Online Learning in Control Theory

by

Mohammad Akbari Varnousfaderani

A thesis submitted to the
Department of Mathematics and Statistics
in conformity with the requirements for
the degree of Doctor of Philosophy

Queen's University
Kingston, Ontario, Canada
June 2022

Abstract

In this thesis, we study two classes of problems in optimal control theory involving unknown parameters, with focus on Linear-Quadratic-Gaussian systems.

In the first problem, the control system is known and linear, and a sequence of quadratic cost functions is revealed to the controller in hindsight. The controller aims to achieve a sublinear regret, similar to the online optimization setting. A modified online Riccati algorithm is introduced that under some uniform boundedness assumptions results in a logarithmic regret bound. In particular, the logarithmic regret for the scalar case is achieved without any extra condition, while in the vector case we impose a uniform boundedness assumption. In addition to having a better regret bound, the proposed algorithm has reduced complexity compared to the ones in the literature which mainly rely on solving semi-definite programs in each time step.

In the second problem, the true system transition parameters (matrices A and B) are unknown, and the objective is to design and analyze algorithms that generate control policies with sublinear regret. Recent studies show that when the system parameters are fully unknown, for any algorithm that only uses data from the past system trajectory there exists a choice of the unknown parameters such that the algorithm at best achieves a square-root regret bound, providing a hard fundamental

limit on the achievable regret in general. However, it is known that (poly)-logarithmic regret is achievable when only matrix A, or only matrix B is unknown. We prove a result, showing that (poly)-logarithmic regret is achievable when both of these matrices are unknown, but a hint about one of them is given to the learner periodically. This result is the first attempt to achieve poly-logarithmic regret for the case that both A and B are unknown under milder assumptions.

Acknowledgments

First and foremost, I would like to thank my supervisors, Prof. Bahman Gharesifard and Prof. Tamás Linder for their guidance, patience and support. I thank them for giving me the opportunity of studying and doing research under their supervision. I have been very fortunate to work with them and receive unconditional support and guidance. Without their motivation and encouragement, I would never had the courage to pursue my Ph.D.

I would like to thank the department of Mathematics and Statistics of Queen's university for providing me the great opportunity of learning and doing research. In particular, I thank Professor Abdol-Reza Mansouri for his excellent teaching and support in the stochastic calculus course. I thank Professor Fady Alajaji for both accepting to be part of my thesis committee, and his wonderful teaching in information theory course. I would like to thank my supervisory and thesis committee members, Professor Martin Guay, Professor Thomas Barthelmé, Professor Sarah Dean, for their time and effort. I also thank our graduate program assistant Jennifer Read for her kind support and my colleagues, Daniel Adu, Somya Singh, and Annika Fuernsinn.

I want to also thank my special friends, Hossein and Najmeh, for the joyful moments we have spent together in Kingston.

I am deeply grateful to my wife, Marzieh, for her love and support and the lovely

moments we have had in the past years. I am also thankful to my parents and my family for their unconditional support.

Finally, I would like to thank all my teachers that help me to find my path to the academic world.

Statement Of Originality

The following work is my own and I hereby certify the intellectual content of this thesis is the product of my own work. All references and contributions of other individuals has been cited and sourced appropriately, as defined by the IEEE Citation Reference manual.

Contents

List of Figures

Chapter 1

Introduction

The problem of prediction and decision making using data has many applications in engineering, economy, and social sciences. Examples include, portfolio selection [3, 52], transportation and traffic control [59], power engineering [74], manufacturing and supply chain management. This problem has received substantial attention in recent years [21, 10, 63]. The subject under study in the thesis sits within this general theme of a class of decision making problems, where some properties of the system are not known and the available data is to be used to overcome this lack of information.

Decision making in hindsight has been studied extensively in the machine learning community, especially within the framework of *online optimization* [41, 21]. In this setting, at each time step, a decision maker, or learner, is making a decision on its current state in order to optimize a cost function that is only available in hindsight, i.e., the cost function is only revealed after the decision is made. For this reason, the chosen state does not necessarily correspond to the optimizer of the objective function and the decision maker faces a so-called *regret*. This quantity is defined as the difference between the accumulated cost over time and the cost incurred by the best fixed decision, when all the functions are known in advance, see [75, 40]. The

objective in online optimization is to design an algorithm such that the regret will be sublinear in the time horizon; in other words, the algorithm drives the average regret over time to zero. It turns out, for instance, that under convexity and compactness assumptions, one can guarantee sublinear regret rates by using "greedy" strategies. The literature on online optimization is extremely rich and its connections to many other areas of learning have been explored in recent years [21, 41, 64, 40, 42, 37, 16]. Let us review some facts regarding regret analysis in the context of online optimization here. It is known that for the time horizon T, the best regret bound for convex cost functions is $\mathcal{O}(\sqrt{T})$, and $\mathcal{O}(\log(T))$ for strongly convex cost functions, and there is a fundamental limit for the case of linear cost functions; in this case no algorithm can have a regret bound better than $\Omega(\sqrt{T})$ [41].

Unlike the classical setting of online optimization, where the decisions of the learner are chosen according to a cost function, in many realistic scenarios, the learner's decisions are inputs to a *control system*, and the control system is faced with a cost function, which has been studied in the context of optimal control theory. In this thesis, the class of linear-quadratic-Gaussian (LQG) systems, where the state and control actions are coupled through a linear dynamical system and a quadratic cost function, is considered. In addition to the fact that this setup is a cornerstone benchmark in optimal control and decision theory, it formulates many practical engineering problems [69, 49, 61, 58, 31]. It is well-known that when the costs and the dynamics of the control system are known, the optimal controller is a linear feedback of the control state that can be derived by dynamic programming for the finite-horizon problem or by solving an algebraic Riccati equation for the infinite-horizon problem [13].

The scenario where the dynamics of the control system or the cost functions are *unknown*, which is the subject of this work, are much more challenging. Here, two different problems are considered: the first problem has the assumption that the dynamics of the system is known to the controller, but that the cost functions are time-varying and *unknown*, and the second problem considers settings where the dynamics of the system is fixed but *unknown* to the controller.

For the first problem, one can think about the underlying control system as a dynamic constraint on the online optimization problem. Examples include power supply management in the presence of time-varying energy costs due to demand fluctuations and tracking of an adversarial target. In such scenarios, decisions are usually assumed to be a function of the current state, which is referred to as a *policy*. As usual, the regret is defined as the difference between the accumulated costs incurred by control actions made in hindsight using previous states and the cost incurred by the best fixed admissible policy when all the cost functions are known in advance. Similar to online optimization, the objective is to design algorithms to generate policies which make the regret function grow sublinearly. This setting is connected to online optimization in dynamic environments [39], where the decisions are constrained in dynamics chosen by the environment.

The second problem, the scenario that the dynamics of the control system is unknown, is in fact a core benchmark in adaptive control theory. Most of the classical literature of adaptive control has focused on asymptotic results [47, 23, 24, 19, 15]; however, with the emergence and success of machine learning techniques in strategic decision making settings [65], robotics [50], and biology [53], this classical control problem has been revisited from a learning-theoretic perspective using new tools from

online learning and reinforcement learning [60]. A standard notion for studying the non-asymptotic performance of algorithms for the adaptive control of LQG systems is the regret. Here, the regret is defined as the difference between the accumulated cost of the controller policy generated by the algorithm and the optimal accumulated cost which is achievable by a controller that knows the costs and dynamics of the system and the objective is again to design algorithms that achieve sublinear regret. Recently, there has been a lot of interest in designing algorithms that are computationally efficient and achieve a low regret. In what follows, we review the literature on the problems that we have discussed.

1.1 Literature Review

The problem setting with unknown cost functions is similar to the one studied in [25] where an online version of LQG control is introduced. In particular, in [25] an online gradient descent algorithm with a fixed learning rate is proposed, where in each iteration, a projection onto a bounded set of positive-definite matrices is taken, which itself relies on solving a semi-definite program. Under the assumptions that the underlying system is controllable, the cost functions are bounded, and the covariance of the disturbance is positive definite, it is proved that the regret is sublinear, and grows as $\mathcal{O}(\sqrt{T})$, where T is the time horizon. Other closely related works are [4] and [5], where the cost functions are assumed to be convex and globally Lipschitz functions. In contrast to [25], the noise assumed in [4] is adversarial, and [5] achieves a regret bound of $\mathcal{O}((\log(T))^7)$. In these works, the generated control actions, which lead to a sublinear regret bound, are linear feedbacks which rely on a finite history of the past disturbances. Similarly, in another recent work [34] a fixed (known) system

with adversarial disturbances and fixed (known) quadratic cost functions is assumed, and a regret bound of $\mathcal{O}((\log(T))^3)$ is achieved.

The first study on regret analysis of adaptive control of LQG systems, the problem setting with unknown dynamics, is contained in [2]. In this work, a so-called "optimism in the face of uncertainty" approach is used to design an online algorithm that achieves a regret bound of $\mathcal{O}(\sqrt{T})$, with T being the time horizon. This work has initiated major developments in studying regret minimization for adaptive control problems, including the works [44, 32, 1, 55, 54, 26, 72, 46]. Following this work, the dependency on dimensionality was improved by [44]. A new algorithm using semi-definite programming relaxation, is designed by [26], and [54] employs an ϵ-greedy exploration approach to gain further computationally efficiency. In [66], it is shown that $\mathcal{O}(\sqrt{T})$ is a fundamental limit for the worst-case regret of adaptive LQG. Moreover, [66] proved that this regret bound is achievable and presented an algorithm where the regret bound dependence on the system dimension is optimal.

Here, we point out a wider set of literature related to our work. First, we note that one can think about the underlying control system as a dynamical constraint on the optimization problem. Considering control systems as constraints is also classical in the context of *model predictive control* [36]. Although we tackle dynamic constraints in this work, we should emphasize that online optimization problems with static constraints, known only in hindsight, also play a key role in various settings and have generated interest in recent years [73, 57, 45].

Our work is also related to the framework of Markov decision processes (MDPs), where the system transition to the next state is defined through a probability distribution. Moreover, a reward is given to the decision maker for each action at each

state. This framework is classical in *reinforcement learning*, where the objective is to learn the optimal policy which yields the maximum reward [68]. Finally, our setting is also related to online optimization in dynamic environments [39], where the decisions are constrained in dynamics chosen by the environment. However, the objective of [39] is to study the impact of model mismatch on the overall regret, whereas in this work the decisions are input to a control system, which impacts the way the decisions affect the future outcomes through its dynamics.

1.2 Contributions and Organizations of the Thesis

The rest of this thesis is organized as follows. Chapter 2 contains some mathematical preliminaries and background on online convex optimization, optimal control theory, and the discrete-time algebraic Riccati equation.

In Chapter 3, we consider the problem of online LQG control, where the control system is linear and known and the cost functions are quadratic and time-varying and only become available in hindsight. In contrast to [25], where an online algorithm using semi-definite programming update is designed to generate the control policies, we employ a control-theoretic approach and introduce an online version of a classical iterative Riccati update. Before introducing our algorithm, we present two iterative methods for solving the discrete-time algebraic Riccati equation; the Riccati difference equation and the Newton-Hewer dynamics. Then, we present monotonicity results on the Newton-Hewer dynamics. Using the Newton-Hewer dynamics, which is less known than the classical Riccati difference equation [18], is key in developing our algorithm. This algorithm, which employs only a few matrix addition and multiplication operations in each time step, has reduced complexity compared to the

one using semi-definite programming in each time step and is easier to implement. Our main result is a $\mathcal{O}(\log T)$ regret bound for the online LQG control problem, improving the $\mathcal{O}(\sqrt{T})$ bound of [25] and the $\mathcal{O}((\log(T))^7)$ bound of [5] for time horizon T, under some boundedness assumption. Indeed, the technical part of our result relies on characterizing the interplay between a notion of stability for the sequence of control policies and boundedness of the solutions of the proposed Riccati update. Our observation shows that the latter boundedness property, which follows for the Riccati difference equation from monotonicity with respect to the underlying parameters, cannot be obtained using monotonicity in our case; in fact, the Newton-Hewer updates can fail to be monotone in this setting. This observation is presented in Chapter 3.

For the scalar case, we are able to prove the mentioned boundedness property, yielding the stronger result that initializing the control policy to be stabilizing is enough to guarantee boundedness of the solutions of the proposed online Riccati update. We will demonstrate why the argument for the scalar case cannot be extended to the non-scalar case. We provide numerical results to illustrate this issue while presenting numerical results to show that the uniform boundedness assumption is not a strong assumption for our algorithm which achieves a sublinear regret in practical problems. In addition, we compare the regret of our algorithm with the regret of the Follow-the-Leader Algorithm through numerical examples. This results of Chapter 3 appeared in [8, 9, 7].

In Chapter 4, we introduce the problem of online adaptive LQG control, where the dynamics of the system is unknown. Before stating the contributions of our work in Chapter 4, we review some results on regret minimization for online adaptive

LQG control here. The analysis of regret minimization for various algorithms for the adaptive control of LQG systems brings about the question of finding a fundamental lower bound to the regret of adaptive LQG. Simchowitz and Foster in [66] proved that for any adaptive LQG problem with unknown system parameters A and B, and for any algorithm, there is a choice of the system parameters for which the algorithm must suffer a regret at least $\Omega(\sqrt{T})$. In the positive direction, [20] showed that (poly)-logarithmic regret bound is achievable if one makes the extra assumption that B is known or A is known (along with a minor additional assumption). These studies, however, leave open the question whether one can achieve a (poly)-logarithmic regret under milder assumptions. The main objective of this work is to investigate if (poly)-logarithmic regret can be achieved when both of these matrices are unknown, but some information about the system is given to the controller as a *hint*.

We consider the setting of the adaptive LQG control in a scenario where the true system parameters of the transition dynamics (matrices A and B) are *unknown*, but a hint about the matrix B is given to the controller periodically. This hint, which can be viewed as a noisy directional information pointing toward B, will help the controller to achieve logarithmic regret, even though it does not know the true system parameters A and B. This extra directional information is in the same spirit of the notion of hint by [30] for online optimization. Our algorithm, which uses a regularized least squared error estimate, is adopted from the work of [20]. However, the analysis of our algorithm with the hint is naturally more complicated. We also prove that the results of [20], where B is known, can be obtained from our setting.

Finally, Chapter 5 gives conclusions and ideas for future work.

Chapter 2

Background

In this section, we provide the mathematical preliminaries and background in online convex optimization and optimal control theory.

2.1 Mathematical Notation

We let $\mathbb{R}$ denote the set of real numbers. We use lowercase letters for vectors and uppercase letters for matrices. For a matrix A, we use $A^\top$ and $\mathrm{Tr}(A)$ to denote the transpose and trace of A, respectively. We denote by $\|\cdot\|$ the Euclidean norm on vectors and its corresponding operator norm on matrices. Thus $\|A\| = \sigma_{\max}(A) = \sqrt{\lambda_{\max}(A^\top A)}$, where $\sigma_{\max}(A)$ is the largest singular value of A and $\lambda_{\max}(A^\top A)$ is the largest eigenvalue of $A^\top A$. We also use $\|\cdot\|_F$ to denote the Frobenius norm on matrices. We let I_n denote the $n \times n$ identity matrix. The Gaussian distribution with mean vector $s \in \mathbb{R}^d$ and covariance matrix $R \in \mathbb{R}^{d\times d}$ is denoted by $\mathcal{N}(s, R)$. For a matrix A, $\rho(A)$ denotes its spectral radius. We use the notation $A \succeq 0$ ($A \succ 0$) to indicate that A is positive semi-definite (positive definite). We also use $A \succeq B$ ($A \succ B$) to indicate that $A - B$ is positive semi-definite (positive definite). The indicator function of event $\mathcal{E}$ is denoted by $\mathbf{1}\{\mathcal{E}\}$. We use $\mathrm{poly}(z)$ to denote a polynomial function of

variable z.

2.2 Online Optimization

We start by describing the problem of online convex optimization. Let $\{f_1, f_2, \ldots, f_T\}$ be a sequence of convex cost functions, where $f_t : \mathcal{U} \to \mathbb{R}$ for all $t \in \{1, 2, \ldots, T\}$, and $\mathcal{U} \subset \mathbb{R}^m$ is a convex set. At each time $t \in \{1, 2, \ldots T\}$, a decision maker chooses an action $u_t \in \mathcal{U}$. After that, the convex cost function f_t is revealed and the decision maker suffers the cost $f_t(u_t)$.

In this setting, since the cost functions are not known in advance, the decision does not necessarily correspond to the minimizers of the cost function and the decision maker is faced with a so-called regret. Regret is defined as the difference between the accumulated cost over time and the cost incurred by the best fixed decision, when all the functions are known in advance, see [75, 41]. Formally, the regret is

$$R(T) = \sum_{t=1}^{T} f_t(u_t) - \min_{u^\star \in \mathcal{U}} \sum_{t=1}^{T} f_t(u^\star).$$

The objective here is to design an algorithm for picking u_t so that it achieves a regret which is sublinear in T, i.e., $\lim_{T\to\infty} \frac{R(T)}{T} = 0$.

2.3 Online Optimization over a Control System

The problem of online optimization can be extended to the situation where the decisions are control actions or inputs in a dynamical system. This problem can be modeled as follows. Let $\{f_1, f_2, \ldots, f_T\}$ be a sequence of cost functions, where $f_t : \mathcal{X} \times \mathcal{U} \to \mathbb{R}^{>0}$, $\mathcal{X} \subset \mathbb{R}^n$ and $\mathcal{U} \subset \mathbb{R}^m$. At time t, the decision maker chooses u_t

after observing state x_t. After committing to this action, a convex cost function f_t on both x and u is revealed and the decision maker suffers the cost $f_t(x_t, u_t)$. The system transitions to the next state according to the dynamics $x_{t+1} = g_t(x_t, u_t)$, where $g_i : \mathcal{X} \times \mathcal{U} \to \mathcal{X}$. This problem can be formulated as an optimal control problem:

$$\begin{aligned} \underset{u_1,u_2,\ldots,u_T \in \mathcal{U}}{\text{minimize}} \quad & \mathcal{J}(T) = \sum_{t=1}^{T} f_t(x_t, u_t) \\ \text{s.t.} \quad & x_{t+1} = g_t(x_t, u_t), \qquad t = 1, 2, \ldots, T-1. \end{aligned}$$

In this scenario, the decisions not only incur a cost at that time, but also change the future states which have effect on the cost in the future. The decision maker uses the information of past states and actions to choose the new action. We define a policy $\pi_t : \mathcal{X} \to \mathcal{U}$ as a mapping of current state x_t to the action u_t. The decisions are made by choosing a policy among a class of policies. Similar to the online optimization problem, the cost functions are not known in advance and the decision maker actions do not correspond to the minimizer of $\mathcal{J}(T)$, so the decision maker faces a regret. Here, the regret is defined as the difference between the accumulated cost of chosen actions and the cost incurred by the best fixed policy, i.e.,

$$R_{\{\pi_t\}}(T) = \sum_{t=1}^{T} f_t(x_t, \pi_t(x_t)) - \min_{\pi \in \Pi} \sum_{t=1}^{T} f_t(x_t, \pi(x_t)),$$

where $\Pi = \{\pi | \pi = (\pi_1, \pi_2, \ldots, \pi_T),\ \pi_i = \pi_j \text{ for all } i, j = 1, \ldots, T\}$ is the set of admissible policies. The objective is to design algorithms to choose policies so that the regret be sublinear in T, i.e., the average regret over time converges to zero. We now proceed to analyze this problem on a special class of control systems.

2.4 Discrete-Time Linear Quadratic Gaussian Control

The discrete-time linear quadratic Gaussian (LQG) control problem is defined as follows, see for instance [67]: Let $x_t \in \mathbb{R}^n$ and $u_t \in \mathbb{R}^m$ be the control state and the control action at time t, respectively, with initial state x_1. The system dynamics are given by

$$x_{t+1} = Ax_t + Bu_t + w_t, \qquad t \geq 1 \tag{2.1}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, and $\{w_t\}_{t\geq 1}$ are i.i.d. Gaussian noise vectors with zero mean and covariance $W \in \mathbb{R}^{n\times n}$ ($w_t \sim \mathcal{N}(0, W)$). We either assume that the initial state value is Gaussian $x_1 \sim \mathcal{N}(m, X_1)$ and is independent of the noise sequence $\{w_t\}_{t\geq 1}$, or that it is $x_1 = 0$. The cost incurred in each time step t is a quadratic function of the state and control action given by $x_t^\top Q_t x_t + u_t^\top R_t u_t$, where $Q_t \in \mathbb{R}^{n\times n}$ and $R_t \in \mathbb{R}^{m\times m}$ are positive-definite matrices. The total cost after T time steps is given by

$$J_T(x_1, u_1, \ldots, u_T) = \mathbb{E}\Big[x_T^\top Q_T x_T + \sum_{t=1}^{T-1} \big(x_t^\top Q_t x_t + u_t^\top R_t u_t\big)\Big].$$

The optimal controller that minimizes the finite-horizon cost can be obtained by dynamic programming that will be described next.

For the infinite-horizon problem, we assume that the cost matrices Q_t and R_t are fixed and the average total cost is given by

$$J(\{u_t\}_{t\geq 1}) = \lim_{T\to\infty} \mathbb{E}\Big[\frac{1}{T}\sum_{t=1}^{T} \big(x_t^\top Q_t x_t + u_t^\top R_t u_t\big)\Big],$$

provided the limit exists. In this setting, it is well-known that under the assumption that the control system is stabilizable, and the cost matrices Q_t and R_t are positive-definite, the optimal policy is a stabilizing linear feedback of the state, which will be described next.

2.5 Discrete Algebraic Riccati Equation

In the classical LQG problem, where all the cost functions are known and for the finite-horizon problem, the optimal policy can be obtained by dynamic programming, and is a linear function of the state. In particular, $u_t = K_t x_t$, where K_t is given by the equation

$$K_t = -(B^\top P_{t+1} B + R_t)^{-1} B^\top P_{t+1} A,$$

and P_{t+1} is a sequence of positive-definite matrices obtained iteratively, backwards in time, from the dynamic Riccati equation:

$$P_t = A^\top P_{t+1} A - A^\top P_{t+1} B (B^\top P_{t+1} B + R_t)^{-1} B^\top P_{t+1} A + Q_t \tag{2.2}$$

with the terminal condition $P_T = Q_T$.

For the infinite-horizon problem with the assumption that $Q_t = Q$ and $R_t = R$ are fixed, and under the assumptions that

1. R is positive-definite,
2. (A, B) is stabilizable, i.e., there exists a linear policy $\pi(x) = Kx$ such that the closed-loop system $x_{t+1} = (A+BK)x_t$ is asymptotically stable: $\rho(A+BK) < 1$,
3. (A, C) is detectable where $Q = C^\top C$, [i.e., if $u_t \to 0$ and $Cx_t \to 0$ then, $x_t \to 0$],

it is well-known that the optimal policy is unique, time invariant, and is a linear function of the state [12], i.e., $u_t = K^\star x_t$. Here $K^\star$ is given by

$$K^\star = -(B^\top P^\star B + R)^{-1} B^\top P^\star A, \tag{2.3}$$

where $P^\star$ satisfies the discrete algebraic Riccati equation (DARE):

$$P^\star = A^\top P^\star A - A^\top P^\star B (B^\top P^\star B + R)^{-1} B^\top P^\star A + Q. \tag{2.4}$$

Moreover, P_t given by (2.2) converges to $P^\star$ as $t \to \infty$ [67]. By using the policy $K^\star$, we have that $x_{t+1} = (A + BK^\star)x_t + w_t$. The optimal policy $K^\star$ is guaranteed to be stabilizing, i.e., $\rho(A + BK^\star) < 1$. Here, the distribution of x_t converges to a stationary distribution, i.e., x_t converges weakly to a random variable x which has the same distribution as $(A + BK^\star)x + w_t$, so that we have $\mathbb{E}[x] = \mathbb{E}[(A + BK^\star)x + w_t]$, which implies $\mathbb{E}[x] = 0$, and the covariance matrix $X = \mathbb{E}[xx^\top]$ satisfies $X = (A + BK^\star)X(A + BK^\star)^\top + W$, see e.g., [25].

Chapter 3

Online Linear Quadratic Control

3.1 Problem Statement

We now define the problem we study in this chapter, following [25]. In *online linear quadratic control*, the sequence of cost matrices $\{Q_t\}_{t\geq 1}$ and $\{R_t\}_{t\geq 1}$ are not known in advance and Q_t and R_t are only revealed after choosing the control action u_t. Since it is not possible to find the optimal policy before observing the whole sequence of cost matrices $\{Q_t\}_{t\geq 1}$ and $\{R_t\}_{t\geq 1}$, the decision maker faces a *regret*. Here, we assume that the control system (A, B) is stabilizable, and the cost matrices Q_t and R_t are positive-definite and uniformly bounded over $t \geq 1$. As the optimal policy for the system with these assumptions is given by a stabilizing linear feedback, we use the set of stabilizing linear feedback functions as the set of admissible policies.

Let $x_t \in \mathbb{R}^n$ and $u_t \in \mathbb{R}^m$ be the control state and controller action at time $t \geq 1$. The controller uses a linear feedback policy $u_t = K_t x_t$ and commits to this action after observing x_t. Then the controller receives the positive-definite matrices $Q_t \in \mathbb{R}^{n\times n}$

and $R_t \in \mathbb{R}^{m\times m}$, and suffers the cost

$$J_t(K_t) = \mathbb{E}\Big[x_t^\top Q_t x_t + u_t^\top R_t u_t\Big]. \tag{3.1}$$

The objective is to design an algorithm to generate a sequence of policies $\{K_t\}_{t\geq 1}$ such that the regret function, which is defined as

$$\mathcal{R}(T) = \sum_{t=1}^{T} J_t(K_t) - \min_{K\in\mathcal{K}} \sum_{t=1}^{T} J_t(K), \tag{3.2}$$

where $\mathcal{K}$ is the set of stabilizing policies, grows sublinearly in T. In other words, the average regret over time converges to zero. We use the Newton-Hewer dynamics that is presented next.

3.2 Iterative Methods for Solving the Discrete Algebraic Riccati Equation

Several methods for solving DARE exist in the literature, including iterative methods [18], algebraic methods [62], and semi-definite programming [11]. Our work is based on iterative methods, and in particular, two techniques that we review here.

3.2.1 Riccati Difference Equation

The first technique in solving DARE, given in [18], is called the Riccati difference equation. This technique simply uses the recursion

$$P_{t+1} = A^\top P_t A - A^\top P_t B(B^\top P_t B + R)^{-1} B^\top P_t A + Q.$$

It is shown that under the assumption that (A, B) is stabilizable and (A, C) is detectable, where $Q = C^\top C$, the sequence $\{P_t\}$ converges to the unique solution of DARE.

3.2.2 Newton-Hewer Dynamics

A second approach, studied by [43], uses the following idea: Let P_t be the solution of the equation

$$P_t = (A + BK_t)^\top P_t (A + BK_t) + K_t^\top R K_t + Q, \tag{3.3}$$

where

$$K_t = -(B^\top P_{t-1} B + R)^{-1} B^\top P_{t-1} A,$$

starting from a stabilizing policy K_1. Then under the assumption that (A, B) is stabilizable and (A, C) is detectable, where $Q = C^\top C$, the sequence $\{P_t\}$ converges to the solution of DARE and the rate of convergence is quadratic, i.e.,

$$\|P_t - P^\star\| \leq C \|P_{t-1} - P^\star\|^2$$

where $C > 0$ is a constant.

3.2.3 Monotonicity Properties

This section investigates the monotonicity properties of iterative methods for solving the discrete-time algebraic Riccati equation (DARE). The monotonicity property of

the Riccati difference equation has been used to derive a robust stability condition for finite-horizon robust LQR problem [76]. In addition, a boundedness result for the solution of the Riccati difference equation has been derived using this property [35, 28]. These applications motivate us to ask the natural question whether the Newton-Hewer dynamics has the monotonicity property that Riccati difference equation enjoys. This property can be used in studying boundedness issues in Section 3.7.

3.2.4 Lack of Monotonicity for Newton-Hewer Dynamics

We recall the Riccati difference equation given by

$$P_{t+1} = A^\top P_t A - A^\top P_t B(B^\top P_t B + R)^{-1} B^\top P_t A + Q.$$

It has been shown that the right hand side of this dynamics is monotone as a function of P_t, in the sense that $P_t \succeq \widehat{P}_t \succeq 0$ implies $P_{t+1} \succeq \widehat{P}_{t+1} \succeq 0$, see [29, Lemma 3.1]. Furthermore, this dynamics is monotone as a function of (A, B, Q, R) in the following sense: If $P_t \succeq \widehat{P}_t \succeq 0$ and

$$\begin{pmatrix} Q & A^\top \\ A & -BR^{-1}B^\top \end{pmatrix} \succeq \begin{pmatrix} \widehat{Q} & \widehat{A}^\top \\ \widehat{A} & -\widehat{B}\widehat{R}^{-1}\widehat{B}^\top \end{pmatrix}, \tag{3.4}$$

then $P_{t+1} \succeq \widehat{P}_{t+1} \succeq 0$, see [35, 71]. Notably and important to the discussion we will have in the next section, as long as (3.4) is satisfied, *this monotonicity property holds even when the parameters A, B, Q, R are time-varying.*

We recall the Newton-Hewer method [43], given by

$$\begin{aligned} P_{t+1} &= A_t^\top P_{t+1} A_t + K_t^\top R K_t + Q, \\ A_t &= A + BK_t, \\ K_t &= -(B^\top P_t B + R)^{-1} B^\top P_t A. \end{aligned} \tag{3.5}$$

It has been shown that if the system (A, B) is controllable, by initializing with a stabilizing K_0; i.e., $\rho(A + BK_0) < 1$, where $\rho(\cdot)$ denotes the spectral radius, P_t converges monotonically, i.e., $P_1 \succeq P_2 \succeq \ldots \succeq P^*$, where P^* is the solution of (2.4).

We will show next that the Newton-Hewer dynamics does not have the monotonicity property, i.e., $P_t \succeq \widehat{P}_t \succeq 0$ does not necessarily imply $P_{t+1} \succeq \widehat{P}_{t+1} \succeq 0$, by providing two counterexamples, each aimed to demonstrate a facet of this lack of monotonicity.

3.2.5 Counterexamples

The construction of our examples is done for the scalar case, and for this reason, we write the Newton-Hewer dynamics in this scenario. We assume that Q are R are positive real numbers, and do not change with time. By setting $n = m = 1$, the dynamics (3.5) can be written as:

$$P_{t+1} = \frac{A^2 B^2 P_t^2 R + Q B^4 P_t^2 + 2Q B^2 P_t R + Q R^2}{(P_t B^2 + R + AR)(P_t B^2 + R - AR)}.$$

By taking derivative, it can be shown that P_{t+1} as a function of P_t is increasing for $P_t > P^*$ and decreasing for $P_t < P^*$, where P^* is the solution to (2.4). For a stable policy K_t, P_t will be larger than P^* and monotonicity holds [43]. We have depicted

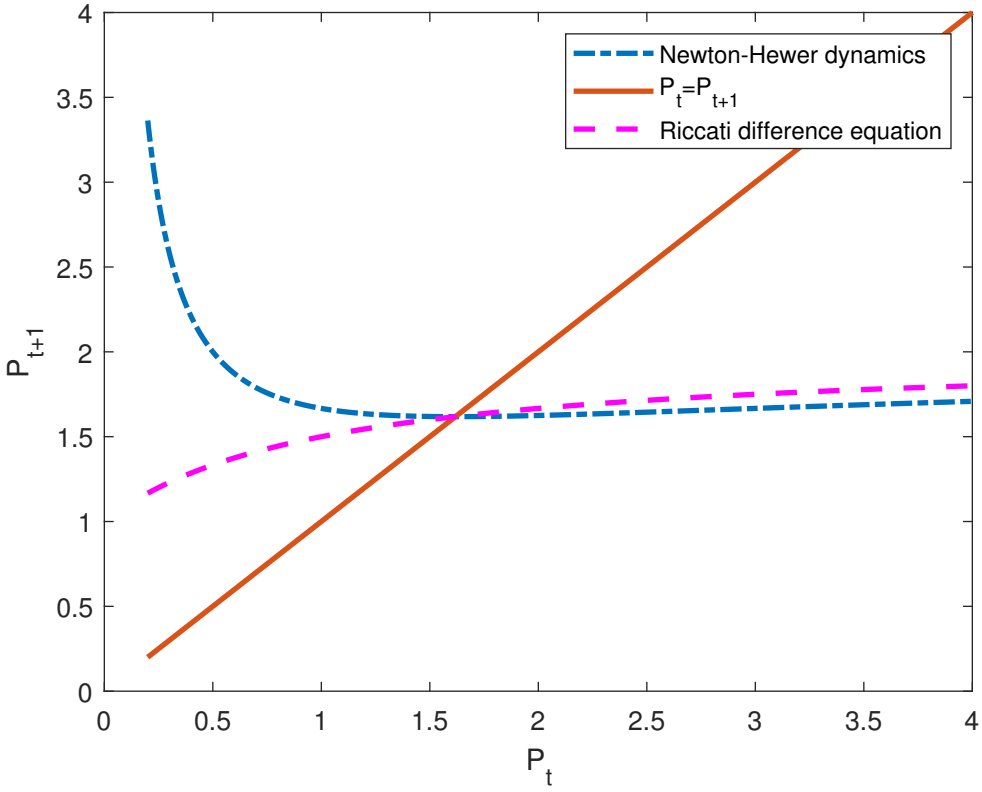

Figure 3.1: P_{t+1} as a function of P_t is shown for $(A, B, R, Q) = (1, 1, 1, 1)$ of Newton-Hewer dynamics and Riccati difference equation

P_{t+1} as a function of P_t in Fig. 3.1, and it can be observed that the Newton-Hewer dynamics is increasing for $P_t \geq P^*$, where P^* is at the intersection of the line $P_t = P_{t+1}$ and Newton-Hewer dynamics. We now show that if the system has time-varying Q and R, the stabilizability properties of the controller do not necessarily imply that the system is monotone, drawing a contrast with the Riccati difference equation.

To this end, note that this graph depends on A, B, R and Q, and if one of these parameters changes, P^* and the graph will change. To elaborate on this, we use Fig. 3.2 where we have depicted P_{t+1} as a function of P_t for two different Newton-Hewer dynamics with $(A, B, R, Q) = (1, 1, 1, 1)$ and $(A, B, R, Q) = (1, 1, 1, 2)$. In Fig. 3.2, the

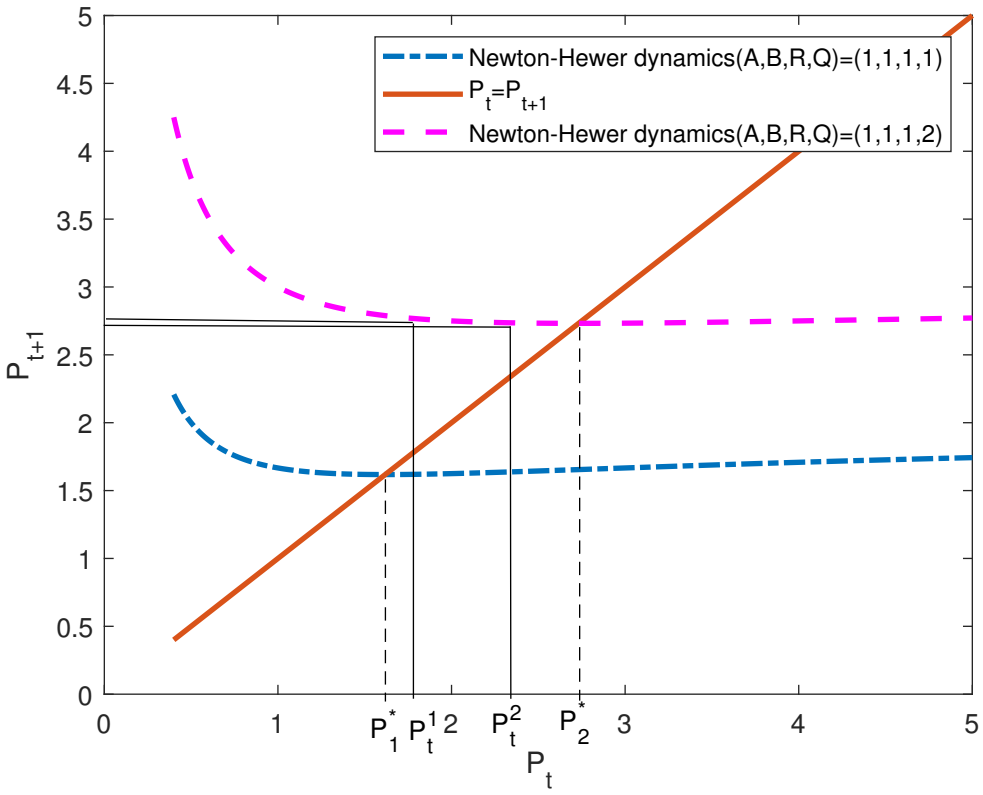

Figure 3.2: P_{t+1} as a function of P_t is shown for $(A, B, R, Q) = (1, 1, 1, 1)$ and $(A, B, R, Q) = (1, 1, 1, 2)$ of Newton-Hewer dynamics

P_1^* and P_2^* refer to the solution to the DARE (2.4) for the systems $(A, B, R, Q) = (1, 1, 1, 1)$ and $(A, B, R, Q) = (1, 1, 1, 2)$, respectively. If $Q_t = 1, Q_{t+1} = 2$ and K_t are such that $P_1^* < P_t < P_2^*$, then the system for the next time step uses the orange graph to update P_{t+1}, and the reader can observe – we prove this with carefully chosen numerical values below – that this can lead to failure of monotonicity, i.e., $P_t \leq \widehat{P}_t$ does not necessarily imply $P_{t+1} \leq \widehat{P}_{t+1}$. Note that the system will remain monotone if $Q_{t+1} < Q_t$ for all t, in case the other system parameters A, B, and R remain fixed. Using this observation, we now explicitly construct the counterexample.

Example 3.2.1. Consider the dynamics (3.5). Let the system be scalar, i.e., $n =$

$m = 1$, and let $A = 1$, $B = 1$, $R = 1$ be fixed and Q_t be time-varying. Let P_t be the sequence generated by (3.5) at each time step. Given that A, B, R are fixed, P_t is a function of $\{Q_1, Q_2, \cdots, Q_t\}$ and K_0, where K_0 is a stable policy at time 0. Let $\widehat{P}_t$ be the sequence generated by (3.5) with $A = 1, B = 1, R = 1$ and $\widehat{Q}_t$ and K_0. We claim that $P_t \geq \widehat{P}_t$ does not necessarily imply that $P_{t+1} \geq \widehat{P}_{t+1}$.

To prove this claim, we need to chose K_0 properly. Let

$$K_0 = 1 - \sqrt{3},$$

which is a stabilizing policy. Hence, by (3.5) we have

$$P_1 = \frac{K_0^2 R_1 + Q_1}{1 - (A + BK_0)^2} = \frac{4 - 2\sqrt{3} + Q_1}{4\sqrt{3} - 6}.$$

Given this

$$\begin{aligned}
K_1 =& \frac{-BP_1A}{B^2P_1 + R_1} = \frac{-4 + 2\sqrt{3} - Q_1}{2\sqrt{3} - 2 + Q_1}, \\
P_2 =& \frac{K_1^2 R_2 + Q_2}{1 - (A + BK_1)^2} \\
=& \frac{(8 - 4\sqrt{3})Q_1 + (16 - 8\sqrt{3})Q_2 + (Q_2 + 1)Q_1^2}{4(\sqrt{3} - 1)Q_1 + Q_1^2 - 68 + 40\sqrt{3}} \\
&+ \frac{4(\sqrt{3} - 1)Q_1Q_2 + 28 - 16\sqrt{3}}{4(\sqrt{3} - 1)Q_1 + Q_1^2 - 68 + 40\sqrt{3}}.
\end{aligned}$$

Now let $Q_1 = 1$ and $Q_2 = 2$ then $P_1 = 1.6547$, and $P_2 = 2.7835$. If we choose $\widehat{Q}_1 = 2$ and $\widehat{Q}_2 = 2$, then $\widehat{P}_1 = 2.7321$, and $\widehat{P}_2 = 2.7321$. This demonstrates that given $\widehat{P}_t \geq P_t$, it does not follow that $\widehat{P}_{t+1} \geq P_{t+1}$. Fig. 3.9 shows the sequence P_t

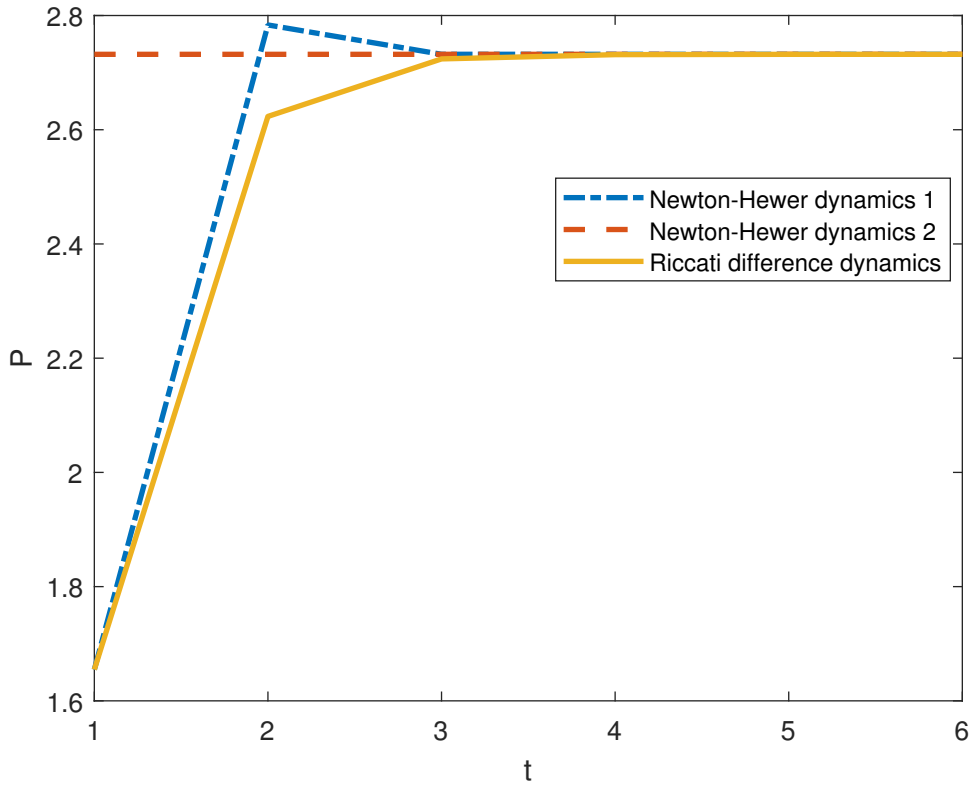

Figure 3.3: This graph shows the Newton-Hewer dynamics for a system starts with $Q_1 = 1$ and $Q_t = 2$ for $t > 1$ (dotted line), Newton-Hewer dynamics for a system with $Q_t = 2$ for all t (dashed line) and Riccati difference dynamics with $Q_1 = 1$ and $Q_t = 2$ for $t > 1$ (solid line).

(dotted line) and $\widehat{P}_t$ (dashed line) where $Q_1 = 1$ and $Q_t = 2$ for $t \geq 2$ and $\widehat{Q}_t = 2$. Furthermore, the sequence $\tilde{P}_t$ which is generated by the Riccati difference equation with initialization $\tilde{P}_1 = P_1$ and the same parameters $\tilde{A} = A, \tilde{B} = B, \tilde{Q}_t = Q_t, \tilde{R} = R$ is shown (solid line) for six time steps. •

We conclude with providing an example which demonstrates another aspect of lack of monotonicity of Newton-Hewer dynamics.

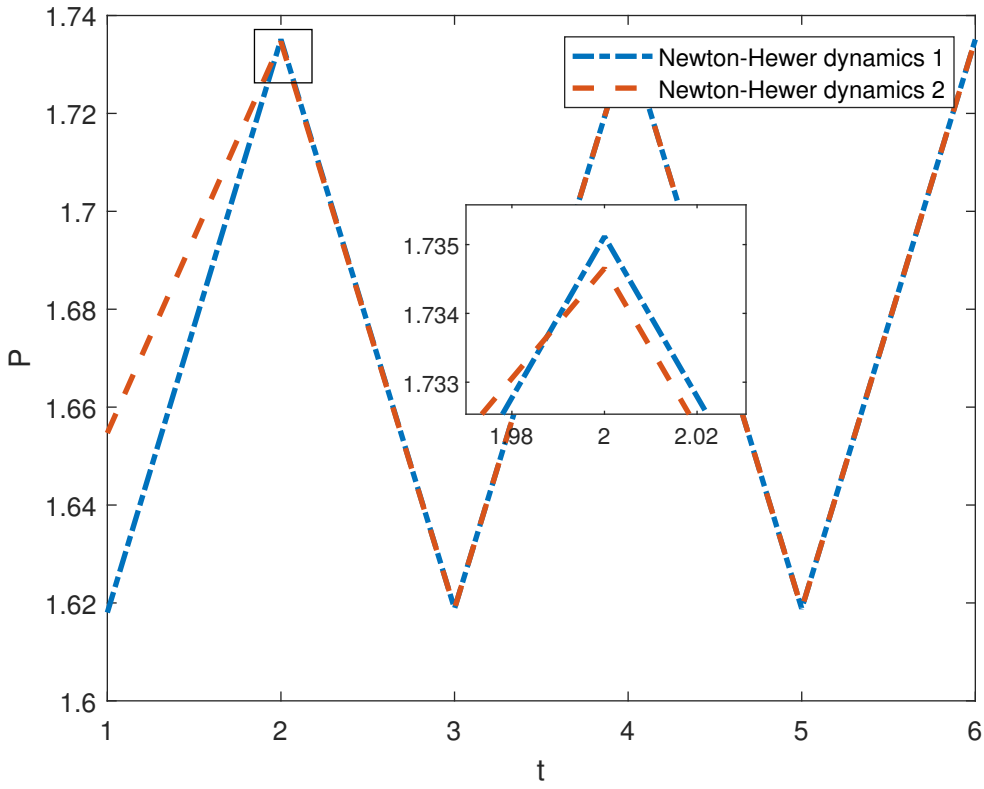

Figure 3.4: Newton-Hewer dynamics for two systems with the same (A, B, R, Q_t) and different initial condition K_0.

Example 3.2.2. We consider two dynamics with the same Q and R, albeit time-varying, but with different initial conditions K_0. Similar to the previous example, we assume $n = m = 1$, and $A = 1$, $B = 1$, $R = 1$ are fixed and Q_t is time-varying. We assume Q_t is 1 for odd time steps and 1.1 for even time steps. If we choose $K_0 = -0.7321$ for the first system and $\widehat{K}_0 = -0.6180$ for the second system, we will have $P_1 = 1.6180$ and $\widehat{P}_1 = 1.6547$, and for the next time, we have $P_2 = 1.7351$ and $\widehat{P}_2 = 1.7347$, which shows that the monotonicity does not hold. Fig. 3.4 illustrates the behaviour of two dynamics at the next time steps. •

3.3 Strong Stability

A key property that we require before introducing our algorithm is the notion of strong stability and sequential strong stability which are similar to the ones in [25]. The notion of strong stability is defined as follows.

Definition 3.3.1. *A policy K is called stabilizing if $\rho(A + BK) < 1$. A policy K is (κ, γ)-strongly stabilizing (for $\kappa > 0$ and $0 < \gamma \leq 1$) if $\|K\| \leq \kappa$, and there exist matrices L and H such that $A+BK = HLH^{-1}$, with $\|L\| \leq 1-\gamma$ and $\|H\|\|H^{-1}\| \leq \kappa$.*

Note that every (κ, γ)-strongly stabilizing policy K is stabilizing, since the matrices $A + BK$ and L are similar and hence $\rho(A + BK) = \rho(L) \leq (1 - \gamma)$. Lemma 3.3.2 shows that every stabilizing policy is (κ, γ)-strongly stabilizing for some $\kappa > 0$ and $0 < \gamma \leq 1$.

Lemma 3.3.2. *[25, Lemma B.1.] Suppose that for a linear system defined by A, B, a policy K is stabilizing. Then there are parameters $\kappa > 0$, $0 < \gamma \leq 1$ for which it is (κ, γ)-strongly stabilizing.*

We refer the reader to [25, Lemma B.1] for a proof of this lemma.

Under the assumption of (κ, γ)-strong stability of policy K, the state covariance matrices $X_t = \mathbb{E}[x_t x_t^\top]$ converge exponentially to a steady-state covariance matrix $\widehat{X}$, which satisfies

$$\widehat{X} = (A + BK)\widehat{X}(A + BK)^\top + W.$$

Lemma 3.3.3 provides the details.

Lemma 3.3.3. *[25, Lemma 3.2] Let the pair (A, B) be stabilizable, and assume the controller uses a fixed (κ, γ)-strongly stabilizing policy K, i.e., for $t \geq 1$, we have*

$u_t = Kx_t$. Let X_t be the covariance matrix of x_t. Then the sequence $\{X_t\}_{t\geq 1}$ converges to the steady-state covariance matrix $\widehat{X}$, and in particular, for any $t \geq 1$,

$$\|X_{t+1} - \widehat{X}\| \leq \kappa^2 e^{-2\gamma t}\|X_1 - \widehat{X}\|.$$

We refer the reader to [25, Lemma 3.2] for a proof.

In order to obtain a similar result for the change of the state covariance matrices using a sequence of different (κ, γ)-strongly stabilizing policies $\{K_t\}_{t\geq 1}$, we need to define a notion of sequential strong stability, which is presented next.

Definition 3.3.4. *A sequence of policies $\{K_t\}_{t\geq 1}$ is sequentially (κ, γ)-strongly stabilizing, for $\kappa > 0$ and $0 < \gamma \leq 1$, if there exist sequences of matrices $\{H_t\}_{t\geq 1}$ and $\{L_t\}_{t\geq 1}$ such that*

$$A + BK_t = H_t L_t H_t^{-1}$$

for all $t \geq 1$, with the following properties:

- $\|L_t\| \leq 1 - \gamma$ *and* $\|K_t\| \leq \kappa$;
- $\|H_t\| \leq \beta$ *and* $\|H_t^{-1}\| \leq 1/\alpha$ *with* $\kappa = \beta/\alpha$ *and* $\alpha > 0$ *and* $\beta > 0$;
- $\|H_{t+1}^{-1} H_t\| \leq 1 + \gamma$.

The importance of this notion of stability is demonstrated in Lemma 3.3.5.

Lemma 3.3.5. *Let the pair (A, B) be stabilizable, and suppose that the controller uses $u_t = K_t x_t$ for $t \geq 1$ and where $\{K_t\}_{t\geq 1}$ is sequentially (κ, γ)-strongly stabilizing with $\kappa > 0$ and $0 < \gamma \leq 1$. For each K_t, let $\widehat{X}_t$ be the corresponding steady-state covariance matrix, i.e., $\widehat{X}_t$ satisfies $\widehat{X}_t = (A + BK_t)\widehat{X}_t(A + BK_t)^\top + W$ and assume*

that $\|\widehat{X}_{t+1} - \widehat{X}_t\| \leq \eta_t$ *with* $\eta_t > 0$*, for all* $t \geq 1$*. Let* X_t *be the corresponding state covariance matrix at time* t*, starting from some initial* $X_1 \succeq 0$*. Then for* $t \geq 1$*,*

$$\|X_{t+1} - \widehat{X}_{t+1}\| \leq \kappa^2 e^{-2\gamma^2 t}\|X_1 - \widehat{X}_1\| + \kappa^2 \sum_{s=0}^{t-1} e^{-2\gamma^2 s}\eta_{t-s}.$$

The proof is similar to [25, Lemma 3.5], but we include it for completeness.

Proof. By definition, for all $t \geq 1$, we have that

$$\begin{aligned} X_{t+1} =& (A + BK_t)X_t(A + BK_t)^\top + W, \\ \widehat{X}_t =& (A + BK_t)\widehat{X}_t(A + BK_t)^\top + W. \end{aligned}$$

Subtracting the equations, substituting $A + BK_t = H_t L_t H_t^{-1}$ and rearranging yields

$$H_t^{-1}(X_{t+1} - \widehat{X}_t)(H_t^{-1})^\top = L_t H_t^{-1}(X_t - \widehat{X}_t)(H_t^{-1})^\top L_t^\top.$$

Let $\Delta_t = H_t^{-1}(X_t - \widehat{X}_t)(H_t^{-1})^\top$ for all $t \geq 1$. Then the above can be written as

$$\begin{aligned} \Delta_{t+1} =& (H_{t+1}^{-1} H_t L_t)\Delta_t(H_{t+1}^{-1} H_t L_t)^\top \\ &+ (H_{t+1}^{-1})(\widehat{X}_t - \widehat{X}_{t+1})(H_{t+1}^{-1})^\top. \end{aligned}$$

Taking the norms yeilds

$$\begin{aligned} \|\Delta_{t+1}\| \leq& \|L_t\|^2\|H_{t+1}^{-1}H_t\|^2\|\Delta_t\| + \|H_{t+1}^{-1}\|^2\|\widehat{X}_t - \widehat{X}_{t+1}\| \\ \leq& (1-\gamma)^2(1+\gamma)^2\|\Delta_t\| + \frac{\eta_t}{\alpha^2} \\ \leq& (1-\gamma^2)^2\|\Delta_t\| + \frac{\eta_t}{\alpha^2}, \end{aligned}$$

and by unfolding the recursion, we obtain

$$\begin{aligned}\|\Delta_{t+1}\| \leq & (1-\gamma^2)^{2t}\|\Delta_1\| + \frac{1}{\alpha^2}\sum_{s=0}^{t-1}(1-\gamma^2)^{2s}\eta_{t-s} \\ \leq & e^{-2\gamma^2 t}\|\Delta_1\| + \frac{1}{\alpha^2}\sum_{s=0}^{t-1} e^{-2\gamma^2 s}\eta_{t-s}.\end{aligned}$$

Using $X_t - \widehat{X}_t = H_t \Delta_t H_t^\top$ now, we have that

$$\begin{aligned}\|X_{t+1} - \widehat{X}_{t+1}\| \leq & e^{-2\gamma^2 t}\|\Delta_1\|\|H_{t+1}\|^2 + \frac{\|H_{t+1}\|^2}{\alpha^2}\sum_{s=0}^{t-1} e^{-2\gamma^2 s}\eta_{t-s} \\ \leq & \kappa^2 e^{-2\gamma^2 t}\|X_1 - \widehat{X}_1\| + \kappa^2 \sum_{s=0}^{t-1} e^{-2\gamma^2 s}\eta_{t-s},\end{aligned}$$

which concludes the proof.

□

We now proceed with some key results that we later use to ensure strong stability for the sequence of policies generated. Suppose that a sequence of positive-definite matrices P_t is generated recursively as

$$P_t = (A + BK_t)^\top P_t (A + BK_t) + \bar{Q}_t + K_t^\top \bar{R}_t K_t, \tag{3.6}$$

where

$$K_{t+1} = -(B^\top P_t B + \bar{R}_t)^{-1} B^\top P_t A \tag{3.7}$$

and where $\bar{R}_t \in \mathbb{R}^{m\times m}$ and $\bar{Q}_t \in \mathbb{R}^{n\times n}$ are given positive-definite matrices for all $t \geq 1$, and K_1 is an initial stabilizing policy. The reason for this update will become clear as part of our algorithm in Section 3.4. The key point we wish to make here is that

under the assumption of uniform boundedness of the matrix sequence $\{P_t\}_{t\geq 1}$, and the stability of matrix K_t, for all $t \geq 1$, the sequence $\{K_t\}_{t\geq 1}$ is uniformly (κ, γ)-strongly stabilizing, with appropriate choices of κ and γ.

Proposition 3.3.6. *Assume that for $t \geq 1$, $Q_t, R_t \succeq \mu I$ and $P_t \preceq \nu I$, where $\mu, \nu > 0$ and $\{P_t\}_{t\geq 1}$ is the sequence of matrices obtained as the solution of (3.6), and assume that the policy K_t given by (3.7) is stabilizing for all $t \geq 1$. Define $\bar{\kappa} = \sqrt{\frac{\nu}{\mu}}$. Then the sequence $\{K_t\}_{t\geq 1}$ is uniformly $(\bar{\kappa}, 1/2\bar{\kappa}^2)$-strongly stabilizing.*

Proof. By the assumption of stability and since $Q_t \succeq \mu I$, we have that

$$\begin{aligned} P_t =& (A+BK)^\top P_t (A+BK) + \bar{Q}_t + K^\top \bar{R}_t K \\ \succeq& (A+BK)^\top P_t (A+BK) + \mu I, \end{aligned} \tag{3.8}$$

where we have used the positive-definiteness of $K^\top \bar{R}_t K$. In particular, this means that $P_t \succeq \mu I$ for all t. On the other hand, assuming $P_t \preceq \nu I$, we have

$$\mu I \preceq P_t \preceq \nu I. \tag{3.9}$$

Given that P_t is positive-definite and nonsingular, we can define $L_t = P_t^{1/2}(A + BK)P_t^{-1/2}$. Multiplying (3.8) by $P_t^{-1/2}$ from both sides, we obtain $I \succeq L_t^\top L_t + \mu P_t^{-1} \succeq L_t^\top L_t + \bar{\kappa}^{-2} I$. Thus $L_t^\top L_t \preceq (1 - \bar{\kappa}^{-2})I$, so $\|L_t\| \leq \sqrt{1-\bar{\kappa}^{-2}} \leq 1 - \bar{\kappa}^{-2}/2$. Also, using (3.9) we have that

$$\|P_t^{1/2}\| \|P_t^{-1/2}\| \leq \bar{\kappa},$$

which finishes the proof. □

We now present a second useful result, where we show that under the additional

property that the rate of changes of sequence P_t is small (which we will be able to establish for our proposed algorithm, see Lemma 3.5.3), one can obtain that the sequence $\{K_t\}_{t\geq 1}$ is sequentially strongly stabilizing.

Proposition 3.3.7. *Assume that for $t \geq 1$, $Q_t, R_t \succeq \mu I$ and $P_t \preceq \nu I$, where $\mu, \nu > 0$ and $\{P_t\}_{t\geq 1}$ is the sequence of matrices obtained as the solution of (3.6), and assume that the policy K_t given by (3.7) is stabilizing for $t \geq 1$. Let $\bar{\kappa} = \sqrt{\frac{\nu}{\mu}}$, and suppose that $\|P_{t+1} - P_t\| \leq \eta$ for $t \geq 1$ for some $\eta \leq \mu/\bar{\kappa}^2$. Then the sequence $\{K_t\}_{t\geq 1}$ is sequentially $(\bar{\kappa}, 1/2\bar{\kappa}^2)$-strongly stabilizing.*

Proof. Proceeding as in the proof of Proposition 3.3.6, one can show that the matrix $L_t = P_t^{1/2}(A+BK_t)P_t^{-1/2}$ satisfies $\|L_t\| \leq 1-1/2\bar{\kappa}^2$ with $\|P_t^{1/2}\| \leq \sqrt{\nu}$ and $\|P_t^{-1/2}\| \leq 1/\sqrt{\mu}$. To establish the sequential strong stability stated by Definition 3.3.4 it thus suffices to show that $\|P_{t+1}^{-1/2}P_t^{1/2}\| \leq 1 + 1/2\bar{\kappa}^2$ for $t \geq 1$. To this end, observe that $\|P_{t+1} - P_t\| \leq \eta$, and that

$$\begin{aligned}
\|P_{t+1}^{-1/2}P_t^{1/2}\|^2 &= \|P_{t+1}^{-1/2}P_tP_{t+1}^{-1/2}\| \\
&\leq \|P_{t+1}^{-1/2}P_{t+1}P_{t+1}^{-1/2}\| + \|P_{t+1}^{-1/2}(P_{t+1} - P_t)P_{t+1}^{-1/2}\| \\
&\leq 1 + \|P_{t+1}^{-1/2}\|^2\|P_{t+1} - P_t\| \\
&\leq 1 + \frac{\eta}{\mu},
\end{aligned}$$

where the second inequality follow by the sub-multiplicative of matrix operator norm. Hence, since $\eta \leq \mu/\bar{\kappa}^2$, then $\|P_{t+1}^{-1/2}P_t^{1/2}\| \leq \sqrt{1 + 1/\bar{\kappa}^2} \leq 1 + 1/2\bar{\kappa}^2$ as required. □

The above results rely on uniform boundedness of the sequence $\{P_t\}_{t\geq 1}$, which we assume throughout this chapter. However, we can show that stability of K_1 is enough to guarantee this property in the scalar case, see Proposition 3.7.1. Based on our

extensive simulation studies, one of which is shown in Example 3.7.3, we believe that this property should hold only by assuming stability of K_1 for the general case. One of the main reason for the difficulty of establishing this result is the lack of monotonicity of the evolutions of the Newton-Hewer dynamics with respect to the underlying system parameter, a sharp contrast with the Riccati difference updates [18], which we have presented in Section 3.2.3. In this sense, the proof of Proposition 3.7.1 for the scalar case establishes the boundedness property of the sequence $\{P_t\}_{t\geq 1}$ without relying on monotonicity. We have outlined further details in Remark 3.7.2.

3.4 The Online Riccati Algorithm

We outline our main algorithm in this section. Our assumptions are as follows:

Assumption 3.4.1. *Throughout we assume that*

- *The pair (A, B) is stabilizable.*
- *The cost matrices Q_t and R_t are positive-definite and $\mu I \preceq Q_t$, $\mu I \preceq R_t$, and $\mathrm{Tr}(Q_t) \leq \sigma$, $\mathrm{Tr}(R_t) \leq \sigma$, for some $\sigma > \mu > 0$ for all $t \geq 1$.*
- *For the noise covariance matrix W we have that $\omega = \mathrm{Tr}(W) < \infty$.*

A formal description is given in Algorithm 1. We provide an informal description. We start from a stabilizing policy K_1; the existence of K_1 is provided by the assumption of stabilizability of the control system. At each time step $t \geq 1$, the controller uses the policy $u_t = K_t x_t$ after observing x_t, then the cost matrices Q_t and R_t are revealed, and the controller updates P_t and K_t using the average of the history of Q_ts and R_ts through (3.3). There is a technical step in our algorithm, which we call the

Algorithm 1: Online Riccati Update

input : The system matrices A and B, initial state x_1, time horizon T, parameters $\nu, \mu, \kappa = \sqrt{\nu/\mu}, \gamma = 1/(2\kappa^2), \sigma$

output: A sequence of stabilizing policies $\{K_t\}_{t=1}^{T}$

Initialize K_1 to be stabilizing;

for *each* $t = 1, 2, \cdots, T$ **do**

  receive x_t;

  use controller $u_t = K_t x_t$ and receive Q_t and R_t;

  update $\bar{R}_t = \frac{t-1}{t}\bar{R}_{t-1} + \frac{1}{t}R_t$, $\bar{Q}_t = \frac{t-1}{t}\bar{Q}_{t-1} + \frac{1}{t}Q_t$;

  update P_t as the solution of

$$P_t = (A + BK_t)^\top P_t (A + BK_t) + \bar{Q}_t + K_t^\top \bar{R}_t K_t$$

  ;

  Reset: if $t = t^\star := \left\lceil \frac{4\kappa^3\|B\|}{\gamma\mu}(2\sigma\kappa + \frac{2\kappa^3\|B\|\sigma(1+\kappa^2)}{\gamma}) + 1 \right\rceil$ **then**

    Initialize $\ell = 0$, $\widehat{P}_0 = P_{t^\star}$, and $\widehat{K}_0 = K_{t^\star}$;

    while $\|\widehat{P}_\ell - \widehat{P}_{\ell-1}\| > (\frac{2\sigma}{\|B\|} + \frac{4\kappa^2\sigma(1+\kappa^2)}{\gamma})/t^\star$ **do**

      $\ell \leftarrow \ell + 1$;

      $\widehat{K}_\ell = -(B^\top \widehat{P}_{\ell-1} B + \bar{R}_{t^\star})^{-1} B^\top \widehat{P}_{\ell-1} A$;

      $\widehat{P}_\ell$ satisfies $\widehat{P}_\ell = (A + B\widehat{K}_\ell)^\top \widehat{P}_\ell (A + B\widehat{K}_\ell) + \bar{Q}_{t^\star} + \widehat{K}_\ell^\top \bar{R}_{t^\star} \widehat{K}_\ell$;

    Return: $P_{t^\star} = \widehat{P}_\ell$;

  Return: $K_{t+1} = -(B^\top P_t B + \bar{R}_t)^{-1} B^\top P_t A$;

"reset" step and describe in detail later in the proof; this step allows us to show that using these updates the change of the norm of the policies is $\mathcal{O}(1/t)$, and this gives a regret bound $\mathcal{O}(\log(T))$. Before we state the algorithm, we need to elaborate on the parameters used.

Remark 3.4.2 (Parameters used in Algorithm 1)**.** *Our algorithm naturally uses parameters μ and σ, stated in Assumption 3.4.1. For the reset step, we also need (an estimate on) the strong stability parameters κ and γ, which are defined in Algorithm 1. Proposition 3.3.6 plays a key role in that regard, as it states that as long as we can estimate a uniform bound on the sequence P_t, we can obtain these parameters. In the*

scalar case, we know this uniform bound by Proposition 3.7.1; in other cases, given that the parameters are not needed in the early steps of the algorithm, one can envision that we can run our algorithm with a large estimate on this bound and adjust it if necessary.

3.5 Main Result

We are now in a position to state our main contribution, providing a logarithmic bound for the regret (3.2).

Theorem 3.5.1. *Suppose that the tuple* $(A, B, \{Q_t\}_{t=1}^T, \{R_t\}_{t=1}^T, W)$ *satisfies Assumption 3.4.1. Suppose that the matrices* P_t *generated by Algorithm 1 are uniformly bounded, i.e.,* $P_t \leq \nu I$. *Let* $\kappa = \sqrt{\frac{\nu}{\mu}}$ *and* $\gamma = 1/2\kappa^2$. *Then for* $T \geq \frac{4\kappa^3\|B\|}{\gamma\mu}(2\sigma\kappa + \frac{2\kappa^3\|B\|\sigma(1+\kappa^2)}{\gamma}) + 1$, *we have that*

$$\begin{aligned}\mathcal{R}(T) \leq & \Big(2\kappa^4\sigma\frac{M}{1-e^{-2\gamma^2}} + \frac{\kappa^4\omega}{\gamma\mu^3}(\|B\|\hat{m} + 2\sigma)^2\Big)\log(T) \\ & - 2\kappa^4\sigma\frac{M}{1-e^{-2\gamma^2}}\log(t^\star) + t^\star\sigma(1+\kappa^2)\max_{0<t\leq t^\star}\|(X_t - \widehat{X}_t)\| \\ & + 2\kappa^4\sigma\Big(\|X_{t^\star} - \widehat{X}_{t^\star}\|\frac{e^{-2\gamma^2 t^\star}}{1-e^{-2\gamma^2}} + \frac{M'\pi^2}{6(1-e^{-2\gamma^2})}\Big) + \omega l\hat{m} \\ & + \frac{\kappa^4\omega}{\gamma\mu^3}(\|B\|\hat{m} + 2\sigma)^2 + \frac{\sigma(1+\kappa^2)\kappa^2}{1-e^{-2\gamma}}\|\widehat{X}^\star - X_1^\star\|,\end{aligned}$$

where $t^\star = \frac{4\kappa^3\|B\|}{\gamma\mu}(2\sigma\kappa + \frac{2\kappa^3\|B\|\sigma(1+\kappa^2)}{\gamma}) + 1$,

$$M = \frac{2\kappa^6\omega}{\mu\gamma^2}\|B\|\big(\|B\|\hat{m} + 2\sigma\big),\ M' = \frac{\kappa^6\omega}{\mu\gamma^2}\|B\|^2(\|B\|\hat{m} + 2\sigma)^2,$$

and $\hat{m}$ *and* l *are constants defined in Lemmas 3.5.3 and 3.5.6, respectively.*

The rest of this section is devoted to proving Theorem 3.5.1. The proof is quite involved, and for this reason we find it useful to provide a brief description to help the reader navigate through it. Our first technical result Lemma 3.5.2 shows that Algorithm 1, as long as it is initialized at a stabilizing policy, iteratively produces stabilizing polices. This step is analogous to the classical result of [43] for the case where the cost objective matrices Q_t and R_t are fixed. Recall that, by Proposition 3.3.6, stability of policies K_t is required to establish strong stability. A technical part of this proof demonstrates the reason why we need the reset step of the algorithm to ensure that the sequence of policies $\{P_{t+1} - P_t\}$ decay as m/t, for some $m > 0$. Using this and by rewriting the regret using trace products, we establish a set of bounds in Lemmas 3.5.5, 3.5.6, and 3.5.7 which eventually yield the result.

Proof of Theorem 3.5.1:

We first provide a straightforward reformulation of the regret function. For matrices A and B of appropriate size, let $A \bullet B = \mathrm{Tr}(A^\top B)$. Then

$$\begin{aligned}\mathcal{R}(T) &= \sum_{t=1}^{T} \mathbb{E}\Big[x_t^\top Q_t x_t + u_t^\top R_t u_t\Big] - \sum_{t=1}^{T} \mathbb{E}\Big[{x_t^\dagger}^\top Q_t x_t^\dagger + {x_t^\dagger}^\top {K^\dagger}^\top R_t K^\dagger x_t^\dagger\Big] \\ &= \sum_{t=1}^{T} (Q_t + K_t^\top R_t K_t) \bullet X_t - \sum_{t=1}^{T} (Q_t + {K^\dagger}^\top R_t K^\dagger) \bullet X_t^\dagger \end{aligned} \tag{3.10}$$

As a result, we have that

$$\mathcal{R}(T) = \sum_{t=1}^{T}(Q_t + K_t^\top R_t K_t) \bullet (X_t - \widehat{X}_t) \tag{3.11}$$

$$+ \sum_{t=1}^{T}(Q_t + K_t^\top R_t K_t) \bullet \widehat{X}_t - \sum_{t=1}^{T}(Q_t + K^{\star\top} R_t K^\star) \bullet \widehat{X}^\star \tag{3.12}$$

$$+ \sum_{t=1}^{T}(Q_t + K^{\star\top} R_t K^\star) \bullet \widehat{X}^\star - \sum_{t=1}^{T}(Q_t + K^{\dagger\top} R_t K^\dagger) \bullet \widehat{X}^\dagger \tag{3.13}$$

$$+ \sum_{t=1}^{T}(Q_t + K^{\dagger\top} R_t K^\dagger) \bullet (\widehat{X}^\dagger - X_t^\dagger), \tag{3.14}$$

where $K^\dagger$ is the fixed optimal policy for the system $(A, B, \{Q_t\}_{t=1}^T, \{R_t\}_{t=1}^T, W)$, $X_t = \mathbb{E}[x_t x_t^\top]$ is the covariance matrix of x_t when the system follows policies K_t generated by Algorithm 1, $\widehat{X}_t$ is the steady-state covariance matrix using the policy K_t, i.e., $\widehat{X}_t$ satisfies

$$\widehat{X}_t = (A + BK_t)\widehat{X}_t(A + BK_t)^\top + W,$$

and

$$X_t^\dagger = \mathbb{E}[x_t^\dagger x_t^{\dagger\top}]$$

is the covariance matrix of the state $x_t^\dagger$ at time t when the system uses policy $K^\dagger$ at each time t; similarly, $\widehat{X}^\dagger$ is the steady-state covariance matrix using the policy $K^\dagger$, i.e., $\widehat{X}^\dagger$ satisfies

$$\widehat{X}^\dagger = (A + BK^\dagger)\widehat{X}^\dagger(A + BK^\dagger)^\top + W. \tag{3.15}$$

$K^\star$ is the solution to DARE and $\widehat{X}^\star$ is the steady-state covariance matrix using policy $K^\star$. From now on, we use the notation $A_t = A + BK_t$ to simplify the presentation.

Note that by the following computation we show that (3.13) is negative. Since $\widehat{X}^\star$

and $\widehat{X}^\dagger$ are fixed, we have that

$$\begin{aligned}
\sum_{t=1}^{T}(Q_t + K^{\star\top} R_t K^\star) \bullet \widehat{X}^\star - \sum_{t=1}^{T}(Q_t + K^{\dagger\top} R_t K^\dagger) \bullet \widehat{X}^\dagger \\
&= T(\bar{Q}_T + K^{\star\top} \bar{R}_T K^\star) \bullet \widehat{X}^\star - T(\bar{Q}_T + K^{\dagger\top} \bar{R}_T K^\dagger) \bullet \widehat{X}^\dagger \\
&= T(P^\star - A^{\star\top} P^\star A^\star) \bullet \widehat{X}^\star - T(P^\dagger - A^{\dagger^\top} P^\dagger A^\dagger) \bullet \widehat{X}^\dagger \\
&= T(P^\star \bullet \widehat{X}^\star - P^\star \bullet A^\star \widehat{X}^\star A^{\star\top}) - T(P^\dagger \bullet \widehat{X}^\dagger - P^\dagger \bullet A^\dagger \widehat{X}^\dagger A^{\dagger^\top}) \\
&= T(P^\star \bullet \widehat{X}^\star - P^\star \bullet (\widehat{X}^\star - W)) - T(P^\dagger \bullet \widehat{X}^\dagger - P^\dagger \bullet (\widehat{X}^\dagger - W) \\
&= T(P^\star - P^\dagger) \bullet W \leq 0,
\end{aligned}$$

where $P^\star$ and $P^\dagger$ satisfies $P = (A + BK)^\top P (A + BK) + \bar{Q}_T + K^\top \bar{R}_T K$ for $K = K^\star$ and $K = K^\dagger$, respectively, and we have used this fact in the second equality, the cyclic property of the trace in the third equality, and (3.15) in the forth equality. By [43, Theorem 1], $P^\star \preceq P^\dagger$ and we have the result.

We start with our first technical result, which shows that Algorithm 1 produces stabilizing polices. This step is similar to the classical result of [43] for the case where the cost objective matrices Q_t and R_t are fixed. Recall that stability of policies K_t is required to establish strong stability, see Proposition 3.3.6.

Lemma 3.5.2. *Suppose that the pair (A, B) is stabilizable and let the sequence $\{K_t\}_{t\geq 1}$ be generated by Algorithm 1, starting from a stabilizing policy K_1. Then policy K_t remains stabilizing for all $t \geq 1$.*

Proof. We proceed by an induction argument. First, since the system is stabilizable, there exists a stabilizing policy and hence we can choose K_1 to be stabilizing, i.e., such that $\rho(A + BK_1) < 1$. Assume now that K_t is stabilizing, for some $t \geq 1$. Then,

using (3.6), P_t is uniquely determined by

$$P_t = \sum_{i=0}^{\infty} (A_t^\top)^i (\bar{Q}_t + K_t^\top \bar{R}_t K_t) A_t^i. \tag{3.16}$$

By a straightforward computation, we have that

$$\begin{aligned}
A_t^\top P_t A_t + K_t^\top \bar{R}_t K_t =& (A + BK_t)^\top P_t (A + BK_t) + K_t^\top \bar{R}_t K_t \\
=& A^\top P_t A + K_t^\top B^\top P_t A + A^\top P_t B K_t + K_t^\top (B^\top P_t B + \bar{R}_t) K_t \\
=& A^\top P_t A - K_t^\top (B^\top P_t B + \bar{R}_t) K_{t+1} - K_{t+1}^\top (B^\top P_t B + \bar{R}_t) K_t \\
& + K_t^\top (B^\top P_t B + \bar{R}_t) K_t \\
=& A^\top P_t A + (K_{t+1} - K_t)^\top (B^\top P_t B + \bar{R}_t)(K_{t+1} - K_t) \\
& - K_{t+1}^\top (B^\top P_t B + \bar{R}_t) K_{t+1} \\
=& A^\top P_t A + (K_{t+1} - K_t)^\top (B^\top P_t B + \bar{R}_t)(K_{t+1} - K_t) \\
& - K_{t+1}^\top B^\top P_t A - A^\top P_t B K_{t+1} + K_{t+1}^\top (B^\top P_t B + \bar{R}_t) K_{t+1} \\
=& A_{t+1}^\top P_t A_{t+1} + K_{t+1}^\top \bar{R}_t K_{t+1} \\
& + (K_{t+1} - K_t)^\top (B^\top P_t B + \bar{R}_t)(K_{t+1} - K_t),
\end{aligned}$$

where we have used $(B^\top P_t B + \bar{R}_t) K_{t+1} = -B^\top P_t A$ in the third and fifth equalities. Therefore, using this and (3.6), we have that

$$P_t = A_{t+1}^\top P_t A_{t+1} + V, \tag{3.17}$$

where

$$V = K_{t+1}^\top \bar{R}_t K_{t+1} + (K_{t+1} - K_t)^\top (B^\top P_t B + \bar{R}_t)(K_{t+1} - K_t) + \bar{Q}_t.$$

As a result,

$$P_t = \sum_{i=0}^{\infty} (A_{t+1}^\top)^i (V) A_{t+1}^i. \tag{3.18}$$

It is easy to observe that V is positive-definite. Now, using (3.16), since K_t is stabilizing, the matrix P_t is finite. Using (3.18), and the fact that the left side of (3.18) is finite, we have that $\rho(A_{t+1}) < 1$, i.e., K_{t+1} is stabilizing, otherwise the sum on the right side of (3.18) will diverge. □

In order to get a $\log(T)$ regret bound, we need to have bounds of order $\mathcal{O}(1/t)$ on $\|P_t - P_{t-1}\|$, $\|\widehat{X}_t - \widehat{X}_{t-1}\|$ and $\|K_t - K_{t-1}\|$. Also, recall that such bounds are essential for obtaining sequential strong stability using Proposition 3.3.7. The next lemma and its corollary serves this purpose.

Lemma 3.5.3. *Suppose that $\mu I \preceq Q_t, R_t$ and $\mathrm{Tr}(Q_t), \mathrm{Tr}(R_t) \leq \sigma$. Let $\{P_t\}_{t\geq 1}$ and $\{K_t\}_{t\geq 1}$ be the sequences of matrices generated by Algorithm 1, and assume that the sequence $\{K_t\}_{t\geq 1}$ is (κ, γ)-strongly stabilizing. Then we have $\|P_{t+1} - P_t\| \leq m/t$ for some $m > 0$, for $t \geq 1$.*

Proof. Note that using (3.17), we have

$$\begin{aligned} P_{t+1} - P_t =& A_{t+1}^\top (P_{t+1} - P_t) A_{t+1} + K_{t+1}^\top (\bar{R}_{t+1} - \bar{R}_t) K_{t+1} + (\bar{Q}_{t+1} - \bar{Q}_t) \\ & - (K_{t+1} - K_t)^\top (B^\top P_t B + \bar{R}_t)(K_{t+1} - K_t). \end{aligned} \tag{3.19}$$

By the definition of K_t, we have the following identity:

$$K_{t+1} - K_t = (B^\top P_t B + \bar{R}_t)^{-1}\big[B^\top (P_t - P_{t-1})A_t + (\bar{R}_{t-1} - \bar{R}_t)K_t\big]. \tag{3.20}$$

Using this along with (3.19), we have that

$$\begin{aligned} P_{t+1} - P_t =& A_{t+1}^\top (P_{t+1} - P_t)A_{t+1} + K_{t+1}^\top (\bar{R}_{t+1} - \bar{R}_t)K_{t+1} + (\bar{Q}_{t+1} - \bar{Q}_t) \\ &- \big[B^\top (P_t - P_{t-1})A_t + (\bar{R}_{t-1} - \bar{R}_t)K_t\big]^\top (B^\top P_t B + \bar{R}_t)^{-1} \qquad (3.21) \\ &\times \big[B^\top (P_t - P_{t-1})A_t + (\bar{R}_{t-1} - \bar{R}_t)K_t\big]. \qquad (3.22) \end{aligned}$$

By the stability of K_{t+1}, we have that

$$\begin{aligned} P_{t+1} - P_t =& \sum_{i=0}^{\infty} (A_{t+1}^\top)^i M_t A_{t+1}^i \\ &\leq \|M_t\| \sum_{i=0}^{\infty} (A_{t+1}^\top)^i A_{t+1}^i, \end{aligned} \tag{3.23}$$

where

$$\begin{aligned} M_t =& K_{t+1}^\top (\bar{R}_{t+1} - \bar{R}_t)K_{t+1} + (\bar{Q}_{t+1} - \bar{Q}_t) \\ &- \big[B^\top (P_t - P_{t-1})A_t + (\bar{R}_{t-1} - \bar{R}_t)K_t\big]^\top (B^\top P_t B + \bar{R}_t)^{-1} \\ &\times \big[B^\top (P_t - P_{t-1})A_t + (\bar{R}_{t-1} - \bar{R}_t)K_t\big]. \end{aligned}$$

Given the strong stability of K_{t+1}, we can write $A_{t+1} = H_{t+1} L_{t+1} H_{t+1}^{-1}$. Hence, we

have that

$$\begin{aligned}\|\sum_{i=0}^{\infty}(A_{t+1}^{\top})^i A_{t+1}^i\| &\leq \sum_{i=0}^{\infty}\|(A_{t+1}^{\top})^i A_{t+1}^i\| \\ &\leq \sum_{i=0}^{\infty}\|H_{t+1}\|^2\|H_{t+1}^{-1}\|^2\|L_{t+1}\|^{2i} \\ &\leq \sum_{i=0}^{\infty}\kappa^2(1-\gamma)^{2i} = \frac{\kappa^2}{1-(1-\gamma)^2} \leq \frac{\kappa^2}{\gamma},\end{aligned}$$

where we used $\|H_{t+1}\|\|H_{t+1}^{-1}\| \leq \kappa$ and $\|L_{t+1}\| \leq 1-\gamma$. We now proceed to bound M_t. We can write

$$\begin{aligned}\|M_t\| =&\|K_{t+1}^{\top}(\bar{R}_{t+1}-\bar{R}_t)K_{t+1} + (\bar{Q}_{t+1}-\bar{Q}_t)\| \\ &+ \|(B^{\top}P_tB+\bar{R}_t)^{-1}\|(\|B\|\|A_t\|\|P_t-P_{t-1}\| + \|(\bar{R}_{t-1}-\bar{R}_t)K_t\|)^2. \quad (3.24)\end{aligned}$$

Using (3.23) and (3.24), we also have

$$z_{t+1} \leq c_t(h_t z_t + d_t)^2 + e_{t+1}, \quad (3.25)$$

where $z_t = \|P_t - P_{t-1}\|$, and

$$\begin{aligned}c_t &= \frac{\kappa^2}{\gamma}\|(B^{\top}P_tB+\bar{R}_t)^{-1}\| \\ d_t &= \|(\bar{R}_{t-1}-\bar{R}_t)K_t\| \\ e_{t+1} &= \frac{\kappa^2}{\gamma}\|K_{t+1}^{\top}(\bar{R}_{t+1}-\bar{R}_t)K_{t+1} + (\bar{Q}_{t+1}-\bar{Q}_t)\| \\ h_t &= \|B\|\|A_t\|.\end{aligned}$$

Using the fact that

$$\|\bar{Q}_{t+1} - \bar{Q}_t\| = \frac{1}{t+1}\|(Q_t - \bar{Q}_t)\| \leq \frac{2}{t+1}\max_{t\geq 0}\|Q_t\| \leq \frac{2\sigma}{t+1},$$

along with

$$\|\bar{R}_{t+1} - \bar{R}_t\| = \frac{1}{t+1}\|(R_t - \bar{R}_t)\| \leq \frac{2}{t+1}\max_{t\geq 0}\|R_t\| \leq \frac{2\sigma}{t+1},$$

and

$$\|(B^\top P_t B + \bar{R}_t)^{-1}\| \leq (\lambda_{\min}(R_t))^{-1} \leq \mu^{-1},$$

and $\|A_t\| \leq \kappa$, we conclude

$$c_t \leq \kappa^2/\gamma\mu,\ d_t \leq \frac{2\sigma\kappa}{t},\ \text{and}\ e_t \leq \frac{2\kappa^2\sigma(1+\kappa^2)}{\gamma t},\ h_t \leq \|B\|\kappa, \tag{3.26}$$

for $t \geq 1$. We next claim that there exists a time $t^\star$ and a constant $m > 0$ such that $z_t \leq m/t$ for all $t > t^\star$. We use an inductive argument to prove this statement. The base case will be proved later. Assume now that $z_t \leq m/t$; we show that $z_{t+1} \leq m/(t+1)$. First, note that if

$$m \leq \frac{2\sigma}{\|B\|} + \frac{4\kappa^2\sigma(1+\kappa^2)}{\gamma},$$

for $t \geq t^\star = \frac{4\kappa^3\|B\|}{\gamma\mu}(2\sigma\kappa + \frac{2\kappa^3\|B\|\sigma(1+\kappa^2)}{\gamma}) + 1$, using an elementary calculation, one can observe that

$$\frac{\kappa^2}{\gamma\mu}(\kappa\|B\|\frac{m}{t} + \frac{2\sigma\kappa}{t})^2 + \frac{2\kappa^2\sigma(1+\kappa^2)}{\gamma(t+1)} \leq \frac{m}{t+1}.$$

The claim then follows by noting that

$$z_{t+1} \leq c_t(h_t z_t + d_t)^2 + e_t \leq \frac{\kappa^2}{\gamma\mu}(\kappa\|B\|\frac{m}{t} + \frac{2\sigma\kappa}{t})^2 + \frac{2\kappa^2\sigma(1+\kappa^2)}{\gamma(t+1)},$$

where we have used (3.26).

It remains to show that the condition we placed to obtain the last inequality, i.e., that $z_{t^\star+1} \leq m/(t^\star+1)$, is satisfied. To proceed with this, first note that $t^\star$ is exactly the reset time in Algorithm 1. Also, the evolution of $\widehat{P}_\ell$ in the reset part of the algorithm is still according to (3.19). Since the matrices Q_t and R_t are fixed in the reset part, $\{\widehat{P}_\ell\}$ is a Cauchy sequence. Hence, by choosing ℓ large enough, we have that $\|\widehat{P}_\ell - \widehat{P}_{\ell-1}\| \leq m/t^\star$, terminating the reset stage of the algorithm; with slight abuse of notation, we let $\widehat{P}_\ell$ be the outcome of the reset part of the algorithm. Note that at time $t^\star$ the algorithm implements $P_{t^\star} = \widehat{P}_\ell$. In the next time step $t^\star+1$, the algorithm updates $P_{t^\star+1}$ as usual, using (3.6). We know by the previous part of the proof that $\|P_{t^\star+1} - P_{t^\star}\| \leq m/(t^\star+1)$, which shows that $z_{t^\star+1} \leq m/(t^\star+1)$ is satisfied. To conclude the proof, note that we can show that $z_t \leq \hat{m}/t$, for all $t \geq 1$, simply by selecting $\hat{m} = \max\{m, tz_t | t \leq t^\star\}$. □

Corollary 3.5.4. *Let $\widehat{X}_t$ be the steady-state covariance matrix using policy K_t generated by Algorithm 1. Then we have $\|\widehat{X}_t - \widehat{X}_{t-1}\| \leq M/t + M'/t^2$ for some $M > 0$ and $M' > 0$ and for $t \geq 1$.*

Proof. By the definition of $\widehat{X}_t$, we have that

$$\begin{aligned}\widehat{X}_t - \widehat{X}_{t-1} =& A_t\widehat{X}_t A_t^\top - A_{t-1}\widehat{X}_{t-1}A_{t-1}^\top \\ =& A_t(\widehat{X}_t - \widehat{X}_{t-1})A_t^\top + (A_t - A_{t-1})\widehat{X}_{t-1}(A_t - A_{t-1})^\top\end{aligned}$$

$$
\begin{aligned}
&+ A_{t-1}\widehat{X}_{t-1}(A_t - A_{t-1})^\top + (A_t - A_{t-1})\widehat{X}_{t-1}A_{t-1}^\top \\
=&A_t(\widehat{X}_t - \widehat{X}_{t-1})A_t^\top + B(K_t - K_{t-1})\widehat{X}_{t-1}(K_t - K_{t-1})^\top B^\top \\
&+ A_{t-1}\widehat{X}_{t-1}(K_{t-1} - K_t)^\top B^\top + B(K_{t-1} - K_t)\widehat{X}_{t-1}A_{t-1}.
\end{aligned}
$$

Note that Lemma 3.5.3 can be used to bound $K_t - K_{t-1}$. Using (3.20), we have that

$$
\begin{aligned}
\|K_{t+1} - K_t\| \leq& \|(B^\top P_t B + \bar{R}_t)^{-1}\| \big[\|B\|\|P_t - P_{t-1}\|\|A_t\| + \|\bar{R}_{t-1} - \bar{R}_t\|\|K_t\|\big] \\
\leq& \frac{\kappa}{\mu}(\|B\|\hat{m} + 2\sigma)/t, \qquad (3.27)
\end{aligned}
$$

where we have used $\|(B^\top P_t B + \bar{R}_t)^{-1}\| \leq \mu^{-1}$, $\|A_t\| \leq \kappa$, $\|K_t\| \leq \kappa$, and $\hat{m}$ is given in the proof of Lemma 3.5.3. Using this $\|\widehat{X}_t - \widehat{X}_{t+1}\|$ is bounded by $M/t + M'/t^2$, where

$$
M = \frac{2\kappa^6\omega}{\mu\gamma^2}\|B\|\big(\|B\|\hat{m} + 2\sigma\big), \qquad (3.28)
$$

and

$$
M' = \frac{\kappa^6\omega}{\mu\gamma^2}\|B\|^2(\|B\|\hat{m} + 2\sigma)^2, \qquad (3.29)
$$

where we have used

$$
\|\widehat{X}_{t-1}\| \leq \|W\| \sum_{i=0}^{\infty} \|(A_{t-1}^\top)^i (A_{t-1}^i)\| \leq \frac{\omega\kappa^2}{\gamma}
$$

□

The following lemmas will be used to derive bounds on the redundancy terms (3.11), (3.12), and (3.14).

Lemma 3.5.5. *Suppose that the tuple $(A, B, \{Q_t\}_{t=1}^T, \{R_t\}_{t=1}^T, W)$ satisfies Assumption 3.4.1. Suppose that the matrices P_t generated by Algorithm 1 are uniformly bounded, i.e., $P_t \leq \nu I$. Let $\kappa = \sqrt{\frac{\nu}{\mu}}$ and $\gamma = 1/2\kappa^2$. Then for the covariance matrices X_t and $\widehat{X}_t$, we have*

$$\begin{aligned}\sum_{t=1}^{T}(Q_t + K_t^\top R_t K_t) \bullet (X_t - \widehat{X}_t) \leq & t^\star \sigma(1+\kappa^2) \max_{0<t\leq t^\star} \|(X_t - \widehat{X}_t)\| \\ & + 2\kappa^4\sigma\bigg(\|X_{t^\star} - \widehat{X}_{t^\star}\|\frac{e^{-2\gamma^2 t^\star}}{1-e^{-2\gamma^2}} + \frac{M'\pi^2}{6(1-e^{-2\gamma^2})} \\ & + \frac{M}{1-e^{-2\gamma^2}}\log\left(\frac{T}{t^\star}\right)\bigg).\end{aligned}$$

Proof. For $t \geq t^\star$, we have that $\|P_{t+1} - P_t\| \leq m/t \leq \mu/\kappa^2$. Then, using Proposition 3.3.7, the matrices K_t are sequentially (κ, γ)-strongly stabilizing for $t \geq t^\star$ ($\kappa = \sqrt{\frac{\nu}{\mu}}$ and $\gamma = 1/(2\kappa^2)$). Using this by Lemma 3.3.5, we conclude that for $t \geq t^\star$

$$\|X_{t+1} - \widehat{X}_{t+1}\| \leq \kappa^2 e^{-2\gamma^2(t+1-t^\star)}\|X_{t^\star} - \widehat{X}_{t^\star}\| + \kappa^2 \sum_{s=0}^{t-t^\star} e^{-2\gamma^2 s}\eta_{t-s}; \tag{3.30}$$

hence we can separate (3.11) into two parts as follows:

$$\begin{aligned}\sum_{t=1}^{T}(Q_t + K_t^\top R_t K_t) \bullet (X_t - \widehat{X}_t) = & \sum_{t=1}^{t^\star}(Q_t + K_t^\top R_t K_t) \bullet (X_t - \widehat{X}_t) \\ & + \sum_{t=t^\star}^{T}(Q_t + K_t^\top R_t K_t) \bullet (X_t - \widehat{X}_t).\end{aligned}$$

By stability of policies K_t, the matrices X_t and $\widehat{X}_t$ are bounded and we have that

$$\sum_{t=1}^{t^\star}(Q_t + K_t^\top R_t K_t) \bullet (X_t - \widehat{X}_t) = \sum_{t=1}^{t^\star}\mathrm{Tr}\big[(Q_t + K_t^\top R_t K_t)(X_t - \widehat{X}_t)\big]$$

$$\leq \sum_{t=1}^{t^\star} \mathrm{Tr}(Q_t + K_t^\top R_t K_t) \|(X_t - \widehat{X}_t)\|$$
$$\leq t^\star \sigma(1+\kappa^2) \max_{0<t\leq t^\star} \|(X_t - \widehat{X}_t)\|, \qquad (3.31)$$

where we have used

$$\mathrm{Tr}(Q_t + K_t^\top R_t K_t) \leq \sigma(1+\kappa^2)$$

Using (4.13), we have that

$$\begin{aligned}
\sum_{t=t^\star}^{T} (Q_t + K_t^\top R_t K_t) \bullet (X_t - \widehat{X}_t) \leq & \sum_{t=t^\star}^{T} \mathrm{Tr}(Q_t + K_t^\top R_t K_t) \|(X_t - \widehat{X}_t) \\
\leq & \sum_{t=t^\star}^{T} \sigma(1+\kappa^2) \|X_t - \widehat{X}_t\| \\
\leq & (\sigma(1+\kappa^2))\kappa^2 \sum_{t=t^\star}^{T} \Big(e^{-2\gamma^2 t} \|X_{t^\star} - \widehat{X}_{t^\star}\| + \sum_{s=0}^{t-t^\star} e^{-2\gamma^2 s} \eta_{t-s} \Big) \\
\leq & 2\kappa^4 \sigma (\|X_{t^\star} - \widehat{X}_{t^\star}\| \frac{e^{-2\gamma^2 t^\star}}{1-e^{-2\gamma^2}} + \sum_{t=t^\star}^{T} \sum_{s=0}^{t-t^\star} e^{-2\gamma^2 s} \eta_{t-s}).
\end{aligned}$$

Note that by using Corollary 3.5.4, we have $\eta_t = M/t + M'/t$, where M and M' are given by (3.28) and (3.29). Consequently,

$$\begin{aligned}
\sum_{t=t^\star}^{T} (Q_t + K_t^\top R_t K_t) \bullet (X_t - \widehat{X}_t) \leq & 2\kappa^4 \sigma \|X_{t^\star} - \widehat{X}_{t^\star}\| \frac{e^{-2\gamma^2 t^\star}}{1-e^{-2\gamma^2}} \\
& + 2\kappa^4 \sigma \sum_{t=t^\star}^{T} \sum_{s=0}^{t-t^\star} e^{-2\gamma^2 s} \Big(\frac{M}{t-s} + \frac{M'}{(t-s)^2} \Big). \qquad (3.32)
\end{aligned}$$

Next, by changing the order of summation we obtain

$$\begin{aligned}\sum_{t=t^\star}^{T}\sum_{s=0}^{t-t^\star} e^{-2\gamma^2 s}\Big(\frac{M}{t-s}+\frac{M'}{(t-s)^2}\Big) &= \sum_{s=0}^{T-t^\star} e^{-2\gamma^2 s}\sum_{t=s+t^\star}^{T}\Big(\frac{M}{t-s}+\frac{M'}{(t-s)^2}\Big)\\ &\le \sum_{s=0}^{T-t^\star} e^{-2\gamma^2 s}\Big(M\log\Big(\frac{T-s}{t^\star}\Big)+\frac{M'\pi^2}{6}\Big)\\ &\le \frac{M'\pi^2}{6(1-e^{-2\gamma^2})}+\sum_{s=0}^{T-t^\star} Me^{-2\gamma^2 s}\log\Big(\frac{T}{t^\star}\Big)\\ &\le \frac{M'\pi^2}{6(1-e^{-2\gamma^2})}+\frac{M}{1-e^{-2\gamma^2}}\log\Big(\frac{T}{t^\star}\Big),\end{aligned}$$

where we have used a logarithmic upper bound for $\sum_{t=t^\star}^{T-s} 1/t$ and the identity $\sum_{t=1}^{\infty} 1/t^2 = \pi^2/6$ in the second inequality. The third and fourth inequalities follow by manipulating geometric series. Therefore, by substituting this inequality in Equation (3.32) we obtain

$$\begin{aligned}\sum_{t=t^\star}^{T}(Q_t+K_t^\top R_t K_t)\bullet(X_t-\widehat{X}_t) \le 2\kappa^4\sigma\Big(\|X_{t^\star}-\widehat{X}_{t^\star}\|\frac{e^{-2\gamma^2 t^\star}}{1-e^{-2\gamma^2}}+\frac{M'\pi^2}{6(1-e^{-2\gamma^2})}\\ +\frac{M}{1-e^{-2\gamma^2}}\log\Big(\frac{T}{t^\star}\Big)\Big)\end{aligned} \tag{3.33}$$

The result follow by adding (3.31) and (3.33).

□

Lemma 3.5.6. *Suppose that the tuple $(A, B, \{Q_t\}_{t=1}^T, \{R_t\}_{t=1}^T, W)$ satisfies Assumption 3.4.1. Suppose that the matrices P_t generated by Algorithm 1 are uniformly bounded, i.e., $P_t \le \nu I$. Let $\kappa = \sqrt{\frac{\nu}{\mu}}$ and $\gamma = 1/2\kappa^2$. Then the covariance matrices*

$\widehat{X}_t$ and $\widehat{X}^\star$ satisfy

$$\sum_{t=1}^{T}(Q_t + K_t^\top R_t K_t) \bullet \widehat{X}_t - \sum_{t=1}^{T}(Q_t + K^{\star\top} R_t K^\star) \bullet \widehat{X}^\star \leq \omega l \hat{m} + \frac{\kappa^4 \omega}{\gamma \mu^3}(\|B\|\hat{m} + 2\sigma)^2(1 + \log(T)).$$

Proof. Using the fact that $Q_t = t\bar{Q}_t - (t-1)\bar{Q}_{t-1}$ and $R_t = t\bar{R}_t - (t-1)\bar{R}_{t-1}$, we have

$$\begin{aligned}(Q_t + K_t^\top R_t K_t) =& (t\bar{Q}_t - (t-1)\bar{Q}_{t-1}) + K_t^\top (t\bar{R}_t - (t-1)\bar{R}_{t-1})K_t \\ =& t(\bar{Q}_t + K_t^\top \bar{R}_t K_t) - (t-1)(\bar{Q}_{t-1} + K_t^\top \bar{R}_{t-1} K_t) \\ =& t(P_t - A_t^\top P_t A_t) - (t-1)(P_{t-1} - A_t^\top P_{t-1} A_t) \\ &+ (t-1)(K_t - K_{t-1})^\top (B^\top P_{t-1} B + \bar{R}_{t-1})^{-1}(K_t - K_{t-1}), \quad (3.34)\end{aligned}$$

where we have used (3.6) and (3.17) in the third equality. Note that

$$\begin{aligned}A_t^\top P_t A_t \bullet \widehat{X}_t =& \mathrm{Tr}(A_t^\top P_t A_t \widehat{X}_t) \\ =& \mathrm{Tr}(P_t A_t \widehat{X}_t A_t^\top) \\ =& P_t \bullet A_t \widehat{X}_t A_t^\top \\ =& P_t \bullet (\widehat{X}_t - W) \\ =& P_t \bullet \widehat{X}_t - P_t \bullet W. \quad (3.35)\end{aligned}$$

Therefore, by multiplying (3.34) and $\widehat{X}_t$ we obtain

$$(Q_t + K_t^\top R_t K_t) \bullet \widehat{X}_t = tP_t \bullet W - (t-1)P_{t-1} \bullet W$$

$$+ (t-1)(K_t - K_{t-1})^\top (B^\top P_{t-1} B + \bar{R}_{t-1})^{-1} (K_t - K_{t-1}) \bullet \widehat{X}_t,$$

where we have used (3.35) to cancel out some terms. Summing over t and using the telescopic series for $tP_t \bullet W - (t-1)P_{t-1} \bullet W$, we obtain

$$\begin{aligned}\sum_{t=1}^{T} (Q_t + K_t^\top R_t K_t) \bullet \widehat{X}_t \leq\ & T P_T \bullet W \\ & + \sum_{t=1}^{T} (t-1)(K_t - K_{t-1})^\top (B^\top P_{t-1} B + \bar{R}_{t-1})^{-1} (K_t - K_{t-1}) \bullet \widehat{X}_t. \end{aligned} \tag{3.36}$$

On the other hand,

$$\begin{aligned}\sum_{t=1}^{T} (Q_t + K^{\star\top} R_t K^\star) \bullet \widehat{X}^\star =& T(\bar{Q}_T + K^{\star\top} \bar{R}_T K^\star) \bullet \widehat{X}^\star \\ =& T(P^\star - A^{\star\top} P^\star A^\star) \bullet \widehat{X}^\star \\ =& T(P^\star \bullet \widehat{X}^\star - P^\star \bullet A^\star \widehat{X}^\star A^{\star\top}) \\ =& T(P^\star \bullet \widehat{X}^\star - P^\star \bullet \widehat{X}^\star + P^\star \bullet W) \\ =& T P^\star \bullet W. \end{aligned} \tag{3.37}$$

Therefore, by subtracting (3.37) from (3.36) we have

$$\begin{aligned}\sum_{t=1}^{T} (Q_t + K_t^\top R_t K_t) \bullet \widehat{X}_t - \sum_{t=1}^{T} (Q_t + K^{\star\top} R_t K^\star) \bullet \widehat{X}^\star = T(P_T - P^\star) \bullet W + \\ + \sum_{t=1}^{T} (t-1)(K_t - K_{t-1})^\top (B^\top P_{t-1} B + \bar{R}_{t-1})^{-1} (K_t - K_{t-1}) \bullet \widehat{X}_t.\end{aligned}$$

Note that $P^\star$ is the solution of DARE when the cost matrices $Q_t = \bar{Q}_T$ and $R_t = \bar{R}_T$ are chosen to be fixed; it is the limit of the sequence P_t when Q_t and R_t are chosen

to be $\bar{Q}_T$ and $\bar{R}_T$, respectively. The rate of convergence is quadratic [43], i.e. there exists $C > 0$ such that for all $t \geq 2$,

$$\|P_t - P^\star\| \leq C\|P_{t-1} - P^\star\|^2 \tag{3.38}$$

and by a similar analysis, we also have

$$\|P_{t+1} - P_t\| \leq C\|P_t - P_{t-1}\|^2. \tag{3.39}$$

Here we use a similar technique to bound $\|P_T - P^\star\|$. We can update the sequence P_t after time T by starting at P_T using (3.6), with $\bar{Q}_t = \bar{Q}_T$ and $\bar{R}_t = \bar{R}_T$ fixed for $t \geq T$. We hence have that

$$\begin{aligned}
\|P_T - P^\star\| &= \lim_{t\to\infty} \|P_T - P_t\| \\
&\leq \lim_{t\to\infty} \sum_{i=0}^{t-1} \|P_{T+i} - P_{T+i+1}\| \\
&\leq \lim_{t\to\infty} \sum_{i=0}^{t-1} C^{2^i-1}\|P_T - P_{T+1}\|^{2^i} \\
&= \|P_T - P_{T+1}\| \lim_{t\to\infty} \sum_{i=0}^{t-1} C^{2^i-1}\|P_T - P_{T+1}\|^{2^i-1},
\end{aligned}$$

where we have used (3.38). For $T \geq t^\star$, $C\|P_T - P_{T+1}\| < 1$ and thus the sum $\sum_{i=0}^{\infty} C^{2^i-1}\|P_T - P_{T+1}\|^{2^i-1}$ is bounded by some finite value $l > 0$. Hence we have

$$T(P_T - P^\star) \bullet W \leq T\omega\|P_T - P^\star\| \leq T\omega l\|P_T - P_{T+1}\| \leq \frac{T\omega l\hat{m}}{T} = \omega l\hat{m}, \tag{3.40}$$

where we have used $\omega = \mathrm{Tr}(W)$ and $\|P_T - P_{T+1}\| \leq \hat{m}/T$ by Lemma 3.5.3. We now

proceed by noting that

$$\begin{aligned}\sum_{t=2}^{T}(t-1)(K_t-K_{t-1})^\top(B^\top P_{t-1}B+\bar{R}_{t-1})^{-1}(K_t-K_{t-1})\bullet\widehat{X}_t\\ \leq&\sum_{t=2}^{T}(t-1)\mathrm{Tr}(\widehat{X}_t)\frac{1}{(t-1)^2\mu^3}(\|B\|\kappa\hat{m}+2\sigma\kappa)^2\\ \leq&\frac{\kappa^2}{\gamma}\omega(\|B\|\kappa\hat{m}+2\sigma\kappa)^2\sum_{t=2}^{T}\frac{1}{(t-1)\mu^3}\\ \leq&\frac{\kappa^4\omega}{\gamma\mu^3}(\|B\|\hat{m}+2\sigma)^2(1+\log(T)),\end{aligned}\tag{3.41}$$

where we have used the bound in (3.27) on $\|K_t-K_{t-1}\|$, and the bound for $\mathrm{Tr}(\widehat{X}_t)$. Adding (3.41) and (3.40) completes the proof. □

Lemma 3.5.7. *Suppose that the tuple $(A,B,\{Q_t\}_{t=1}^T,\{R_t\}_{t=1}^T,W)$ satisfies Assumption 3.4.1. Suppose that the matrices P_t generated by Algorithm 1 are uniformly bounded, i.e., $P_t\leq\nu I$. Let $\kappa=\sqrt{\frac{\nu}{\mu}}$ and $\gamma=1/2\kappa^2$. Then we have*

$$\sum_{t=1}^{T}(Q_t+K^{\star\top}R_tK^\star)\bullet(\widehat{X}^\star-X_t^\star)\leq\frac{\sigma(1+\kappa^2)\kappa^2}{1-e^{-2\gamma}}\|\widehat{X}^\star-X_1^\star\|.\tag{3.42}$$

Proof. $\|K^\star\|\leq\kappa$ implies that $\mathrm{Tr}(Q_t+K^{\star\top}R_tK^\star)\leq\sigma(1+\kappa^2)$. Moreover, by Lemma 3.3.3, we have that

$$\begin{aligned}\sum_{t=1}^{T}(Q_t+K^{\star\top}R_tK^\star)\bullet(\widehat{X}^\star-X_t^\star)\leq&\sigma(1+\kappa^2)\sum_{t=1}^{T}\|\widehat{X}^\star-X_t^\star\|\\ \leq&\sigma(1+\kappa^2)\sum_{t=1}^{T}\kappa^2e^{-2\gamma(t-1)}\|\widehat{X}^\star-X_1^\star\|\\ \leq&\sigma(1+\kappa^2)\kappa^2\frac{1}{1-e^{-2\gamma}}\|\widehat{X}^\star-X_1^\star\|,\end{aligned}$$

as claimed. □

To conclude, by summing the right hand side of (3.31), (3.33), (3.41), (3.40), and (3.42), we obtain the regret bound as follows,

$$\begin{aligned}\mathcal{R}(T) \leq & \Big(2\kappa^4\sigma\frac{M}{1-e^{-2\gamma^2}} + \frac{\kappa^4\omega}{\gamma\mu^3}(\|B\|\hat{m}+2\sigma)^2\Big)\log(T) \\ & - 2\kappa^4\sigma\frac{M}{1-e^{-2\gamma^2}}\log(t^\star) + t^\star\sigma(1+\kappa^2)\max_{0<t\leq t^\star}\|(X_t-\widehat{X}_t)\| \\ & + 2\kappa^4\sigma\Big(\|X_{t^\star}-\widehat{X}_{t^\star}\|\frac{e^{-2\gamma^2 t^\star}}{1-e^{-2\gamma^2}} + \frac{M'\pi^2}{6(1-e^{-2\gamma^2})}\Big) + \omega l\hat{m} \\ & + \frac{\kappa^4\omega}{\gamma\mu^3}(\|B\|\hat{m}+2\sigma)^2 + \frac{\sigma(1+\kappa^2)\kappa^2}{1-e^{-2\gamma}}\|\widehat{X}^\star - X_1^\star\|, \end{aligned} \tag{3.43}$$

which finishes the proof of Theorem 3.5.1

Note that the assumption of (κ,γ)-strongly stability in Theorem 3.5.1 will be satisfied as long as the solutions to the online Riccati equation are uniformly bounded. In particular, we do not need this assumption for the scalar case, see Proposition 3.7.1.

3.6 Simulations

We provide simulation results for the proposed algorithm to illustrate its performance. The control system dynamics are given by $x_{t+1} = Ax_t + Bu_t + w_t$, where the pair (A,B) is stabilizable, and $A \in \mathbb{R}^{10\times 10}$ and $B \in \mathbb{R}^{10\times 7}$, and w_t is a Gaussian noise. The matrices A and B are chosen randomly with entry-wise i.i.d uniform distribution on $[-3,3]$ and $[-2,2]$ respectively. We have considered three scenarios for the cost functions. For the first experiment, the matrices Q_t and R_t are generated randomly with the Wishart distribution with unit variance and 20 degrees of freedom. For the second and third experiment, we followed the experiment setting of [25], where

$Q_t = Q$ is fixed as the identity matrix, while R_t is diagonal where some diagonal entries are 1, while others are r_t. For the second experiment, we assume that r_t is randomly changing over time with i.i.d uniform distribution on $[0.1, 1]$, and for the third experiment, we assume that r_t is changing over time according to a random walk restricted to $[0.1, 1]$ taking steps of size $0.1, -0.1, 0$, with probability $0.1, 0.1, 0.8$, respectively. We ran the algorithm with the stabilizing matrix K_0 and $X_0 = 0$ to generate a sequence of matrices K_t, and we computed the regret and the average regret over time. We compared our results with the ones stated in [25].

Figure 3.5 shows the regret over time for the online Riccati algorithm and the follow the lazy leader (FLL) algorithm given in [25] for the first experiment. The results show that both algorithms behave similarly. Although [25] found a regret bound of $\mathcal{O}(\sqrt{T})$ while we have achieved a regret bound of $\mathcal{O}(\log T)$, this simulation result is expected, since FLL uses the average cost matrices Q_t and R_t over time and finds the optimal K_t and uses it for the next time step, and the online Riccati algorithm uses a Riccati update of the average cost matrices Q_t and R_t over time.

Figures 3.6, 3.7 and 3.8 show the average regret of the online Riccati algorithm, FLL algorithm, and the recent cost policy, where the optimal policy of recent cost matrices is used for the next time step. The graphs show that the online Riccati algorithm works well for different scenarios, and as expected the recent cost policy is not a good strategy and only works for the random walk scenario where the change in the cost function is slow, as also indicated in [25], and we have plotted these for comparison.

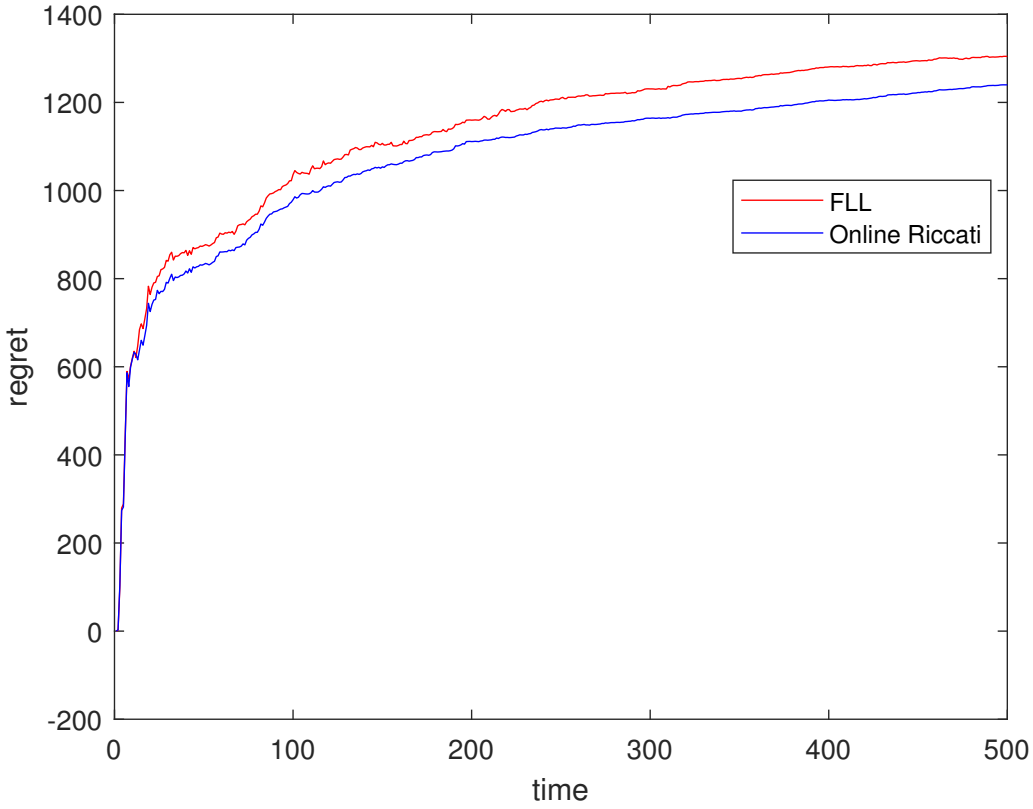

Figure 3.5: The regret over time using the policies generated by Algorithm 1 and FLL algorithm

3.7 Boundedness Assumptions

Proposition 3.7.1. *Let $n = m = 1$ and let $\{P_t\}_{t=1}^T$ be a sequence of positive numbers generated by Equations (3.6) and (3.7) recursively, and assume that policy K_t is stabilizing for all $t \geq 1$. Then there exists $\nu > 0$ such that $P_t \leq \nu$ for all $t \geq 1$.*

Proof. Note that

$$P_t = (A + BK_t)^2 P_t + \bar{Q}_t + K_t^2 \bar{R}_t,$$

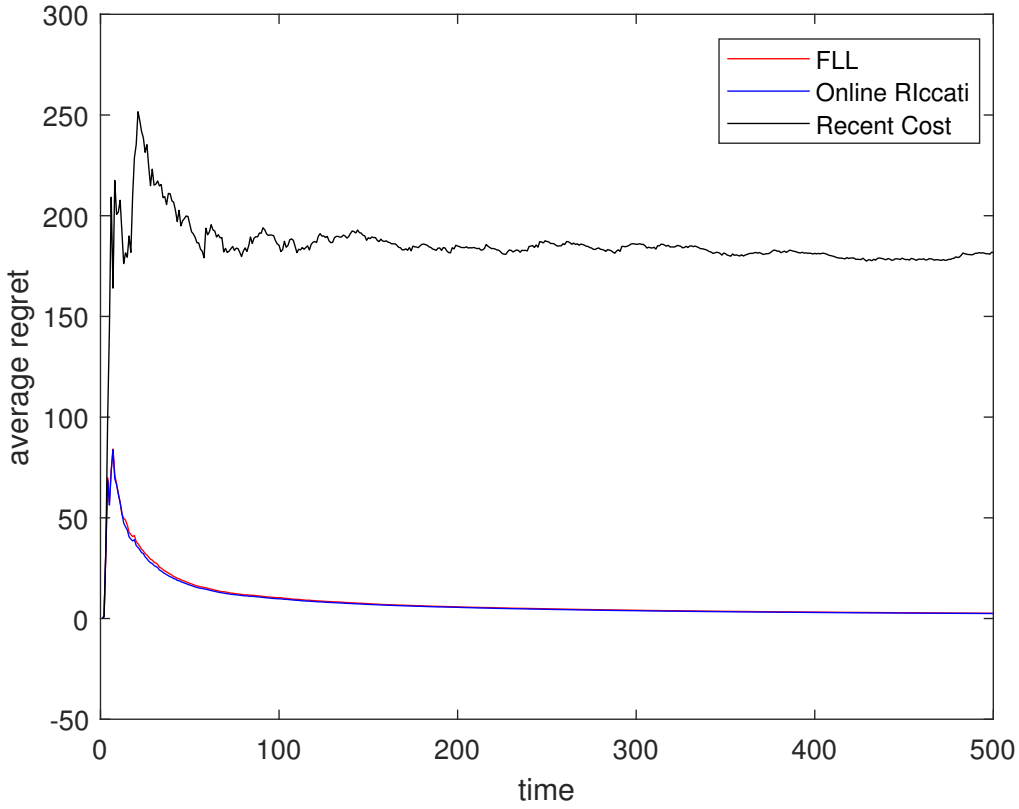

Figure 3.6: The average regret over time using the policies generated by Algorithm 1, FLL algorithm and recent cost policy for the first experiment

Since K_t is stabilizing using the stability of K_1, c.f. Lemma 3.5.2, we have that

$$P_t = \frac{\bar{Q}_t + K_t^2 \bar{R}_t}{1 - (A + BK_t)^2}.$$

Now if you consider P_t as a function of K_t, by taking derivative of P_t with respect to K_t and setting it to zero, we have that

$$K_t = \frac{\bar{R}_t - A^2\bar{R}_t + B^2\bar{Q}_t - \sqrt{(\bar{R}_t - A^2\bar{R}_t + B^2\bar{Q}_t)^2 + 4A^2B^2\bar{R}_t\bar{Q}_t}}{2AB\bar{R}_t}$$

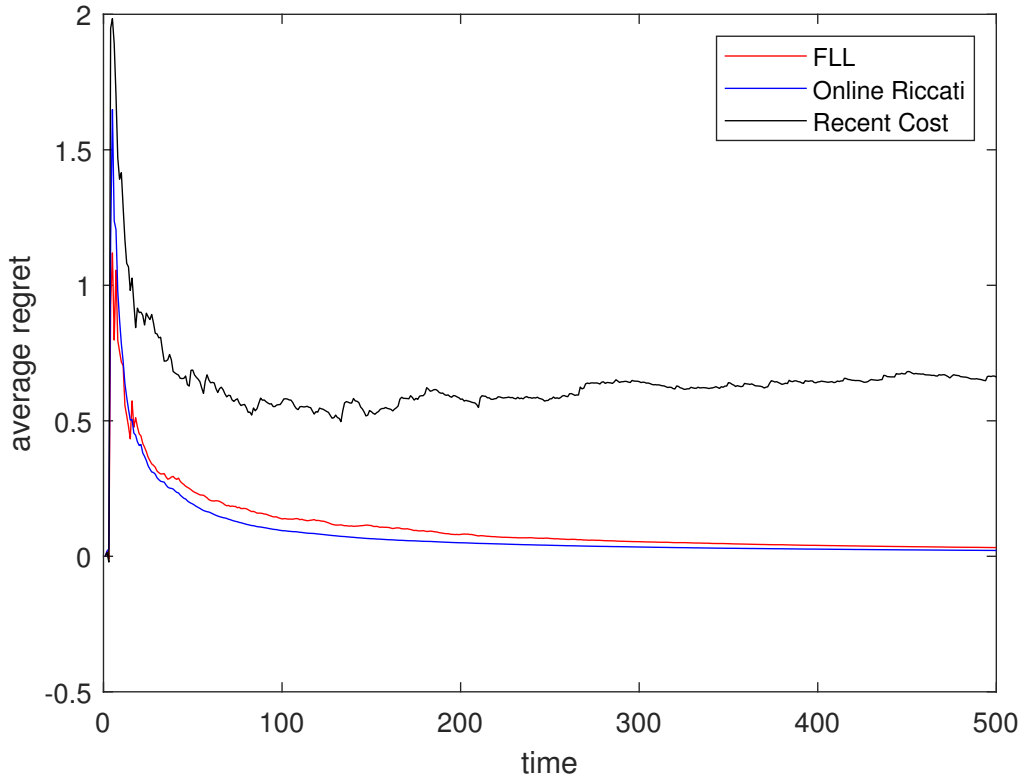

Figure 3.7: The average regret over time using the policies generated by Algorithm 1, FLL algorithm and recent cost policy for the second experiment

minimizes the P_t and the minimum admissible P_t which we denote by $\tilde{P}_t$ is given by

$$\tilde{P}_t = \frac{A^2\bar{R}_t - \bar{R}_t + \bar{Q}_t B^2 + \sqrt{(\bar{R}_t - A^2\bar{R}_t - \bar{Q}_t B^2)^2 + 4B^2\bar{Q}_t\bar{R}_t}}{2B^2}.$$

Now if we write P_{t+1} as a function of P_t we have that

$$\begin{aligned} P_{t+1} &= \frac{\bar{Q}_{t+1} + K_{t+1}^2\bar{R}_{t+1}}{1-(A+BK_{t+1})^2} \\ &= \frac{\bar{Q}_{t+1} + ((B^2P_t + \bar{R}_t)^{-1}BP_tA)^2\bar{R}_{t+1}}{1-(A\bar{R}_t(B^2P_t+\bar{R}_t)^{-1})^2} \end{aligned}$$

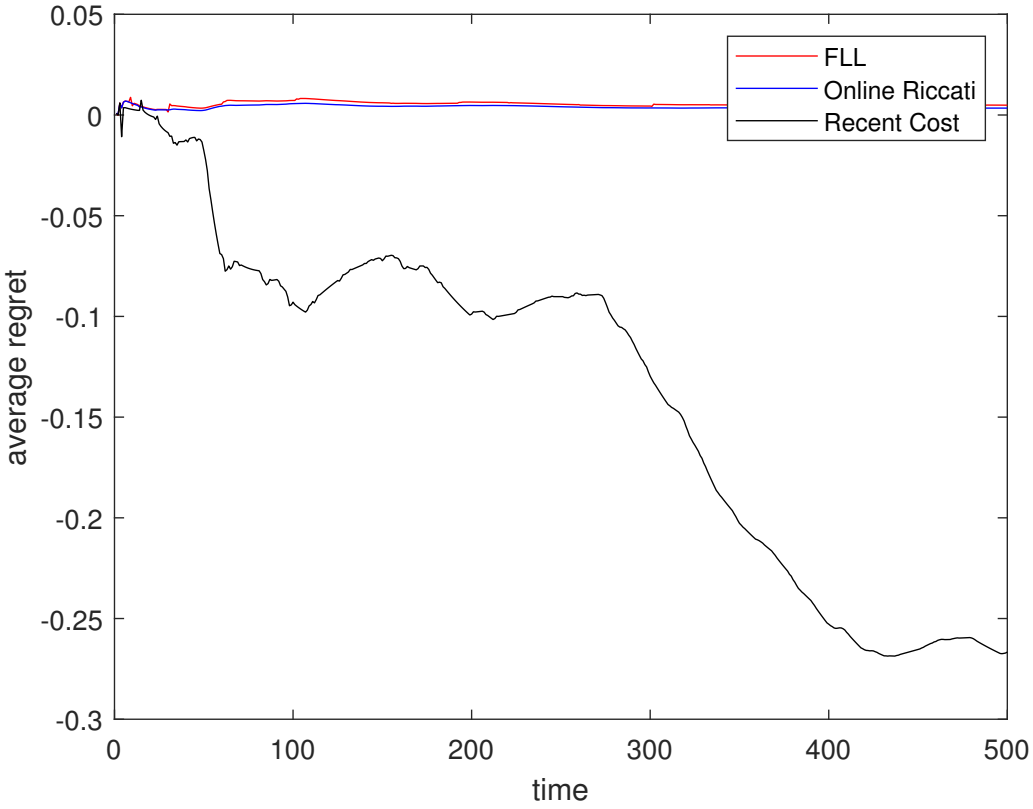

Figure 3.8: The average regret over time using the policies generated by Algorithm 1, FLL algorithm and recent cost policy for the third experiment

$$= \frac{\bar{Q}_{t+1}(B^2P_t + \bar{R}_t)^2 + B^2P_t^2A^2\bar{R}_{t+1}}{(B^2P_t + \bar{R}_t)^2 - A^2\bar{R}_t^2}.$$

By taking derivative of P_{t+1} with respect to P_t, we conclude that for the admissible P_t, i.e., $P_t \geq \tilde{P}_t$, the function P_{t+1} is decreasing for $P_t \leq \breve{P}_t$ and increasing for $P_t \geq \breve{P}_t$ [7], where $\breve{P}_t$ is given by

$$\breve{P}_t = \frac{(A^2-1)\bar{R}_{t+1}\bar{R}_t + B^2\bar{Q}_{t+1}\bar{R}_t + \sqrt{((A^2-1)\bar{R}_{t+1}\bar{R}_t + B^2\bar{Q}_{t+1}\bar{R}_t)^2 + 4B^2\bar{Q}_{t+1}\bar{R}_{t+1}\bar{R}_t^2}}{2B^2\bar{R}_{t+1}}.$$

Since P_{t+1} is decreasing for $P_t \leq \breve{P}_t$ and increasing for $P_t \geq \breve{P}_t$, its maximum is

achieved on the boundary. So we will check the value of P_{t+1} for the point P_t at infinity and at its admissible minimum $\tilde{P}_t$. Now letting P_t goes to infinity, we have

$$P_{t+1} = \lim_{P_t \to \infty} \frac{\bar{Q}_{t+1}(B^2 P_t + \bar{R}_t)^2 + B^2 P_t^2 A^2 \bar{R}_{t+1}}{(B^2 P_t + \bar{R}_t)^2 - A^2 \bar{R}_t^2} = \frac{A^2}{B^2}\bar{R}_{t+1} + \bar{Q}_{t+1},$$

and for $P_t = \tilde{P}_t$, we have

$$P_{t+1} = \frac{\bar{Q}_{t+1}(B^2 \tilde{P}_t + \bar{R}_t)^2 + B^2 \tilde{P}_t^2 A^2 \bar{R}_{t+1}}{(B^2 \tilde{P}_t + \bar{R}_t)^2 - A^2 \bar{R}_t^2}$$

One can observe that P_{t+1} as a function of R_t has a similar behaviour. So for P_{t+1} to achieve its maximum, $(\tilde{P}_t, R_t)$ should be minimum and (Q_{t+1}, R_{t+1}) should be maximum. So if we let $Q_{\max} = \max\{\bar{Q}_1, \bar{Q}_2, \cdots, \bar{Q}_T\}$, $Q_{\min} = \min\{\bar{Q}_1, \bar{Q}_2, \cdots, \bar{Q}_T\}$, $R_{\max} = \max\{\bar{R}_1, \bar{R}_2, \cdots, \bar{R}_T\}$, $R_{\min} = \min\{\bar{R}_1, \bar{R}_2, \cdots, \bar{R}_T\}$, and

$$\tilde{P}_{\min} = \frac{A^2 R_{\min} - R_{\min} + Q_{\min} B^2 + \sqrt{(R_{\min} - A^2 R_{\min} - Q_{\min} B^2)^2 + 4B^2 Q_{\min} R_{\min}}}{2B^2},$$

we obtain that for all $t > 0$

$$P_t \leq \max\left\{\frac{A^2}{B^2} R_{\max} + Q_{\max}, \frac{Q_{\max}(B^2 \tilde{P}_{\min} + R_{\min})^2 + B^2 \tilde{P}_{\min}^2 A^2 R_{\max}}{(B^2 \tilde{P}_{\min} + R_{\min})^2 - A^2 R_{\min}^2}\right\}$$

□

We illustrate in the next remark as to why the argument that we have used above cannot be readily extended to non-scalar cases.

Remark 3.7.2. The procedure that we have used above to prove boundedness of P_t relied on studying the evolutions of P_{t+1} as a function of P_t. When these quantities

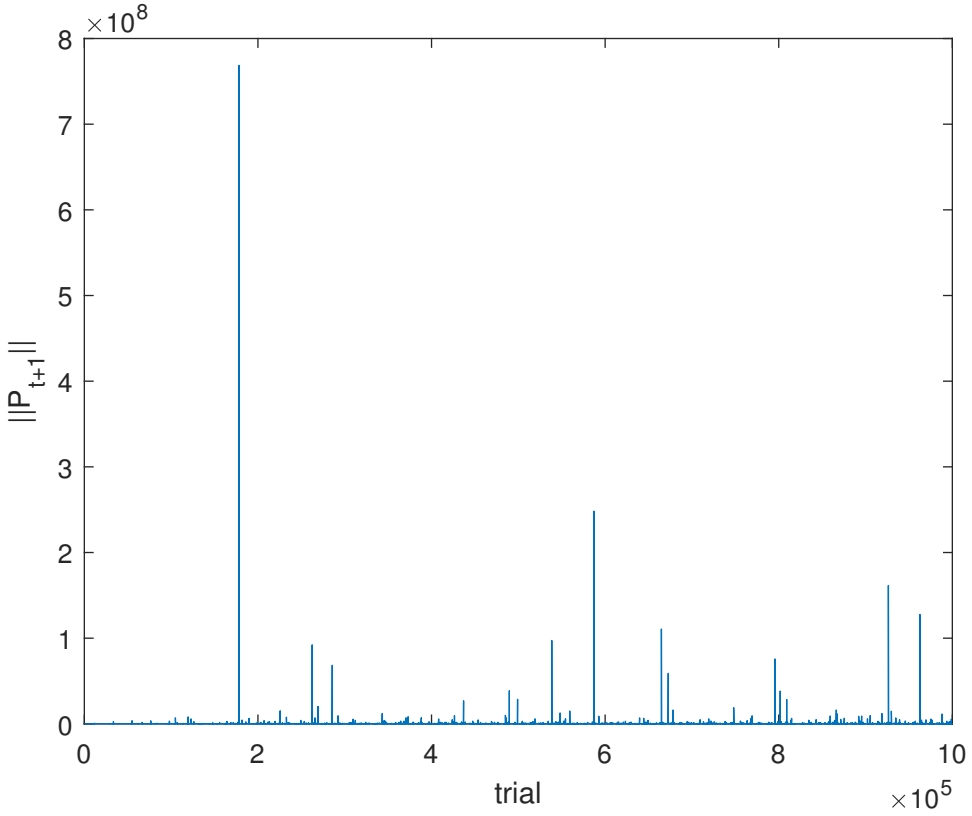

Figure 3.9: This graph shows the norm of P_{t+1} for different values of $P_t \succeq P^*$. P_t can be near the boundary that makes K_{t+1} unstabilizing, and hence P_{t+1} gets very large.

are not scalars, one naturally aims to consider the norm of P_{t+1} as a function of the norm of P_t. However, an example can be constructed where P_{t+1} as a function of P_t becomes unbounded as P_t approaches the boundary of the set positive-definite matrices that make K_{t+1} unstabilizing. This does not happen in the scalar case since this boundary is smaller than $\tilde{P}_t$, the minimum achievable P_t. Figure 3.9 depicts the norm of P_{t+1} for different trials of selecting P_t. For each trial, the P_t is chosen as $P_t = P^* + \Omega$, where P^* is the minimum achievable P_t for a stabilizing matrix K_t, and Ω is a positive definite matrix. It can be seen that the norm of P_{t+1} for some trials

gets very large. For example, for P_t

$$P_t = \begin{pmatrix} [r]18714 & -312 & 291 \\ -312 & 82149 & -144 \\ 291 & -144 & 14220 \end{pmatrix},$$

the matrix $A + BK_{t+1}$ has the eigenvalues

$$\lambda(A + BK_{t+1}) = \begin{pmatrix} -0.999996 \\ 0.002971 \\ -0.000047 \end{pmatrix},$$

and the first eigenvalue that is near 1, which makes the norm of P_{t+1} around the order of 7.7×10^8. However, in several simulations of online Riccati algorithm, we observed that changes in P_t as a result of changes in bounded $\bar{Q}_t$ and $\bar{R}$ do not make K_{t+1} to get close to the unstabilizing policy boundary, and hence P_{t+1} cannot get unbounded. We will show this behaviour in the following experiment.

Example 3.7.3. In order to observe the behaviour of matrices P_t over time, a linear discrete-time control system with $n = 7$ states and $m = 5$ control actions is considered, where the matrices (A, B) are fixed. We used several trials, where for each trial a sequence of positive definite random matrices Q_t and R_t with Wishart distribution is generated and we used the online Riccati algorithm with different initialization K_1 to generate the sequence P_t. Figure 3.10 shows the graph of the norm of P_t over time for each trial. Clearly, P_t stays bounded. Similar property is observed in all our simulation studies. Understanding why this boundedness occurs and if this is generally true is an important open problem, and appears to be difficult in light of

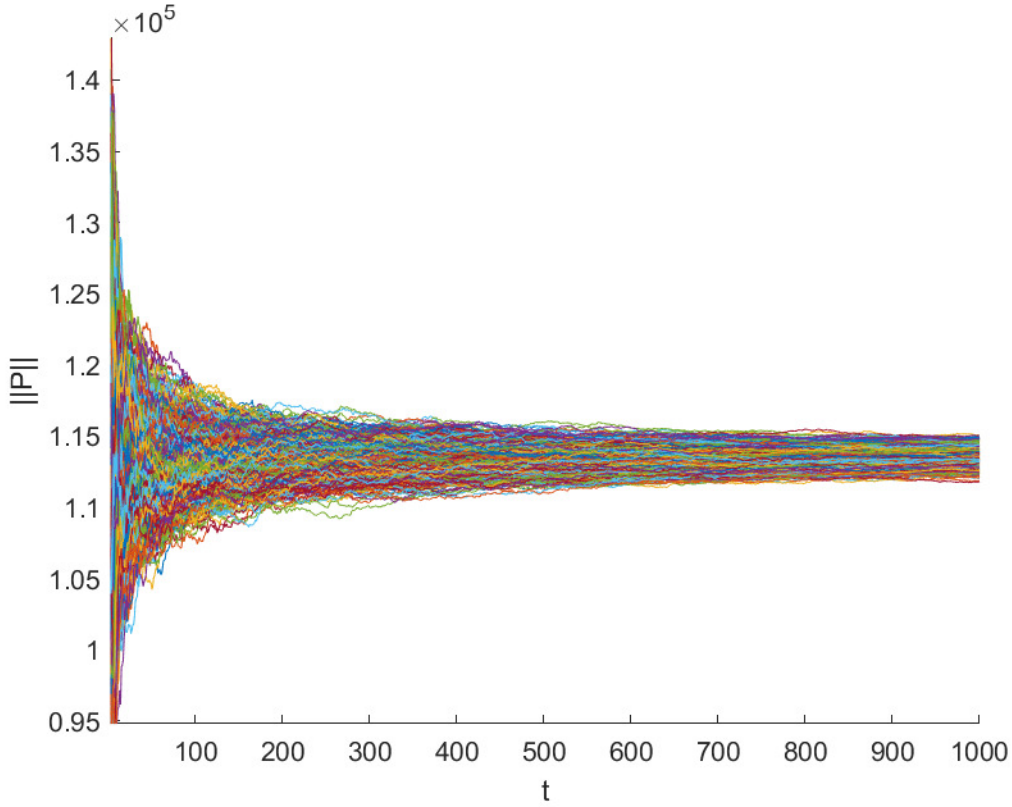

Figure 3.10: The norm of P_t over time for 1000 trials is shown. For each trial, a sequence of matrices Q_t and R_t with Wishart distribution is generated and the sequence P_t is generated using the online Riccati algorithm.

the previous remark.

Chapter 4

Online Adaptive Linear Quadratic Gaussian Control

4.1 Problem Statement

Here, we review the problem of the adaptive control of discrete-time linear quadratic Gaussian (LQG) systems. We recall the LQG problem and the notation we use in this chapter.

4.2 Discrete-Time Linear Quadratic Gaussian Control

The LQG problem is modelled as follows. Let $x_t \in \mathbb{R}^n$ be the system state and let $u_t \in \mathbb{R}^m$ be the control action at time $t \geq 1$ with initial state x_1. The system states evolve over time according to the difference equation

$$x_{t+1} = A_* x_t + B_* u_t + w_t, \tag{4.1}$$

where $A_* \in \mathbb{R}^{n \times n}$, $B_* \in \mathbb{R}^{n \times m}$ are the state-state and state-action real matrices, respectively, and $\{w_t\}_{t \geq 1}$ is an i.i.d Gaussian noise sequence ($w_t \sim \mathcal{N}(0, \sigma^2 I_n)$). At

each time step t, there is a cost of the form $x_t^\top Q x_t + u_t^\top R u_t$, where $Q \in \mathbb{R}^{n\times n}$ and $R \in \mathbb{R}^{m\times m}$ are positive definite matrices. The objective is to find the optimal controller that minimizes the infinite horizon cost

$$J(\{u_t\}_{t\geq 1}) = \lim_{T\to\infty} \mathbb{E}\Big[\frac{1}{T}\sum_{t=1}^{T} x_t^\top Q x_t + u_t^\top R u_t\Big] \tag{4.2}$$

It can be proved that when the pair (A_*, B_*) is stabilizable and $(A_*, Q^{1/2})$ is detectable, the optimal controller is a linear feedback of the state, see [14]. In particular, the optimal controller is given by $u_t^* = K_* x_t$, where $K_* \in \mathbb{R}^{m\times n}$ is the so-called optimal feedback gain matrix given by

$$K_* = -(B_*^\top P_* B_* + R)^{-1} B_*^\top P_* A_*, \tag{4.3}$$

where $P_* \in \mathbb{R}^{n\times n}$ is the solution to the algebraic Riccati equation

$$P_* = A_*^\top P_* A_* - A_*^\top P_* B_* (B_*^\top P_* B_* + R)^{-1} B_*^\top P_* A_* + Q. \tag{4.4}$$

Furthermore the linear feedback gain K_* is stabilizing (i.e., $(A_* + B_* K_*)$ is stable: $\rho(A_* + B_* K_*) < 1$), and the optimal infinite horizon cost is

$$J_* = \min_{\{u_t\}_{t\geq 1}} J = \sigma^2 \mathrm{Tr}(P_*). \tag{4.5}$$

We will use the notation $K_* = \mathrm{dare}(A_*, B_*, Q, R)$ for (4.3).

4.3 Adaptive Control

In adaptive control, the system parameters (A_*, B_*) are not known, and the controller uses the past history of data $\{(x_t, u_t)\}_{t\geq 1}$ to update its policy. In this setting, although the optimal controller is potentially not achievable, we are interested in approaching the optimal infinite horizon cost. We define the *regret* as the difference between the cumulative cost over time when the controller uses policy $u_t = \pi_t(x_t)$ and the optimal infinite horizon cost, given by

$$\mathcal{R}_T(\pi) = \sum_{t=1}^{T} (x_t^\top Q x_t + u_t^\top R u_t) - T J_*, \tag{4.6}$$

where $\pi = (\pi_1, \pi_2, \ldots, \pi_T)$ and π_t depends on the history of the data $\{(x_s, u_s)\}_{s=1}^{t-1}$.

The objective here is to design algorithm that generates the policy π that has a low regret. Note that in (4.6), we are comparing the cost of the controller policy with the infinite-horizon optimal cost. Although it is natural to consider the finite-horizon optimal cost in the definition of regret, the difference between these two costs is negligible as far as the regret bound in Theorem 4.6.1 is concerned. This is due to the fact that the finite-horizon optimal policy converges to the infinite-horizon optimal policy as T goes to infinity with an exponential rate [22]; in particular, the sequence of matrices P_t computed by the dynamic Riccati equation (2.2) for the finite-horizon problem converges exponentially fast to P_*. Furthermore, the difference between the finite-horizon total cost given by $J_T = \sum_{s=1}^{T} \sigma^2 \mathrm{Tr}(P_t)$ [13], and the infinite-horizon total cost TJ_* is bounded by a constant and hence does not impact the regret bound in terms of T.

4.4 Main Result

We consider the problem of the adaptive LQG control in a scenario where the true system parameters of the transition dynamics A_* and B_* are *unknown*, but a hint for the matrix B_* is given to the controller periodically. This hint, which roughly speaking includes some noisy information about the "direction" towards B_*, will help the controller achieve logarithmic regret, even though it does not know the true system parameters A_* and B_*. Our proposed algorithm which uses a regularized least squares estimate is adopted from [20]. We now specify our assumptions.

We assume that the matrices Q and R are known to the learner and there are constants $\alpha_0, \alpha_1 > 0$ such that

$$\alpha_0 I_n \preceq Q, R \preceq \alpha_1 I_m.$$

We also assume that the learner knows a stabilizing feedback gain K_0 with finite cost such that $J(K_0) \leq \nu$ for some $\nu > 0$ and hence $J_* \leq \nu$. We also define ϕ as

$$\phi = \max(\|A_*\|_F,\ \|B_*\|_F).$$

All these assumptions are standard and widely used in this setting, see [20].

4.5 Online Adaptive Control Algorithm with Hint

In this section, we present our algorithm. The algorithm start with a warm-up stage where the controller uses the policy $u_t = K_0 x_t + \eta_t$ for $1 \leq t < \tau_1$, where K_0 is a (k, ℓ) strongly stabilizing feedback gain (with k and ℓ determined by the assumption

$J(K_0) \leq \nu$, see Lemma 4.7.7), η_t is i.i.d. Gaussian perturbation ($\eta_t \sim \mathcal{N}(0, \sigma^2 I)$), and τ_1 is the length of the warm-up. The perturbation η_t ensures with high probability the persistency of excitation of the controller [70]. This is classically being used for the purpose of identification, see [27, 38]. The data $\{(x_t, u_t)\}_{t=1}^{\tau_1 - 1}$ collected after the warm-up period is used to obtain an estimate of (A_*, B_*) using a regularized least squares method. Let us call this estimate $(\widehat{A}_{\tau_1}, \widehat{B}_{\tau_1})$. After this step, an external "hint", to be made precise shortly, is given to the learner that improves the estimate of (A_*, B_*). This new estimate will be denoted by (A_{τ_1}, B_{τ_1}). The learner uses this estimate to find a better feedback gain K_{τ_1}, and applies the control $u_t = K_{\tau_1} x_t$ for $\tau_1 \leq t < \tau_2$ if K_{τ_1} is a (k, ℓ) strongly stabilizing controller for the system and the state of the system remains bounded. Otherwise, the controller uses the policy $u_t = K_0 x_t$ for the remaining time. This process will be repeated for each time period of length τ_i, and after period τ_i, the learner achieves a better feedback gain K_{τ_i}. We will show that the generated feedback gain results in a logarithmic regret bound.

We now provide a rough description of the notion of hint and how it is used by the learner; the idea of hint used here is similar to the one in [30]. We assume that there is an external hint for B_* given to the learner periodically. In particular, we assume that after the warm-up, the learner has the estimate $\widehat{B}_{\tau_1}$ of B_*. The hint is given as $\gamma_1(B_* - \widehat{B}_{\tau_1}) + E_1$, where $\gamma_1 \in (0, 1)$ and $E_1 \in \mathbb{R}^{n \times m}$ are not known to the learner, the assumption on which we describe later. The learner updates its estimate as

$$B_{\tau_1} = \widehat{B}_{\tau_1} + \gamma_1(B_* - \widehat{B}_{\tau_1}) + E_1. \tag{4.7}$$

Using the new estimate B_{τ_1}, the learner updates A_{τ_1} as a new estimate of A_* and uses the algebraic Riccati equation to compute a new feedback gain K_{τ_1}. Then the

learner uses the policy $u_t = K_{\tau_1} x_t$ to control the system. We assume that at each round of estimation τ_i, after making the estimate $\widehat{A}_{\tau_i}$ and $\widehat{B}_{\tau_i}$, the learner receives the hint $\gamma_i(B_* - \widehat{B}_{\tau_i}) + E_i$ and revises $B_{\tau_i} = \widehat{B}_{\tau_i} + \gamma_i(B_* - \widehat{B}_{\tau_i}) + E_i$. The conditions on γ_i and E_i will be provided in the main result.

Algorithm 2: Online Adaptive Control with Hint

input : a stabilizing controller K_0, time horizon T, time window parameter r, τ_1, k, x_b, λ
output: A sequence of policies $\{K_t\}_{t=1}^T$

Initialize: $n_T = \lfloor \log_r(T/\tau_1) \rfloor$ $\quad \tau_{n_{T+1}} = T + 1$;
set $\tau_i = \tau_1 r^{i-1}$ for all $i = 1, \ldots, n_T$;
for *each* $t = 1, \ldots, \tau_1 - 1$ **do**
  receive x_t;
  use controller $u_t = K_0 x_t + \eta_t$;
for *each* $i = 1, 2, \ldots, n_T$ **do**
  $(\widehat{A}_{\tau_i}\ \widehat{B}_{\tau_i}) = \operatorname{argmin}_{(A\ B)} \sum_{t=1}^{\tau_i - 1} \|x_{t+1} - Ax_t - Bu_t\|^2 + \lambda \|(A\ B)\|_F^2$;
  receive hint and update $B_{\tau_i} = \widehat{B}_{\tau_i} - \gamma_i(\widehat{B}_{\tau_i} - B_*) + E_i$;
  update $A_{\tau_i} = \operatorname{argmin}_A \sum_{t=1}^{\tau_i - 1} \|x_{t+1} - Ax_t - B_{\tau_i} u_t\|^2 + \lambda \|A\|_F^2$
   $K_{\tau_i} = \operatorname{dare}(A_{\tau_i}, B_{\tau_i}, Q, R)$;
  for *each* $t = \tau_i, \ldots, \tau_{i+1} - 1$ **do**
    if $\|x_t\|^2 > x_b$ *or* $\|K_{\tau_i}\| > k$ **then**
      abort and play $u_t = K_0 x_t$ forever
    else
      play $u_t = K_{\tau_i} x_t$

4.6 Main Theorem

Our main result, stated below, shows that if partial information about the matrix B_* is provided to the learner periodically, as it is stated in the algorithm, the learner can achieve a logarithmic regret.

Theorem 4.6.1. *Let Algorithm 2 be run with parameters*

$$k = \sqrt{\frac{\nu + \epsilon_0^2 C_0}{\alpha_0 \sigma^2}}, \tau_1 = \left\lceil \frac{240\lambda(1+\phi^2)((1+k^2)/\min\{p,1\}+1)(n+m)}{\epsilon_0^2 \sigma^2} \right\rceil$$
$$x_b = 135nk^2\sigma^2 \max\left\{(1+\phi)^2k^6, 4k^6\right\}\log(4T), \lambda = (1+k)^2 x_b, p = \frac{r}{2+k^2},$$

and assume the γ_i *satisfy* $0 \le 1-\gamma_1 \le \frac{1}{r}$ *and* $(1-\gamma_{i+1}) \le \frac{1}{r}(1-\gamma_i)$ *for all* $i \ge 1$. *Further assume* $\|E_i\|_F^2 \le \frac{1-\gamma_i}{\tau_i}$ *for all* $i \ge 1$. *Then for* $T \ge \mathrm{poly}(\alpha_0, \alpha_1, \phi, \nu, m, n, r)$ *we have* $\mathbb{E}[\mathcal{R}_T] \le \mathrm{poly}(\alpha_0, \alpha_1, \phi, \nu, m, n, r)\log^2(T)$, *and in particular,*

$$\begin{aligned}
\mathbb{E}[\mathcal{R}_T] \le \frac{135}{\log(r)}\Big((r-1)C_0(240(1+k^2)(1+\phi^2)\big(\frac{1+k^2}{\min\{p,1\}}+1\big)(n+m)+\sigma^2) \\
+ 4\alpha_1 k^6\sigma^2\Big)nk^2\max\{(1+\phi)^2k^6, 4k^6\}\log^2(T) \\
+ 135\frac{240(1+k^2)(1+\phi^2)((1+k^2)/\min\{p,1\}+1)(n+m)+\sigma^2}{\epsilon_0^2} \\
nk^2\max\{(1+\phi)^2k^6, 4k^6\}(1+\phi^2)\nu\log(4T) \\
+ \frac{270}{\log(r)}\Big((r-1)C_0(240(1+k^2)(1+\phi^2)(\frac{1+k^2}{\min\{p,1\}}+1)(n+m)+\sigma^2) \\
+ 4\alpha_1 k^6\sigma^2\Big)nk^2\max\{(1+\phi)^2k^6, 4k^6\}\log(T) \\
+ (\nu + 270\alpha_1 nk^4\sigma^2\max\{(1+\phi)^2k^6, 4k^6\}\log(4T))T^{-1} \\
+ 1080\alpha_1(1+8\phi^2)nk^{10}\sigma^2\max\{(1+\phi)^2k^6, 4k^6\}\log(4T)T^{-2}.
\end{aligned}$$

For the special case of $\gamma_i = 1$ and $E_i = 0$, we recover the following known result as a special case of our main theorem.

Corollary 4.6.2. *[20, Theorem 1] There exists an efficient algorithm that, given matrix* B_* *as a known input, has expected regret* $\mathbb{E}[\mathcal{R}_T] \le \mathrm{poly}(\alpha_0, \alpha_1, \phi, \nu, m, n, r)\log^2(T)$.

It is also worth making some remarks regarding the case where A_* is a known input to the algorithm. As [20, Theorem 2] stated below demonstrates, one can still obtain a logarithmic result. Even though this result is not a direct consequence of Theorem 4.6.1, we point out that it is in the same spirit. In fact, in the proof of the mentioned result, the accurate information on the drift term A_* is utilized to find better estimate of the unknown parameter B, with a smaller distance to B_*, much like the hint that is utilized in Algorithm 2. This being said, this type of hint is naturally different from the one given in our algorithm, making this result not a direct corollary of Theorem 4.6.1.

Theorem 4.6.3. *[20, Theorem 2] Suppose that the optimal policy of the system satisfies $K_*K_*^\top \succeq \mu_* I$. There exists an efficient algorithm that, given matrix A_* as input, has expected regret $\mathbb{E}[\mathcal{R}_T] \leq \mathrm{poly}(\mu_*^{-1}, \alpha_0, \alpha_1, \phi, \nu, m, n, r)\log^2(T)$.*

Remark 4.6.4. *Note that in Algorithm 2 we make an initial estimate $\widehat{B}_{\tau_i}$ on Line 7 using a least-square error estimate before receiving the hint. An interesting question is if this estimate is necessary to achieve a logarithmic regret bound. One can consider an alternative algorithm that uses an arbitrary estimate $\widehat{B}_{\tau_i}$ and receives the hint on its direction towards B_* (Line 8 of Algorithm 2) and updates the estimate A_{τ_i} using Line 9 of Algorithm 2. Our simulation studies suggest that this algorithm may lead to a similar regret bound; however, the analysis for this setting appears to require a new set of tools as the current proof does not apply. We leave investigating this for future.*

4.7 Proof of Theorem 4.6.1

The proof of the theorem is organized as follows. We start by reviewing some results from the literature that play major role in proving Theorem 4.6.1. Next, we state our results on how the estimated parameters $(A_{\tau_i}\ B_{\tau_i})$ are related to the history of observed data. Then we use these results and Lemma 4.7.2 below to find a suitable event with high probability on which the difference between $(A_{\tau_i}\ B_{\tau_i})$ and $(A_*\ B_*)$ is small, which allows us to use Lemma 4.7.1 below to bound the regret. Then we define an event on which the system noise and controller perturbation is bounded and we have stabilizing controller and bounded states. The rest of the proof bounds the expected regret by conditioning on this event and its complement.

The following two lemmas play an important role in proving our main result.

Lemma 4.7.1. *[54, Theorem 2] There are explicit constant* $C_0 > 0, \epsilon_0 = \text{poly}(\alpha_0, \alpha_1, \phi, \nu, n, m)$ *such that, for any* $\epsilon \in (0, \epsilon_0)$ *and matrices* $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$ *such that* $\|A-A_*\| \leq \epsilon$ *and* $\|B-B_*\| \leq \epsilon$, *the policy* $K = \text{dare}(A, B, Q, R)$ *satisfies*

$$J(K) - J_* \leq C_0\epsilon^2 \qquad \text{and} \qquad \|K - K_*\| \leq C_0\epsilon. \tag{4.8}$$

This lemma is used to bound the regret, when the estimated matrices $(A_{\tau_i}\ B_{\tau_i})$ are in a small neighbourhood of the true pair $(A_*\ B_*)$. The next lemma is used to get a bound on the difference of $(A_{\tau_i}\ B_{\tau_i})$ and $(A_*\ B_*)$.

Lemma 4.7.2. *[2, Theorem 1] Let $\{\mathcal{F}_t\}_{t=0}^{\infty}$ be a filtration and let $\{\xi_t\}_{t=1}^{\infty}$ be a real-valued martingale difference sequence adapted to this filtration such that ξ_t is S-sub-Gaussian conditioned on $\mathcal{F}_{t-1}$, that is,*

$$\mathbb{E}[e^{\lambda\xi_t}|\mathcal{F}_{t-1}] \leq e^{\lambda^2 S^2/2}, \qquad t \geq 1. \tag{4.9}$$

Further, let $\{u_t\}_{t=1}^{\infty}$ be an $\mathbb{R}^n$-valued stochastic process adapted to $\{\mathcal{F}_{t-1}\}_{t=1}^{\infty}$, let $V \in \mathbb{R}^{n\times n}$ be a positive-definite matrix, and define

$$U_t = \sum_{s=1}^{t-1} \xi_s u_s, \qquad V_t = V + \sum_{s=1}^{t-1} u_s u_s^\top, \qquad t \geq 1. \tag{4.10}$$

Then, for any $\delta \in (0,1)$, we have that with probability at least $1-\delta$,

$$U_t^\top V_t^{-1} U_t \leq 2S^2 \log\Big(\frac{1}{\delta}\frac{\det(V_t)}{\det(V)}\Big), \qquad t \geq 1. \tag{4.11}$$

Throughout, we denote

$$\Delta_t = (A_t\ B_t) - (A_*\ B_*), \tag{4.12}$$

and $z_s = (x_s^\top\ u_s^\top)^\top$. We also define $W_t \in \mathbb{R}^{(n+m)\times(n+m)}$ by

$$\begin{aligned} W_t &= \sum_{s=1}^{t-1} z_s z_s^\top + \lambda I_{n+m} \\ &= \begin{pmatrix} \sum_{s=1}^{t-1} x_s x_s^\top & \sum_{s=1}^{t-1} x_s u_s^\top \\ \sum_{s=1}^{t-1} u_s x_s^\top & \sum_{s=1}^{t-1} u_s u_s^\top \end{pmatrix} + \lambda I_{n+m}, \end{aligned}$$

and

$$V_{\tau_i} = \sum_{s=1}^{\tau_i-1} x_s x_s^\top + \lambda I_n, \tag{4.13}$$
$$Y_{\tau_i} = \sum_{s=1}^{\tau_i-1} u_s u_s^\top + \lambda I_m - (\sum_{s=1}^{\tau_i-1} u_s x_s^\top) V_{\tau_i}^{-1} (\sum_{s=1}^{\tau_i-1} x_s u_s^\top),$$

and

$$\widehat{W}_{\tau_i} = \begin{pmatrix} \sum_{s=1}^{\tau_i-1} x_s x_s^\top & \sum_{s=1}^{\tau_i-1} x_s u_s^\top \\ \sum_{s=1}^{\tau_i-1} u_s x_s^\top & \sum_{s=1}^{\tau_i-1} u_s u_s^\top + \frac{\gamma_i}{1-\gamma_i} Y_{\tau_i} \end{pmatrix} + \lambda I_{n+m}. \tag{4.14}$$

The next proposition states the relation of $(A_{\tau_i}\ B_{\tau_i})$ with the history of observed data $\{(x_s, u_s)\}_{s=1}^{\tau_i-1}$ collected as $\widehat{W}_{\tau_i}$.

Proposition 4.7.3. *Let $\{(x_t, u_t)\}_{t=1}^{T}$ be the sequence of states and actions of the system (4.1), generated using the Algorithm 2. If $(A_{\tau_i}\ B_{\tau_i})$ are generated by Algorithm 2, then we have*

$$(A_{\tau_i}\ B_{\tau_i}) = (A_*\ B_*) - \lambda (A_*\ B_*) \widehat{W}_{\tau_i}^{-1} + \big(\sum_{s=1}^{\tau_i-1} w_s (x_s^\top\ u_s^\top)\big) \widehat{W}_{\tau_i}^{-1} + \big(0\ \frac{1}{1-\gamma_i} E_i Y_{\tau_i}\big) \widehat{W}_{\tau_i}^{-1}, \tag{4.15}$$

where $\widehat{W}_{\tau_i}$ is given in (4.14).

Proof. Let $(\widehat{A}_{\tau_i}, \widehat{B}_{\tau_i})$ be generated by Algorithm 2. From Line 7 of the Algorithm 2, we have the following equality [2],

$$(\widehat{A}_{\tau_i}\ \widehat{B}_{\tau_i}) = \text{argmin}_{(A\ B)} \sum_{s=1}^{\tau_i-1} \|x_{s+1} - (A\ B)(x_s^\top\ u_s^\top)^\top\|^2 + \lambda \|(A\ B)\|_F^2$$

$$= \sum_{s=1}^{\tau_i-1} (x_{s+1}x_s^\top \; x_{s+1}u_s^\top) W_{\tau_i}^{-1}, \tag{4.16}$$

where

$$W_{\tau_i} = \begin{pmatrix} (\sum_{s=1}^{\tau_i-1} x_s x_s^\top) + \lambda I_n & \sum_{s=1}^{\tau_i-1} x_s u_s^\top \\ \sum_{s=1}^{\tau_i-1} u_s x_s^\top & (\sum_{s=1}^{\tau_i-1} u_s u_s^\top) + \lambda I_m \end{pmatrix}. \tag{4.17}$$

In order to write the inverse of W_{τ_i} as a block matrix for further computation, we define

$$\begin{aligned} V_{\tau_i} &= \sum_{s=1}^{\tau_i-1} x_s x_s^\top + \lambda I_n, \\ Y_{\tau_i} &= \sum_{s=1}^{\tau_i-1} u_s u_s^\top + \lambda I_m - (\sum_{s=1}^{\tau_i-1} u_s x_s^\top) V_{\tau_i}^{-1} (\sum_{s=1}^{\tau_i-1} x_s u_s^\top), \end{aligned}$$

and using block matrix inverse formula [51], we obtain

$$W_{\tau_i}^{-1} = \begin{pmatrix} V_{\tau_i}^{-1} + V_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} x_s u_s^\top) Y_{\tau_i}^{-1} (\sum_{s=1}^{\tau_i-1} u_s x_s^\top) V_{\tau_i}^{-1} & -V_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} x_s u_s^\top) Y_{\tau_i}^{-1} \\ -Y_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} u_s x_s^\top) V_{\tau_i}^{-1} & Y_{\tau_i}^{-1} \end{pmatrix}. \tag{4.18}$$

Using (4.16), (4.18), and (4.7), we have that

$$\begin{aligned} B_{\tau_i} =& (1-\gamma_i)\widehat{B}_{\tau_i} + \gamma_i B_* + E_i \\ =& -(1-\gamma_i)(\sum_{s=1}^{\tau_i-1} x_{s+1}x_s^\top) V_{\tau_i}^{-1} (\sum_{s=1}^{\tau_i-1} x_s u_s^\top) Y_{\tau_i}^{-1} + (1-\gamma_i)(\sum_{s=1}^{\tau_i-1} x_{s+1}u_s^\top) Y_{\tau_i}^{-1} \\ &+ \gamma_i B_* + E_i. \end{aligned} \tag{4.19}$$

Now for A_{τ_i}, we have

$$\begin{aligned}
A_{\tau_i} &= \text{argmin}_A \sum_{s=1}^{\tau_i-1} \|x_{s+1} - Ax_s - B_{\tau_i}u_s\|^2 + \lambda\|A\|_F^2 \\
&= \Big(\sum_{s=1}^{\tau_i-1}(x_{s+1} - B_{\tau_i}u_s)x_s^\top\Big)\Big(\sum_{s=1}^{\tau_i-1} x_s x_s^\top + \lambda I_n\Big)^{-1} \\
&= (\sum_{s=1}^{\tau_i-1} x_{s+1}x_s^\top)V_{\tau_i}^{-1} - B_{\tau_i}(\sum_{s=1}^{\tau_i-1} u_s x_s^\top)V_{\tau_i}^{-1} \\
&= (\sum_{s=1}^{\tau_i-1} x_{s+1}x_s^\top)V_{\tau_i}^{-1} + (1-\gamma_i)(\sum_{s=1}^{\tau_i-1} x_{s+1}x_s^\top)V_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} x_s u_s^\top)Y_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} u_s x_s^\top)V_{\tau_i}^{-1} \\
&\quad - (1-\gamma_i)(\sum_{s=1}^{\tau_i-1} x_{s+1}u_s^\top)Y_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} u_s x_s^\top)V_{\tau_i}^{-1} - (\gamma_i B_* + E_i)(\sum_{s=1}^{\tau_i-1} u_s x_s^\top)V_{\tau_i}^{-1},
\end{aligned} \tag{4.20}$$

where the first equality follows by the Line 9 of the Algorithm 2, the third equality follows by (4.13), and the last equality follows by plugging in B_{τ_i} from (4.19). Putting (4.20) and (4.19) together, we have

$$(A_{\tau_i}\ B_{\tau_i}) = \Big(\sum_{s=1}^{\tau_i-1} x_{s+1}(x_s^\top\ u_s^\top)\Big)Z_{\tau_i} + \Big(-(\gamma_i B_* + E_i)(\sum_{s=1}^{\tau_i-1} u_s x_s^\top)V_{\tau_i}^{-1}\ \ \gamma_i B_* + E_i\Big), \tag{4.21}$$

where

$$Z_{\tau_i} = \begin{pmatrix} V_{\tau_i}^{-1} + (1-\gamma_i)V_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} x_s u_s^\top)Y_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} u_s x_s^\top)V_{\tau_i}^{-1} & -(1-\gamma_i)V_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} x_s u_s^\top)Y_{\tau_i}^{-1} \\ -(1-\gamma_i)Y_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} u_s x_s^\top)V_{\tau_i}^{-1} & (1-\gamma_i)Y_{\tau_i}^{-1} \end{pmatrix}. \tag{4.22}$$

After some simplifications, we obtain

$$(A_{\tau_i}\ B_{\tau_i}) = \big(\sum_{s=1}^{\tau_i-1} x_{s+1}x_s^\top\ \ \sum_{s=1}^{\tau_i-1} x_{s+1}u_s^\top + \frac{1}{1-\gamma_i}(\gamma_i B_* + E_i)Y_{\tau_i}\big)Z_{\tau_i}.$$

Now using (4.1), we have

$$
\begin{aligned}
&(A_{\tau_i}\ B_{\tau_i}) \\
&= \big(\sum_{s=1}^{\tau_i-1}(A_*x_s + B_*u_s + w_s)x_s^\top \quad \sum_{s=1}^{\tau_i-1}(A_*x_s + B_*u_s + w_s)u_s^\top + \frac{1}{1-\gamma_i}(\gamma_i B_* + E_i)Y_{\tau_i}\big)Z_{\tau_i} \\
&= (A_*\ B_*)\begin{pmatrix} \sum_{s=1}^{\tau_i-1} x_s x_s^\top & \sum_{s=1}^{\tau_i-1} x_s u_s^\top \\ \sum_{s=1}^{\tau_i-1} u_s x_s^\top & \sum_{s=1}^{\tau_i-1} u_s u_s^\top + \frac{\gamma_i}{1-\gamma_i}Y_{\tau_i} \end{pmatrix} Z_{\tau_i} + \big(\sum_{s=1}^{\tau_i-1} w_s(x_s^\top\ u_s^\top)\big)Z_{\tau_i} \\
&\quad + \big(0\ \frac{1}{1-\gamma_i}E_iY_{\tau_i}\big)Z_{\tau_i}. \qquad (4.23)
\end{aligned}
$$

For further simplifications, let

$$
\widehat{W}_{\tau_i} = \begin{pmatrix} \sum_{s=1}^{\tau_i-1} x_s x_s^\top & \sum_{s=1}^{\tau_i-1} x_s u_s^\top \\ \sum_{s=1}^{\tau_i-1} u_s x_s^\top & \sum_{s=1}^{\tau_i-1} u_s u_s^\top + \frac{\gamma_i}{1-\gamma_i}Y_{\tau_i} \end{pmatrix} + \lambda I_{n+m}, \qquad (4.24)
$$

and by computing the inverse of the $\widehat{W}_{\tau_i}$, we obtain

$$
\widehat{W}_{\tau_i}^{-1} = \begin{pmatrix} V_{\tau_i}^{-1} + V_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} x_s u_s^\top)\widehat{Y}_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} u_s x_s^\top)V_{\tau_i}^{-1} & -V_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} x_s u_s^\top)\widehat{Y}_{\tau_i}^{-1} \\ -\widehat{Y}_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} u_s x_s^\top)V_{\tau_i}^{-1} & \widehat{Y}_{\tau_i}^{-1} \end{pmatrix}, \qquad (4.25)
$$

where V_{τ_i} is given by (4.13) and $\widehat{Y}_{\tau_i}$ is the Schur complement of matrix $\widehat{W}_{\tau_i}$, given by

$$
\begin{aligned}
\widehat{Y}_{\tau_i} &= \sum_{s=1}^{\tau_i-1} u_s u_s^\top + \lambda I_m + \frac{\gamma_i}{1-\gamma_i}Y_{\tau_i} - (\sum_{s=1}^{\tau_i-1} u_s x_s^\top)V_{\tau_i}^{-1}(\sum_{s=1}^{\tau_i-1} x_s u_s^\top) \\
&= (1 + \frac{\gamma_i}{1-\gamma_i})Y_{\tau_i} = \frac{1}{1-\gamma_i}Y_{\tau_i}.
\end{aligned}
$$

Using this, we conclude that $\widehat{Y}_{\tau_i}^{-1} = (1-\gamma_i)Y_{\tau_i}^{-1}$, and

$$Z_{\tau_i} = \widehat{W}_{\tau_i}^{-1}. \tag{4.26}$$

Plugging (4.26) into (4.23), we conclude that

$$(A_{\tau_i}\ B_{\tau_i}) = (A_*\ B_*) - \lambda(A_*\ B_*)\widehat{W}_{\tau_i}^{-1} + \big(\sum_{s=1}^{\tau_i-1} w_s(x_s^\top\ \ u_s^\top)\big)\widehat{W}_{\tau_i}^{-1} + \big(0\ \frac{1}{1-\gamma_i}E_iY_{\tau_i}\big)\widehat{W}_{\tau_i}^{-1}.$$

□

Lemma 4.7.4. *Let $\{(x_t,u_t)\}_{t=1}^T$ be the sequence of states and actions of the system (4.1), and let (A_{τ_i}, B_{τ_i}) be the corresponding pair generated by the Algorithm 2. Then we have, with probability $1-\delta$,*

$$\mathrm{Tr}(\Delta_{\tau_i}\widehat{W}_{\tau_i}\Delta_{\tau_i}^\top) \le 6n\sigma^2\log\Big(\frac{n}{\delta}\frac{\det(W_{\tau_i})}{\det(\lambda I)}\Big) + 3\lambda\|(A_\star\ B_\star)\|_F^2 + \frac{3}{1-\gamma_i}\mathrm{Tr}(E_iY_{\tau_i}E_i^\top),$$

where Δ_{τ_i} is defined by (4.12).

Proof. The proof strategy is similar to [20, Lemma 6]. Let

$$U_{\tau_i} = \sum_{s=1}^{\tau_i-1} w_s(x_s^\top\ \ u_s^\top).$$

Using Proposition 4.7.3 and by multiplying both side of (4.15) by $\widehat{W}_{\tau_i}$ and then multiplying each side by the transpose of corresponding side of (4.15), we have that

$$\mathrm{Tr}(\Delta_{\tau_i}\widehat{W}_{\tau_i}\Delta_{\tau_i}^\top) \le 3\mathrm{Tr}(U_{\tau_i}\widehat{W}_{\tau_i}^{-1}U_{\tau_i}^\top) + 3\lambda^2\mathrm{Tr}((A_\star\ B_\star)\widehat{W}_{\tau_i}^{-1}(A_\star\ B_\star)^\top) + \frac{3}{1-\gamma_i}\mathrm{Tr}(E_iY_{\tau_i}E_i^\top). \tag{4.27}$$

Note that

$$\widehat{W}_{\tau_i} - W_{\tau_i} = \begin{pmatrix} 0 & 0 \\ 0 & \frac{\gamma_i}{1-\gamma_i} Y_{\tau_i} \end{pmatrix} \succeq 0, \tag{4.28}$$

since Y_{τ_i} is the Schur complement of $W_{\tau_i} \succeq \lambda I$ and is positive definite, and $0 < \gamma_i < 1$. Hence, we have $\widehat{W}_{\tau_i} \succeq W_{\tau_i}$ and therefore $\mathrm{Tr}(U_{\tau_i} \widehat{W}_{\tau_i}^{-1} U_{\tau_i}^\top) \leq \mathrm{Tr}(U_{\tau_i} W_{\tau_i}^{-1} U_{\tau_i}^\top)$. Thus

$$\begin{aligned} \mathrm{Tr}(\Delta_{\tau_i} \widehat{W}_{\tau_i} \Delta_{\tau_i}^\top) \leq & 3\mathrm{Tr}(U_{\tau_i} W_{\tau_i}^{-1} U_{\tau_i}^\top) + 3\lambda^2 \mathrm{Tr}((A_\star \ B_\star) \widehat{W}_{\tau_i}^{-1} (A_\star \ B_\star)^\top) \\ & + \frac{3}{1-\gamma_i} \mathrm{Tr}(E_i Y_{\tau_i} E_i^\top) \\ \leq & 3\mathrm{Tr}(U_{\tau_i} W_{\tau_i}^{-1} U_{\tau_i}^\top) + 3\lambda \|(A_\star \ B_\star)\|_F^2 + \frac{3}{1-\gamma_i} \mathrm{Tr}(E_i Y_{\tau_i} E_i^\top), \end{aligned} \tag{4.29}$$

where we have used $\widehat{W}_{\tau_i} \succeq \lambda I$ in the last inequality. In order to bound the first term, we use Theorem 4.7.2. Let $U_t(i) = \sum_{s=1}^{t-1} w_s(i)(x_s^\top \ u_s^\top)$ for all $i = 1, \ldots, n$. For each i, by Theorem 4.7.2, with probability $1 - \delta/n$, we have

$$U_t(i) W_t^{-1} U_t^\top(i) \leq 2\sigma^2 \log\Big(\frac{n}{\delta} \frac{\det(W_t)}{\det(\lambda I_{n+m})}\Big). \tag{4.30}$$

From this, by applying the union bound over the events for which (4.30) is satisfied, we have, with probability $1 - \delta$,

$$\mathrm{Tr}(U_t W_t^{-1} U_t^\top) = \sum_{i=1}^{n} U_t(i) W_t^{-1} U_t^\top(i) \leq 2n\sigma^2 \log\Big(\frac{n}{\delta} \frac{\det(W_t)}{\det(\lambda I_{n+m})}\Big),$$

for all $i = 1, \ldots, n$. The result now follows by plugging this into (4.29). □

We recall the following result.

Lemma 4.7.5. *[20, Theorem 20] Let $\{z_t\}_{t=1}^\infty$ be a sequence of random variables*

adapted to a filtration $\{\mathcal{F}_t\}_{t=0}^{\infty}$. *Suppose that the* z_t *are conditionally Gaussian given* $\mathcal{F}_{t-1}$ *and that* $\mathbb{E}[z_t z_t^\top | \mathcal{F}_{t-1}] \succeq \sigma^2 I$ *for some fixed* $\sigma^2 > 0$. *Then for* $t \geq 200n \log \frac{12}{\delta}$ *we have, with probability at least* $1 - \delta$,

$$\sum_{s=1}^{t} z_s z_s^\top \succeq \frac{t\sigma^2}{40} I. \tag{4.31}$$

Lemma 4.7.6. *Let* $K \in \mathbb{R}^{m \times n}$ *be such that* $\|K\| \leq k$ *and let* $p > 0$. *Then*

$$\begin{pmatrix} I_n & K^\top \\ K & KK^\top + pI_m \end{pmatrix} \succeq \frac{1}{(1+k^2)/p + 1} I_{n+m}. \tag{4.32}$$

Proof. Since $\|K\| \leq k$, we have $KK^\top \preceq k^2 I_m$, and hence

$$\begin{aligned} KK^\top &\preceq k^2 I_m + \frac{p}{1+k^2+p} I_m \\ &= \frac{(1+k^2)(p+k^2)}{1+k^2+p} I_m. \end{aligned}$$

Multiplying both sides by $\frac{-p}{1+k^2}$, we obtain

$$\begin{aligned} \frac{-p(p+k^2)}{1+k^2+p} I_m &\preceq \frac{-p}{1+k^2} KK^\top \\ &= KK^\top - KK^\top (1 - \frac{p}{p+1+k^2})^{-1}, \end{aligned} \tag{4.33}$$

now rearranging (4.33) gives

$$KK^\top + (p - \frac{p}{1+k^2+p}) I_m - KK^\top (1 - \frac{p}{p+1+k^2})^{-1} \succeq 0.$$

Note that the left-hand side is the Schur complement of

$$\begin{pmatrix} (1-\frac{p}{p+1+k^2})I_n & K^\top \\ K & KK^\top + (p-\frac{p}{1+k^2+p})I_m \end{pmatrix}. \tag{4.34}$$

Given now the fact that $(1-\frac{p}{p+1+k^2})I_n$ is positive-definite, the claim follows. □

The next result is used throughout this chapter.

Lemma 4.7.7. *[20, Lemma 41] Suppose $J(K) < J$, then K is (k,ℓ)-strongly stable with $k = \frac{J}{\alpha_0\sigma^2}$ and $\ell = \frac{\alpha_0\sigma^2}{2J}$*

Lemma 4.7.5 and Lemma 4.7.6 are used to obtain the next result.

Lemma 4.7.8. *Let $\{(x_t,u_t)\}_{t=1}^{\tau_1}$ be the sequence of states and actions of the system (4.1), and let $\tau_1 \geq 200n\log\frac{12}{\delta}$. Then we have, with probability $1-\delta$,*

$$W_{\tau_1} \succeq \frac{\tau_1\sigma^2}{40(2+k^2)}I_{n+m}. \tag{4.35}$$

Proof. Let $z_t = (x_t\ u_t)$, for $t = 1,\ldots,\tau_1-1$, and consider $u_t = K_0x_t + \eta_t$, where K_0 is a stabilizing feedback gain, and $\eta_t \sim \mathcal{N}(0,\sigma^2I_m)$. Given that $J(K_0) \leq \nu$, by Lemma 4.7.7, K_0 is (k,ℓ)-strongly stable, with $k = \frac{\nu}{\alpha_0\sigma^2}$ and $\ell = \frac{1}{2k^2}$. Now we can

write

$$\begin{aligned}
\mathbb{E}[z_t z_t^\top | \mathcal{F}_{t-1}] &= \begin{pmatrix} \mathbb{E}[x_s x_s^\top | \mathcal{F}_{t-1}] & \mathbb{E}[x_s u_s^\top | \mathcal{F}_{t-1}] \\ \mathbb{E}[u_s x_s^\top | \mathcal{F}_{t-1}] & \mathbb{E}[u_s u_s^\top | \mathcal{F}_{t-1}] \end{pmatrix} \\
&= \begin{pmatrix} \mathbb{E}[x_s x_s^\top | \mathcal{F}_{t-1}] & \mathbb{E}[x_s x_s^\top | \mathcal{F}_{t-1}] K_0^\top \\ K_0 \mathbb{E}[x_s x_s^\top | \mathcal{F}_{t-1}] & K_0 \mathbb{E}[x_s x_s^\top | \mathcal{F}_{t-1}] K_0^\top + \mathbb{E}[\eta_t \eta_t^\top | \mathcal{F}_{t-1}] \end{pmatrix} \\
&= \begin{pmatrix} I_n \\ K_0 \end{pmatrix} \mathbb{E}[x_s x_s^\top | \mathcal{F}_{t-1}] \begin{pmatrix} I_n & K_0 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & \mathbb{E}[\eta_t \eta_t^\top | \mathcal{F}_{t-1}] \end{pmatrix} \\
&\succeq \sigma^2 \begin{pmatrix} I_n & K_0^\top \\ K_0 & K_0 K_0^\top + I_m \end{pmatrix},
\end{aligned}$$

where we have used the facts that $\mathbb{E}[x_s x_s^\top | \mathcal{F}_{t-1}] \succeq \mathbb{E}[w_s w_s^\top | \mathcal{F}_{t-1}] = \sigma^2 I_n$ and $\mathbb{E}[\eta_t \eta_t^\top | \mathcal{F}_{t-1}] = \sigma^2 I_m$. Now using $\|K_0\| \leq k$ and applying Lemma 4.7.6, we have that

$$\mathbb{E}[z_t z_t^\top | \mathcal{F}_{t-1}] \succeq \frac{\sigma^2}{2+k^2} I_{n+m}$$

Given that $W_{\tau_1} = \lambda I_{n+m} + \sum_{s=1}^{\tau_1 - 1} z_s z_s^\top$, and using Lemma 4.7.5, with probability $1-\delta$, we have

$$W_{\tau_1} \succeq \lambda I_{n+m} + \frac{(\tau_1 - 1)\sigma^2}{40(2+k^2)} I_{n+m} \succeq \frac{\tau_1 \sigma^2}{40(2+k^2)} I_{n+m}, \tag{4.36}$$

as claimed. □

The next lemmas are used in the proof of Theorem 4.6.1.

Lemma 4.7.9. *[20, Lemma 34] Let $w_t \in \mathbb{R}^n$ for $t = 1, \ldots, T$ be i.i.d. random*

variables with distribution $\mathcal{N}(0, \sigma^2 I_n)$. *Suppose that* $T > 2$. *Then with probability at least* $1 - \delta$ *we have that*

$$\max_{1 \le t \le T} \|w_t\| \le \sigma \sqrt{5n \log \frac{T}{\delta}} \tag{4.37}$$

Lemma 4.7.10. *[20, Lemma 37] Let* $z_s \in \mathbb{R}^m$ *for* $s = 1, \ldots, t-1$ *be such that* $\|z_s\|^2 \le \lambda$ *and define* $W_t = \lambda I + \sum_{s=1}^{t-1} z_s z_s^\top$. *Then we have that*

$$\log \frac{\det(W_t)}{\det(\lambda I)} \le m \log t. \tag{4.38}$$

Lemma 4.7.11. *[20, Lemma 38] Suppose* K *is* (k, ℓ)*-strongly stable controller and* s_0, s_1 *are integers such that* $1 \le s_0 < s_1 \le T$. *Let* x_s *for* $s = s_0, \ldots, s_1$ *be the sequence of states generated under the controller* K *starting from* x_{s_0}. *Then we have*

$$\|x_t\| \le k(1-l)^{t-s_0} \|x_{s_0}\| + \frac{k}{l} \max_{1 \le t \le T} \|w_t\|, \qquad \text{for all } s_0 \le t \le s_1. \tag{4.39}$$

Lemma 4.7.12. *[20, Lemma 39] Suppose* $K_1, \ldots, K_r$ *are* (k, ℓ)*-strongly stable feedback gains and* $\{t_i\}_{i=1}^{r+1}$ *are integers such that* $1 \le t_1 < \ldots < t_{r+1} \le T$. *For each* t_i *Let* $\{x_t\}$ *be the sequence of states generated by starting from* x_{t_i} *and playing controller* K_i *at times* $t_i \le t < t_{i+1}$, *i.e.,* $x_{t+1} = (A_* + B_* K_i) x_t + w_t$ *for all* $t_i \le t < t_{i+1}$. *Denote* $\tau = \min_i \{t_{i+1} - t_i\}$ *and suppose that* $\tau \ge l^{-1} \log(2k)$. *Then we have*

$$\|x_t\| \le 3k \max \left\{ \frac{1}{2} \|x_{t_1}\|, \frac{k}{l} \max_{1 \le s \le T} \|w_s\| \right\}, \quad \text{for all } t_1 \le t \le t_{r+1}$$

Lemma 4.7.13. *[20, Lemma 40] Suppose* K *is a* (k, ℓ)*-strongly stable controller and let* x_s *for* $s = 1, \ldots, t$ *be the sequence of states generated under the control* K *starting from* x_1, *i.e.,* $x_{s+1} = (A_* + B_* K) x_s + w_s$ *for all* $1 \le s \le t$. *Then we have that*

$\mathbb{E}\Big[\sum_{s=1}^{t} x_s^\top (Q + K^\top R K) x_s \Big| x_1\Big] \leq tJ(K) + \frac{2\alpha_1 k^4}{l}\|x_1\|^2.$

Lemma 4.7.14. *[20, Lemma 42] Let $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$ and denote $\Delta = \|(A - A_\star \quad B - B_\star)\|$. Taking $K = \mathrm{dare}(A, B, Q, R)$ and denoting $k = \sqrt{\frac{\nu + C_0\epsilon_0^2}{\alpha_0 \sigma^2}}$, and $l = \frac{1}{2k^2}$ where $J_\star \leq \nu$, and $\min\{Q, R\} \geq \alpha_0$, we have that If $\Delta \leq \epsilon_0$ then K is (k, ℓ)-strongly stable.*

Lemma 4.7.15. *Let X_1 and X_2 be two positive definite block matrices as follows.*

$$X_1 = \begin{pmatrix} A_1 & B_1 \\ B_1^\top & C_1 \end{pmatrix} \qquad X_2 = \begin{pmatrix} A_2 & B_2 \\ B_2^\top & C_2 \end{pmatrix},$$

If $X_1 - X_2$ is positive semi-definite, then we have that

$$\big(C_1 - B_1^\top A_1^{-1} B_1\big) - \big(C_2 - B_2^\top A_2^{-1} B_2\big) \succeq 0$$

Proof. Note that for a positive definite block matrix we have that

$$\begin{pmatrix} x^\top & y^\top \end{pmatrix} \begin{pmatrix} A & B \\ B^\top & C \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = (x + A^{-1}By)^\top A (x + A^{-1}By) + y^\top (C - B^\top A^{-1} B) y,$$

for all real vectors $(x\ y)$. Given this and by assumption $X_1 - X_2 \succeq 0$, we have that

$$(x + A_1^{-1}B_1 y)^\top A_1 (x + A_1^{-1}B_1 y) + y^\top (C_1 - B_1^\top A_1^{-1} B_1) y \geq$$
$$(x + A_2^{-1}B_2 y)^\top A_2 (x + A_2^{-1}B_2 y) + y^\top (C_2 - B_2^\top A_2^{-1} B_2) y,$$

Now for any y and choosing $x = -A_1^{-1}B_1 y$, we have that

$$\begin{aligned} y^\top (C_1 - B_1^\top A_1^{-1} B_1) y &\geq (-A_1^{-1}B_1 y + A_2^{-1}B_2 y)^\top A_2 (-A_1^{-1}B_1 y + A_2^{-1}B_2 y) \\ &\quad + y^\top (C_2 - B_2^\top A_2^{-1} B_2) y \\ &\geq y^\top (C_2 - B_2^\top A_2^{-1} B_2) y \end{aligned}$$

where we have used the fact that A_2 is positive definite. This implies the result.

□

In order to find the bound on the expected value of the regret, we consider an event with high probability on which the growth of the regret is small. For this event, the estimated parameters of the system are close to the true system parameters, and the feedback gains generated by the algorithm is strongly stable. We give the specifics of this.

We define the following events:

$$\mathcal{E}_x = \Big\{ \sum_{t=\tau_i}^{\tau_{i+1}-1} x_s x_s^\top \succeq \frac{(\tau_{i+1} - \tau_i)\sigma^2}{40} I_n, \text{ for } 1 \leq i \leq n_T \Big\}, \tag{4.40a}$$

$$\mathcal{E}_W = \Big\{ W_{\tau_1} \succeq \frac{\tau_1 \sigma^2}{40(2+k^2)} I_{n+m} \Big\}, \tag{4.40b}$$

$$\mathcal{E}_w = \Big\{ \max_{1 \leq t \leq T} \|w_t\| \leq \sigma \sqrt{15 n \log 4T} \Big\}, \tag{4.40c}$$

$$\mathcal{E}_\eta = \Big\{ \max_{1 \leq t \leq T} \|\eta_t\| \leq \sigma \sqrt{15 n \log 4T} \Big\}, \tag{4.40d}$$

$$\begin{aligned} \mathcal{E}_\Delta = \Big\{ \mathrm{Tr}(\Delta_{\tau_i} \widehat{W}_{\tau_i} \Delta_{\tau_i}^\top) &\leq 6n\sigma^2 \log \Big(4T^3 \frac{\det(W_{\tau_i})}{\det(\lambda I)} \Big) + 3\lambda \|(A_\star, B_\star)\|_F^2 \\ &\quad + \frac{3}{1-\gamma_i} \mathrm{Tr}(E_i Y_{\tau_i} E_i^\top), \quad \text{for } i = 1, \ldots, n_T \Big\}. \end{aligned} \tag{4.40e}$$

Lemma 4.7.16. *Let* $\mathcal{E} = \mathcal{E}_x \cap \mathcal{E}_W \cap \mathcal{E}_w \cap \mathcal{E}_\eta \cap \mathcal{E}_\Delta$, *and* $\tau_1 \geq 600 n \log 48T$. *Then*

$\mathbb{P}(\mathcal{E}) \geq 1 - T^{-2}$.

Proof. Let $\delta = \frac{1}{4}T^{-3}$. Using Lemma 4.7.8, we have that for $\tau_1 \geq 600n \log 48T$, $\mathbb{P}(\mathcal{E}_W) \geq 1 - \frac{1}{4}T^{-3}$. Note that $\tau_{i+1} - \tau_i \geq \tau_1$, and hence using Lemma 4.7.5 n_T times with $n_{T+1} \leq T$ and taking a union bound, we get $\mathbb{P}(\mathcal{E}_x \cap \mathcal{E}_W) \geq 1 - \frac{1}{4}T^{-2}$.

Next, by Lemma 4.7.9 and letting $\delta = \frac{1}{4}T^{-2}$ with probability $1 - \frac{1}{4}T^{-2}$, we have that $\max_{1 \leq t \leq T} \|w_t\| \leq \sigma\sqrt{15n \log 4T}$. Similarly, $\max_{1 \leq t \leq T} \|\eta_t\| \leq \sigma\sqrt{15n \log 4T}$.

For the last part, we use Lemma 4.7.4 by selecting $\delta = \frac{1}{4}T^{-2}$. By taking a union bound, we obtain the result.

□

Lemma 4.7.17. *Let $\{(x_t, u_t)\}_{t=1}^T$ be the sequence of states and actions of the system (4.1) generated by Algorithm 2. Then on the event $\mathcal{E}_w \cap \mathcal{E}_\eta$, we have $\|x_t\| \leq \sigma k^3(1+\phi)\sqrt{60n \log 4T}$ and $\|u_t\| \leq k^4(2+\phi)\sqrt{60n \log 4T}$, for all $1 \leq t \leq \tau_1 - 1$.*

Proof. For $t = 1, \ldots, \tau_1 - 1$, the controller uses $u_t = K_0 x_t + \eta_t$. By plugging this into (4.1), and defining $\tilde{w}_t = w_t + B_* \eta_t$, we have

$$x_{t+1} = A_* x_t + B_* K_0 x_t + B_* \eta_t + w_t = (A_* + B_* K_0) x_t + \tilde{w}_t. \tag{4.41}$$

By the assumption $J(K_0) \leq \nu$, and applying Lemma 4.7.7, we have that K_0 is (k, ℓ)-strongly stable with $k = \frac{\nu}{\alpha_0 \sigma^2}$ and $\ell = 1/2k^2$. Applying now Lemma 4.7.11 with $x_0 = 0$, we conclude that

$$\|x_t\| \leq 2k^3 \max_{1 \leq s \leq T} \|\tilde{w}_s\|. \qquad 0 \leq t \leq \tau_1 - 1 \tag{4.42}$$

Now, we can bound $\max_{1\le s\le T}\|\tilde{w}_s\|$ on the event $\mathcal{E}_w \cap \mathcal{E}_\eta$:

$$\max_{1\le s\le T}\|\tilde{w}_s\| \le \max_{1\le s\le T}\|w_s\| + \|B_*\| \max_{1\le s\le T}\|\eta_s\| \le \sigma(1+\phi)\sqrt{15n\log 4T}.$$

The bound for $\|x_t\|$ follows by plugging this into (4.42).

Using $u_t = K_0 x_t + \eta_t$, we have

$$\begin{aligned}
\|u_t\| &\le \|K_0\|\|x_t\| + \|\eta_t\| \\
&\le \sigma k^4(1+\phi)\sqrt{60n\log 4T} + \max_{1\le s\le T}\|\eta_s\| \\
&\le \sigma k^4(1+\phi)\sqrt{60n\log 4T} + \sigma\sqrt{15n\log 4T} \\
&\le \sigma k^4(2+\phi)\sqrt{60n\log 4T},
\end{aligned}$$

as claimed. □

Lemma 4.7.18. *Let $0 \le 1-\gamma_1 \le \frac{1}{r}$ and $(1-\gamma_{i+1}) \le \frac{1}{r}(1-\gamma_i)$ for all $i \ge 1$, and let $p = \frac{r}{2+k^2}$. Further assume $\|E_i\|_F^2 \le \frac{1-\gamma_i}{\tau_i}$ for all $i \ge 1$. On the event $\mathcal{E} = \mathcal{E}_x \cap \mathcal{E}_W \cap \mathcal{E}_w \cap \mathcal{E}_\eta \cap \mathcal{E}_\Delta$ we have that,*

1. *K_{τ_i} is (k,ℓ)-strongly stable, for all $1 \le i \le n_T$;*
2. *$\|x_t\|^2 \le x_b$, for all $1 \le t \le T$;*
3. *$\|\Delta_{\tau_i}\| \le \epsilon_0 r^{-i+1}$, for all $1 \le i \le n_T$;*
4. *$\widehat{W}_{\tau_i} \succeq \frac{\sigma^2\tau_i}{40((1+k^2)/\min\{p,1\}+1)} I_{n+m}$ for all $1 \le i \le n_T$,*

where

$$x_b = 135nk^2\sigma^2 \max\left\{(1+\phi)^2k^6, 4k^6\right\}\log(4T). \tag{4.43}$$

Proof. We use an induction on i. Let $i = 1$, using (4.1), for all $s \geq 1$, we have that

$$\mathbb{E}[x_s x_s^\top] \succeq \mathbb{E}[w_s w_s^\top] \succeq \sigma^2 I_n. \tag{4.44}$$

For the event $\mathcal{E}_W$, we have $W_{\tau_1} \succeq \frac{\tau_1 \sigma^2}{40(2+k^2)} I_{n+m}$. Using this, we have

$$\widehat{W}_{\tau_1} = W_{\tau_1} + \begin{pmatrix} 0 & 0 \\ 0 & \frac{\gamma_1}{1-\gamma_1} Y_{\tau_1} \end{pmatrix} \succeq \frac{\tau_1 \sigma^2}{40(2+k^2)} I_{n+m}, \tag{4.45}$$

which proves the statement 4 of Lemma 4.7.18 for $i = 1$. Next we show that

$$\|\Delta_{\tau_1}\| \leq \epsilon_0,$$

proving statement 3. First note that $\|x_t\|^2 \leq x_b$, for all $1 \leq t < \tau_1$ by Lemma 4.7.17. By defining $z_s = (x_s^\top \ u_s^\top)^\top$, and using Lemma 4.7.17 again, we have that

$$\begin{aligned} \|z_s\| &\leq \|x_s\| + \|u_s\| \leq k^3(1+\phi)\sqrt{60n \log 4T} + k^4(2+\phi)\sqrt{60n \log 4T} \\ &\leq 2k^4(2+\phi)\sqrt{60n \log 4T} \leq \sqrt{\lambda}, \end{aligned}$$

for $1 \leq s \leq \tau_1$. Then by Lemma 4.7.10, we have

$$\log \frac{\det(W_{\tau_1})}{\det(\lambda I_{n+m})} \leq (n+m) \log T. \tag{4.46}$$

Using $\|z_s\| \leq \sqrt{\lambda}$ and since $W_{\tau_1} = \lambda I_{n+m} + \sum_{s=1}^{\tau_1 - 1} z_s z_s^\top$ we obtain

$$\|W_{\tau_1}\| \leq \lambda + \sum_{s=1}^{\tau_1 - 1} \|z_s\|^2 \leq \tau_1 \lambda$$

and since Y_{τ_1} is the Schur complement of W_{τ_1}, we have $Y_{\tau_1} \leq \tau_1 \lambda$, and therefore

$$\begin{aligned}\frac{3}{1-\gamma_1}\mathrm{Tr}(E_1 Y_{\tau_1} E_1^\top) &\leq \frac{3}{1-\gamma_1}\|Y_{\tau_1}\|\mathrm{Tr}(E_1 E_1^\top) \\ &\leq \frac{3\tau_1\lambda}{1-\gamma_1}\frac{1-\gamma_1}{\tau_1} \leq 3\lambda,\end{aligned}$$

where we have used the assumption that $\mathrm{Tr}(E_1 E_1^\top) \leq \frac{1-\gamma_1}{\tau_1}$. For the event $\mathcal{E}_\Delta$, we have

$$\begin{aligned}\|\Delta_{\tau_1}\|^2 &\leq \mathrm{Tr}\Big(\Delta_{\tau_1}\Delta_{\tau_1}^\top\Big) \\ &\leq \frac{40(2+k^2)}{\tau_1\sigma^2}\mathrm{Tr}\Big(\Delta_{\tau_1}\widehat{W}_{\tau_1}\Delta_{\tau_1}^\top\Big) \\ &\leq \frac{40(2+k^2)}{\tau_1\sigma^2}\Big(6n\sigma^2\log\Big(4T^3\frac{\det(W_{\tau_1})}{\det(\lambda I)}\Big) + 6\lambda\|(A_\star\ B_\star)\|_F^2\Big),\end{aligned}$$

where we have used (4.45) in the second inequality and Lemma 4.7.4 in the last inequality. Using this and (4.46), we can write

$$\begin{aligned}\|\Delta_{\tau_1}\|^2 &\leq \frac{2+k^2}{\tau_1}\Big(240n(n+m)\log(4T^4) + \frac{240\lambda(n+m)\phi^2}{\sigma^2}\Big) \\ &\leq \frac{(2+k^2)(n+m)}{\tau_1}\Big(960n\log(4T) + \frac{240\lambda\phi^2}{\sigma^2}\Big) \\ &\leq \frac{(2+k^2)(n+m)}{\tau_1}\frac{240\lambda(1+\phi^2)}{\sigma^2} \leq \frac{\epsilon_0^2\tau_1}{\tau_1} \leq \epsilon_0^2,\end{aligned}$$

where we have used the definition of τ_1 in the last inequality.

We can now apply Lemma 4.7.14 to conclude that K_{τ_1} is (k,ℓ)-strongly stable, with $k = \sqrt{\frac{\nu+C_0\epsilon_0^2}{\alpha_0\sigma^2}}$, and $\ell = \frac{1}{2k^2}$. This proves the statement 1 for $i = 1$.

Next we prove the statements of Lemma 4.7.18 for $i = 2, \dots, n_T$. Assume that

$$\widehat{W}_{\tau_s} \succeq \frac{\sigma^2 \tau_s}{40((1+k^2)/\min\{p,1\}+1)} I_{n+m}, \tag{4.47}$$

and K_{τ_s} is (k,ℓ)-strongly stable for $s \in \{1, \dots, i\}$, we prove that this is also true for $s = i+1$.

Note that Y_{τ_1} is the Schur complement of W_{τ_1} and we have that $Y_{\tau_1} \succeq \frac{\tau_1 \sigma^2}{40(2+k^2)} I_m$. Also note that $W_{\tau_{i+1}} \succeq W_{\tau_i}$ for all $i \geq 1$, since

$$\begin{aligned} W_{\tau_{i+1}} &= W_{\tau_i} + \begin{pmatrix} \sum_{s=\tau_i}^{\tau_{i+1}-1} x_s x_s^\top & \sum_{s=\tau_i}^{\tau_{i+1}-1} x_s u_s^\top \\ \sum_{s=\tau_i}^{\tau_{i+1}-1} u_s x_s^\top & \sum_{s=\tau_i}^{\tau_{i+1}-1} u_s u_s^\top \end{pmatrix} \\ &= W_{\tau_i} + \sum_{s=\tau_i}^{\tau_{i+1}-1} \begin{pmatrix} x_s \\ u_s \end{pmatrix} \begin{pmatrix} x_s^\top & u_s^\top \end{pmatrix} \\ &\succeq W_{\tau_i}, \end{aligned}$$

and hence by Lemma 4.7.15, we have that $Y_{\tau_{i+1}} \succeq Y_{\tau_i}$.

Now for $\widehat{W}_{\tau_{i+1}}$, we have that

$$\begin{aligned} \widehat{W}_{\tau_{i+1}} &= \begin{pmatrix} \sum_{s=1}^{\tau_{i+1}-1} x_s x_s^\top + \lambda I_n & \sum_{s=1}^{\tau_{i+1}-1} x_s u_s^\top \\ \sum_{s=1}^{\tau_{i+1}-1} u_s x_s^\top & \sum_{s=1}^{\tau_{i+1}-1} u_s u_s^\top + \lambda I_m + \frac{\gamma_{i+1}}{1-\gamma_{i+1}} Y_{\tau_{i+1}} \end{pmatrix} \\ &\succeq \begin{pmatrix} \sum_{s=1}^{\tau_{i+1}-1} x_s x_s^\top + \lambda I_n & \sum_{s=1}^{\tau_{i+1}-1} x_s u_s^\top \\ \sum_{s=1}^{\tau_{i+1}-1} u_s x_s^\top & \sum_{s=1}^{\tau_{i+1}-1} u_s u_s^\top + \lambda I_m + \frac{\gamma_{i+1}}{1-\gamma_{i+1}} Y_{\tau_i} \end{pmatrix} \end{aligned}$$

$$
\begin{aligned}
&= \begin{pmatrix} \sum_{s=1}^{\tau_i-1} x_s x_s^\top + \lambda I_n & \sum_{s=1}^{\tau_i-1} x_s u_s^\top \\ \sum_{s=1}^{\tau_i-1} u_s x_s^\top & \sum_{s=1}^{\tau_i-1} u_s u_s^\top + \lambda I_m + \frac{\gamma_i}{1-\gamma_i} Y_{\tau_i} \end{pmatrix} \\
&\quad + \begin{pmatrix} \sum_{s=\tau_i}^{\tau_{i+1}-1} x_s x_s^\top & \sum_{s=\tau_i}^{\tau_{i+1}-1} x_s u_s^\top \\ \sum_{s=\tau_i}^{\tau_{i+1}-1} u_s x_s^\top & \sum_{s=\tau_i}^{\tau_{i+1}-1} u_s u_s^\top + \frac{\gamma_{i+1}-\gamma_i}{(1-\gamma_{i+1})(1-\gamma_i)} Y_{\tau_i} \end{pmatrix} \\
&\succeq \widehat{W}_{\tau_i} + \begin{pmatrix} \sum_{s=\tau_i}^{\tau_{i+1}-1} x_s x_s^\top & \sum_{s=\tau_i}^{\tau_{i+1}-1} x_s u_s^\top \\ \sum_{s=\tau_i}^{\tau_{i+1}-1} u_s x_s^\top & \sum_{s=\tau_i}^{\tau_{i+1}-1} u_s u_s^\top + \frac{\gamma_{i+1}-\gamma_i}{(1-\gamma_{i+1})(1-\gamma_i)} Y_{\tau_1} \end{pmatrix} \\
&= \widehat{W}_{\tau_i} + \begin{pmatrix} I_n \\ K_{\tau_i} \end{pmatrix} \sum_{s=\tau_i}^{\tau_{i+1}-1} x_s x_s^\top \begin{pmatrix} I_n & K_{\tau_i}^\top \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & \frac{\gamma_{i+1}-\gamma_i}{(1-\gamma_{i+1})(1-\gamma_i)} Y_{\tau_1} \end{pmatrix}
\end{aligned}
$$

where we have used the fact that $Y_{\tau_{i+1}} \succeq Y_{\tau_i}$ for the first inequality and $Y_{\tau_i} \succeq Y_{\tau_1}$ and $u_s = K_{\tau_i} x_s$ for the last inequality and the last equality, respectively. Using Lemma 4.7.5 and the fact that $\mathbb{E}[x_s x_s^\top] \succeq \sigma^2 I_n$, for the event $\mathcal{E}_x$, we have $\sum_{s=\tau_i}^{\tau_{i+1}-1} x_s x_s^\top \succeq \frac{(\tau_{i+1}-\tau_i)\sigma^2}{40} I_n$, and hence we have that

$$
\begin{aligned}
\widehat{W}_{\tau_{i+1}} &\succeq \widehat{W}_{\tau_i} + \frac{(\tau_{i+1}-\tau_i)\sigma^2}{40} \begin{pmatrix} I_n \\ K_{\tau_i} \end{pmatrix} \begin{pmatrix} I_n & K_{\tau_i}^\top \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & \frac{\gamma_{i+1}-\gamma_i}{(1-\gamma_{i+1})(1-\gamma_i)} \frac{\tau_1 \sigma^2}{40(2+k^2)} I \end{pmatrix} \\
&= \widehat{W}_{\tau_i} + \frac{\sigma^2(\tau_{i+1}-\tau_i)}{40} \begin{pmatrix} I_n & K_{\tau_i}^\top \\ K_{\tau_i} & K_{\tau_i} K_{\tau_i}^\top + \frac{\gamma_{i+1}-\gamma_i}{(1-\gamma_{i+1})(1-\gamma_i)} \frac{\tau_1}{(\tau_{i+1}-\tau_i)} \frac{1}{2+k^2} I_m \end{pmatrix}.
\end{aligned}
$$

Now given that $\tau_i = r^{i-1}\tau_1$, we have $\frac{\tau_1}{(\tau_{i+1}-\tau_i)} = \frac{1}{(r-1)r^{i-1}}$. Now let γ_i be such that

$(1-\gamma_i) \leq \frac{1}{r^i}$ and $1-\gamma_{i+1} \leq \frac{1}{r}(1-\gamma_i)$. Then we have

$$\begin{aligned}\frac{\gamma_{i+1}-\gamma_i}{(1-\gamma_{i+1})(1-\gamma_i)} &= \frac{1}{1-\gamma_{i+1}} - \frac{1}{1-\gamma_i} \\ &\geq \frac{r-1}{1-\gamma_i} \geq (r-1)r^i.\end{aligned}$$

Now by defining $p = \frac{r}{(2+k^2)}$ we have that

$$p \leq \frac{\gamma_{i+1}-\gamma_i}{(1-\gamma_{i+1})(1-\gamma_i)} \frac{\tau_1}{(\tau_{i+1}-\tau_i)} \frac{1}{2+k^2},$$

and also

$$\widehat{W}_{\tau_{i+1}} \succeq \widehat{W}_{\tau_i} + \frac{\sigma^2(\tau_{i+1}-\tau_i)}{40} \begin{pmatrix} I_n & K_{\tau_i}^\top \\ K_{\tau_i} & K_{\tau_i}K_{\tau_i}^\top + pI_m \end{pmatrix}.$$

By Lemma 4.7.6 and using the induction hypothesis (4.47), we have that

$$\begin{aligned}\widehat{W}_{\tau_{i+1}} &\succeq \frac{\sigma^2\tau_i}{40((1+k^2)/\min\{p,1\}+1)} I_{n+m} + \frac{\sigma^2(\tau_{i+1}-\tau_i)}{40((1+k^2)/p+1)} I_{n+m} \\ &\succeq \frac{\sigma^2\tau_{i+1}}{40((1+k^2)/\min\{p,1\}+1)} I_{n+m}.\end{aligned}$$

Next we show that $\|x_t\|^2 \leq x_b$, $\|u_t\|^2 \leq k^2 x_b$, and $\|z_t\|^2 \leq \lambda$ for all $t_i \leq t < t_{i+1}$, and $\|\Delta_{\tau_{i+1}}\| \leq \epsilon r^{i-1}$, where x_b is given by (4.43) and $\lambda = (1+k)^2 x_b$. First, by induction hypothesis the algorithm does not abort and the controller uses the (k,ℓ)-strongly stable feedback gain K_{τ_i} for $\tau_i \leq t < \tau_{i+1}$. We can hence use Lemma 4.7.12

to obtain

$$\|x_t\| \leq 3k \max\left\{\frac{1}{2}\|x_{\tau_1}\|, \frac{k}{\ell}\max_{1\leq s\leq T}\|w_s\|\right\}, \quad \text{for all } \tau_1 \leq t < \tau_{i+1}.$$

Letting $\ell = 1/2k^2$, and using the bound for $\|x_{\tau_1}\|$ from Lemma 4.7.17, we have

$$\|x_t\| \leq k\sigma \max\left\{k^3(1+\phi), 2k^3\right\}\sqrt{135n\log 4T} \leq \sqrt{x_b}.$$

Now for $t = \tau_i, \ldots, \tau_{i+1} - 1$, we have $u_t = K_{\tau_i}x_t$ and hence $\|u_t\| = k\sqrt{x_b}$. Therefore, for $t = \tau_i, \ldots, \tau_{i+1} - 1$, we have

$$\|z_t\| \leq \|x_t\| + \|u_t\| \leq (1+k)\sqrt{x_b} \leq \sqrt{\lambda}.$$

Using Lemma 4.7.10, we obtain

$$\log\frac{\det(W_{\tau_{i+1}})}{\det(\lambda I_{n+m})} \leq (m+n)\log T.$$

Given that $\|z_t\|^2 \leq \lambda$ and using a similar argument used for Y_{τ_1}, we have that $Y_{\tau_i} \leq \tau_i\lambda$, and hence

$$\begin{aligned}\frac{3}{1-\gamma_i}\mathrm{Tr}(E_iY_{\tau_i}E_i^\top) &\leq \frac{3}{1-\gamma_i}\|Y_{\tau_i}\|\mathrm{Tr}(E_iE_i^\top)\\ &\leq \frac{3\tau_i\lambda}{1-\gamma_i}\frac{1-\gamma_i}{\tau_i} \leq 3\lambda,\end{aligned}$$

where we have used the assumption that $\mathrm{Tr}(E_iE_i^\top) \leq \frac{1-\gamma_i}{\tau_i}$. Using a similar argument

for Δ_{τ_1}, we obtain

$$\begin{aligned}\|\Delta_{\tau_{i+1}}\|^2 &\le \frac{((1+k^2)/\min\{p,1\}+1)}{\tau_{i+1}}\Big(160n(n+m)\log(4T^4)+\frac{80\lambda(n+m)\phi^2}{\sigma^2}\Big)\\ &\le \frac{((1+k^2)/\min\{p,1\}+1)(n+m)}{\tau_{i+1}}\Big(640n\log(4T)+\frac{80\lambda\phi^2}{\sigma^2}\Big)\\ &\le \frac{((1+k^2)/\min\{p,1\}+1)(n+m)}{\tau_{i+1}}\frac{80\lambda(1+\phi)^2}{\sigma^2} \le \frac{\epsilon_0^2\tau_1}{\tau_{i+1}} \le \epsilon_0^2 r^{-i}.\end{aligned}$$

Finally, by applying Lemma 4.7.14, we conclude that $K_{\tau_{i+1}}$ is (k,ℓ)-strongly stable.

□

Proof of Theorem 4.6.1. Let $\mathcal{E} = \mathcal{E}_x \cap \mathcal{E}_W \cap \mathcal{E}_w \cap \mathcal{E}_\eta \cap \mathcal{E}_\Delta$ be the event defined in (4.40a). The regret can be written as

$$\mathbb{E}[\mathcal{R}_T] = J_1 + J_2 + J_3 - TJ_*,$$

where

$$\begin{aligned}J_1 &= \mathbb{E}\Big[\mathbf{1}\{\mathcal{E}\}\sum_{i=1}^{n_T}\sum_{t=\tau_i}^{\tau_{i+1}-1} x_t^\top Q x_t + u_t^\top R u_t\Big],\\ J_2 &= \mathbb{E}\Big[\mathbf{1}\{\mathcal{E}^c\}\sum_{t=\tau_1}^{T} x_t^\top Q x_t + u_t^\top R u_t\Big],\\ J_3 &= \mathbb{E}\Big[\sum_{t=0}^{\tau_1-1} x_t^\top Q x_t + u_t^\top R u_t\Big].\end{aligned}$$

In the following lemmas, we will bound each term.

Lemma 4.7.19. *$J_1 \le TJ_* + n_T((r-1)C_0\epsilon_0^2\tau_1 + 4\alpha_1 k^6 x_b)$, where C_0 and ϵ_0 are positive constants given in Lemma 4.7.1, and x_b is given in* (4.43).

Proof. For each $i = 1, \dots, n_T$, we define the events $\mathcal{E}_{\tau_i} = \left\{\|\Delta_{\tau_i}\| \leq \epsilon_0 r^{-i+1}\right\}$ and $\mathcal{S}_{\tau_i} = \left\{\|x_{\tau_i}\|^2 \leq x_b\right\}$. We have $\mathcal{E} \subset \mathcal{E}_{\tau_i} \cap \mathcal{S}_{\tau_i}$, and

$$\mathbf{1}\{\mathcal{E}\} \sum_{t=\tau_i}^{\tau_{i+1}-1} x_t^\top Q x_t + u_t^\top R u_t \leq \mathbf{1}\{\mathcal{E}_{\tau_i} \cap \mathcal{S}_{\tau_i}\} \sum_{t=\tau_i}^{\tau_{i+1}-1} x_t^\top (Q + K_{\tau_i}^\top R K_{\tau_i}) x_t$$

Note that $\mathcal{E}_{\tau_i}$, $\mathcal{S}_{\tau_i}$ and K_{τ_i} are completely determined by x_{τ_i}, A_{τ_i}, and B_{τ_i}. By computing a total expectation, we have that

$$\mathbb{E}\Big[\mathbf{1}\{\mathcal{E}\} \sum_{t=\tau_i}^{\tau_{i+1}-1} x_t^\top Q x_t + u_t^\top R u_t\Big] \leq \mathbb{E}\Big[\mathbf{1}\{\mathcal{E}_{\tau_i} \cap \mathcal{S}_{\tau_i}\}\mathbb{E}\Big[\sum_{t=\tau_i}^{\tau_{i+1}-1} x_t^\top (Q + K_{\tau_i}^\top R K_{\tau_i}) x_t \Big| x_{\tau_i}, A_{\tau_i}, B_{\tau_i}\Big]\Big]$$

Using Lemma 4.7.14, for the event $\mathcal{E}_{\tau_i}$, we conclude that K_{τ_i} is (k, ℓ)-strongly stable. Therefore, by Lemma 4.7.13, we have that

$$\begin{aligned}\mathbb{E}\Big[\mathbf{1}\{\mathcal{E}\} \sum_{t=\tau_i}^{\tau_{i+1}-1} x_t^\top Q x_t + u_t^\top R u_t\Big] &\leq (\tau_{i+1} - \tau_i)\mathbb{E}[\mathbf{1}\{\mathcal{E}_{\tau_i}\} J(K_{\tau_i})] + \frac{2\alpha_1 k^4}{l}\mathbb{E}[\mathbf{1}\{\mathcal{S}_{\tau_i}\}\|x_{\tau_i}\|^2] \\ &\leq (\tau_{i+1} - \tau_i)\mathbb{E}[\mathbf{1}\{\mathcal{E}_{\tau_i}\} J(K_{\tau_i})] + 4\alpha_1 k^6 x_b.\end{aligned}$$

Now by Lemma 4.7.1, we obtain

$$\begin{aligned}(\tau_{i+1} - \tau_i)\mathbb{E}[\mathbf{1}\{\mathcal{E}_{\tau_i}\} J(K_{\tau_i})] &\leq (\tau_{i+1} - \tau_i)(J_* + C_0 \epsilon_0^2 r^{-i+1}) \\ &\leq (\tau_{i+1} - \tau_i) J_* + (r-1) C_0 \epsilon_0^2 \tau_1.\end{aligned}$$

Summing over i, and since $\sum_{i=1}^{n_T} \tau_{i+1} - \tau_i \leq T$, we conclude that

$$J_1 \leq T J_* + n_T((r-1) C_0 \epsilon_0^2 \tau_1 + 4\alpha_1 k^6 x_b).$$

□

The next result can be proved using the same proof as in Lemma 9 of [20], and hence we omit the proof.

Lemma 4.7.20. $J_2 \leq (J(K_0) + 2\alpha_1 k^2 x_b)T^{-1} + 8\alpha_1 k^8(1+8\phi^2)x_b T^{-2}$.

Sketch of the proof. Note that J_2 is the expected cost of the complement of the event $\mathcal{E}$, where the system states become unbounded or the controller feedback gain is not (k,ℓ)-stabilizing. For this event, the controller uses the policy $u_t = K_0 x_t$. The rest of the proof is computing the expected cost conditioning on the time that the norm of the system state becomes greater than x_b. □

Lemma 4.7.21. $J_3 \leq \tau_1(1+\phi^2)J(K_0)$

Proof. Note that J_3 is the expected cost of warm-up part of Algorithm 1. For $t = 0, \ldots, \tau_1$, the controller uses $u_t = K_0 x_t + \eta_t$, and by plugging this into (4.1), we obtain

$$x_{t+1} = (A_* + B_* K_0)x_t + (B_*\eta_t + w_t).$$

This is equivalent to the system (A_*, B_*) with noise covariance $\sigma^2(I + B_* B_*^\top)$, where the controller uses $u_t = K_0 x_t$. Let $J(K_0, W)$ be the infinite horizon cost for system (4.1) with noise covariance W, where the controller uses $u_t = K_0 x_t$. Now we have that

$$J(K_0, \sigma^2(I + B_* B_*^\top)) = \mathrm{Tr}(\sigma^2(I + B_* B_*^\top)P) \leq (1+\phi^2)\mathrm{Tr}(\sigma^2 P) = (1+\phi^2)J(K_0, \sigma^2). \tag{4.48}$$

By applying Lemma 4.7.13, we conclude that

$$\begin{aligned} J_3 &= \mathbb{E}\Big[\sum_{t=0}^{\tau_1-1} x_t^\top Q x_t + u_t^\top R u_t\Big] \\ &\le \tau_1 J(K_0, \sigma^2(I + B_* B_*^\top)) + \frac{2\alpha_1 k^4}{\ell}\|x_1\|^2 \\ &\le \tau_1(1+\phi^2) J(K_0), \end{aligned}$$

where we have used the assumption that $x_1 = 0$. □

By applying Lemmas 4.7.19, 4.7.20, and 4.7.21, we conclude that

$$\begin{aligned} \mathbb{E}[\mathcal{R}_T] = \;& J_1 + J_2 + J_3 - TJ_* \\ \le \;& n_T((r-1)C_0\epsilon_0^2\tau_1 + 4\alpha_1 k^6 x_b) \\ & + (J(K_0) + 2\alpha_1 k^2 x_b)T^{-1} + 8\alpha_1 k^8(1+8\phi^2)x_b T^{-2} \\ & + \tau_1(1+\phi^2)\nu. \end{aligned} \tag{4.49}$$

By substituting the values of τ_1, x_b, and n_T in (4.49), we have that

$$\begin{aligned} \mathbb{E}[\mathcal{R}_T] \le\; & 135\frac{\log(T)}{\log(r)}\Big((r-1)C_0(240(1+k^2)(1+\phi^2)(\frac{1+k^2}{\min\{p,1\}}+1)(n+m)+\sigma^2) \\ & + 4\alpha_1 k^6\sigma^2\Big) nk^2 \max\{(1+\phi)^2k^6, 4k^6\}\log(4T) \\ & + (J(K_0) + 270\alpha_1 nk^4\sigma^2 \max\{(1+\phi)^2k^6, 4k^6\}\log(4T))T^{-1} \\ & + 1080\alpha_1(1+8\phi^2)nk^{10}\sigma^2\max\{(1+\phi)^2k^6, 4k^6\}\log(4T)T^{-2} \\ & + 135\frac{240(1+k^2)(1+\phi^2)((1+k^2)/\min\{p,1\}+1)(n+m)+\sigma^2}{\epsilon_0^2} \\ & nk^2\max\{(1+\phi)^2k^6, 4k^6\}\log(4T)(1+\phi^2)\nu. \end{aligned}$$

By rearranging the terms we get

$$\begin{aligned}
\mathbb{E}[\mathcal{R}_T] \leq & \frac{135}{\log(r)}\Big((r-1)C_0(240(1+k^2)(1+\phi^2)(\frac{1+k^2}{\min\{p,1\}}+1)(n+m)+\sigma^2) \\
& + 4\alpha_1 k^6\sigma^2\Big)nk^2\max\{(1+\phi)^2k^6, 4k^6\}\log^2(T) \\
& + 135\frac{240(1+k^2)(1+\phi^2)((1+k^2)/\min\{p,1\}+1)(n+m)+\sigma^2}{\epsilon_0^2} \\
& nk^2\max\{(1+\phi)^2k^6, 4k^6\}(1+\phi^2)\nu\log(4T) \\
& + \frac{135}{\log(r)}\Big((r-1)C_0(240(1+k^2)(1+\phi^2)(\frac{1+k^2}{\min\{p,1\}}+1)(n+m)+\sigma^2) \\
& + 4\alpha_1 k^6\sigma^2\Big)nk^2\max\{(1+\phi)^2k^6, 4k^6\}\log(4)\log(T) \\
& + (\nu + 270\alpha_1 nk^4\sigma^2\max\{(1+\phi)^2k^6, 4k^6\}\log(4T))T^{-1} \\
& + 1080\alpha_1(1+8\phi^2)nk^{10}\sigma^2\max\{(1+\phi)^2k^6, 4k^6\}\log(4T)T^{-2}. \qquad (4.50)
\end{aligned}$$

Overall, noting that the dominant term is $\log^2(T)$ in the first two lines of (4.50), we conclude that

$$\mathbb{E}[\mathcal{R}_T] \leq \text{poly}(\alpha_0, \alpha_1, \phi, \nu, m, n, r)\log^2(T).$$

□

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this thesis, we studied linear quadratic Gaussian systems with unknown parameters and we introduced two classes of online algorithms which achieve logarithmic regrets. In particular, in Chapter 3, the problem of online linear quadratic control, where the control system is linear and known and the cost functions are quadratic and time-varying and only become available in hindsight, was studied. We developed an online algorithm using the Newton-Hewer dynamics, an iterative method for solving the Riccati equation which generates stabilizing policies at each iteration. We have proved that under some uniform boundedness assumption, the algorithm can achieve a $\mathcal{O}(\log T)$ regret bound, improving the works in the literature.

We also showed that the uniform boundedness property cannot be obtained using monotonicity in Newton-Hewer dynamics, since this dynamics is not monotone in the underlying parameters, see Chapter 3. Although for the scalar case we are able to prove the uniform boundedness property, this argument cannot be readily extended to the non-scalar case. We have added various numerical examples to demonstrate that

the uniform boundedness assumption is not a strong assumption for our algorithm to achieve a logarithmic regret.

In Chapter 4, a problem setting of the adaptive LQG control was studied in which the true system parameters of the transition dynamics (matrices A and B) are *unknown*. We reviewed the existing results in the literature which state that there exists a fundamental lower bound to the regret of adaptive LQG: when the true system parameters are unknown, for any algorithm there is a choice of A and B that the algorithm must suffer a regret at least $\Omega(\sqrt{T})$. We also noted that a (poly)-logarithmic regret bound is achievable if one makes the extra assumption that B is known or A is known. By observing these results, we investigated the question whether a poly-logarithmic regret is achievable with weaker assumptions. We defined a notion of hint about the matrix B, which can be seen as noisy directional information for B, that is given to the controller periodically. We developed an algorithm using a regularized least squared error estimate and the given hint to achieve logarithmic regret. We also stated that the results of [20], where B is known, can be obtained from our setting.

5.2 Future Work

In this section, we mention possible directions for future research.

1. For the problem presented in Chapter 3, where the cost functions are unknown and time-varying, one can think of relaxing the uniform boundedness assumption of the solutions to the discrete algebraic Riccati equation. We have shown that the logarithmic regret can be achieved for our algorithm without the boundedness assumption for the scalar case, and the numerical results suggest that it

might be true for the non-scalar case; however, this needs to be explored.

2. One of the topics that one can explore is the application of a variety of greedy schemes in the online control setting. To elaborate on this, let us start by reviewing a few key points from some of the recent related work. A gradient method programming for the infinite-horizon linear-quadratic problem has been studied in [17]. Here the cost function of an infinite-horizon linear-quadratic problem is rewritten as a matrix function of stabilizing feedback gains, and the convergence of first-order gradient methods are investigated. Three types of gradient flows over the set of stabilizing policies are studied, namely, gradient flow, natural gradient flow, and the quasi-Newton flow. For each of these flows, it is shown that some Lyapunov functions decay at an exponential rate, and the corresponding trajectories are exponentially stable. Utilizing such gradient methods can potentially improve the results presented in Chapter 3.

3. Another key part of our future research will deal with the notion of gradient flow on the cone of symmetric positive definite matrices. Like any other gradient flow defined on a space that is not flat, this gradient flow relies on the inner product that is defined on the tangent space. For the symmetric cone of positive definite matrices, this is typically done using the Frobenius inner product, which is an Euclidean inner product. However, the metric corresponding to this inner product does not possess the necessary basic properties of a metric on a manifold, and only works because the underlying set is convex. A "natural" metric for the symmetric positive-definite matrices, which has the property of being invariant under congruent transformations by elements of general linear group is introduced in [56] which has led to many interesting applications, as

the gradient flow respects the geometry of the underlying set. An important question, both with respect to the treatment of gradient flows for Riccati equation [33, 17] as well as what we have studied here, is if this flow can lead to more efficient dynamics.

4. The problem of online adaptive control of linear quadratic Gaussian systems, introduced in Chapter 4, can be further studied. The hint that is introduced in this chapter is a noisy direction toward the unknown B. One interesting question is if one can find a similar approach when a hint of unknown A is given to the controller periodically, and combining these two hints is another interesting approach. The controller in the algorithm uses a linear feedback of the state, another approach is to investigate if adding perturbation to the controller policy can weaken the assumptions on hints.

5. There is also another line of research for online adaptive control of partially observable LQG systems [48]. In this setting, the controller does not have access to the state of the system and it has only access to the output of the system; however, the system dynamics is unknown and the controller faces a regret of not choosing the optimal controller. Finding a lower and upper regret bound for this setting is among the future works.

6. Decentralization and distributed systems have received considerable attention in recent years. In our previous work [6], we investigated the problem of online distributed optimization in time-varying networks, where a set of agents cooperatively make decisions and observe local cost functions. The goal is to bound an individual regret function which is defined as the difference between sum of

the costs incurred by each individual's decisions and the cost incurred by the best fixed decision when all the functions are known in advance. Extending this problem to an online optimal distributed control system are of interest.

Bibliography

[1] Y. Abbasi-Yadkori, N. Lazic, and C. Szepesvári. Model-free linear quadratic control via reduction to expert prediction. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3108–3117. PMLR, 2019.

[2] Y. Abbasi-Yadkori and C. 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.

[3] A. Agarwal, E. Hazan, S. Kale, and R. E. Schapire. Algorithms for portfolio management based on the Newton method. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 9–16, 2006.

[4] N. Agarwal, B. Bullins, E. Hazan, S. Kakade, and K. Singh. Online control with adversarial disturbances. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 111–119, 2019.

[5] N. Agarwal, E. Hazan, and K. Singh. Logarithmic regret for online control. *arXiv preprint arXiv:1909.05062*, 2019.

[6] M. Akbari, B. Gharesifard, and T. Linder. Distributed online convex optimization on time-varying directed graphs. *IEEE Transactions on Control of Network Systems*, 4:417–428, 2015.

[7] M. Akbari, B. Gharesifard, and T. Linder. On the lack of monotonicity of Newton-Hewer updates for Riccati equations. *Automatica*, 132:109788, 2020.

[8] M. Akbari, B. Gharesifard, and T. Linder. Riccati updates for online linear quadratic control. In *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, volume 120, pages 476–485. PMLR, 2020.

[9] M. Akbari, B. Gharesifard, and T. Linder. Logarithmic regret in online linear quadratic control using Riccati updates. *Mathematics of Control, Signals, and Systems*, pages 1–32, 2022.

[10] O. Anava, E. Hazan, S. Mannor, and O. Shamir. Online learning for time series prediction. In *Proceedings of the 26th Annual Conference on Learning Theory*, volume 30, pages 172–184, 2013.

[11] V. Balakrishnan and L. Vandenberghe. Semidefinite programming duality and linear time-invariant systems. *IEEE Transactions on Automatic Control*, 48(1):30–41, 2003.

[12] D. P. Bertsekas. Stable optimal control and semicontractive dynamic programming. *SIAM Journal on Control and Optimization*, 56(1):231–252, 2018.

[13] D. P. Bertsekas et al. Dynamic programming and optimal control 3rd edition, volume I. *Belmont, MA: Athena Scientific*, 2005.

[14] D. P. Bertsekas and S. E. Shreve. *Stochastic optimal control: the discrete-time case*, volume 5. Athena Scientific, 1996.

[15] S. Bittanti and M. C. Campi. Adaptive control of linear time invariant systems: the "bet on the best" principle. *Communications in Information & Systems*, 6(4):299–320, 2006.

[16] A. Blum and Y. Mansour. From external to internal regret. *Journal of Machine Learning Research*, 8:1307–1324, 2007.

[17] J. Bu, A. Mesbahi, M. Fazel, and M. Mesbahi. LQR through the lens of first order methods: Discrete-time case. *arXiv preprint arXiv:1907.08921*, 2019.

[18] P. E. Caines and D. Q. Mayne. On the discrete time matrix Riccati equation of optimal control. *International Journal of Control*, 12(5):785–794, 1970.

[19] M. C. Campi and P. R. Kumar. Adaptive linear quadratic Gaussian control: the cost-biased approach revisited. *SIAM Journal on Control and Optimization*, 36(6):1890–1907, 1998.

[20] A. Cassel, A. Cohen, and T. Koren. Logarithmic regret for learning linear quadratic regulators efficiently. In *International Conference on Machine Learning*, pages 1328–1337. PMLR, 2020.

[21] N. Cesa-Bianchi and G. Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

[22] S. Chan, G. Goodwin, and K. Sin. Convergence properties of the Riccati difference equation in optimal filtering of nonstabilizable systems. *IEEE Transactions on Automatic Control*, 29(2):110–118, 1984.

[23] H. Chen and L. Guo. Optimal adaptive control and consistent parameter estimates for ARMAX model with quadratic cost. *SIAM Journal on Control and Optimization*, 25(4):845–867, 1987.

[24] H. Chen and J. Zhang. Identification and adaptive control for systems with unknown orders, delay, and coefficients. *IEEE Transactions on Automatic Control*, 35(8):866–877, 1990.

[25] A. Cohen, A. Hasidim, T. Koren, N. Lazic, Y. Mansour, and K. Talwar. Online linear quadratic control. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 1029–1038, 2018.

[26] A. Cohen, T. Koren, and Y. Mansour. Learning linear-quadratic regulators efficiently with only $\sqrt{T}$ regret. In *International Conference on Machine Learning*, pages 1300–1309. PMLR, 2019.

[27] R. Cristi. Internal persistency of excitation in indirect adaptive control. *IEEE Transactions on Automatic Control*, 32(12):1101–1103, 1987.

[28] G. De Nicolao. On the time-varying Riccati difference equation of optimal filtering. *SIAM Journal on Control and Optimization*, 30(6):1251–1269, 1992.

[29] C. E. De Souza. On stabilizing properties of solutions of the Riccati difference equation. *IEEE Transactions on Automatic Control*, 34(12):1313–1316, 1989.

[30] O. Dekel, A. Flajolet, N. Haghtalab, and P. Jaillet. Online learning with a hint. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[31] V. V. Dombrovskii and E. A. Lyashenko. A linear quadratic control for discrete systems with random parameters and multiplicative noise and its application to

investment portfolio optimization. *Automation and remote control*, 64(10):1558–1570, 2003.

[32] M. K. S. Faradonbeh, A. Tewari, and G. Michailidis. Finite time analysis of optimal adaptive policies for linear-quadratic systems. *arXiv preprint arXiv:1711.07230*, 2017.

[33] M. Fazel, R. Ge, S. M. Kakade, and M. Mesbahi. Global convergence of policy gradient methods for the linear quadratic regulator. *arXiv preprint arXiv:1801.05039*, 2018.

[34] D. Foster and M. Simchowitz. Logarithmic regret for adversarial online control. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pages 3211–3221. PMLR, 2020.

[35] G. Freiling, G. Jank, and H. Abou-Kandil. Generalized Riccati difference and differential equations. *Linear algebra and its applications*, 241:291–303, 1996.

[36] C. E. Garcia, D. M. Prett, and M. Morari. Model predictive control: Theory and practice; a survey. *Automatica*, 25(3):335–348, 1989.

[37] E. Gofer, N. Cesa-Bianchi, C. Gentile, and Y. Mansour. Regret minimization for branching experts. In *Conference on Learning Theory*, pages 618–638, 2013.

[38] Michael Green and John B Moore. Persistence of excitation in linear systems. *Systems & control letters*, 7(5):351–360, 1986.

[39] E. C. Hall and R. M. Willett. Online convex optimization in dynamic environments. *IEEE Journal of Selected Topics in Signal Processing*, 9(4):647–662, 2015.

[40] E. Hazan, A. Agarwal, and S. Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2-3):169–192, 2007.

[41] E. Hazan et al. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

[42] E. Hazan and S. Kale. Beyond the regret minimization barrier: optimal algorithms for stochastic strongly-convex optimization. *The Journal of Machine Learning Research*, 15(1):2489–2512, 2014.

[43] G. Hewer. An iterative technique for the computation of the steady state gains for the discrete optimal regulator. *IEEE Transactions on Automatic Control*, 16(4):382–384, 1971.

[44] M. Ibrahimi, A. Javanmard, and B. Roy. Efficient reinforcement learning for high dimensional linear quadratic systems. *Advances in Neural Information Processing Systems*, 25, 2012.

[45] R. Jenatton, J. Huang, and C. Archambeau. Adaptive algorithms for online convex optimization with long-term constraints. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48, pages 402–411, 2016.

[46] A. Karimi and C. Kammer. A data-driven approach to robust control of multivariable systems by convex optimization. *Automatica*, 85:227 – 233, 2017.

[47] T. L. Lai and C. Z. Wei. Least squares estimates in stochastic regression models with applications to identification and control of dynamic systems. *The Annals of Statistics*, 10(1):154–166, 1982.

[48] S. Lale, K. Azizzadenesheli, B. Hassibi, and A. Anandkumar. Logarithmic regret bound in partially observable linear dynamical systems, 2020.

[49] J. M. Lemos and L. F. Pinto. Distributed linear-quadratic control of serially chained systems: application to a water delivery canal [applications of control]. *IEEE Control Systems Magazine*, 32(6):26–38, 2012.

[50] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[51] T. T. Lu and S. H. Shiou. Inverses of 2×2 block matrices. *Computers & Mathematics with Applications*, 43(1-2):119–129, 2002.

[52] H. Luo, C. Wei, and K. Zheng. Efficient online portfolio with logarithmic regret. In *Advances in Neural Information Processing Systems 31*, pages 8235–8245. Curran Associates, Inc., 2018.

[53] M. Mahmud, M. S. Kaiser, A. Hussain, and S. Vassanelli. Applications of deep learning and reinforcement learning to biological data. *IEEE transactions on neural networks and learning systems*, 29(6):2063–2079, 2018.

[54] H. Mania, S. Tu, and B. Recht. Certainty equivalence is efficient for linear quadratic control. *arXiv preprint arXiv:1902.07826*, 2019.

[55] S. Dean H. Mania, N. Matni, B. Recht, and S. Tu. Regret bounds for robust adaptive control of the linear quadratic regulator. *arXiv preprint arXiv:1805.09388*, 2018.

[56] M. Moakher. A differential geometric approach to the geometric mean of symmetric positive-definite matrices. *SIAM Journal on Matrix Analysis and Applications*, 26(3):735–747, 2005.

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

[58] K. Z. Ostergaard, P. Brath, and J. Stoustrup. Gain-scheduled linear quadratic control of wind turbines operating at high wind speed. In *2007 IEEE International Conference on Control Applications*, pages 276–281. IEEE, 2007.

[59] M. Patel and N. Ranganathan. IDUTC: an intelligent decision-making system for urban traffic-control applications. *IEEE Transactions on Vehicular Technology*, 50(3):816–829, 2001.

[60] B. Recht. A tour of reinforcement learning: The view from continuous control. *Annual Review of Control, Robotics, and Autonomous Systems*, 2:253–279, 2019.

[61] F. Rinaldi, S. Chiesa, and F. Quagliotti. Linear quadratic control for quadrotors UAVs dynamics and formation flight. *Journal of Intelligent & Robotic Systems*, 70(1):203–220, 2013.

[62] L. Rodman and P. Lancaster. *Algebraic Riccati Equations*. Oxford Mathematical Monographs. Clarendon press, 1995.

[63] S. Ross, G. Gordon, and D. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 627–635, 2011.

[64] S. Shalev-Shwartz. *Online Learning and Online Convex Optimization*, volume 12 of *Foundations and Trends in Machine Learning*. Now Publishers Inc, 2012.

[65] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

[66] M. Simchowitz and D. Foster. Naive exploration is optimal for online LQR. In *International Conference on Machine Learning*, pages 8937–8948. PMLR, 2020.

[67] T. Soderstrom. *Discrete-Time Stochastic Systems: Estimation and Control*. Springer-Verlag, 2nd edition, 2002.

[68] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.

[69] V. Van Breusegem, L. Chen, G. Bastin, V. Wertz, V. Werbrouck, and C. de Pierpont. An industrial application of multivariable linear quadratic control to a cement mill circuit. *IEEE Transactions on Industry Applications*, 32(3):670–677, 1996.

[70] J. C. Willems, P. Rapisarda, I. Markovsky, and B. L.M. De Moor. A note on persistency of excitation. *Systems & Control Letters*, 54(4):325–329, 2005.

[71] H. K. Wimmer. Monotonicity and maximality of solutions of discrete-time algebraic Riccati equations. *Mathematical Systems, Estimation, and Control*, 2:219–235, 1992.

[72] Y. Yang, Z. Guo, H. Xiong, D. Ding, Y. Yin, and D. C. Wunsch. Data-driven robust control of discrete-time uncertain linear systems via off-policy reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2019.

[73] H. Yu, M. Neely, and X. Wei. Online convex optimization with stochastic constraints. *Advances in Neural Information Processing Systems*, 30:1428–1438, 2017.

[74] J. Zhai, Y. Li, and H. Chen. An online optimization for dynamic power management. In *2016 IEEE International Conference on Industrial Technology (ICIT)*, pages 1533–1538, 2016.

[75] M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the Twentieth International Conference on Machine Learning*, pages 928–936, Washighton, D.C., 2003.

[76] J. Zou and Y. P. Gupta. Robust stabilizing solution of the Riccati difference equation. *European Journal of Control*, 6(4):384–391, 2000.