_t} \sigma^2 \log \eta(t)/\sqrt{\eta(t)}} \), we obtain:

\[
\sum_{t \in \mathcal{T}^N} (13, b)
\]

\[
= \frac{L_{h,u}K \mathcal{M}}{c_u} \mathcal{E}[\mathbb{T}^N \times\mathcal{U}]
\]

To bound the second term in (13), recall \( \mathcal{E}[M_t] \leq \mathcal{M} \). Then:

\[
\sum_{t \in \mathcal{T}^N} (13, b) \leq \mathcal{M} \sum_{t \in \mathcal{T}^N} \exp \left( -\mathcal{M}c(1 - \log(c + 1) + \log T) + \frac{L_{h,u}K \mathcal{M}}{c_u} \sqrt{3T \log T} + \frac{\mathcal{M}c(1 - \log(c + 1) + \log T)}{c_u} \sqrt{3T \log T} + \frac{L_{h,u}K \mathcal{M}}{c_u} \sqrt{\frac{2 \log T}{L^2}} \right)
\]
\[ M\sum_{t \in T_{L^N}} \exp\left( -\frac{\delta^2 (t-1)/2 + \gamma^2 (d_x, d_y)}{2L^2_t, \sigma^2} \right) \]
\[ \leq M \exp \left( \frac{\gamma^2 (d_x, d_y)}{2L^2_t, \sigma^2} \right) \]
\[ \left( \sum_{t=0}^{T-1} \exp(-\log t) + \sum_{t \in T_{L^N}} \exp(-\log t) \right) \]
\[ \leq M \exp \left( \frac{\gamma^2 (d_x, d_y)}{2L^2_t, \sigma^2} \right) (C + \log T) \]
where (21) follows by Theorem 2 and C can be approximated as 2.2232. Finally, we bound the first term in (12):
\[ M\mathbb{E}[\#T^{\text{non}}] = M\sum_{t=0}^{T} \min\{1, \frac{t}{T}\} \leq M(e + \sum_{t=1}^{T} \frac{t}{T}) \leq M \left( 1 - \log(e+1) + \log T \right). \]

Substituting these into (12) as
\[ \left( \sum_{t=0}^{T} \mathbb{E}[\mathcal{M}] \right)^{\text{max}} \leq M \exp \left( \frac{\gamma^2 (d_x, d_y)}{2L^2_t, \sigma^2} \right) (C + \log T) \]
and taking the expectation gives us the desired result.

**Theorem 4:** For \( 0 < c \leq \sqrt{\frac{T}{10}} \), the expected N-step dynamic regret \( R_{N,T}(\mathcal{F}_t) \) for a policy \( \Lambda_t^{O,N}(\mathcal{F}_t) \) computed by Algorithm 1 satisfies
\[ R_{N,T}(\mathcal{F}_t) \leq 2 L_{h,x} \sqrt{T} \log(M) + 2 L_{h,x} \sqrt{\log(M)} \sqrt{T} \frac{1}{\sqrt{10}} \]
and with probability at least
\[ 1 - (\frac{T}{2} - 2\sqrt{T}) \exp\left( -\frac{2 \log(2\sqrt{T} + 1) + \frac{2 \log^2(\sqrt{T} + 1)}{1 + 2 \log \cdot \frac{\log^2(\sqrt{T} + 1)}{\log^2(\sqrt{T} + 1)}} - \frac{2 \log^2(\sqrt{T} + 1)}{1 + 2 \log \cdot \frac{\log^2(\sqrt{T} + 1)}{\log^2(\sqrt{T} + 1)}}}{\log^2(\sqrt{T} + 1)} \right), \]
where \( C = c_{\lambda, C} \left( (\mathcal{B} + L_{f,u} \alpha)c \right) \sqrt{\log^2(\sqrt{T} + 1)} \).

**Proof:** By Assumption 4 and the upper bound in (11), we have
\[ R_{N,T}(\mathcal{F}_t) \leq \sum_{t=0}^{T} h(x_t, \Lambda_t^{O,N}(\mathcal{F}_t), \theta_0) - h(x_t, \Lambda_t^{O,N}(\mathcal{F}_t), \theta_0) \]
\[ + \sum_{t=0}^{T} h(x_t, \Lambda_t^{O,N}(\mathcal{F}_t), \theta_0) - h(x_t, \Lambda_t^{O,N}(\mathcal{F}_t), \theta_0) \]
\[ \leq L_{h,x} \sum_{t=0}^{T} \| x_t - x_t' \| + (11). \]

Algorithm 1 performs exploration at random times according to a nonstationary stochastic process over \( \mathcal{T} \). We divide \( \mathcal{T} \) into “inter-explore intervals” composed of an exploration and the subsequent exploitations until the next one is reached. Let \( I_k = [I_{k-1}, I_k] \) be the kth subinterval such that \( I_{-1} = [0, 2\sqrt{T}] \), \( I_0 = [2\sqrt{T}, 1, 1, \ldots] \), and \( I_k = [\delta_{I_k}^T, \delta_{I_k}^T + 1] \) for \( k \in \{1, K \} \), where \( \delta_{I_k}^T \) is the kth exploration step after time \( 2\sqrt{T} \). Then, \( \delta_{I_k}^T \) is an equilibrium for the LBMPC \( \mathcal{F}_t \) for \( t \in [I_k, I_{k+1}] \), where \( I_{k+1} = [S_k, S_{k+1}] = [I_k, I_{k+1}] \) that includes a single exploration at time \( I_k \) followed by exploitation steps thereafter up to \( I_{k+1} \).

Describe Algorithm 1 uses \( \Lambda_t^{O,N}(\mathcal{F}_t) \) at all greedy exploitation steps of \( S_k, k \in [0, K] \). Since \( x_t \) is \( \mathcal{T} \) for \( t \in \mathcal{T}^{\text{max}} \) and \( \mathcal{T} \) is compact, \( \| x_t - x_{eq} \| \leq \text{diam}(\mathcal{X}) \), \( t \in \mathcal{T}^{\text{max}} \). Then, by Assumption 7, \( \| x_t - x_{eq} \| \leq \sum_{t=1}^{T} \text{diam}(\mathcal{X}) + \sum_{t=S_k}^{T} \text{diam}(\mathcal{X}) \leq (1/\alpha) \text{diam}(\mathcal{X}) \) (1/\alpha). Next, suppose instead that \( \Lambda_t^{E,N}(\mathcal{F}_t) \) is used at all greedy exploitation steps of \( S_k, k \in [0, K] \). Observe the convergence of \( \Lambda_t^{E,N}(\mathcal{F}_t) \) to
Fig. 1. Expected ten-step dynamic regret.

\[
\leq \frac{2}{1-\alpha} \text{diam}(X) + \frac{C_C}{\alpha} \sum_{t=1}^{T} \sum_{t'=t+1}^{T-1} \frac{\log t}{\sqrt{t}} \\
\leq \frac{2}{1-\alpha} \text{diam}(X) + \frac{2C_C}{\alpha} \sum_{t=1}^{T} (\log t)^2 \sqrt{t} \\
\leq \frac{2}{1-\alpha} \text{diam}(X) + \frac{2C_C}{\alpha} \sqrt{T} (\log T)^2. \tag{28}
\]

Note \( \mathbb{E} \sum_{j=1}^{t} s_j \geq c \log \frac{e^{(t+1)}}{t} \) and \( \text{Var}(\sum_{j=1}^{t} s_j) = \sum_{j=1}^{t-1} \frac{1}{j} - c \log \frac{e^{(t+1)}}{t} \). Then, by Bernstein’s inequality (see [60, Corollary 2.11]), it follows that (28) holds with probability \( P(\gamma_{t+1-k}^{(T)}) \geq 1 - \sum_{t=1}^{T} \text{diam}(X) + \frac{2C_C}{\alpha} \sqrt{T} (\log T)^2 \geq 1 - (T-2\sqrt{T}) \exp\left(-\frac{4c^2(\log (\frac{e^{(t+1)}}{t}))^2}{2c^2(\log (\frac{e^{(t+1)}}{t}))^2}\right) \). The above bounds for \( \Delta^{+N}(F) \) and (28) for \( \Delta^{+N}(F) \) allow us to bound the deviation of the system trajectory under the learning policy from the one under the oracle policy over \( I_{k, k} \geq 0 \) as \( \sum_{t'=t+1}^{T} \frac{1}{t} - c \log \frac{e^{(t+1)}}{t} \leq 2\sqrt{T} \text{diam}(X) \) and over \( I_{k, k} \geq 0 \) as \( \sum_{t'=t+1}^{T} \frac{1}{t} - c \log \frac{e^{(t+1)}}{t} \leq 2\sqrt{T} \text{diam}(X) \) and \( \text{Var}(\sum_{j=1}^{t} s_j) = \sum_{j=1}^{t-1} \frac{1}{j} - c \log \frac{e^{(t+1)}}{t} \). Bernstein’s inequality (see [60, Corollary 2.11]), it follows that (28) holds with probability \( P(\gamma_{t+1-k}^{(T)}) \geq 1 - \sum_{t=1}^{T} \text{diam}(X) + \frac{2C_C}{\alpha} \sqrt{T} (\log T)^2 \geq 1 - (T-2\sqrt{T}) \exp\left(-\frac{4c^2(\log (\frac{e^{(t+1)}}{t}))^2}{2c^2(\log (\frac{e^{(t+1)}}{t}))^2}\right) \). The above bounds for \( \Delta^{+N}(F) \) and (28) for \( \Delta^{+N}(F) \) allow us to bound the deviation of the system trajectory under the learning policy from the one under the oracle policy over \( I_{k, k} \geq 0 \) as \( \sum_{t'=t+1}^{T} \frac{1}{t} - c \log \frac{e^{(t+1)}}{t} \leq 2\sqrt{T} \text{diam}(X) \) and over \( I_{k, k} \geq 0 \) as \( \sum_{t'=t+1}^{T} \frac{1}{t} - c \log \frac{e^{(t+1)}}{t} \leq 2\sqrt{T} \text{diam}(X) \) and \( \text{Var}(\sum_{j=1}^{t} s_j) = \sum_{j=1}^{t-1} \frac{1}{j} - c \log \frac{e^{(t+1)}}{t} \). Bernstein’s inequality (see [60, Corollary 2.11]), it follows that (28) holds with probability \( P(\gamma_{t+1-k}^{(T)}) \geq 1 - \sum_{t=1}^{T} \text{diam}(X) + \frac{2C_C}{\alpha} \sqrt{T} (\log T)^2 \geq 1 - (T-2\sqrt{T}) \exp\left(-\frac{4c^2(\log (\frac{e^{(t+1)}}{t}))^2}{2c^2(\log (\frac{e^{(t+1)}}{t}))^2}\right) \).

VIII. CONCLUSION

This article studies the intersection of nonlinear MPC and RL. Stability is one of the unique (and not previously well-studied) issues that arises with RL for nonlinear systems. We develop a new class of LBMPc policies that we prove achieves low regret, which is supported by our numerical experiments.

REFERENCES

[1] Y. Abbasi-Yadkori, N. Lazie, and C. Szepesvari, “Model-free linear quadratic control via reduction to expert prediction,” in Proc. 22nd Int. Conf. Artif. Intell. Statist., 2019, pp. 3108–3117.
[2] C. Chen, H. Modares, K. Xie, F. L. Lewis, Y. Wan, and S. Xie, “Reinforcement learning-based adaptive optimal exponential tracking control of linear systems with unknown dynamics,” IEEE Trans. Autom. Control, vol. 64, no. 11, pp. 4425–4443, Nov. 2019.
[3] N. Agarwal, N. Brukhim, E. Hazan, and Z. Lu, “Boosting for control of dynamical systems,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 96–103.
[4] R. Negenborn, B. De Schutter, M. Wiering, and J. Hellendoorn, “Experience-based model predictive control using reinforcement learning,” in Proc. 8th TRAIL Congr., 2004, pp. 1–17.
[5] A. Aswani, H. Gonzalez, S. S. Sastry, and C. Tomlin, “Provably safe and robust learning-based model predictive control,” Automatica, vol. 49, pp. 1216–1226, 2013.
[6] N. Karnchanachari, M. I. Valls, D. Hoeller, and M. Hutter, “Practical reinforcement learning for MPC: Learning from sparse objectives in under an hour on a real robot,” in Proc. 2nd Conf. Learn. Dyn. Control, 2020, pp. 211–224.
[7] S. Gros and M. Zanon, “Reinforcement learning for mixed-integer problems based on MPC,” IFAC-PapersOnLine, vol. 53, no. 2, pp. 5219–5224, 2020.
[8] A. Aswani, N. Master, J. Taneja, D. Culler, and C. Tomlin, “Reducing transient and steady state electricity consumption in HVAC using learning-based model-predictive control,” Proc. IEEE, vol. 100, no. 1, pp. 240–253, Jan. 2012.
[9] A. Afram and F. Janabi-Sharifi, “Theory and applications of HVAC control systems—A review of model predictive control (MPC),” Building Environ., vol. 72, pp. 343–355, 2014.
[10] M. Ostadijafari and A. Dubey, “Linear model-predictive controller (LMPC) for building’s heating ventilation and air conditioning (HVAC) system,” in *Proc. IEEE Conf. Control Technol.*, 2019, pp. 617–623.

[11] J. Fang, R. Ma, and Y. Deng, “Identification of the optimal control strategies for the energy-efficient ventilation under the model predictive control,” *Sustain. Cities Soc.*, vol. 53, 2020, Art. no. 101908.

[12] A. Aswani, Z.-J. Shen, and A. Siddiq, “Inverse optimization with noisy data,” *Oper. Res.*, vol. 66, no. 3, pp. 870–892, 2018.

[13] P. Velarde, J. Jauregui, I. Jurado, I. Fernandez, B. I. Tejera, and J. del Prado, “Application of robust model predictive control to inventory management in hospital pharmacy,” in *Proc. IEEE Emerg. Technol. Factory Autom.*, 2014, pp. 1–6.

[14] G. Schildbach and M. Morari, “Scenario-based model predictive control for multi-echelon supply chain management,” *Eur. J. Oper. Res.*, vol. 252, no. 2, pp. 540–549, 2016.

[15] J. Jauregui, M. Fernandez, and I. Jurado, “An application of economic model predictive control to inventory management in hospitals,” *Control Eng. Pract.*, vol. 71, pp. 120–128, 2018.

[16] I. F. Garcia, P. Chanfreut, I. Jurado, and J. M. Jauregui, “A data-based model predictive decision support system for inventory management in hospitals,” *IEEE J. Biomed. Health Inform.*, vol. 25, no. 6, pp. 2227–2236, Jun. 2021.

[17] A. Mesbah, “Stochastic model predictive control with active uncertainty learning: A survey on dual control,” *Annua. Rev. Control.*, vol. 45, pp. 107–117, 2018.

[18] W. R. Thompson, “On the likelihood that one unknown probability exceeds another in view of the evidence of two samples,” *Biometrika*, vol. 25, nos. 3/4, pp. 285–294, 1933.

[19] S. Agrawal and N. Goyal, “Thompson sampling for contextual bandits with linear payoffs,” in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 127–135.

[20] M. Heger, “Consideration of risk in reinforcement learning,” in *Machine Learning Proceedings*. Amsterdam, The Netherlands: Elsevier, 1994, pp. 105–111.

[21] E. Biyik, J. Margolishi, S. R. Alimo, and D. Sadigh, “Efficient and safe exploration in deterministic Markov decision processes with unknown transition models,” in *Proc. Amer. Control Conf.*, 2019, pp. 1792–1799.

[22] M. Budd, B. Lacerda, P. Duckworth, A. B. Lennox, and N. Hawes, “Markov decision processes with unknown state feature values for safe exploration using Gaussian processes,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 7344–7350.

[23] S. J. Bradtke, B. E. Ydstie, and A. G. Barto, “Adaptive linear quadratic control using policy iteration,” in *Proc. Amer. Control Conf.*, 1994, pp. 3475–3479.

[24] B. Kiumarsi-Khomartash, F. Lewis, and Z. Jiang, “H∞ control of linear discrete-time systems: Off-policy reinforcement learning,” *Automatica*, vol. 78, pp. 144–152, 2017.

[25] A. Cohen, T. Koren, and Y. Mansour, “Learning linear-quadratic regulators efficiently with only $\sqrt{T}$ regret,” in *Proc. 36th Int. Conf. Mach. Learn.*, 2019, pp. 1300–1309.

[26] M. Simchowitz and D. Foster, “Naive exploration is optimal for online LQR,” in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 8937–8948.

[27] C. H. Papadimitrion and J. N. Tsitsiklis, “The complexity of optimal queuing network control,” *Math. Oper. Res.*, vol. 24, pp. 293–305, 1999.

[28] T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause, “Learning-based model predictive control for safe exploration,” in *Proc. IEEE Conf. Decis. Control*, 2018, pp. 6059–6066.

[29] S. Gros and M. Zanón, “Data-driven economic NMPC using reinforcement learning,” *IEEE Trans. Autom. Control*, vol. 65, no. 2, pp. 636–648, Feb. 2020.

[30] S. Kakade, A. Krishnamurthy, K. Lowrey, M. Ohnishi, and W. Sun, “Information theoretic regret bounds for online nonlinear control,” *Adv. Neural Inf. Proc. Syst.*, vol. 33, pp. 15312–15325, 2020.

[31] K. P. Wabersich and M. N. Zeilinger, “Performance and safety of Bayesian model predictive control: Scalable model-based RL with guarantees,” 2020, arXiv:2006.03483.

[32] Y. Fan and Y. Ming, “Efficient exploration for model-based reinforcement learning with continuous states and actions,” in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 3078–3087.

[33] N. M. Boﬁ, S. Tu, and J.-J. E. Slotine, “Regret bounds for adaptive nonlinear control,” in *Proc. 3rd Conf. Learn. Dyn. Control*, 2021, pp. 471–483.

[34] F. Borelli, A. Bemporad, and M. Morari, “Constrained optimal control and predictive control for linear and hybrid systems,” *PredictivE Control for Linear and Hybrid Systems*. Cambridge University Press, 2017.