2021 American Control Conference (ACC)
New Orleans, USA, May 25-28, 2021

# Distributed Online Linear Quadratic Control for Linear Time-invariant Systems

Ting-Jui Chang and Shahin Shahrampour, *Senior Member*, *IEEE*

*Abstract*— **Classical linear quadratic (LQ) control centers around linear time-invariant (LTI) systems, where the control-state pairs introduce a quadratic cost with time-invariant parameters. Recent advancement in online optimization and control has provided novel tools to study LQ problems that are robust to time-varying cost parameters. Inspired by this line of research, we study the distributed online LQ problem for identical LTI systems. Consider a multi-agent network where each agent is modeled as an LTI system. The LTI systems are associated with decoupled, time-varying quadratic costs that are revealed sequentially. The goal of the network is to make the control sequence of all agents competitive to that of the best centralized policy in hindsight, captured by the notion of regret. We develop a distributed variant of the online LQ algorithm, which runs distributed online gradient descent with a projection to a semi-definite programming (SDP) to generate controllers. We establish a regret bound scaling as the square root of the finite time-horizon, implying that agents reach consensus as time grows. We further provide numerical experiments verifying our theoretical result.**

## I. Introduction

In recent years, there has been a significant interest on problems arising at the interface of control and machine learning. Modern statistical and optimization algorithms have opened new avenues to rethink classical control problems, where linear quadratic (LQ) control ( [1]–[3]) is a prominent point in case. In its classical form, LQ control centers around LTI systems, where the control-state pairs introduce a quadratic cost with *time-invariant* parameters. For the infinite-horizon problem, the optimal controller has a closed-form solution, and it can be derived by solving the algebraic Riccati equation.

Fueled by applications in practical control problems, *online* LQ control has received a great deal of attention [4]. In this scenario, the environment is subject to unpredictable dynamics, making the cost functions *time-varying* in an arbitrary fashion. Examples include variable-supply electricity production and building climate control with time-varying energy costs. Motivated by the online optimization literature, online LQ casts the time-varying problem as a regret minimization, where the performance of the online algorithm is compared with that of the best fixed control policy in hindsight.

In this paper, we address the distributed online LQ problem for a network of identical LTI systems. Each system is modeled as an agent in a multi-agent network, associated with a decoupled, time-varying quadratic cost. The cost sequence for each agent can be chosen in an adversarial fashion and the agent observes the sequence on-the-fly. The goal of the network is to make the control sequence of all agents competitive to that of the best centralized policy in hindsight, captured by the notion of *regret*. We develop a distributed variant of the online LQ algorithm. At each iteration, agents run distributed online gradient descent [5] to maintain an ideal steady-state covariance matrix. To do so, they need to perform a projection to an SDP and extract a feasible policy to generate the controllers. We prove that the individual regret can be bounded by $O(\sqrt{T})$, where $T$ is the total number of iterations. This implies that the agents reach consensus and collectively compete with the best fixed controller in hindsight. We finally provide simulation results verifying this theoretical property.

### A. Related Work

**Distributed LQ Control:** Distributed linear quadratic regulator (LQR) has been widely studied in the control literature. Some works focus on multi-agent systems with known, identical decoupled dynamics. In [6], a distributed control design is proposed by solving a single LQR problem whose size matches the maximum vertex degree of the underlying graph plus one. The authors of [7] derive the necessary condition for the optimal distributed controller, resulting in a non-convex optimization problem. The work of [8] addresses a multi-agent network, where the dynamics of each agent is a single integrator. The authors of [8] show that the computation of the optimal controller requires the knowledge of the graph and the initial information of all agents. Given the difficulty of precisely solving the optimal distributed controller, Jiao et al. [9] provide the sufficient conditions to obtain sub-optimal controllers. All of the aforementioned works need global information such as network topology to compute the controllers. On the other hand, Jiao et al. [10] propose a decentralized way to compute the controllers and show that the system will reach consensus. For the case of unknown dynamics, Alemzadeh et al. [11] propose a distributed Q-learning algorithm for dynamically decoupled systems. There are other works focusing on distributed control without assuming identical decoupled sub-systems. Fattahi et al. [12] study distributed controllers for unknown and sparse LTI systems. Furieri et al. [13] address model-free methods for distributed LQ problems and provide sample-complexity bounds for problems with local gradient dominance property (e.g., quadratically-invariant problems). The work of [14] investigates the convergence of distributed controllers to a global minimum for quadratically invariant problems with first-order methods.

T.J. Chang and S. Shahrampour are with Wm Michael Barnes '64 Department of Industrial and Systems Engineering, Texas A&M University, College Station, TX 77843, USA. email:{tingjui.chang,shahin}@tamu.edu.

This work was supported by NSF ECCS-1933878 Award.

2021 American Control Conference (ACC) | 978-1-6654-4197-1/21/$31.00 ©2021 AACC | DOI: 10.23919/ACC50511.2021.9483391

978-1-6654-4197-1/$31.00 ©2021 AACC

**Classical LQ with Unknown Dynamics:** There is a recent line of research dealing with LQ control problems with unknown dynamics. Several techniques are proposed using (i) gradient estimation (see e.g., [15]–[18]) (ii) the estimation of dynamics matrices and derivation of the controller by considering the estimation uncertainty [19], and (iii) wave-filtering [20], [21].

**Online LQ Control:** Recently, there has been a significant interest in studying linear dynamical systems with time-varying cost functions, where online learning techniques are applied. This literature investigates two scenarios: **1) Known Systems:** As mentioned before, Cohen et al. [4] study the SDP relaxation for online LQ control and establish a regret bound of $O(\sqrt{T})$ for known LTI systems with time-varying quadratic costs. Agarwal et al. [22] propose the disturbance-action policy parameterization and reduce the online control problem to online convex optimization with memory. They show that for adversarial disturbances and arbitrary time-varying convex functions, the regret is $O(\sqrt{T})$. Agarwal et al. [23] consider the case of time-varying strongly-convex functions and improve the regret bound to $O(\text{poly}(\log T))$. Simchowitz et al. [24] further extend the $O(\text{poly}(\log T))$ regret bound to partially observable systems with semi-adversarial disturbances. **2) Unknown Systems:** For fully observable systems, Hazan et al. [25] derive the regret of $O(T^{2/3})$ for time-varying convex functions with adversarial noises. For partially observable systems, the work of [24] addresses the cases of (i) convex functions with adversarial noises and (ii) strongly-convex functions with semi-adversarial noises, and provide regret bounds of $O(T^{2/3})$ and $O(\sqrt{T})$, respectively. Lale et al. [26] establish an $O(\text{poly}(\log T))$ regret bound for the case of stochastic perturbations, time-varying strongly-convex functions, and partially observed states.

Our work lies precisely at the interface of distributed LQR and online LQ, addressing distributed online LQ.

## II. Problem Formulation

### A. Notation

We use the following notation in this work:

| | |
|---|---|
| $[n]$ | The set of $\{1, 2, \ldots, n\}$ for any integer $n$ |
| $\text{Tr}(\cdot)$ | The trace operator |
| $\lVert\cdot\rVert$ | Euclidean (spectral) norm of a vector (matrix) |
| $\text{E}[\cdot]$ | The expectation operator |
| $[\mathbf{A}]_{ij}$ | The entry in the $i$-th row and $j$-th column of $\mathbf{A}$ |
| $\mathbf{A} \bullet \mathbf{B}$ | $\text{Tr}(\mathbf{A}^\top \mathbf{B})$ |
| $\mathbf{A} \succeq \mathbf{B}$ | $(\mathbf{A} - \mathbf{B})$ is positive semi-definite |

### B. Distributed Online LQ Control

We consider a multi-agent network of $m$ identical LTI systems. The dynamics of agent $i$ is given as,

$$\mathbf{x}_{i,t+1} = \mathbf{A}\mathbf{x}_{i,t} + \mathbf{B}\mathbf{u}_{i,t} + \mathbf{w}_{i,t}, \quad i \in [m]$$

where $\mathbf{x}_{i,t} \in \mathrm{R}^d$ and $\mathbf{u}_{i,t} \in \mathrm{R}^k$ represent agent $i$'s state and control (or action) at time $t$, respectively. Furthermore, $\mathbf{A} \in \mathrm{R}^{d\times d}$, $\mathbf{B} \in \mathrm{R}^{d\times k}$, and $\mathbf{w}_{i,t}$ is a Gaussian noise with zero mean and covariance $\mathbf{W} \succ 0$. The noise sequence $\{\mathbf{w}_{i,t}\}$ is independent over time and agents.

Departing from the classical LQ control, we consider the *online* distributed LQ problem in this work. At round $t$, agent $i$ receives the state $\mathbf{x}_{i,t}$ and applies the action $\mathbf{u}_{i,t}$. Then, positive definite cost matrices $\mathbf{Q}_{i,t}$ and $\mathbf{R}_{i,t}$ are revealed, and the agent incurs the cost $\mathbf{x}_{i,t}^\top \mathbf{Q}_{i,t}\mathbf{x}_{i,t} + \mathbf{u}_{i,t}^\top \mathbf{R}_{i,t}\mathbf{u}_{i,t}$. Throughout this paper, we assume that $\text{Tr}(\mathbf{Q}_{i,t}), \text{Tr}(\mathbf{R}_{i,t}) \le C$ for all $i, t$ and some $C > 0$. Agent $i$ follows a policy that selects the control $\mathbf{u}_{i,t}$ based on the observed cost matrices $\mathbf{Q}_{i,1}, \ldots, \mathbf{Q}_{i,t-1}$ and $\mathbf{R}_{i,1}, \ldots, \mathbf{R}_{i,t-1}$, as well as the information received from the agents in its neighborhood.

**Centralized Benchmark:** In order to gauge the performance of any distributed LQ algorithm, we require a centralized benchmark. In this work, we focus on the *finite-horizon* problem, where for a centralized policy $\pi$, the cost after $T$ steps is given as

$$J_T(\pi) = \text{E}\left[\sum_{t=1}^{T} \mathbf{x}_t^{\pi\top} \mathbf{Q}_t \mathbf{x}_t^{\pi} + \mathbf{u}_t^{\pi\top} \mathbf{R}_t \mathbf{u}_t^{\pi}\right], \tag{1}$$

where $\mathbf{Q}_t = \sum_{i=1}^m \mathbf{Q}_{i,t}$ and $\mathbf{R}_t = \sum_{i=1}^m \mathbf{R}_{i,t}$, and the expectation is over the possible randomness of the policy as well as the noise. The superscript $\pi$ in $\mathbf{u}_t^\pi$ and $\mathbf{x}_t^\pi$ alludes that the state-control pairs are chosen by the policy $\pi$, given full access to cost matrices of all agents. Notice that in the *infinite-horizon* version of the problem with time-invariant cost matrices $(\mathbf{Q}, \mathbf{R})$, where the goal is to minimize $\lim_{T\to\infty} J_T(\pi)/T$, it is well-known that for a controllable LTI system $(\mathbf{A}, \mathbf{B})$, the optimal policy is given by the constant linear feedback, i.e., $\mathbf{u}_t^\pi = \mathbf{K}\mathbf{x}_t^\pi$ for a matrix $\mathbf{K} \in \mathrm{R}^{k\times d}$.

**Regret Definition:** The goal of a distributed LQ algorithm $\mathcal{A}$ is to mimic the performance of an ideal centralized algorithm using only *local* information. More formally, each agent $j$ locally generates the control sequence $\{\mathbf{u}_{j,t}\}_{t=1}^T$, that is competitive to the best policy among a benchmark policy class $\Pi$. This can be formulated as minimizing the individual regret, which is defined as follows

$$\text{Regret}_T^j(\mathcal{A}) = J_T^j(\mathcal{A}) - \min_{\pi\in\Pi} J_T(\pi), \tag{2}$$

for agent $j \in [m]$, where

$$\begin{aligned} J_T^j(\mathcal{A}) &= \text{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{m} \mathbf{x}_{j,t}^{\mathcal{A}\,\top} \mathbf{Q}_{i,t}\mathbf{x}_{j,t}^{\mathcal{A}} + \mathbf{u}_{j,t}^{\mathcal{A}\,\top} \mathbf{R}_{i,t}\mathbf{u}_{j,t}^{\mathcal{A}}\right] \\ &= \text{E}\left[\sum_{t=1}^{T} \mathbf{x}_{j,t}^{\mathcal{A}\,\top} \mathbf{Q}_{t}\mathbf{x}_{j,t}^{\mathcal{A}} + \mathbf{u}_{j,t}^{\mathcal{A}\,\top} \mathbf{R}_{t}\mathbf{u}_{j,t}^{\mathcal{A}}\right]. \end{aligned}$$

A successful distributed algorithm is one that keeps the regret sublinear with respect to $T$. Of course, this also depends on the choice of the benchmark policy class $\Pi$, which is assumed to be the set of strongly stable policies (to be defined precisely in Section II-C).

**Network Structure:** The agents communicate locally to minimize the cost. The network topology is defined by a time-invariant doubly stochastic matrix $\mathbf{P}$, where $[\mathbf{P}]_{ji} > 0$ if agent $j$ communicates with agent $i$; otherwise $[\mathbf{P}]_{ji} = 0$. The network is assumed to be connected, and there exists a geometric mixing bound for $\mathbf{P}$ [27], such that

$\sum_{j=1}^{m} \left|[\mathbf{P}^k]_{ji} - 1/m\right| \leq \sqrt{m}\beta^k$, $i \in [m]$, where $\beta$ is the second largest singular value of $\mathbf{P}$.

### C. Strong Stability and Sequential Strong Stability

We consider the set of strongly stable linear (i.e., $\mathbf{u} = \mathbf{Kx}$) controllers as the benchmark policy class. Following [4], we define the notion of strong stability as follows.

*Definition 1:* (Strong Stability) A linear policy $\mathbf{K}$ is $(\kappa, \gamma)$-strongly stable (for $\kappa > 0$ and $0 < \gamma \leq 1$) for the LTI system $(\mathbf{A}, \mathbf{B})$, if $\|\mathbf{K}\| \leq \kappa$, and there exist matrices $\mathbf{L}$ and $\mathbf{H}$ such that $\mathbf{A} + \mathbf{BK} = \mathbf{HLH}^{-1}$, with $\|\mathbf{L}\| \leq 1 - \gamma$ and $\|\mathbf{H}\| \, \|\mathbf{H}^{-1}\| \leq \kappa$.

Intuitively, a strongly stable policy ensures fast mixing and exponential convergence to a steady-state distribution. In particular, for the LTI system $\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t + \mathbf{w}_t$, if a $(\kappa, \gamma)$-strongly stable policy $\mathbf{K}$ is applied ($\mathbf{u}_t = \mathbf{K}\mathbf{x}_t$), $\widehat{\mathbf{X}}_t$ (the state covariance matrix of $\mathbf{x}_t$) converges to $\mathbf{X}$ (the steady-state covariance matrix) with the following exponential rate

$$\left\|\widehat{\mathbf{X}}_t - \mathbf{X}\right\| \leq \kappa^2 e^{-2\gamma t} \left\|\widehat{\mathbf{X}}_0 - \mathbf{X}\right\|.$$

See Lemma 3.2 in [4] for details. The *sequential* nature of *online* LQ control requires another notion of strong stability, called *sequential strong stability* [4], defined as follows.

*Definition 2:* (Sequential Strong Stability) A sequence of linear policies $\{\mathbf{K}_t\}_{t=1}^{T}$ is $(\kappa, \gamma)$-strongly stable if there exist matrices $\{\mathbf{H}_t\}_{t=1}^{T}$ and $\{\mathbf{L}_t\}_{t=1}^{T}$ such that $\mathbf{A} + \mathbf{B}\mathbf{K}_t = \mathbf{H}_t\mathbf{L}_t\mathbf{H}_t^{-1}$ for all $t$ with the following properties:

1) $\|\mathbf{L}_t\| \leq 1 - \gamma$ and $\|\mathbf{K}_t\| \leq \kappa$.
2) $\|\mathbf{H}_t\| \leq \beta'$ and $\left\|\mathbf{H}_t^{-1}\right\| \leq 1/\alpha'$ with $\kappa = \beta'/\alpha'$.
3) $\left\|\mathbf{H}_{t+1}^{-1}\mathbf{H}_t\right\| \leq 1 + \gamma/2$.

Sequential strong stability generalizes strong stability to the time-varying scenario. On the technical level, it helps with characterizing the convergence of the state covariance matrices when a sequence of policies $\{\mathbf{K}_t\}_{t=1}^{T}$ is used instead of a fixed policy $\mathbf{K}$, which is the case in this work.

### D. SDP Relaxation for LQ Control

For the following dynamical system

$$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t + \mathbf{w}_t, \quad \mathbf{w}_t \sim \mathcal{N}(0, \mathbf{W}),$$

the infinite-horizon version of (1), i.e., minimize $\lim_{T\to\infty} J_T(\pi)/T$, with fixed cost matrices $\mathbf{Q}$ and $\mathbf{R}$ can be relaxed via a semi-definite programming when the steady-state distribution exists. For $\nu > 0$, the SDP relaxation is formulated as [4]

$$\begin{aligned} \text{minimize} \quad & J(\Sigma) = \begin{pmatrix} \mathbf{Q} & 0 \\ 0 & \mathbf{R} \end{pmatrix} \bullet \Sigma \\ \text{subject to} \quad & \Sigma_{\mathbf{xx}} = (\mathbf{A}\,\mathbf{B})\Sigma(\mathbf{A}\,\mathbf{B})^{\top} + \mathbf{W}, \\ & \Sigma \succeq 0, \quad \mathrm{Tr}(\Sigma) \leq \nu, \end{aligned} \tag{3}$$

where $\Sigma = \begin{pmatrix} \Sigma_{\mathbf{xx}} & \Sigma_{\mathbf{xu}} \\ \Sigma_{\mathbf{ux}} & \Sigma_{\mathbf{uu}} \end{pmatrix}$. Recall that in the online LQ problem, we deal with time-varying cost matrices $(\mathbf{Q}_t, \mathbf{R}_t)$, and for any $t \in [T]$, the above SDP yields different solutions. In fact, for any feasible solution $\Sigma$ of the above SDP, a strongly stable controller $\mathbf{K} = \Sigma_{\mathbf{xu}}^{\top}\Sigma_{\mathbf{xx}}^{-1}$ can be extracted. The steady-state covariance matrix induced by this controller is also feasible for the SDP and its cost is at most that of $\Sigma$ (see Theorem 4.2 in [4]).

Moreover, for any (slowly-varying) sequence of feasible solutions to the above SDP, the induced controller sequence is sequentially strongly-stable. This implies that the covariance matrix of the state converges to the steady-state in a rapid sense as the following.

*Lemma 1:* (Lemma 4.4 in [4]) Assume that $\mathbf{W} \succeq \sigma^2\mathbf{I}$ and let $\kappa = \sqrt{\nu}/\sigma$. Let $\{\Sigma_t\}$ be a sequence of feasible solutions of the SDP, and suppose that $\|\Sigma_{t+1} - \Sigma_t\| \leq \eta$ for all $t$ and for some $\eta \leq \sigma^2/\kappa^2$. Then, the control matrix $\mathbf{K}_t = (\Sigma_t)_{\mathbf{xu}}^{\top}(\Sigma_t)_{\mathbf{xx}}^{-1}$ is $(\kappa, \frac{1}{2\kappa^2})$-strongly stable for all $t$.

Furthermore, it can be shown that given the sequence $\mathbf{X}_t = (\Sigma_t)_{\mathbf{xx}}$, if we follow the policy sequence $\pi_t(\mathbf{x}) = \mathbf{K}_t\mathbf{x} + v_t$ where $v_t \sim \mathcal{N}\left(0, (\Sigma_t)_{\mathbf{uu}} - \mathbf{K}_t(\Sigma_t)_{\mathbf{xx}}\mathbf{K}_t^{\top}\right)$, the following relationship holds:

$$\left\|\widehat{\mathbf{X}}_{t+1} - \mathbf{X}_{t+1}\right\| \leq \kappa^2 e^{-(\frac{1}{2\kappa^2})t} \left\|\widehat{\mathbf{X}}_1 - \mathbf{X}_1\right\| + 4\eta\kappa^4,$$

where $\widehat{\mathbf{X}}_t$ is the state covariance matrix on round $t$ [4].

## III. Algorithm and Theoretical Results

We now lay out the distributed online LQ algorithm and provide its theoretical regret bound.

### A. Algorithm

In the distributed online LQ, each agent $i$ at time $t$ maintains an ideal steady-state covariance matrix $\Sigma_{i,t}$ by running a distributed online gradient descent on the SDP (3). Then, a control matrix $\mathbf{K}_{i,t}$ is extracted from $\Sigma_{i,t}$ and is used to determine the action. In particular, the action $\mathbf{u}_{i,t}$ is sampled from a Gaussian distribution $\mathcal{N}(\mathbf{K}_{i,t}\mathbf{x}_{i,t}, \mathbf{V}_{i,t})$, which entails $\mathrm{E}[\mathbf{u}_{i,t}|\mathcal{F}_t] = \mathbf{K}_{i,t}\mathbf{x}_{i,t}$, where $\mathcal{F}_t$ is the smallest $\sigma$-field containing the information about all agents up to time $t$. This stochastic policy ensures the fast convergence of the covariance matrix of $\mathbf{x}_{i,t}$ and $\mathbf{u}_{i,t}$ to the iterate $\Sigma_{i,t}$ generated by the algorithm. The proposed method is outlined in Algorithm 1.

### B. Theoretical Result: Regret Bound

In this section, we present our main theoretical result. By applying algorithm 1, we show that for a multi-agent network of identical LTI systems (with a connected communication graph), the individual regret of an arbitrary agent is upper-bounded by $O(\sqrt{T})$, which implies that the performance of all agents would converge to that of the best fixed controller in hindsight for large enough $T$.

*Theorem 2:* Assume that the network is connected, $\mathrm{Tr}(\mathbf{W}) \leq \lambda^2$ and $\mathbf{W} \succeq \sigma^2\mathbf{I}$. Given $\kappa > 1$ and $0 \leq \gamma < 1$, set $\nu = 2\kappa^4\lambda^2/\gamma$ and $\eta = 1/\sqrt{\rho T}$, where

$$\rho = \left[4mC^2\left(3 + \frac{4\sqrt{m}}{1-\beta}\right) + mC(1 + \frac{\nu}{\sigma^2})\frac{16\sqrt{2m}C\nu^2}{(1-\beta)\sigma^4}\right].$$

By running Algorithm 1, the expected individual regret of agent $j$ with respect to any $(\kappa, \gamma)$-strongly stable controller $\mathbf{K}^s$ is bounded as follows

$$\mathrm{Regret}_T^j(\mathcal{A}) = J_T^j(\mathcal{A}) - J_T(\mathbf{K}^s) = O\left((1-\beta)^{-0.5}\sqrt{T}\right),$$

**Algorithm 1** Online Distributed LQ Control

```
Require: number of agents $m$, doubly stochastic matrix $\mathbf{P} \in \mathrm{R}^{m\times m}$, parameter $\nu$, step size $\eta$, system matrices $(\mathbf{A}, \mathbf{B})$.
Initialize: $\Sigma_{i,1}$ is identically initialized with a feasible point and $\mathbf{x}_{i,1}$ is drawn from normal distribution with mean zero for $i \in [m]$.
for $t = 1, 2, \dots, T$ do
    for $i = 1, 2, \dots, m$ do
        Receive $\mathbf{x}_{i,t}$
        Compute $\mathbf{K}_{i,t} = (\Sigma_{i,t})_{\mathbf{ux}}(\Sigma_{i,t})_{\mathbf{xx}}^{-1}, \mathbf{V}_{i,t} = (\Sigma_{i,t})_{\mathbf{uu}} - \mathbf{K}_{i,t}(\Sigma_{i,t})_{\mathbf{xx}}\mathbf{K}_{i,t}^{\top}$
        Predict $\mathbf{u}_{i,t} \sim \mathcal{N}(\mathbf{K}_{i,t}\mathbf{x}_{i,t}, \mathbf{V}_{i,t})$ and Observe $\mathbf{Q}_{i,t}, \mathbf{R}_{i,t}$
        Communicate $\Sigma_{i,t}$ with agents in the neighborhood and obtain their parameters
        $\Sigma_{i,t+1} = \Pi_{\mathcal{S}}\left[\sum_{j=1}^{m}\mathbf{P}_{ji}\Sigma_{j,t} - \eta\begin{pmatrix}\mathbf{Q}_{i,t} & 0\\ 0 & \mathbf{R}_{i,t}\end{pmatrix}\right]$,
        where
        $\mathcal{S} = \left\{\Sigma \in \mathrm{R}^{(d+k)\times(d+k)} \middle| \begin{array}{l}\Sigma \succeq 0, \quad \mathrm{Tr}(\Sigma) \leq \nu, \\ \Sigma_{\mathbf{xx}} = (\mathbf{A}\;\mathbf{B})\Sigma(\mathbf{A}\;\mathbf{B})^{\top} + \mathbf{W}\end{array}\right\}$
    end for
end for
```

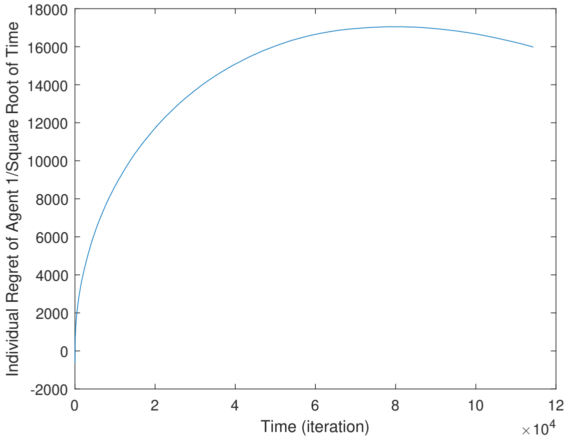

Fig. 1: This plot shows that the individual regret of agent 1 is of $O(\sqrt{T})$ when $T$ is large enough.

for $T \geq \left(\frac{4\sqrt{2}\nu C}{\sigma^4(1-\beta)\rho^{1/2}}\right)^2$.

The dependence of regret bound to the spectral gap $1 - \beta$ is perhaps not surprising, as it has been previously observed in distributed online algorithms (see e.g., [28] Corollary 4).

## IV. Numerical Experiments

**Experiment Setup:** We consider a distributed network of five agents where $d = k = 3$. The network topology is a cycle, where each agent has a self-weight of 0.6, and the rest of the weight is evenly distributed between its neighborhood as 0.2. The other hyper-parameters are set as follows: $\kappa = 1.5$, $\gamma = 0.4$, $C = 30$. We let matrices $\mathbf{A} = (1 - 2\gamma)\mathbf{I}$ and $\mathbf{B} = (\gamma/\kappa)\mathbf{I}$. We set the cost matrix $\mathbf{Q}_{i,t}$ (respectively, $\mathbf{R}_{i,t}$) as a diagonal matrix with each diagonal entry sampled from the uniform distribution over $[0, C/d]$ (respectively, $[0, C/k]$) to ensure that $\mathrm{Tr}(\mathbf{Q}_{i,t}), \mathrm{Tr}(\mathbf{R}_{i,t}) \leq C$. The noise $\mathbf{w}_{i,t}$ is sampled from a standard Gaussian distribution, and thus $\lambda^2 = d = 3$ and $\sigma^2 = 1$.

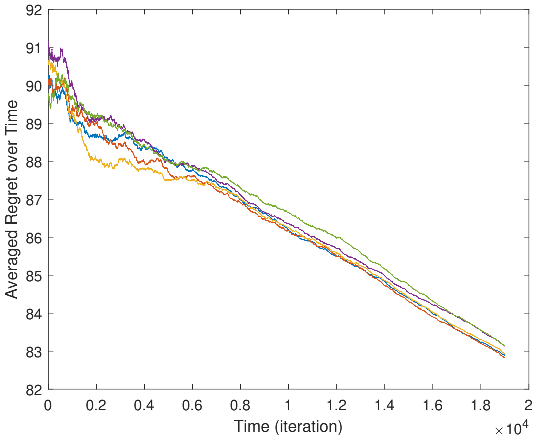

Fig. 2: The averaged regrets over time for all agents converge as time grows.

**Simulation:** The total iteration number $T$ is set as 30 times of the theoretical lower bound in Theorem 2 in order to better see the performance. We let $\mathbf{K}^s = (1e-2)(-\kappa)\mathbf{I}$ which is $(\kappa, \gamma)$-strongly stable with $\mathbf{A}, \mathbf{B}$, and leads to a small enough cumulative cost to be the benchmark. Noting that we apply Dykstra's projection algorithm for the projection step, the matrix $\mathbf{V}_{i,t}$ for action-sampling may not be positive semi-definite (PSD) due to floating-point computations, so we do some tuning by adding to it a small term $((1e-25)\mathbf{I})$ to keep it PSD. The parameters $\Sigma_{i,1}$ are identically initialized and the initial states of all agents are sampled from normal distribution. The entire process is repeated for 30 Monte-Carlo simulations.

**Performance:** To see the sub-linearity of individual regret (Theorem 2), we plot the regret normalized by the root-square of time in Fig. 1. We observe that for large enough $T$, the slope of the curve is non-positive, which verifies that the regret is upper-bounded by $O(\sqrt{T})$. In Fig. 2, it can also be seen that the time-averaged regrets for all agents are decreasing over time.

## V. Conclusion

In this paper, we considered the distributed online LQ problem with known identical LTI systems and decoupled, time-varying quadratic cost functions. We developed a fully distributed algorithm to minimize the finite-horizon cost, which can be recast as a regret minimization. We proved that the individual regret, which is the performance of the control sequence of any agent compared to the best (linear and strongly stable) controller in hindsight, is upper bounded by $O(\sqrt{T})$. Possible future directions include extending the setup to *unknown* dynamics or assuming *coupled* time-varying cost functions.

## Appendix

***Proof of Theorem 2:*** Recall the definition of regret (2). For a fixed arbitrary $(\kappa,\gamma)$-strongly stable controller $\mathbf{K}^s$ and agent $j$, the regret is expressed as the following:

$$J_T^j(\mathcal{A}) - J_T(\mathbf{K}^s) = \mathrm{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{m}(\mathbf{x}_{j,t}^\top\mathbf{Q}_{i,t}\mathbf{x}_{j,t} + \mathbf{u}_{j,t}^\top\mathbf{R}_{i,t}\mathbf{u}_{j,t})\right] - \mathrm{E}\left[\sum_{t=1}^{T}(\mathbf{x}_t^{s\top}\mathbf{Q}_t\mathbf{x}_t^s + \mathbf{u}_t^{s\top}\mathbf{R}_t\mathbf{u}_t^s)\right], \tag{4}$$

where $\mathbf{u}_t^s = \mathbf{K}^s\mathbf{x}_t^s$ for all $t$. Let us denote

$$\mathbf{L}_{i,t} = \begin{pmatrix}\mathbf{Q}_{i,t} & 0\\ 0 & \mathbf{R}_{i,t}\end{pmatrix} \quad \text{and} \quad \mathbf{L}_t = \begin{pmatrix}\mathbf{Q}_t & 0\\ 0 & \mathbf{R}_t\end{pmatrix},$$

where $\mathbf{L}_t = \sum_{i=1}^m \mathbf{L}_{i,t}$. Also let,

$$\widehat{\Sigma}_{j,t} = \mathrm{E}\left[[\mathbf{x}_{j,t}^\top\ \mathbf{u}_{j,t}^\top]^\top[\mathbf{x}_{j,t}^\top\ \mathbf{u}_{j,t}^\top]\right]$$
$$\widehat{\Sigma}_t^s = \mathrm{E}\left[[\mathbf{x}_t^{s\top}\ \mathbf{u}_t^{s\top}]^\top[\mathbf{x}_t^{s\top}\ \mathbf{u}_t^{s\top}]\right].$$

We can then write (4) as

$$\begin{aligned}&\sum_{t=1}^{T}\sum_{i=1}^{m}\mathbf{L}_{i,t}\bullet\widehat{\Sigma}_{j,t} - \sum_{t=1}^{T}\mathbf{L}_t\bullet\widehat{\Sigma}_t^s\\ =&\sum_{t=1}^{T}\sum_{i=1}^{m}\mathbf{L}_{i,t}\bullet(\widehat{\Sigma}_{j,t}-\Sigma_{j,t})\\ &+\sum_{t=1}^{T}\sum_{i=1}^{m}\mathbf{L}_{i,t}\bullet\Sigma_{j,t} - \sum_{t=1}^{T}\mathbf{L}_t\bullet\Sigma^s\\ &+\sum_{t=1}^{T}\mathbf{L}_t\bullet(\Sigma^s-\widehat{\Sigma}_t^s),\end{aligned} \tag{5}$$

where $\Sigma^s$ is the steady-state covariance matrix induced by $\mathbf{K}^s$, and $\Sigma_{j,t}$ is generated by Algorithm 1. Now, we show how each term in (5) is bounded.

**(I)** For the term $\sum_{t=1}^T\sum_{i=1}^m \mathbf{L}_{i,t}\bullet\Sigma_{j,t} - \sum_{t=1}^T\mathbf{L}_t\bullet\Sigma^s$: Based on Lemma 3.3 in [4], it can be shown that $\mathrm{Tr}(\Sigma^s) = \mathrm{Tr}(\Sigma^s_{\mathbf{xx}}) + \mathrm{Tr}(\Sigma^s_{\mathbf{uu}}) \leq 2\kappa^4\lambda^2/\gamma = \nu$. Then, by Lemma 4.1 in [4], $\Sigma^s$ is a feasible solution to the SDP (3). Based on the definition of the feasible set $\mathcal{S}$, the diameter $\sup_{\Sigma,\Sigma'\in\mathcal{S}}\|\Sigma-\Sigma'\|_F \leq 2\sup_{\Sigma\in\mathcal{S}}\|\Sigma\|_F = 2\sup_{\Sigma\in\mathcal{S}}\sqrt{\mathrm{Tr}(\Sigma^2)} \leq 2\sup_{\Sigma\in\mathcal{S}}\sqrt{\mathrm{Tr}(\Sigma)^2} \leq 2\nu$. And the norm of the gradient of the linear loss function $\mathcal{S} \rightarrow \mathbf{L}_{i,t}\bullet\mathcal{S}$ is upper bounded by $\sqrt{2}C$ since $\sqrt{\mathrm{Tr}(\mathbf{Q}_{i,t}^\top\mathbf{Q}_{i,t}) + \mathrm{Tr}(\mathbf{R}_{i,t}^\top\mathbf{R}_{i,t})} \leq \sqrt{2}C$.

Let $\Sigma^* = \mathrm{argmin}_{\Sigma\in\mathcal{S}}\sum_{t=1}^T\mathbf{L}_t\bullet\Sigma$. Based on the regret bound of distributed online gradient descent [5], we have

$$\begin{aligned}&\sum_{t=1}^{T}\sum_{i=1}^{m}\mathbf{L}_{i,t}\bullet\Sigma_{j,t} - \sum_{t=1}^{T}\mathbf{L}_t\bullet\Sigma^s\\ \leq&\sum_{t=1}^{T}\sum_{i=1}^{m}\mathbf{L}_{i,t}\bullet\Sigma_{j,t} - \sum_{t=1}^{T}\mathbf{L}_t\bullet\Sigma^*\\ \leq&\frac{m\nu}{\eta} + \left(3+\frac{4\sqrt{m}}{1-\beta}\right)4mC^2\eta T,\end{aligned} \tag{6}$$

where $\beta\in[0,1)$ is the second largest singular value of $\mathbf{P}$. Also, based on Lemma 3 in [5], the variation $\|\Sigma_{j,t+1}-\Sigma_{j,t}\|_F$ is upper bounded as the following:

$$\|\Sigma_{j,t+1}-\Sigma_{j,t}\|_F \leq \frac{4\sqrt{2m}C\eta}{1-\beta}. \tag{7}$$

**(II)** For the term $\sum_{t=1}^T\sum_{i=1}^m\mathbf{L}_{i,t}\bullet(\widehat{\Sigma}_{j,t}-\Sigma_{j,t})$: Based on Algorithm 1, we have

$$\begin{aligned}\Sigma_{j,t} &= \begin{pmatrix}(\Sigma_{j,t})_{\mathbf{xx}} & (\Sigma_{j,t})_{\mathbf{xu}}\\ (\Sigma_{j,t})_{\mathbf{ux}} & (\Sigma_{j,t})_{\mathbf{uu}}\end{pmatrix}\\ &= \begin{pmatrix}(\Sigma_{j,t})_{\mathbf{xx}} & (\Sigma_{j,t})_{\mathbf{xx}}\mathbf{K}_{j,t}^\top\\ \mathbf{K}_{j,t}(\Sigma_{j,t})_{\mathbf{xx}} & \mathbf{K}_{j,t}(\Sigma_{j,t})_{\mathbf{xx}}\mathbf{K}_{j,t}^\top\end{pmatrix} + \begin{pmatrix}0 & 0\\ 0 & \mathbf{V}_{j,t}\end{pmatrix}\end{aligned}$$

and

$$\widehat{\Sigma}_{j,t} = \begin{pmatrix}(\widehat{\Sigma}_{j,t})_{\mathbf{xx}} & (\widehat{\Sigma}_{j,t})_{\mathbf{xx}}\mathbf{K}_{j,t}^\top\\ \mathbf{K}_{j,t}(\widehat{\Sigma}_{j,t})_{\mathbf{xx}} & \mathbf{K}_{j,t}(\widehat{\Sigma}_{j,t})_{\mathbf{xx}}\mathbf{K}_{j,t}^\top\end{pmatrix} + \begin{pmatrix}0 & 0\\ 0 & \mathbf{V}_{j,t}\end{pmatrix}.$$

Therefore, we get

$$\begin{aligned}&\mathbf{L}_{i,t}\bullet(\widehat{\Sigma}_{j,t}-\Sigma_{j,t}) = \mathbf{Q}_{i,t}\bullet\big((\widehat{\Sigma}_{j,t})_{\mathbf{xx}}-(\Sigma_{j,t})_{\mathbf{xx}}\big)\\ &+\mathbf{R}_{i,t}\bullet\mathbf{K}_{j,t}\big((\widehat{\Sigma}_{j,t})_{\mathbf{xx}}-(\Sigma_{j,t})_{\mathbf{xx}}\big)\mathbf{K}_{j,t}^\top\\ &=(\mathbf{Q}_{i,t}+\mathbf{K}_{j,t}^\top\mathbf{R}_{i,t}\mathbf{K}_{j,t})\bullet\big((\widehat{\Sigma}_{j,t})_{\mathbf{xx}}-(\Sigma_{j,t})_{\mathbf{xx}}\big)\\ &\leq\mathrm{Tr}(\mathbf{Q}_{i,t}+\mathbf{K}_{j,t}^\top\mathbf{R}_{i,t}\mathbf{K}_{j,t})\left\|(\widehat{\Sigma}_{j,t})_{\mathbf{xx}}-(\Sigma_{j,t})_{\mathbf{xx}}\right\|\\ &\leq\left[\mathrm{Tr}(\mathbf{Q}_{i,t})+\mathrm{Tr}(\mathbf{R}_{i,t})\left\|\mathbf{K}_{j,t}\mathbf{K}_{j,t}^\top\right\|\right]\left\|(\widehat{\Sigma}_{j,t})_{\mathbf{xx}}-(\Sigma_{j,t})_{\mathbf{xx}}\right\|\\ &\leq C(1+\frac{\nu}{\sigma^2})\left\|(\widehat{\Sigma}_{j,t})_{\mathbf{xx}}-(\Sigma_{j,t})_{\mathbf{xx}}\right\|,\end{aligned} \tag{8}$$

where the third inequality holds since $\mathrm{Tr}(\mathbf{Q}_{i,t}), \mathrm{Tr}(\mathbf{R}_{i,t}) \leq C$ and $\mathbf{K}_{j,t}$ is $(\frac{\sqrt{\nu}}{\sigma}, \frac{\sigma^2}{2\nu})$-strongly stable based on Lemma 4.3 in [4]. Choosing $\eta$ such that $\frac{4\sqrt{2m}C\eta}{1-\beta} \leq \frac{\sigma^4}{\nu}$, based on (7) and Lemma 1, we have

$$\begin{aligned}&\left\|(\widehat{\Sigma}_{j,t})_{\mathbf{xx}}-(\Sigma_{j,t})_{\mathbf{xx}}\right\|\\ \leq&\frac{\nu}{\sigma^2}e^{-\frac{\sigma^2}{2\nu}(t-1)}\left\|(\widehat{\Sigma}_{j,1})_{\mathbf{xx}}-(\Sigma_{j,1})_{\mathbf{xx}}\right\| + \frac{16\sqrt{2m}C\eta\nu^2}{(1-\beta)\sigma^4}.\end{aligned} \tag{9}$$

Substituting (9) into (8) and summing over $t\in[T]$, we get

$$\begin{aligned}&\sum_{t=1}^{T}\mathbf{L}_{i,t}\bullet(\widehat{\Sigma}_{j,t}-\Sigma_{j,t})\\ \leq&C(1+\frac{\nu}{\sigma^2})\left(\frac{\nu}{\sigma^2}\left\|(\widehat{\Sigma}_{j,1})_{\mathbf{xx}}-(\Sigma_{j,1})_{\mathbf{xx}}\right\|\sum_{t=1}^{T}e^{-\frac{\sigma^2}{2\nu}(t-1)}\right)\\ &+C(1+\frac{\nu}{\sigma^2})\left(\frac{16\sqrt{2m}C\eta\nu^2}{(1-\beta)\sigma^4}T\right)\\ \leq&C(1+\frac{\nu}{\sigma^2})(\frac{2\nu^2}{\sigma^4}+\frac{\nu}{\sigma^2})\left\|(\widehat{\Sigma}_{j,1})_{\mathbf{xx}}-(\Sigma_{j,1})_{\mathbf{xx}}\right\|\\ &+C(1+\frac{\nu}{\sigma^2})\left(\frac{16\sqrt{2m}C\eta\nu^2}{(1-\beta)\sigma^4}T\right),\end{aligned} \tag{10}$$

where the second inequality comes from the fact that $\sum_{t=1}^T e^{-\alpha t} \leq \int_0^\infty e^{-\alpha t}dt = 1/\alpha$ for $\alpha > 0$. Summing (10) over $i$, the result is obtained.

**(III)** For the term $\sum_{t=1}^{T} \mathbf{L}_t \bullet (\Sigma^s - \widehat{\Sigma}_t^s)$:

By denoting $\Sigma^s = \begin{pmatrix} \Sigma_{\mathbf{xx}}^s & \Sigma_{\mathbf{xx}}^s \mathbf{K}^{s\top} \\ \mathbf{K}^s \Sigma_{\mathbf{xx}}^s & \mathbf{K}^s \Sigma_{\mathbf{xx}}^s \mathbf{K}^{s\top} \end{pmatrix}$ and $\widehat{\Sigma}_t^s = \begin{pmatrix} (\widehat{\Sigma}_t^s)_{\mathbf{xx}} & (\widehat{\Sigma}_t^s)_{\mathbf{xx}} \mathbf{K}^{s\top} \\ \mathbf{K}^s (\widehat{\Sigma}_t^s)_{\mathbf{xx}} & \mathbf{K}^s (\widehat{\Sigma}_t^s)_{\mathbf{xx}} \mathbf{K}^{s\top} \end{pmatrix}$, we have

$$\begin{aligned} &\mathbf{L}_t \bullet (\Sigma^s - \widehat{\Sigma}_t^s) \\ =&\sum_{i=1}^{m} (\mathbf{Q}_{i,t} + \mathbf{K}^s \mathbf{R}_{i,t} \mathbf{K}^{s\top}) \bullet \left( \Sigma_{\mathbf{xx}}^s - (\widehat{\Sigma}_t^s)_{\mathbf{xx}} \right) \\ \leq& \sum_{i=1}^{m} \mathrm{Tr}(\mathbf{Q}_{i,t} + \mathbf{K}^s \mathbf{R}_{i,t} \mathbf{K}^{s\top}) \left\| \Sigma_{\mathbf{xx}}^s - (\widehat{\Sigma}_t^s)_{\mathbf{xx}} \right\| \\ \leq& mC(1+\kappa^2) \left\| \Sigma_{\mathbf{xx}}^s - (\widehat{\Sigma}_t^s)_{\mathbf{xx}} \right\|, \end{aligned} \tag{11}$$

where the second inequality comes from the fact that $\mathrm{Tr}(\mathbf{Q}_{i,t}), \mathrm{Tr}(\mathbf{R}_{i,t}) \leq C$ and $\mathbf{K}^s$ is $(\kappa, \gamma)$-strongly stable.

Based on Lemma 3.2 in [4], we get

$$\left\| (\widehat{\Sigma}_t^s)_{\mathbf{xx}} - \Sigma_{\mathbf{xx}}^s \right\| \leq \kappa^2 e^{-2\gamma(t-1)} \left\| (\widehat{\Sigma}_1^s)_{\mathbf{xx}} - \Sigma_{\mathbf{xx}}^s \right\|. \tag{12}$$

Substituting (12) into (11) and summing over $t$, we have

$$\begin{aligned} &\sum_{t=1}^{T} \mathbf{L}_t \bullet (\Sigma^s - \widehat{\Sigma}_t^s) \\ \leq& mC(1+\kappa^2)\kappa^2 \left\| (\widehat{\Sigma}_1^s)_{\mathbf{xx}} - \Sigma_{\mathbf{xx}}^s \right\| \sum_{t=1}^{T} e^{-2\gamma(t-1)} \\ \leq& \frac{mC(\kappa^2+\kappa^4)(1+2\gamma)}{2\gamma} \left\| (\widehat{\Sigma}_1^s)_{\mathbf{xx}} - \Sigma_{\mathbf{xx}}^s \right\|. \end{aligned} \tag{13}$$

Based on (6), (10) and (13), we have

$$\begin{aligned} &J_T^j(\mathcal{A}) - J_T(\mathbf{K}^s) \\ \leq& \frac{m\nu}{\eta} + mC(1+\frac{\nu}{\sigma^2})(\frac{2\nu^2}{\sigma^4} + \frac{\nu}{\sigma^2}) \left\| (\widehat{\Sigma}_{j,1})_{\mathbf{xx}} - (\Sigma_{j,1})_{\mathbf{xx}} \right\| \\ +& \frac{mC(\kappa^2+\kappa^4)(1+2\gamma)}{2\gamma} \left\| (\widehat{\Sigma}_1^s)_{\mathbf{xx}} - \Sigma_{\mathbf{xx}}^s \right\| + \rho\eta T, \end{aligned} \tag{14}$$

where

$$\rho \triangleq \left[ 4mC^2 \left( 3 + \frac{4\sqrt{m}}{1-\beta} \right) + mC(1+\frac{\nu}{\sigma^2}) \frac{16\sqrt{2m}C\nu^2}{(1-\beta)\sigma^4} \right].$$

By setting $\eta = 1/\sqrt{\rho T}$, the upper bound in (14) is $O(\sqrt{\rho T})$. (Here it is assumed that $T \geq \left( \frac{4\sqrt{2m}\nu C}{\sigma^4(1-\beta)\rho^{1/2}} \right)^2$ to make sure $\frac{4\sqrt{2m}C\eta}{1-\beta} \leq \frac{\sigma^4}{\nu}$ and (9) holds.) ■

## References

[1] B. D. O. Anderson, J. B. Moore, and B. P. Molinari, "Linear optimal control," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-2, no. 4, pp. 559–559, 1972.

[2] D. P. Bertsekas, *Dynamic programming and optimal control*, vol. 1, no. 2.

[3] K. Zhou, J. C. Doyle, K. Glover *et al.*, *Robust and optimal control*. Prentice hall New Jersey, 1996, vol. 40.

[4] A. Cohen, A. Hasidim, T. Koren, N. Lazic, Y. Mansour, and K. Talwar, "Online linear quadratic control," in *International Conference on Machine Learning*, 2018, pp. 1029–1038.

[5] F. Yan, S. Sundaram, S. Vishwanathan, and Y. Qi, "Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 11, pp. 2483–2493, 2012.

[6] F. Borrelli and T. Keviczky, "Distributed lqr design for identical dynamically decoupled systems," *IEEE Transactions on Automatic Control*, vol. 53, no. 8, pp. 1901–1912, 2008.

[7] A. Mosebach and J. Lunze, "Synchronization of autonomous agents by an optimal networked controller," in *2014 European Control Conference (ECC)*, 2014, pp. 208–213.

[8] Y. Cao and W. Ren, "Optimal linear-consensus algorithms: An lqr perspective," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 40, no. 3, pp. 819–830, 2010.

[9] J. Jiao, H. L. Trentelman, and M. K. Camlibel, "A suboptimality approach to distributed linear quadratic optimal control," *IEEE Transactions on Automatic Control*, vol. 65, no. 3, pp. 1218–1225, 2020.

[10] ——, "Distributed linear quadratic optimal control: Compute locally and act globally," *IEEE Control Systems Letters*, vol. 4, no. 1, pp. 67–72, 2020.

[11] S. Alemzadeh and M. Mesbahi, "Distributed q-learning for dynamically decoupled systems," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 772–777.

[12] S. Fattahi, N. Matni, and S. Sojoudi, "Efficient learning of distributed linear-quadratic control policies," *SIAM Journal on Control and Optimization*, vol. 58, no. 5, pp. 2927–2951, 2020.

[13] L. Furieri, Y. Zheng, and M. Kamgarpour, "Learning the globally optimal distributed lq regulator," in *Learning for Dynamics and Control*, 2020, pp. 287–297.

[14] L. Furieri and M. Kamgarpour, "First order methods for globally optimal distributed controllers beyond quadratic invariance," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 4588–4593.

[15] M. Fazel, R. Ge, S. Kakade, and M. Mesbahi, "Global convergence of policy gradient methods for the linear quadratic regulator," in *International Conference on Machine Learning*, 2018, pp. 1467–1476.

[16] D. Malik, A. Pananjady, K. Bhatia, K. Khamaru, P. Bartlett, and M. Wainwright, "Derivative-free methods for policy optimization: Guarantees for linear quadratic systems," in *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, 2019, pp. 2916–2925.

[17] H. Mohammadi, M. Soltanolkotabi, and M. R. Jovanović, "On the linear convergence of random search for discrete-time lqr," *IEEE Control Systems Letters*, vol. 5, no. 3, pp. 989–994, 2021.

[18] H. Mohammadi, M. Soltanolkotabi, and M. R. Jovanovic, "Random search for learning the linear quadratic regulator," in *2020 American Control Conference (ACC)*, 2020, pp. 4798–4803.

[19] S. Dean, H. Mania, N. Matni, B. Recht, and S. Tu, "On the sample complexity of the linear quadratic regulator," *Foundations of Computational Mathematics*, pp. 1–47, 2019.

[20] E. Hazan, K. Singh, and C. Zhang, "Learning linear dynamical systems via spectral filtering," in *Advances in Neural Information Processing Systems*, 2017, pp. 6702–6712.

[21] S. Arora, E. Hazan, H. Lee, K. Singh, C. Zhang, and Y. Zhang, "Towards provable control for unknown linear dynamical systems," 2018.

[22] N. Agarwal, B. Bullins, E. Hazan, S. M. Kakade, and K. Singh, "Online control with adversarial disturbances," in *36th International Conference on Machine Learning, ICML 2019*. International Machine Learning Society (IMLS), 2019, pp. 154–165.

[23] N. Agarwal, E. Hazan, and K. Singh, "Logarithmic regret for online control," in *Advances in Neural Information Processing Systems*, 2019, pp. 10 175–10 184.

[24] M. Simchowitz, K. Singh, and E. Hazan, "Improper learning for non-stochastic control," *arXiv preprint arXiv:2001.09254*, 2020.

[25] E. Hazan, S. Kakade, and K. Singh, "The nonstochastic control problem," in *Algorithmic Learning Theory*, 2020, pp. 408–421.

[26] S. Lale, K. Azizzadenesheli, B. Hassibi, and A. Anandkumar, "Logarithmic regret bound in partially observable linear dynamical systems," *arXiv preprint arXiv:2003.11227*, 2020.

[27] J. S. Liu, *Monte Carlo strategies in scientific computing*. Springer Science & Business Media, 2008.

[28] S. Shahrampour and A. Jadbabaie, "Distributed online optimization in dynamic environments using mirror descent," *IEEE Transactions on Automatic Control*, vol. 63, no. 3, pp. 714–725, 2018.