Accelerated Reinforcement Learning Via Dynamic Mode Decomposition

Article *in* IEEE Transactions on Control of Network Systems · March 2023

DOI: 10.1109/TCNS.2023.3259060

CITATION
1

READS
196

4 authors, including:

Vrushabh S. Donge
University of Texas at Arlington
10 PUBLICATIONS **15** CITATIONS
SEE PROFILE

Bosen Lian
Auburn University
25 PUBLICATIONS **207** CITATIONS
SEE PROFILE

Frank Lewis
University of Texas at Arlington
94 PUBLICATIONS **458** CITATIONS
SEE PROFILE

Accelerated Reinforcement Learning via Dynamic Mode Decomposition

Vrushabh S. Donge, *Graduate Student Member, IEEE*, Bosen Lian, *Member, IEEE*, Frank L. Lewis, *Life Fellow, IEEE*, and Ali Davoudi, *Fellow, IEEE*

***Abstract*— This work applies the decomposition principle to discrete-time reinforcement learning to solve the optimal control problems for a network of subsystems. The control design is defined as a linear quadratic regulator graphical problem, where the performance function couples subsystems' dynamics. We first present a model-free discrete-time reinforcement learning algorithm based on online behaviors without using system dynamics. This could become a prohibitively-long learning process for larger networks. To remedy this issue, we develop an efficient model-free reinforcement learning algorithm based on dynamic mode decomposition. This decomposition method reduces the size of the measured data while the dynamic information of the original network is still retained. This algorithm is then implemented online. The proposed methodology is validated using examples of a consensus network and a power system network.**

***Index Terms*— Dynamic mode decomposition, Large-scale systems, Optimal control, Reinforcement learning.**

I. Introduction

Large-scale systems, frequently constructed from lower-dimensional subsystems, are becoming more common [1]. Examples include the air-traffic systems, where the physical agents are decoupled dynamically but are linked via a mutual performance function, and power systems, where physical agents are coupled both dynamically and through a mutual performance function. Conventional optimal control [2] might need the knowledge of system dynamics, whereas reinforcement learning (RL) [3] could provide optimal solutions using behavior data without knowing system dynamics. As a system grows larger, distributed control design via RL is a viable option, but the iterative learning involved could become computationally expensive. It is desired to seek a low-dimensional abstraction of the original large-scale system while retaining essential dynamics.

The work in [4], [5] study model-based and model-free RL algorithms with reduced-order optimal control problems by assuming a time lag in system dynamics. [6]–[8] study RL control algorithms for linear multi-agent systems with dimensionality reduction based on controllability and observability gramians. The majority of existing work on RL with decomposition do not extract complete dynamic information from the original system. This information could become crucial in a networked system where the control objective of each subsystem depends on its states and those of its neighbors. Moreover, the above studies on RL have been conducted in continuous time, which restricts the use of data-driven decomposition. Herein, large-scale system dynamics are formulated in discrete-time as lumped-state dynamics with a defined global performance function that enables interactions among subsystems. This interdependence characteristic makes the system more complex and challenging to analyze.

This paper proposes an off-policy RL approach in discrete-time for the large-scale linear quadratic regulator (LQR) control problem by decomposing it into a lower-dimensional one. The off-policy approach uses two policies: one is used to generate data, namely the behavior policy, while the other is used to evaluate and improve the policy [9]. The work in [10], [11] discuss LQR design for a large-scale network of homogeneous dynamical systems. One could develop a lower-dimensional mapping of the original observations and, then, study its temporal dynamics. Data obtained over space and time could provide more relevant dynamic information compared to those obtained using just spatial data.

We use a data-driven strategy to characterize complex system dynamics having high spatial dimensionality in discrete time. Employing RL could result in limited accuracy due to potentially near-singular rank matrices and longer run times for high-dimensional systems. Dynamic mode decomposition (DMD) is a computationally viable structure to analyze spatial-temporal data that can be depicted as dynamical model realizations [12], [13]. This approach does not rely on the system model, making it appropriate for model-free RL adaptation. When singular values closer to zero are preserved, singular value decomposition (SVD)-based approaches for order reduction could involve near-singular matrices. [14], [15] show thresholding techniques and [16] shows optimal thresholding. The truncation step in SVD-based DMD selects a relatively small threshold, and sets all eigenvalues below this threshold to zero.

Furthermore, we show that DMD can extract spatio-temporal coherent patterns from data. These patterns are called essential modes that oscillate at fixed natural frequencies. Using these modes can help extract exact behaviors of the underlying original high-dimensional system into lower ones [17]. The discrete-time RL algorithm is then designed using this truncated model. This significantly reduces the computational complexity of the discrete-time RL algorithm for optimal control learning.

This work is supported, in part, by ARO grant W911NF-20-1-0132.

Authors are with the Department of Electrical Engineering, University of Texas at Arlington, TX 76010, USA (e-mail: vsd4437@mavs.uta.edu, bosen.lian@mavs.uta.edu, lewis@uta.edu, davoudi@uta.edu).

The prime motivation of this paper is to illustrate that dynamic decomposition and RL can be integrated to provide a computationally-tractable optimal control scheme suitable for large-scale networks. Salient contributions of this paper are summarized as follows:

1) For a large-scale system, we formulate the LQR graphical problem in discrete time. A large-scale system is constructed by assembling linear subsystems, defined for both coupled and decoupled dynamical systems. The stabilizing controller is specified.
2) We show that the model-free discrete-time RL algorithm scales poorly when solving large-scale LQR graphical problems.
3) We develop a computationally efficient discrete-time RL algorithm based on DMD. The algorithm reduces data dimensions needed for optimal control learning while retaining the dynamic information of the original system.
4) We show the efficiency of the proposed algorithm in both theoretical and numerical analysis.

This paper is organized as follows: Section II gives preliminaries and notations. Section III introduces the large-scale LQR graphical problem, and provides a model-free discrete-time RL algorithm. Section IV presents a model-free discrete-time RL algorithm with DMD-preconditioning. Section V offers case studies. The conclusion is drawn in Section VI.

II. Notations and Preliminaries

Notations: The n-dimensional Euclidean space is denoted by $\mathbb{R}^n$. $\otimes$ stands for the Kronecker product. I_n indicates the n-dimensional identity matrix. $|\cdot|$ defines the absolute-value norm of a vector. $\|\cdot\|_F$ indicates the Frobenius norm of a vector or a matrix. Y^{t} stands for the complex conjugate transpose of Y. $Y^{\dagger}$ denotes pseudoinverse of Y matrix. $S(L) = \{\lambda_1(L),\ldots,\lambda_M(L)\}$ denotes spectrum of matrix L where λ_i is the i-th eigenvalue. For vectors a,b and matrix W, $a^T W b = (b^T \otimes a^T) vec(W)$. For a matrix $L = [l_{ij}] \in \mathbb{R}^{n\times n}$, $\mathrm{vec}(L) = [l_{11}, l_{12}, \ldots, l_{1n}, l_{21}, \ldots, l_{2n}, \ldots, l_{nn}]^T \in \mathbb{R}^{n^2}$.

Graph Preliminaries: $\mathscr{G} = (\mathscr{V},\mathscr{E})$ denotes a graph topology with $\mathscr{M} \cong (1,2,\ldots,M)$ vertices where $v_i \in \mathscr{V}$, $\forall i \in \mathscr{M}$. The edges $\mathscr{E} \subseteq \mathscr{V}\times\mathscr{V}$ are between the two vertices with the connectivity weight $e_{ij} > 0$ if $(v_j, v_i) \in \mathscr{E}$; Otherwise, $e_{ij} = 0$. The set of neighbors of vertex v_i is $\mathscr{M}_i \cong \{v_j : e_{ij} > 0\}$. We assume no self-loops in the graph, i.e., $e_{ii} = 0$. The weighted in-degree of vertex i is $d_i^i = \sum_{j=1}^{\mathscr{M}} e_{ij}$, and the weighted out-degree of vertex i is $d_i^o = \sum_{j=1}^{\mathscr{M}} e_{ji}$. Progression of edges $\mathscr{E}$ through vertices $(v_{i_{s-1}}, v_{i_s}) \in \mathscr{E}$ for $s \in (2,\ldots,j)$ constitutes a directed path originating from v_{i_1} to v_{i_j}. A graph is considered balanced and bi-directional where $e_{ij} = e_{ji}, d_i^i = d_i^o, \forall i,j$, i.e., undirected topology where directed path is present between every pair of vertices v_i and v_j. Laplacian matrix $\mathscr{L}$ is defined as $\mathscr{D} - \mathscr{A}$, where $\mathscr{A} = [e_{ij}], \mathscr{A} = \mathscr{A}^{\mathrm{T}}$, is the graph adjacency matrix, and $\mathscr{D} = diag\{d_i\}$ is the graph degree matrix.

III. Large-Scale LQR Graphical Problem

This section introduces a discrete-time LQR graphical problem for a large-scale dynamical system as a network of linear subsystems. The formulation of a large-scale system considers both decoupled and coupled fashions. Then, we propose an off-policy RL algorithm to compute the optimal control solution of large-scale systems.

A. Decoupled Systems

Consider a network having dynamically decoupled discrete-time linear subsystems

$$x_{i_{k+1}} = A_i x_{i_k} + B_i u_{i_k}, \quad i \in \mathscr{M}, \tag{1}$$

where $x_{i_k} \in \mathbb{R}^n$ and $u_{i_k} \in \mathbb{R}^m$ denote the state and control input of subsystem i, respectively. Each subsystem has local identical matrices $A_i = A_1 \in \mathbb{R}^{n\times n}$ and $B_i = B \in \mathbb{R}^{n\times m}$. (A_1, B) is assumed to be stabilizable. Note that subsystems are dynamically decoupled, but with a common objective, i.e., coupled through a performance function.

B. Coupled Systems

Consider a network having dynamically coupled discrete-time linear subsystems. The i-th subsystem dynamics, at the local level, is

$$x_{i_{k+1}} = A_{ii} x_{i_k} + A_{ij} \sum_{j\neq i, j=1}^{M} e_{ij}(x_{i_k} - x_{j_k}) + B_i u_{i_k}, \quad i \in \mathscr{M}, \tag{2}$$

where $x_{i_k} \in \mathbb{R}^n$ and $u_{i_k} \in \mathbb{R}^m$ denote subsystems state and control input, respectively. We assume that the network topology of the coupled system for both physical couplings and communication between subsystems coincide, and are defined by the same graphical topology $\mathscr{G} = (\mathscr{V},\mathscr{E})$ with the Laplacian matrix $\mathscr{L} \in \mathbb{R}^{M\times M}$. The subsystems $i \in \mathscr{V}$ relates to the local state x_{i_k}, whereas edges $(i,j) \in \mathscr{E}$ between subsystems relates to the $(x_{i_k} - x_{j_k})$. Each subsystem has identical matrices as $A_{ii} = A_1 \in \mathbb{R}^{n\times n}$, $A_{ij} = A_2 \in \mathbb{R}^{n\times n}$, and $B_i = B \in \mathbb{R}^{n\times m}$. The pairs (A_1, B) and $(A_1 + MA_2, B)$ are assumed to be stabilizable. Note that we are considering dynamically coupled M subsystems having a mutual performance function.

Define global states $\tilde{x} = (x_1^{\mathrm{T}},\ldots,x_M^{\mathrm{T}})^{\mathrm{T}} \in \mathbb{R}^{nM}$ and $\tilde{u} = (u_1^{\mathrm{T}},\ldots,u_M^{\mathrm{T}})^{\mathrm{T}} \in \mathbb{R}^{mM}$. The global dynamics of (2) and (1) are

$$\tilde{x}_{k+1} = \tilde{A}\tilde{x}_k + \tilde{B}\tilde{u}_k, \tag{3}$$

where global state and control matrices are $\tilde{A} = I_M \otimes A_1 \in \mathbb{R}^{nM\times nM}$ and $\tilde{B} = I_M \otimes B \in \mathbb{R}^{nM\times mM}$ for the decoupled system. Similarly, for the coupled system, the global state and control matrices are $\tilde{A} = (I_M \otimes A_1 + \mathscr{L} \otimes A_2) \in \mathbb{R}^{nM\times nM}$ and $\tilde{B} = I_M \otimes B \in \mathbb{R}^{nM\times mM}$.

C. Large-scale LQR Problem

The performance function couples the dynamic behavior for the set $\mathscr{M}$ of subsystems as

$$J(\tilde{u}_k, \tilde{x}_k) = \sum_{k=0}^{\infty} \Bigg[\sum_{i=1}^{M} x_{i_k}^{\mathrm{T}} Q_1 x_{i_k} + u_{i_k}^{\mathrm{T}} R_{ii} u_{i_k} + \sum_{i=1}^{M}\sum_{j\neq i}^{M} (x_{i_k} - x_{j_k})^{\mathrm{T}} Q_2 (x_{i_k} - x_{j_k}) \Bigg], \tag{4}$$

where $R_{ii} = R_{ii}^{\mathrm{T}} = R > 0$, $Q_1 = Q_1^{\mathrm{T}} \geq 0, \forall i$, and $Q_2 = Q_2^{\mathrm{T}} \geq 0, \forall i \neq j$. The global form of the performance function, J, which integrates the behavior of all subsystems, is

$$J(\tilde{u}_k, \tilde{x}_k) = \sum_{k=0}^{\infty} \left[\tilde{x}_k^{\mathrm{T}} \tilde{Q} \tilde{x}_k + \tilde{u}_k^{\mathrm{T}} \tilde{R} \tilde{u}_k\right]. \tag{5}$$

Herein, $\tilde{Q} = (I_M \otimes Q_1 + \mathscr{L} \otimes Q_2) \in \mathbb{R}^{nM \times nM}$ and $\tilde{R} = I_M \otimes R \in \mathbb{R}^{mM \times mM}$, given by

$$\tilde{R} = \begin{bmatrix} R & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & \dots & R \end{bmatrix}, \tilde{Q} = \begin{bmatrix} \tilde{Q}_{11} & \tilde{Q}_{12} & \dots & \tilde{Q}_{1M} \\ \vdots & \ddots & \vdots & \vdots \\ \tilde{Q}_{M1} & \dots & \dots & \tilde{Q}_{MM} \end{bmatrix},$$

where

$$\tilde{Q}_{ii} = Q_1 + (M-1)Q_2, \quad i \in \mathscr{M}, \tag{6a}$$

$$\tilde{Q}_{ij} = -Q_2, \quad i, j \in \mathscr{M}, i \neq j. \tag{6b}$$

Remark 1. Here, $Q_1, Q_2 \in \mathbb{R}^{n \times n}$ penalize local and relative state difference between vertices $i, j \in V$, respectively, with identical weights. Given (6), it is evident that $|\tilde{Q}_{ii}| > \sum_{i \neq j} |\tilde{Q}_{ij}|$, $\forall i$. This makes a matrix $\tilde{Q}$ strictly diagonally-dominant, and given $Q_1, Q_2 \geq 0$, $\tilde{Q} \geq 0$. Also, $\tilde{R} > 0$ is a block-diagonal matrix, where $R > 0 \in \mathbb{R}^{m \times m}$.

LQR Graphical Problem: Given Remark 1, find a unique stabilizing control policy $\tilde{u}_k$ that minimizes the global $J(\tilde{u}_k, \tilde{x}_k)$ in (5) as

$$\begin{aligned} J^*(\tilde{x}_k) = \min_{\tilde{u}_k} J(\tilde{u}_k, \tilde{x}_k) &= \min_{\tilde{u}_k} \sum_{k=0}^{\infty} \left[\tilde{x}_k^{\mathrm{T}} \tilde{Q} \tilde{x}_k + \tilde{u}_k^{\mathrm{T}} \tilde{R} \tilde{u}_k\right] \\ &= \tilde{x}_k^{\mathrm{T}} \tilde{P} \tilde{x}_k, \end{aligned} \tag{7}$$

where $\tilde{P} = \tilde{P}^{\mathrm{T}} \in \mathbb{R}^{nM \times nM} > 0$ is a cost matrix.

The optimal control policy $\tilde{u}_k^*$ from [2] is given as

$$\tilde{u}_k^* = -\tilde{K}^* \tilde{x}_k, \tag{8}$$

where $\tilde{K} \in \mathbb{R}^{mM \times nM}$ is

$$\tilde{K}^* = (\tilde{R} + \tilde{B}^{\mathrm{T}} \tilde{P} \tilde{B})^{-1} \tilde{B}^{\mathrm{T}} \tilde{P} \tilde{A}, \tag{9}$$

and $\tilde{P}$ satisfies the Algebric Riccati Equation (ARE)

$$\tilde{P} = \tilde{A}^{\mathrm{T}} \tilde{P} \tilde{A} - \tilde{A}^{\mathrm{T}} \tilde{P} \tilde{B} (\tilde{R} + \tilde{B}^{\mathrm{T}} \tilde{P} \tilde{B})^{-1} \tilde{B}^{\mathrm{T}} \tilde{P} \tilde{A} + \tilde{Q}. \tag{10}$$

In the following Theorem 1 and Corollary 1, we prove that $\tilde{P}$ in (10) has a certain structure due to the global formulation of large-scale dynamics and performance function. This is important for the design of stabilizing distributed controllers. We rewrite (10) in an equivalent form, and set $\tilde{B}\tilde{R}^{-1}\tilde{B}^{\mathrm{T}} = H$ for simplification,

$$\tilde{Q} - \tilde{A}^{\mathrm{T}} \tilde{P} H \tilde{P} \tilde{A} - \tilde{P} = 0. \tag{11}$$

For decoupled and coupled large-scale systems, diagonal blocks of $\tilde{A}$ are $\tilde{A}_{ii} = A_1$ and $\tilde{A}_{ii} = A_1 + (M-1)A_2$, respectively.

Theorem 1. *Consider the discrete-time ARE in* (10)*, with a symmetric* $\tilde{P}$ *for large-scale system dynamics in* (3)*, to solve the LQR graphical problem in* (7)*. Then, certain matrix* $W_{ii} = \sum_{h=1}^{M} \tilde{p}_{ih} = P_1, i \in \mathscr{M}$*, where* P_1 *satisfies*

$$A_1^T P_1 A_1 - A_1^T P_1 B (R + B^T P_1 B)^{-1} B^T P_1 A_1 + Q_1 - P_1 = 0. \tag{12}$$

Proof. Refer to Appendix A. □

Corollary 1. Given Theorem 1, the control policy (8) for the solution to (7) gives $\tilde{P}$ in (10), and can be divided into M^2 blocks of $\mathbb{R}^{n \times n}$, denoted by $\tilde{P}_{ih}$. For $i, j, h \in \mathscr{M}$, the followings hold

1) Diagonal blocks of $\tilde{P}$, i.e., $\tilde{P}_{ii}$ is $P_1 - (M-1)P_2$, where P_1 is solution of (12).
2) Off-diagonal blocks of $\tilde{P}$, i.e., $\tilde{P}_{ij}$, $i \neq j$, is P_2, where P_2 is associated with

$$\begin{aligned} &\bar{A}^{\mathrm{T}}(P_1 - MP_2)\bar{A} - \bar{A}^{\mathrm{T}}(P_1 - MP_2)B(R + B^{\mathrm{T}}(P_1 - MP_2)B)^{-1} \\ &\times B^{\mathrm{T}}(P_1 - MP_2)\bar{A} + (Q_1 + MQ_2) = (P_1 - MP_2). \end{aligned} \tag{13}$$

   Note that for decoupled and coupled large-scale systems, $\bar{A} = A_1$ are $\bar{A} = A_1 + MA_2$, respectively.
3) $\tilde{P} > 0$ is the solution to discrete ARE in (10) such that

$$\tilde{x}_k^{\mathrm{T}} \tilde{P} \tilde{x}_k = \tilde{x}_k^{\mathrm{T}} P_1 \tilde{x}_k + \sum_{i=1}^{M} \sum_{j \neq i}^{M} (x_{ik} - x_{jk})^{\mathrm{T}} P_2 (x_{ik} - x_{jk}). \tag{14}$$

   Herein, $\tilde{P} = (I_M \otimes P_1 - \mathscr{L} \otimes P_2) \in \mathbb{R}^{nM \times nM}$ is given by

$$\tilde{P} = \begin{bmatrix} P_1 - (M-1)P_2 & P_2 & \dots & P_2 \\ \vdots & \ddots & \vdots & \vdots \\ P_2 & \dots & \dots & P_1 - (M-1)P_2. \end{bmatrix}. \tag{15}$$

Remark 2. Note that the subsystem's ability to achieve the mutual goal, which is interconnected in the network, is generalized in (3). Therein, each subsystem has independent actuation. This formulation is standard for networked control systems [1], [10], [11], where subsystems are dynamically coupled or decoupled and have mutual performance function. It yields a more general optimal control framework as in (3) and (5) for both coupled and decoupled network systems.

Remark 3. Stabilizing distributed controllers for the global dynamics in (3) of (1) and (2) are constructed in (9), where $\tilde{P}$ in (9) and (10) has the structure given in (15).

In the following subsection, solutions of the LQR graphical problem, i.e., $(\tilde{P}, \tilde{K}^*)$, are computed using RL.

D. Off-Policy Discrete-time RL Algorithm

We present the model-free, off-policy discrete-time RL algorithm based on [9], [18] to solve the LQR graphical problem in Algorithm 1. The system (3) is rewritten as

$$\tilde{x}_{k+1} = \tilde{A}_j \tilde{x}_k + \tilde{B}(\tilde{K}^j \tilde{x}_k + \tilde{u}_k), \tag{16}$$

where $\tilde{A}_j = \tilde{A} - \tilde{B}\tilde{K}^j$. The fixed policy $\tilde{u}_k^j = -\tilde{K}^j \tilde{x}_k$, learned and updated by policy iteration [19], is applied to (3) to generate the behavioral trajectory data used in learning. The off-policy Bellman equation can be expressed as

$$\begin{aligned} &(\tilde{x}_k^{\mathrm{T}} \otimes \tilde{x}_k^{\mathrm{T}}) \mathrm{vec}(\tilde{P}^{j+1}) - (\tilde{x}_{k+1}^{\mathrm{T}} \otimes \tilde{x}_{k+1}^{\mathrm{T}}) \mathrm{vec}(\tilde{P}^{j+1}) \\ &+ 2(\tilde{x}_k^{\mathrm{T}} \otimes (\tilde{u}_k + \tilde{K}^j \tilde{x}_k)^{\mathrm{T}}) \mathrm{vec}(\tilde{B}^{\mathrm{T}} \tilde{P}^{j+1} \tilde{A}) \\ &- ((\tilde{K}^j \tilde{x}_k - \tilde{u}_k)^{\mathrm{T}} \otimes (\tilde{u}_k + \tilde{K}^j \tilde{x}_k)^{\mathrm{T}}) \mathrm{vec}(\tilde{B}^{\mathrm{T}} \tilde{P}^{j+1} \tilde{B}) \\ &= \tilde{x}_k^{\mathrm{T}} \tilde{Q} \tilde{x}_k + \tilde{x}_k^{\mathrm{T}} (\tilde{K}^j)^{\mathrm{T}} \tilde{R} \tilde{K}^j \tilde{x}_k. \end{aligned} \tag{17}$$

Next, we present a data-driven implementation of model-free RL in Algorithm 1 to solve (17) for $(\tilde{P}, \tilde{K}^*)$.

Algorithm 1 Data-driven Implementation of the Model-free RL Algorithm for a Large-scale System

1: **Initialization:** Given N, set $j=0$ and small threshold e. Select stabilizing $\tilde{K}^0$ and control input $\tilde{u}_k=-\tilde{K}\tilde{x}_k+\varepsilon_k$, where ε_k is a probing noise.

2: **Data Collection**: For $j=0,1,2,\ldots,N-1$, collect $\tilde{x}_k$ under a given $\tilde{u}_k$.

3: Compute (φ^j,ψ^j) defined as

$$\varphi^j=\begin{bmatrix}\tilde{x}_k^{\mathrm{T}}\tilde{Q}\tilde{x}_k+\tilde{x}_k^{\mathrm{T}}(\tilde{K}^j)^{\mathrm{T}}\tilde{R}\tilde{K}^j\tilde{x}_k\\ \vdots\\ \tilde{x}_{k+\zeta-1}^{\mathrm{T}}\tilde{Q}\tilde{x}_{k+\zeta-1}+\tilde{x}_{k+\zeta-1}^{\mathrm{T}}(\tilde{K}^j)^{\mathrm{T}}\tilde{R}\tilde{K}^j\tilde{x}_{k+\zeta-1}\end{bmatrix}, \tag{18}$$

$$\psi^j=\begin{bmatrix}\rho_{xx_1} & \rho_{xu_1} & \rho_{uu_1}\\ \vdots & \vdots & \vdots\\ \rho_{xx_\zeta} & \rho_{xu_\zeta} & \rho_{uu_\zeta}\end{bmatrix}, \tag{19}$$

where $s=1,2,\cdots,\zeta$, and

$$\rho_{xx_s}=(\tilde{x}_{k+s-1}^{\mathrm{T}}\otimes\tilde{x}_{k+s-1}^{\mathrm{T}})-(\tilde{x}_{k+s}^{\mathrm{T}}\otimes\tilde{x}_{k+s}^{\mathrm{T}}), \tag{20a}$$

$$\rho_{xu_s}=2(\tilde{x}_{k+s-1}^{\mathrm{T}}\otimes(\tilde{u}_{k+s-1}+\tilde{K}^j\tilde{x}_{k+s-1})^{\mathrm{T}}), \tag{20b}$$

$$\rho_{uu_s}=-(\tilde{K}^j\tilde{x}_{k+s-1}-\tilde{u}_{k+s-1})^{\mathrm{T}}\otimes(\tilde{u}_{k+s-1}+\tilde{K}^j\tilde{x}_{k+s-1})^{\mathrm{T}}. \tag{20c}$$

4: **Policy Evaluation**: Compute $\tilde{P}^{j+1}$ by

$$((\psi^j)^{\mathrm{T}}\psi^j)^{-1}(\psi^j)^{\mathrm{T}}\varphi^j=\begin{bmatrix}\mathrm{vec}(\tilde{P}^{j+1})^{\mathrm{T}}\\ \mathrm{vec}(\tilde{B}^{\mathrm{T}}\tilde{P}^{j+1}\tilde{A})^{\mathrm{T}}\\ \mathrm{vec}(\tilde{B}^{\mathrm{T}}\tilde{P}^{j+1}\tilde{B})^{\mathrm{T}}\end{bmatrix}. \tag{21}$$

5: **Policy Improvement**: Compute $\tilde{K}^{j+1}$ by

$$\tilde{K}^{j+1}=\left(\tilde{R}+\tilde{B}^{\mathrm{T}}\tilde{P}^{j+1}\tilde{B}\right)^{-1}\tilde{B}^{\mathrm{T}}\tilde{P}^{j+1}\tilde{A}. \tag{22}$$

6: **Stop if** $\|\tilde{K}^{j+1}-\tilde{K}^j\|\le e$; Otherwise, set $j=j+1$ and go to step 2.

Note that in (20), $\rho_{xx_s}\in\mathbb{R}^{1\times(nM)^2}$, $\rho_{xu_s}\in\mathbb{R}^{1\times nmM^2}$, and $\rho_{uu_s}\in\mathbb{R}^{1\times(mM)^2}$. Thus, (21) has $d=(nM)^2+(mM)^2+nmM^2$ unknown parameters. The least square (LS) solution to (21) needs a full rank of $((\psi^j)^{\mathrm{T}}\psi^j)$ and, at a minimum, needs $\zeta\ge d$ samples at each iteration. Using solutions from (21), the feedback gain $\tilde{K}^{j+1}$ can be acquired by (22).

Remark 4. The terms containing system dynamics (A,B) are regarded as unknowns and are solved in (21) given measured data in (18)-(20) by using the LS method. Note that $\tilde{x}_k$ in (18)-(20) is collected given $\tilde{u}_k$ as shown in Step 2 of Algorithm 1. $\tilde{P}^{j+1}$ is also solved in (21). $\tilde{K}^{j+1}$ is then updated using the solution of (21). This makes the approach model-free.

Remark 5. Given the large-scale networked system in (3), consider Algorithm 1 for the LQR graphical problem (7). At each iteration j, collected operators φ^j and ψ^j in (18) and (19), where $\tilde{x}_k$ is collected given $\tilde{u}_k$, give $\tilde{P}^{j+1}$ in (21). As seen in [20], a unique solution $(\tilde{P}^{j+1},\tilde{K}^{j+1})$ is obtained using an off-policy algorithm when probing noises are included in the control input for the persistence of excitation. The pair $(\tilde{P}^{j+1},\ \tilde{K}^{j+1})$ can be uniquely solved by LS while satisfying the full-rank condition of ψ^j. Then, Algorithm 1 converges to the optimal solution (9).

Remark 6. In Algorithm 1, LS requires $(n^2+m^2+nm)M^2$ data samples to obtain a unique solution. This requirement scales by M^2 over samples needed for a single subsystem. Thus, for a large M, RL Algorithm 1 scales poorly when finding optimal control using policy iteration. The learning time, i.e., the time required to execute steps 3 and 4 of Algorithm 1, could restrict its use in real-time control.

IV. DMD-BASED OFF-POLICY RL

In this section, we introduce an alternative formulation for the RL algorithm to solve the problem given in Remark 6. We use the DMD method to reduce the learning time by formulating a lower-dimensional dynamic model.

Given the state measurement $\tilde{x}_k\in\mathbb{R}^{nM}$, discretized in space and collected in step 2 of Algorithm 1, a lower-dimensional state measurement ξ_k is acquired by projecting $\tilde{x}_k$ through a projection matrix Y

$$\xi_k\approx Y\tilde{x}_k, \tag{23}$$

where $Y\in\mathbb{R}^{r\times nM}$, $r<<(nM)$.

The following lemma shows that the optimal controller $\tilde{u}_k^*$ in (8) is learned by ξ_k rather than $\tilde{x}_k$ so that the performance function associated with lower-dimensional system model is near to the global J in (5).

Lemma 1. See [21]. Given system dynamics (3) and structure of cost matrix in (15), ξ_k in (23) satisfies

$$\xi_{k+1}=Y\tilde{A}Y^{\dagger}\xi_k+Y\tilde{B}\tilde{u}_k, \tag{24}$$

$$\bar{J}(\tilde{u}_k,\xi_k)=\sum_{k=0}^{\infty}\left[\xi_k^{\mathrm{T}}\tilde{Q}_r\xi_k+\tilde{u}_k^{\mathrm{T}}R\tilde{u}_k\right], \tag{25}$$

where $Y\tilde{A}Y^{\dagger}\in\mathbb{R}^{r\times r}$ is Hurwitz,

$$\tilde{x_k}\approx Y^{\dagger}\xi_k \tag{26}$$

holds for any $\tilde{u}_k,\tilde{x}_k$ and $\tilde{Q}_r=(Y^{\dagger})^{\mathrm{T}}\tilde{Q}Y^{\dagger}\ge 0\in\mathbb{R}^{r\times r}$.

Proof. Refer to Appendix B. □

The balanced truncation [22] can build Y, but it is infeasible for larger systems that require a solution to high-dimensional Lyapunov equations [21], [23] in the computation of discrete controllability Gramian $\Phi_c=\sum_{m=0}^{\infty}\tilde{A}^m\tilde{B}\tilde{B}^{\top}(\tilde{A}^{\top})^m$. Balanced proper orthogonal decomposition (POD) [24] approximates balanced truncation and avoids the computation of Φ_c. This makes the computation of a projection matrix Y tractable for larger systems.

Remark 7. Since the system model is unknown for the model-free algorithm, the matrix Y in (23) will only be built using state measurements $\tilde{x}_k$. The matrix Y indicates the measure of insufficiency for the system (3) to be controlled and reduces the system dimension.

Remark 8. If Y in (23) satisfies controllability Gramian then $(I-Y^{\dagger}Y)\tilde{x}_k \approx 0$ holds for any $\tilde{u}_k, \tilde{x}_k$ [25]. Then, matrix Y can be found from $\min_{Y\in\mathbb{R}^{r\times nM}} \sum_{j=0}^{N-1} \left\|(I-Y^{\dagger}Y)\tilde{x}_k\right\|^2$. This projection, as shown in [25], makes (23) hold, where $Y^{\dagger}$ is specified by the first r columns of the left-singular matrix.

Eigensystem realization algorithm in [26] is similar to balanced POD [24], [27] which is suitable for high dimensional systems. The POD and SVD can identify only the spatial patterns, whereas DMD can identify spatiotemporal patterns in $\tilde{x}_k$ [28]. Thus, firstly, DMD can be used for the balanced POD to obtain Y as shown in Remark 8, where $Y^{\dagger} = \tilde{U}$. Secondly, DMD enables the extraction of dynamic modes from collected $\tilde{x}_k$. These dynamic modes are used to precondition the data vectors in (20) of the underlying large-scale system (3) onto a lower-dimensional space. This retains the complete dynamic information of (3) in lower dimensions.

A. Build Y in (23) and Extract Dynamic Modes via DMD

This section presents the data-driven approach to build Y in (23) for the system (3) while satisfying Remark 7 and extracting DMD modes. DMD is a data-driven method [28] that examines the relationship between measurements of a dynamical system correlated by a linear operator as

$$\tilde{x}_{k+1} = T\tilde{x}_k. \tag{27}$$

To obtain the data-driven model of (3) which is the same as in (27), $\tilde{x}_k$ is collected given $\tilde{u}_k$ in step 2 of Algorithm 1, where $T = \tilde{A} - \tilde{B}\tilde{K}$ and $T \in \mathbb{R}^{nM\times nM}$. Given $\tilde{x}_k$, $(X_1, X_2) \in \mathbb{R}^{nM\times(p-1)}$ is arranged such that X_2 is the time-shifted matrix of X_1

$$X_1 = \begin{bmatrix} | & | & \dots & | \\ x_1 & x_2 & \dots & x_{p-1} \\ | & | & \dots & | \end{bmatrix}, \quad X_2 = \begin{bmatrix} | & | & \dots & | \\ x_2 & x_3 & \dots & x_p \\ | & | & \dots & | \end{bmatrix},$$

where p is the complete number of snapshots. The data-driven dynamic model results in

$$X_2 = TX_1. \tag{28}$$

The LS solution to (28) can be obtained by minimizing the norm of $\|X_2 - TX_1\|_F$. This minimization problem can be efficiently solved via (29), where the pseudoinverse of X_1 is computed by SVD

$$T = X_2X_1^{\dagger}. \tag{29}$$

where $X_1 = USV^{\iota}$, $U \in \mathbb{R}^{nM\times nM}$, $S \in \mathbb{R}^{nM\times nM}$, and $V \in \mathbb{R}^{(p-1)\times nM}$. At this stage, SVD eliminates nonessential singular values in S such that S is a square matrix enabling pseudoinverse of X_1. Note that S is a diagonal matrix holding singular values σ_i of X_1, for $i = 1,\dots,nM$, which represents energy ranking [12], [29]. The thresholding of matrix S at rank r is used to truncate X_1 while retaining a high percentage of energy content. This truncation to rank r results in $\tilde{X}_1 \approx \tilde{U}\tilde{S}\tilde{V}^{\iota}$, where $\tilde{U} \in \mathbb{R}^{nM\times r}$, $\tilde{S} \in \mathbb{R}^{r\times r}$, and $\tilde{V} \in \mathbb{R}^{(p-1)\times r}$. Herein, the projection matrix $Y = \tilde{U}^{\iota}$ is obtained. Note that SVD provides the projection between two different dimensional spaces ($\mathbb{R}^{nM} \to \mathbb{R}^r$).

Remark 9. The percentage energy content in each σ is given by $E_{\sigma_i} = \frac{\sigma_i}{\sum_{i=1}^{nM}\sigma_i}$. The r can be any positive integer between 1 to nM determined by a strict threshold μ. The value of $\mu \geq 0$ is chosen for the matrix S such that $\sigma_i \geq \mu\sum_{i=1}^{nM}\sigma_i$. This thresholding using μ keeps r leading σ's with high energy and eliminates remaining, i.e., $r+1$ to nM with lower energy.

The approximation of matrix T in (29) can be computed as

$$T \approx \tilde{T} = X_2\tilde{V}\tilde{S}^{-1}\tilde{U}^{\iota}, \tag{30}$$

and dynamic model (28) results in

$$X_2 \approx \tilde{T}X_1, \tag{31}$$

where $\tilde{T}$ has the same dimension as T. For the large-scale networked system (3), (5) defines how one subsystem interacts with others. Eigenvalue analysis is critical to retrieve complete dynamic information from state measurement $\tilde{x}_k$. It is possible to extract dynamic modes efficiently from a dynamic model, where $\tilde{x}_k$ is projected onto a linear subdomain of dimension r. The lower-dimensional dynamic model becomes

$$\begin{aligned}\xi_{k+1} &= Y\tilde{T}Y^{\dagger}\xi_k \\ &= YX_2\tilde{V}\tilde{S}^{-1}\xi_k \triangleq \bar{T}\xi_k,\end{aligned} \tag{32}$$

where $\bar{T} \triangleq YX_2\tilde{V}\tilde{S}^{-1}$ and $\bar{T} \in \mathbb{R}^{r\times r}$. Given Lemma 1, one can find a unique stabilizing control policy $\tilde{u}_k$ that minimizes the global $J(\tilde{u}_k, \tilde{x}_k)$ in (5) as

$$\begin{aligned}\bar{J}^*(\tilde{x}_k) = \min_{\tilde{u}_k}\bar{J}(\tilde{u}_k,\tilde{x}_k) &= \sum_{k=0}^{\infty}\left[\xi_k^{\mathrm{T}}\tilde{Q}_r\xi_k + \tilde{u}_k^{\mathrm{T}}R\tilde{u}_k\right] \\ &= \xi_k^{\mathrm{T}}\tilde{P}_r\xi_k,\end{aligned} \tag{33}$$

where $\tilde{P}_r \in \mathbb{R}^{r\times r} > 0$ is a truncated cost matrix.

Remark 10. Note that state measurement at each snapshot can be viewed as a sample of a signal. Since these samples are stacked over p snapshots, (X_1, X_2) forms a time-series signal. Thus, T in (28) provides a temporal mapping of the state measurement signal. Eigenanalysis of T gives dynamic modes to truncate data vectors in (20), so that dynamic information of (3) is preserved [12]. The eigenvalue analysis of $\bar{T}$ is more feasible than that of $T \approx \tilde{T}$, where $r << (nM)$. This analysis provides projection within the same dimensional space ($\mathbb{R}^r \to \mathbb{R}^r$) as obtained in (32).

The next theorem shows the extraction of complete dynamic information via eigenvalue analysis for the system of high dimensions, in contrast to [21]. This information is then used to truncate data vectors in (20) within the RL framework.

Theorem 2. *Given the reduced-order dynamic model in* (32)*, eigendecomposition of $\bar{T}$ provides essential dynamic modes to precondition data vectors in* (20)*, i.e., to project $\rho_{xx_s} \in \mathbb{R}^{1\times(nM)^2}, \rho_{xu_s} \in \mathbb{R}^{1\times nmM^2}$, and $\rho_{uu_s} \in \mathbb{R}^{1\times(mM)^2}$ onto a r-dimensional space.*

Proof. Let $\mathbb{P} = \tilde{U}\tilde{U}^{\iota}$ denote the orthogonal projection onto the space of $\tilde{X}_1$, where $\tilde{X}_1 \approx \tilde{U}\tilde{S}\tilde{V}^{\iota}$. $\tilde{U}^{\iota}\tilde{U}$ is the identity matrix. If X_2 lies in span of X_1, then $\mathbb{P}\tilde{T} = \tilde{T}$.

$$\tilde{U}^{\iota}\tilde{T}\tilde{U} = \tilde{U}^{\iota}X_2\tilde{V}\tilde{S}^{-1}\tilde{U}^{\iota}\tilde{U} = \tilde{U}^{\iota}X_2\tilde{V}\tilde{S}^{-1} = \bar{T}. \tag{34}$$

Eigendecomposition of $\bar{T}$ generates eigenvalues λ and eigenvectors v that can examine essential characteristics of the

system (3), such as oscillation modes and natural frequencies [21], [30],

$$\bar{T}v = \lambda v. \tag{35}$$

It is clear from (30) and (32) that eigenvalues of $\tilde{T}$ and $\bar{T}$ are analogous [12], given $Y = \tilde{U}^\iota$.

$$\tilde{T}v = X_2\tilde{V}\tilde{S}^{-1}\tilde{U}^\iota v = YX_2\tilde{V}\tilde{S}^{-1}v = \bar{T}v = \lambda v. \tag{36}$$

As shown in Remark 10, analyzing $\bar{T}$ is computationally tractable. Given the dynamic model in (32) and (27), dynamic modes of $T \approx \tilde{T}$, i.e., θ, and eigenvectors of $\bar{T}$, i.e., v, are related by a linear transformation,

$$\theta = \frac{1}{\lambda}X_2\tilde{V}\tilde{S}^{-1}v. \tag{37}$$

We have

$$\mathbb{P}\theta = \tilde{U}\tilde{U}^\iota\theta = \tilde{U}\tilde{U}^\iota\frac{1}{\lambda}X_2\tilde{V}\tilde{S}^{-1}v = \tilde{U}\frac{1}{\lambda}\bar{T}v = \tilde{U}v = \theta. \tag{38}$$

The columns of $\theta \in \mathbb{R}^{nM\times r}$ are called the DMD modes for linear systems, and are the exact eigenvectors of $T \approx \tilde{T} \in \mathbb{R}^{nM\times nM}$ [12], [21],

$$\begin{aligned}\tilde{T}\theta = \mathbb{P}\tilde{T}\theta &= (\tilde{U}\tilde{U}^\iota)(X_2\tilde{V}\tilde{S}^{-1}\tilde{U}^\iota)(\frac{1}{\lambda}X_2\tilde{V}\tilde{S}^{-1}v)\\ &= \tilde{U}\tilde{U}^\iota X_2\tilde{V}\tilde{S}^{-1}v = \tilde{U}\bar{T}v = \lambda\theta.\end{aligned} \tag{39}$$

The updated vectors are computed from (20) as

$$\hat{\rho}_{xx} = \rho_{xx_s}(\theta\otimes\theta) \in \mathbb{R}^{1\times r^2}, \tag{40a}$$

$$\hat{\rho}_{xu} = \rho_{xu_s}(I_{mM}\otimes\theta) \in \mathbb{R}^{1\times r(mM)}, \tag{40b}$$

$$\hat{\rho}_{uu} = \rho_{uu_s}(I_{mM}\otimes I_{mM}) \in \mathbb{R}^{1\times (mM)^2}. \tag{40c}$$

□

Remark 11. The truncation of data vectors in (40) is executed with DMD modes, unlike to [6] which uses SVD modes. DMD modes differ from individual SVD modes, and they preserve dynamic information in lower dimensions as given in Remark 10. As shown in Theorem 2, the dynamic model can be decomposed into dynamic modes, where eigenvalues characterize the temporal nature of the associated dynamic modes θ. Theorem 2 pertains to higher dimensional networked systems, in contrast to [21]. Moreover, DMD modes are extracted within the RL algorithm that iteratively truncates learned datasets to lower dimensions.

B. DMD-based Model-free Off-policy RL

This subsection proposes the DMD-preconditioned off-policy discrete-time RL algorithm in Algorithm 2. Unlike Algorithm 1, Algorithm 2 has an additional step of DMD-preconditioning to project large-scale system data vectors to a truncated order while preserving dynamic information. The policy evaluation and improvement steps are the same as Algorithm 1 but use a lower-dimensional state ξ_k.

Note that in (42), $\hat{\rho}_{xx} \in \mathbb{R}^{1\times r^2}, \hat{\rho}_{xu} \in \mathbb{R}^{1\times r(mM)}$, and $\hat{\rho}_{uu} \in \mathbb{R}^{1\times (mM)^2}$. Thus, (43) has $d = r^2 + (mM)^2 + r(mM)$ unknown parameters. The LS solution to (43) needs a full rank of $((\psi_r^j)^{\mathrm{T}}\psi_r^j)$ and, at a minimum, needs $\zeta \geq d$ samples at each iteration.

Algorithm 2 Data-driven Implementation of DMD-based Model-free RL Algorithm for a Large-scale System

1: **Initialization**: Given N, set $j = 0$ and small threshold e. Select stabilizing $\tilde{K}_r^0$, $\tilde{K}^0$, and control input $\tilde{u}_k = -\tilde{K}\tilde{x}_k + \varepsilon_k$, where ε_k is a probing noise.

2: **Data Collection**: Same as step 2 in Algorithm 1 and compute (20).

3: **DMD Preconditioning**: Set strict threshold μ. Compute reduced-order data vectors $(\hat{\rho}_{xx}, \hat{\rho}_{xu}, \hat{\rho}_{uu})$ in (40) and operators (φ_r^j, ψ_r^j) given by

$$\varphi_r^j = \begin{bmatrix} \xi_k^{\mathrm{T}}\tilde{Q}_r\xi_k + \xi_k^{\mathrm{T}}(\tilde{K}_r^j)^{\mathrm{T}}\tilde{R}\tilde{K}_r^j\xi_k \\ \vdots \\ \xi_{k+\zeta-1}^{\mathrm{T}}\tilde{Q}_r\xi_{k+\zeta-1} + \xi_{k+\zeta-1}^{\mathrm{T}}(\tilde{K}_r^j)^{\mathrm{T}}\tilde{R}\tilde{K}_r^j\xi_{k+\zeta-1} \end{bmatrix}, \tag{41}$$

$$\psi_r^j = \begin{bmatrix}\hat{\rho}_{xx} & \hat{\rho}_{xu} & \hat{\rho}_{uu}\end{bmatrix}. \tag{42}$$

4: **Policy Evaluation**: Compute $\tilde{P}_r^{j+1}$ by

$$((\psi_r^j)^{\mathrm{T}}\psi_r^j)^{-1}(\psi_r^j)^{\mathrm{T}}\varphi_r^j = \begin{bmatrix} \mathrm{vec}(\tilde{P}_r{}^{j+1})^{\mathrm{T}} \\ \mathrm{vec}((Y\tilde{B})^{\mathrm{T}}\tilde{P}_r{}^{j+1}Y\tilde{A}Y^\dagger)^{\mathrm{T}} \\ \mathrm{vec}((Y\tilde{B})^{\mathrm{T}}\tilde{P}_r{}^{j+1}Y\tilde{B})^{\mathrm{T}} \end{bmatrix}. \tag{43}$$

5: **Policy Improvement**: Compute $\tilde{K}_r{}^{j+1}$ by

$$\tilde{K}_r{}^{j+1} = \left(R + (Y\tilde{B})^{\mathrm{T}}\tilde{P}_r{}^{j+1}Y\tilde{B}\right)^{-1}(Y\tilde{B})^{\mathrm{T}}\tilde{P}_r{}^{j+1}A. \tag{44}$$

6: **Stop if** $\|\tilde{K}_r{}^{j+1} - \tilde{K}_r{}^j\| \leq e$; Otherwise, set $j = j+1$, and go to step 2.

7: **Collect**:

$$\tilde{K}^{j+1} = Y\tilde{K}_r{}^{j+1}. \tag{45}$$

The following theorem proves that Algorithm 1 and Algorithm 2 have the identical convergence characteristics.

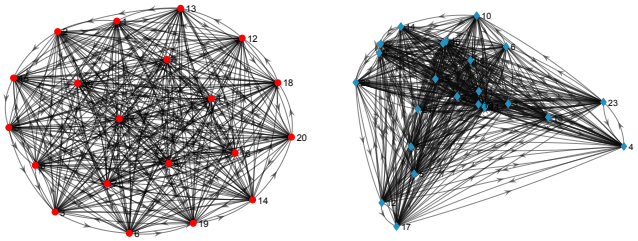
Theorem 3. ***(Convergence of Algorithm 2)*** *Given the networked system* (3)*, consider Algorithm 2 for the LQR graphical problem* (7)*. Then, Algorithm 2 converges to the optimal solution* (9)*, and P satisfies discrete-time ARE* (10)*, i.e, Algorithm 2 converges to Algorithm 1.*

Proof. Refer to Appendix C. □

C. Computation Complexity

In this section, we show that Algorithm 2 has more computational tractability than Algorithm 1. As given in Remark 6, LS requires $(n^2+m^2+nm)M^2$ data samples to obtain a unique solution in Algorithm 1, whereas Algorithm 2 would require $r^2+(mM)^2+r(mM)$ samples.

Considering ζ as data samples and d as unknown parameters, the computational complexity of LS for $\zeta \geq d$ is of the order $O(d^2\zeta)$ [31]. When finding optimal control using policy iteration for large values of M, Algorithm 1 has higher complexity of the order $O((n^2+m^2+nm)M^2)^2\zeta)$ compared to Algorithm 2 with order $O((r^2+(mM)^2+r(mM))^2\zeta)$, where



(a) Consensus network with 20 subsystems (b) 24-area power system

Fig. 1. Topologies considered for dynamically (a) decoupled and (b) coupled networks.

$r << (nM)$. Therefore, Algorithm 2 is more tractable than Algorithm 1. Algorithm 2 conserves a substantial amount of learning time provided that r is small enough. This will be verified through the following numerical simulation studies.

V. Simulation Studies

We verify the proposed DMD-based model-free RL Algorithm 2 by simulating two large-scale networks. First, the consensus network problem is discussed in Section V-A, where each subsystem is dynamically decoupled but linked with a mutual performance function. Second, the load frequency control (LFC) for multi-area power systems [32] is presented in Section V-B, where subsystems are physically coupled and have a mutual performance function.

A. Consensus Network

Figure 1(a) shows a large-scale network with 20 subsystems. Consider the LQR graphical problem in (7) with $M = 20$. The linear large-scale networked dynamical system becomes

$$\tilde{x}_{k+1} = \tilde{A}\tilde{x}_k + \tilde{B}\tilde{u}_k, \tag{46}$$

where $\tilde{x} = (x_1^{\mathrm{T}}, \ldots, x_{20}^{\mathrm{T}})^{\mathrm{T}} \in \mathbb{R}^{60}$, and $x_i \in \mathbb{R}^3$ is the state of the subsystem i. Here, $\tilde{A} = I_{20} \otimes A \in \mathbb{R}^{60\times 60}$ and $\tilde{B} = I_{20} \otimes B \in \mathbb{R}^{60\times 20}$, where $(A \in \mathbb{R}^{3\times 3}, B \in \mathbb{R}^{3\times 1})$ is adopted from [20] as

$$A = \begin{bmatrix} 0.9064 & 0.0816 & -0.0005 \\ 0.0743 & 0.9012 & -0.0007 \\ 0 & 0 & 0.1326 \end{bmatrix}, B = \begin{bmatrix} 0.0015 \\ 0.0096 \\ 0.8673 \end{bmatrix}. \tag{47}$$

One can select penalizing local state weight $Q_1 = I_3$, relative state weight $Q_2 = 0.5I_3$, and control weight $R = 1$. The global performance function in (5) uses $\tilde{Q} = I_{20} \otimes Q_1 + \mathscr{L} \otimes Q_2$ and $\tilde{R} = I_{20} \otimes R$ for the Laplacian matrix $\mathscr{L} \in \mathbb{R}^{20\times 20}$ related with the undirected $\mathscr{G}$.

We first collect state data of the large-scale system under a given $\tilde{u}_k$ in (8), i.e., $\tilde{x}_k$ over $k = \{0 \ldots 2000\}$. In step 1 of Algorithm 2, consider the probing noise $\varepsilon_k = 0.1sin(9.8k) + 0.1sin(10k) + 0.1cos(10k) + 0.1cos(10.2k)$. Compute data vectors $(\rho_{xx_s}, \rho_{xu_s}, \rho_{uu_s})$ in step 2 of Algorithm 2 as in (20). Given the collected $\tilde{x}_k$, time-shifted matrices are generated as shown in (28). The strict threshold μ, selected for singular values, is 10^{-10} for a concise truncation. Figure 2 shows singular values captured for the balanced truncation in DMD. We compute dynamic modes θ using (37) to project $(\rho_{xx_s}, \rho_{xu_s}, \rho_{uu_s})$ to lower dimensions, i.e., $(\hat{\rho}_{xx}, \hat{\rho}_{xu}, \hat{\rho}_{uu})$ in (40), to conclude DMD-preconditioning in step 3 of Algorithm 2.

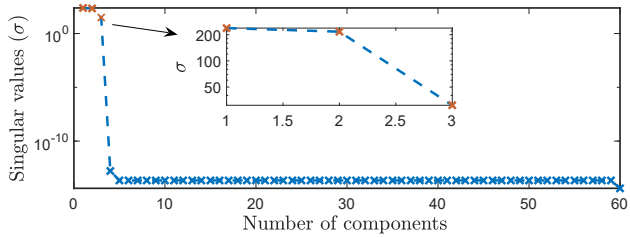


Fig. 2. Low-rank approximation for DMD (3 dominant singular values).

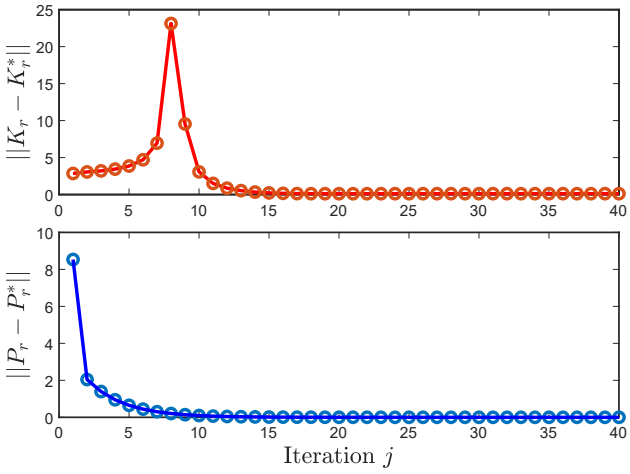


Fig. 3. Convergence of the feedback gain K_r and performance cost P_r.

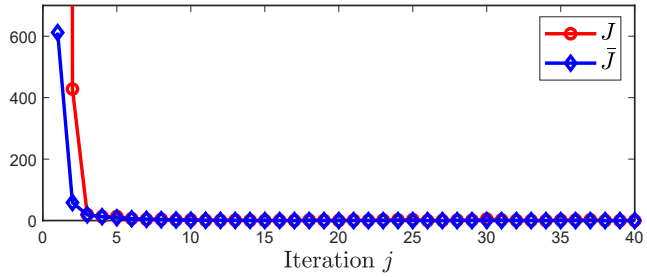


Fig. 4. Convergence of the performance functions J and $\bar{J}$.

The feedback gain $\tilde{K}_r$ for the lower dimension is computed in (44). Figure 3 shows the convergence of feedback gain $\tilde{K}_r$ and performance cost $\tilde{P}_r$ over iteration j. The solution of LQR problem yields the feedback gain matrix $\tilde{K}$ in (9) which is computed in step 7 of Algorithm 2.

Table I shows the computational time for policy improvement with the proposed dimensionality reduction for the two cases of 20 and 12 subsystems. Figure 4 shows the convergence of the performance function J in (5) and $\bar{J}$ in (33) for state matrices of dimensions 60 and 3, respectively. A similar conclusion can be drawn for a case with the state matrices of dimensions 36 and 3. At a minimum, fifteen iterations are required to achieve the optimal performance for $e = 0.003$. Table I and Fig. 4 indicate that the reduction in dimension significantly improves the learning time, whereas comparable optimal performance is achieved for nearly the same number

TABLE I
COMPUTATIONAL TIME FOR POLICY IMPROVEMENT

Number of subsystems	Spatio-temporal dimensions of state matrix before and after truncation	Computational time (*Seconds*)
$M = 20$	$\mathbb{R}^{60\times 2000}$	116
	$\mathbb{R}^{3\times 2000}$	1
$M = 12$	$\mathbb{R}^{36\times 2000}$	70
	$\mathbb{R}^{3\times 2000}$	0.24

TABLE II
DEFINITIONS FOR LFC OF A MULTI-AREA POWER SYSTEM

Variables	Definition
$\Delta f_i, \Delta P_{mi}, \Delta P_{gi}, \Delta P_{tie,i}$	Frequency deviation, Generator output deviation, Governor valve deviation, and Tie-line power deviation, respectively
T_{pi}, T_{gi}, T_{ti}	Time constants of power system, governor and turbine, respectively
K_{pi}	Gain of power system
$u_i = \Delta P_{ci}$ P_{ci}	Control input Automatic generation control
R_i	Speed regulation parameters of governor
T_{ij}	Gain of tie-line interconnection between *i-th* and *j-th* area
ACE_i	Area control error signal
D_i	Load dependency factor

of iterations. All computations are executed on an Intel(R) Xeon(R) W-10855M 2.80 GHz, 32 GB RAM, with MATLAB 2021a.

B. LFC of a Multi-area Power System

In this case study, a large-scale LQR graphical problem is formulated for LFC of multi-area power systems. Figure 1(b) shows a network of 24-areas power system. Load variation in the *i-th* area cause its frequency f_i to deviate from its reference value, and affects *j-th* area due to transient variations in f_j.

Figure 5 illustrates the layout of the *i-th* area LFC model for a multi-area power system. A generator, governor, and turbine are located in each *i-th* area. Using the linearized LFC model for multi-area power systems, adopted from [33], we examine the dynamics of the *i-th* area without load disturbance

$$\Delta \dot{f}_i = \frac{-1}{T_{pi}}\Delta f_i + \frac{K_{pi}}{T_{pi}}\Delta P_{mi} - \frac{K_{pi}}{T_{pi}}\Delta P_{tie,i}, \tag{48a}$$

$$\Delta \dot{P}_{mi} = \frac{-1}{T_{ti}}\Delta P_{mi} + \frac{1}{T_{ti}}\Delta P_{gi}, \tag{48b}$$

$$\Delta \dot{P}_{gi} = \frac{-1}{R_i T_{gi}}\Delta f_i - \frac{1}{T_{gi}}\Delta P_{gi} + \frac{u_i}{T_{gi}}, \tag{48c}$$

$$\Delta \dot{P}_{tie,i} = 2\pi \sum_{j\neq i, j=1}^{M} T_{ij}(\Delta f_i - \Delta f_j), \tag{48d}$$

where definitions for *i-th* area state variables and parameters are summarized in Table II. The input to the controller is $ACE_i = \Delta P_{tie,i} + \beta_i \Delta f_i$, and the usual choice of β_i is $\frac{1}{R_i} + D_i$.

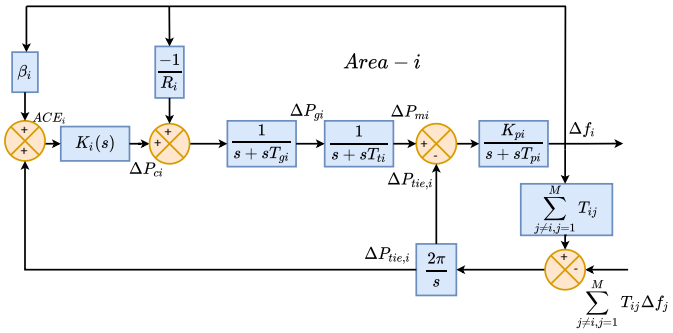


Fig. 5. Block diagram of the LFC model in M-area power systems with tie-line interconnections.

TABLE III
PARAMETERS OF THE MULTI-AREA POWER SYSTEM

Parameter	T_p (s)	T_g (s)	T_t (s)	K_p (MW/MW)	R (Hz/MW)	T_{ij} (MW/Hz)
i-th Area	20	0.08	0.3	120	2.4	0.015

At the local level, the system dynamics of the *i-th* area can be given as

$$\dot{x}_i = A_1 x_i + A_2 \sum_{j\neq i, j=1}^{M} T_{ij}(x_i - x_j) + Bu_i, \quad i \in \mathscr{M}, \tag{49}$$

where $x_i = [\Delta f_i, \Delta P_{mi}, \Delta P_{gi}, \Delta P_{tie,i}]^{\mathrm{T}}$, and

$$A_1 = \begin{bmatrix} \frac{-1}{T_p} & \frac{K_p}{T_p} & 0 & \frac{-K_p}{T_p} \\ 0 & \frac{-1}{T_t} & \frac{1}{T_t} & 0 \\ \frac{-1}{RT_g} & 0 & \frac{-1}{T_g} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 2\pi & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{T_g} \\ 0 \end{bmatrix}. \tag{50}$$

The subscript i is ignored in A_1, A_2, and B, for brevity, as areas are considered to have identical dynamics. The numerical values of parameters given in Table III are adopted from [33].

For $M = 24$, global dynamics with states $\tilde{x} = (x_1^{\mathrm{T}}, \ldots, x_{24}^{\mathrm{T}})^{\mathrm{T}} \in \mathbb{R}^{96}$ and control inputs $\tilde{u} = (u_1^{\mathrm{T}}, \ldots, u_{24}^{\mathrm{T}})^{\mathrm{T}} \in \mathbb{R}^{24}$ are

$$\dot{\tilde{x}} = \tilde{A}\tilde{x} + \tilde{B}\tilde{u}, \tag{51}$$

where $\tilde{A} = (I_M \otimes A_1 + \mathscr{L} \otimes A_2) \in \mathbb{R}^{96\times 96}$, and $\tilde{B} = I_M \otimes B \in \mathbb{R}^{96\times 24}$. Select penalizing local state weight $Q_1 = 2I_4$, relative state weight $Q_2 = 0.01I_4$, and control weight $R = 1$. Performance function in (5) uses $\tilde{Q} = I_{24} \otimes Q_1 + \mathscr{L} \otimes Q_2$ and $\tilde{R} = I_{24} \otimes R$ with the Laplacian matrix $\mathscr{L} \in \mathbb{R}^{24\times 24}$.

In order to design a discrete-time LQR control policy, system (51) is discretized using the Zero-Order-Hold (ZOH) method with a sampling period of $\tau = 1$ seconds. The resulting discrete-time system dynamics become

$$\tilde{x}_{(k+1)\tau} = \tilde{A}_d \tilde{x}_{k\tau} + \tilde{B}_d \tilde{u}_{k\tau}, \tag{52}$$

where $A_d = e^{\tilde{A}\tau} \in \mathbb{R}^{96\times 96}$, and $B_d = \int_0^\tau e^{\tilde{A}\tau}\tilde{B}dt \in \mathbb{R}^{96\times 24}$. The LFC in multi-area power system is an example of a large-scale system having multiple sampling periods. That is, control

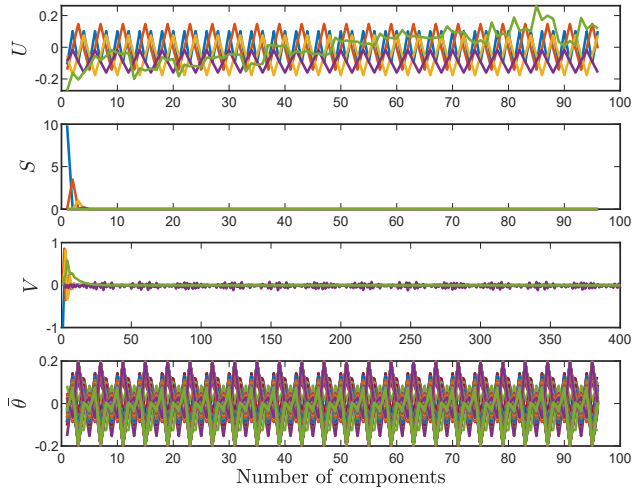


Fig. 6. Total dynamic modes $\bar{\theta}$ (real part only) of M-area power system.

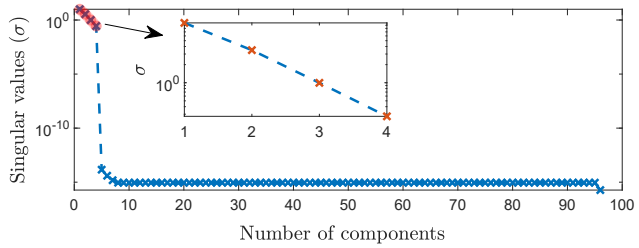


Fig. 7. Low-rank approximation for DMD (4 dominant singular values).

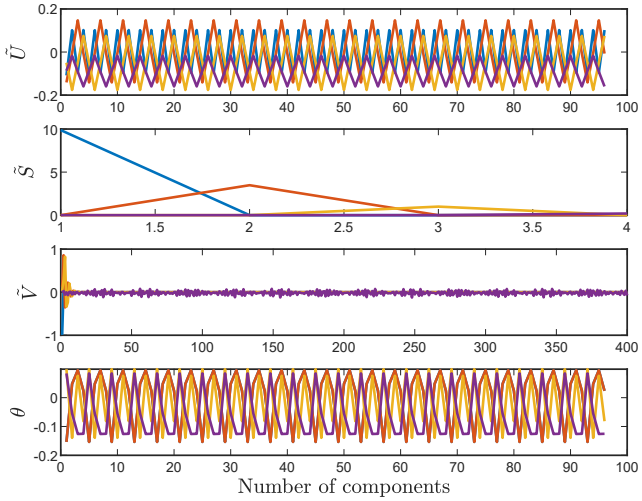


Fig. 8. Essential modes (real part only) for the singular values in Fig. 7.

signals sent to areas are in discrete time with a sampling period of 1-5 seconds (s). Thus, we use discrete-time LFC [34].

The state data of (52) is collected under $\tilde{u}_k$ in (8), i.e., $\tilde{x}_k$ over $k = \{0 \ldots 400\}$ and probing noise $\varepsilon_k = 0.01sin(9.8k) + 0.01sin(10k) + 0.01cos(10k) + 0.01cos(10.2k)$. Figure 6 illustrates the total dynamic modes $\bar{\theta} \in \mathbb{R}^{96\times 96}$ of the original system with the SVD matrices where $U \in \mathbb{R}^{96\times 96}$, $S \in \mathbb{R}^{96\times 96}$, and $V \in \mathbb{R}^{400\times 96}$. Figure 7 shows that 4 dominant singular values need to be captured for the balanced truncation in DMD. Figure 8 presents 4 essential dynamic modes $\theta \in \mathbb{R}^{96\times 4}$ of decomposed system with the truncated SVD matrices where $\tilde{U} \in \mathbb{R}^{96\times 4}$, $\tilde{S} \in \mathbb{R}^{4\times 4}$, and $\tilde{V} \in \mathbb{R}^{400\times 4}$. Figure 9 shows convergence of feedback gain $\tilde{K}_r$ and performance cost P_r over iteration j. As seen in Fig. 10, state deviations of M-area power system i.e., $\Delta f_i, \Delta P_{mi}, \Delta P_{gi}, \Delta P_{tie,i}$ converges to zero under the proposed accelerated control scheme.

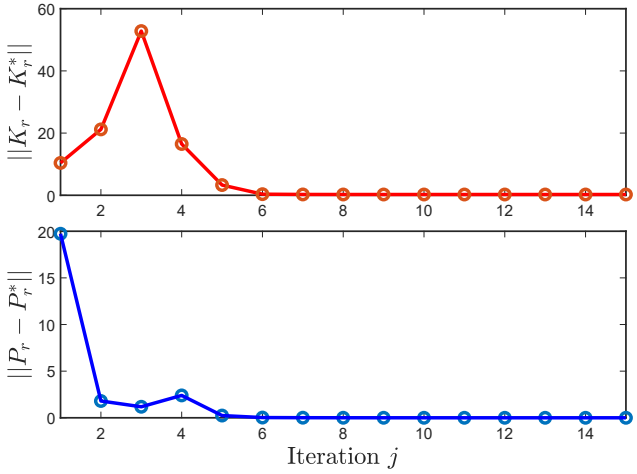


Fig. 9. Convergence of K_r and P_r.

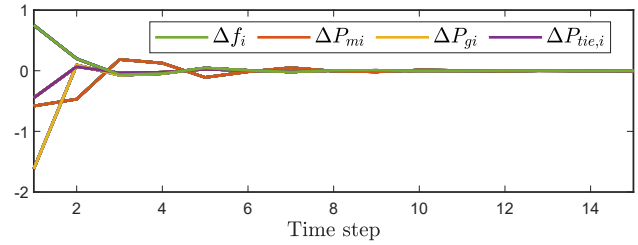


Fig. 10. State trajectories, i.e., deviations of frequency, generator output, governor valve, and tie-line power of i- th area power system.

C. Comparative Studies

We compare the computational features of our Algorithm 2 and the RL algorithm in [6] for the LQR control problem of the large-scale networked system in (3). We consider the case study in Section V-A and use the same parameters. The system model in (47) with M equal to 20 and 12 are considered. We notice the difference in iteration numbers (No.) and computational time (s). Note that Remark 11 is still satisfied. The original dynamic information for the system in (3) is not lost when using DMD for data vectors decomposition instead of SVD modes. Table IV shows that our proposed algorithm has less computational time and iterations than those based on [6].

TABLE IV
COMPUTATION FEATURES OF ALGORITHM 2 AGAINST [6]

Number of subsystems	Iteration No.		Computational Time (s)	
	Algorithm 2	[6]	Algorithm 2	[6]
20	15	18	1	1.2
12	12	15	0.24	0.29

VI. Conclusion & Future Work

This paper proposes a model-free, off-policy discrete-time RL algorithm to solve the optimal control problem for large-scale systems. Using DMD, this approach reduces the efforts of RL control while retaining the dynamic information of the original large-scale system. Since DMD preconditioning is data-driven, the RL algorithm becomes entirely model-free. Potential path forward would treat DMD-based reinforcement learning, for high-dimensional systems having heterogeneous subsystems, extended LQR formulations and DMD-based inverse reinforcement learning [35], [36].

Appendix

A. Proof of Theorem 1

$\tilde{P}$ in (10) is symmetric, i.e., $\tilde{P}_{ij}=\tilde{P}_{ji}$. Then, let a certain matrix W_{ii} as

$$W_{ii}=\tilde{P}_{ii}+\sum_{j\neq i,j=1}^{M}\tilde{P}_{ij}, i\in\mathscr{M}. \tag{A.1}$$

For the diagonal blocks of $\tilde{P}$, i.e., $\tilde{P}_{ii}$ in (10), we have

$$\tilde{Q}_{ii}-\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1}^{M}(\tilde{P}_{ih}H_{ii}\tilde{P}_{ih})\tilde{A}_{ii}-\tilde{P}_{ii}=0. \tag{A.2}$$

Substitute $\tilde{P}_{ii}=W_{ii}-\sum_{j\neq i,j=1}^{M}\tilde{P}_{ij}$ in (A.2) to yield

$$\begin{aligned}&\tilde{Q}_{ii}-\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}(\tilde{P}_{ih}H_{ii}\tilde{P}_{ih})\tilde{A}_{ii}-\tilde{A}_{ii}^{\mathrm{T}}W_{ii}H_{ii}W_{ii}\tilde{A}_{ii}\\&-\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}H_{ii}\sum_{l=1,l\neq i}^{M}\tilde{P}_{il}\tilde{A}_{ii}+\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}H_{ii}W_{ii}\tilde{A}_{ii}\\&+\tilde{A}_{ii}^{\mathrm{T}}W_{ii}H_{ii}\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}\tilde{A}_{ii}-W_{ii}+\sum_{j=1,j\neq i}^{M}\tilde{P}_{ij}=0.\end{aligned} \tag{A.3}$$

For the off-diagonal blocks of $\tilde{P}$, i.e., $\tilde{P}_{ij}$ in (10), we have

$$\tilde{Q}_{ij}-\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1}^{M}(\tilde{P}_{ih}H_{ii}\tilde{P}_{hj})\tilde{A}_{ii}-\tilde{P}_{ij}=0. \tag{A.4}$$

Substituting (A.1) in (A.4) leads to

$$\begin{aligned}&\tilde{Q}_{ij}-\tilde{A}_{ii}^{\mathrm{T}}W_{ii}H_{ii}\tilde{P}_{ij}\tilde{A}_{ii}+\tilde{A}_{ii}^{\mathrm{T}}\Big(\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}\Big)H_{ii}\tilde{P}_{ij}\tilde{A}_{ii}-\tilde{A}_{ii}^{\mathrm{T}}\tilde{P}_{ij}H_{ii}W_{jj}\tilde{A}_{ii}\\&+\tilde{A}_{ii}^{\mathrm{T}}\tilde{P}_{ij}H_{ii}\Big(\sum_{h=1,h\neq j}^{M}\tilde{P}_{jh}\Big)\tilde{A}_{ii}-\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i,j}^{M}(\tilde{P}_{ih}H_{ii}\tilde{P}_{hj})\tilde{A}_{ii}-\tilde{P}_{ij}=0.\end{aligned} \tag{A.5}$$

For $j\neq i$, summing up (A.5) relates to $\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}$, i.e., the summation of the off-diagonal terms leads to

$$\begin{aligned}&\sum_{h=1,h\neq i}^{M}\tilde{Q}_{ih}-\tilde{A}_{ii}^{\mathrm{T}}W_{ii}H_{ii}\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}\tilde{A}_{ii}+\tilde{A}_{ii}^{\mathrm{T}}\sum_{l=1,l\neq i}^{M}\Big(\Big(\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}\Big)\\&\times H_{ii}\tilde{P}_{il}\Big)\tilde{A}_{ii}+\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}\Big(\tilde{P}_{ih}H\Big(\sum_{l=1,l\neq h}^{M}\tilde{P}_{hl}\Big)\Big)\tilde{A}_{ii}\\&-\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}\Big(\sum_{l,h=1,l\neq i}^{M}\Big(\tilde{P}_{ih}H_{ii}\tilde{P}_{il}\Big)\Big)\tilde{A}_{ii}\\&-\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}H_{ii}W_{hh}\tilde{A}_{ii}-\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}=0.\end{aligned} \tag{A.6}$$

Note that

$$\begin{aligned}&\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}\Big(\tilde{P}_{ih}H_{ii}\Big(\sum_{l=1,l\neq h}^{M}\tilde{P}_{hl}\Big)\Big)\tilde{A}_{ii}-\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}\Big(\sum_{l,h=1,l\neq i}^{M}\Big(\tilde{P}_{ih}\\&\times H_{ii}\tilde{P}_{il}\Big)\Big)\tilde{A}_{ii}=\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}\sum_{l=1,l\neq h}^{M}\tilde{P}_{ih}H_{ii}\tilde{P}_{hl}\tilde{A}_{ii}\\&-\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}\sum_{l=1,l\neq i,h}^{M}\tilde{P}_{ih}H_{ii}\tilde{P}_{hl}\tilde{A}_{ii}=\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}H_{ii}\tilde{P}_{hi}\tilde{A}_{ii}.\end{aligned} \tag{A.7}$$

Substituting (A.7) in (A.6) yields

$$\begin{aligned}&\sum_{h=1,h\neq i}^{M}\tilde{Q}_{ih}-\tilde{A}_{ii}^{\mathrm{T}}W_{ii}H_{ii}\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}\tilde{A}_{ii}+\tilde{A}_{ii}^{\mathrm{T}}\sum_{l=1,l\neq i}^{M}\Big(\Big(\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}\Big)\\&\times H_{ii}\tilde{P}_{il}\Big)\tilde{A}_{ii}+\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}H_{ii}\tilde{P}_{hi}\tilde{A}_{ii}-\tilde{A}_{ii}^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}H_{ii}W_{hh}\tilde{A}_{ii}\\&-\sum_{h=1,h\neq i}^{M}\tilde{P}_{ih}=0.\end{aligned} \tag{A.8}$$

Adding (A.8) to (A.3) with (6) gives

$$Q_1-A_1^{\mathrm{T}}W_{ii}H_{ii}W_{ii}A_1+A_1^{\mathrm{T}}\sum_{h=1,h\neq i}^{M}(\tilde{P}_{ih}H_{ii}(W_{ii}-W_{hh})A_1-W_{ii}=0. \tag{A.9}$$

Summing up (A.9) over $i\in\mathscr{M}$ results in

$$\sum_{i=1}^{M}\left[Q_1-A_1^{\mathrm{T}}W_{ii}H_{ii}W_{ii}A_1-W_{ii}\right]=0, \tag{A.10}$$

which is equivalent to

$$\sum_{i=1}^{M}\left[A_1^{\mathrm{T}}W_{ii}A_1-A_1^{\mathrm{T}}W_{ii}\tilde{B}(\tilde{R}+\tilde{B}^{\mathrm{T}}W_{ii}\tilde{B})^{-1}\tilde{B}^{\mathrm{T}}W_{ii}A_1+Q_1-W_{ii}\right]=0. \tag{A.11}$$

Given $\tilde{A}$ and $\tilde{B}$ as block diagonal matrices, having identical blocks, implies that discrete ARE in (10) is a set of M identical discrete AREs. Every $W_{ii}=\sum_{h=1}^{M}\tilde{p}_{ih}$ in (A.1) is identical, and (A.11) is a set of M identical discrete AREs

$$M(A_1^{\mathrm{T}}W_{ii}A_1-A_1^{\mathrm{T}}W_{ii}\tilde{B}(\tilde{R}+\tilde{B}^{\mathrm{T}}W_{ii}\tilde{B})^{-1}\tilde{B}^{\mathrm{T}}W_{ii}A_1+Q_1-W_{ii})=0. \tag{A.12}$$

Putting $W_{ii}=P_1$ in (A.12) gives (12). This finishes the proof. □

B. Proof of Lemma 1

This Lemma implies that the r-dimensional state ξ_k retains the behavior of nM-dimensional state $\tilde{x}_k$, if Y satisfies Remark 8. From (26) and (5), the performance function with respect to the truncated dimension state ξ_k becomes (25).

Given structure of $\tilde{P}>0$ in Corollary 1, the truncated cost matrix $\tilde{P}_r$ spans analogous eigenvalues as $\tilde{P}$, i.e., $S(\tilde{P}_r)=S(Y\tilde{P}Y^{\dagger})$. The spectrum of $\tilde{P}$ [10] is

$$\begin{aligned}S(\tilde{P})&=S(I_M\otimes P_1-\mathscr{L}\otimes P_2)\\&=\bigcup_{i\in\mathscr{M}}S(P_1-\lambda_i(\mathscr{L})P_2),\end{aligned} \tag{B.1}$$

where $\lambda_i(\mathscr{L})\in S(\mathscr{L})$. Then, the optimal control learning for nM-dimensional system in (3), with global performance

function J, is analogous to learning for the truncated system (24) with the performance function in (33). This finishes the proof. □

C. Proof of Theorem 3

It is seen from Lemma 1 that, given the large-scale networked system in (3), ξ_k satisfies (24), i.e., $\tilde{x_k} \approx Y^\dagger \xi_k$ holds $\forall \tilde{u}_k, \tilde{x}_k$ with the stable $Y\tilde{A}Y^\dagger$. Then, the control policy $\tilde{u}_k = -\tilde{K}_r^{j+1}\xi_k$ stabilizes system (24) at every iteration j, where $\tilde{K}^{j+1} = Y\tilde{K}_r{}^{j+1}$. Therefore, $Y\tilde{A}Y^\dagger - Y\tilde{B}\tilde{K}_r$ is Hurwitz. Given Remark 8, if Y is found then, similar to Algorithm 1, decomposed off-policy RL can be achieved, where $\tilde{K}_r = Y^\dagger \tilde{K} \in \mathbb{R}^{mM\times r}$ is

$$\tilde{K_r}^* = (R + (Y\tilde{B})^{\mathrm{T}}\tilde{P}_r^*(Y\tilde{B}))^{-1}(Y\tilde{B})^{\mathrm{T}}\tilde{P}_r^*(Y\tilde{A}), \tag{C.1}$$

and $\tilde{P}_r = Y\tilde{P}Y^\dagger > 0 \in \mathbb{R}^{r\times r}$ satisfies

$$\begin{aligned}\tilde{P_r}^* = {} & (Y\tilde{A})^{\mathrm{T}}\tilde{P_r}^*(Y\tilde{A}) - (Y\tilde{A})^{\mathrm{T}}\tilde{P}_r^*(Y\tilde{B})(R+ \\ & (Y\tilde{B})^{\mathrm{T}}\tilde{P}_r^*(Y\tilde{B})^{-1})(Y\tilde{B})^{\mathrm{T}}\tilde{P}_r^*(Y\tilde{A}) + \tilde{Q}_r.\end{aligned} \tag{C.2}$$

Per Remark 5, Algorithm 1 converges to the optimal solution $(\tilde{K}^*, \tilde{P}^*)$. Let $\tilde{K}_r$ be the initial stabilizing feedback gain matrix. Then, similar to Algorithm 1, given Remark 5, $(\tilde{P}_r{}^{j+1}, \tilde{K}_r{}^{j+1})$ is uniquely obtained by LS in (43) while satisfying the full-rank condition of ψ_r^j in (42) derived from data matrices in (40). Thus, Algorithm 2 converges to Algorithm 1, i.e., to the optimal solution. This finishes the proof. □

REFERENCES

[1] M. S. Mahmoud and Y. Xia, *Networked Control Systems*. Cambridge, MA, USA: Butterworth-Heinemann, 2019.

[2] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal control*. Hoboken, NJ, USA: John Wiley & Sons, 2012.

[3] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. Cambridge, MA, USA: MIT Press, 2018.

[4] S. Mukherjee, H. Bai, and A. Chakrabortty, "On model-free reinforcement learning of reduced-order optimal control for singularly perturbed systems," *IEEE Conf. Decis. Control*, pp. 5288–5293, 2018.

[5] J. Chow and P. Kokotovic, "A decomposition of near-optimum regulators for systems with slow and fast modes," *IEEE Trans. Autom. Control*, vol. 21, no. 5, pp. 701–705, 1976.

[6] T. Sadamoto, A. Chakrabortty, and J. Imura, "Fast online reinforcement learning control using state-space dimensionality reduction," *IEEE Trans. Control Netw. Syst.*, vol. 8, no. 1, pp. 342–353, 2021.

[7] G. Jing, H. Bai, J. George, and A. Chakrabortty, "Model-free optimal control of linear multiagent systems via decomposition and hierarchical approximation," *IEEE Trans. Control Netw. Syst.*, vol. 8, no. 3, pp. 1069–1081, 2021.

[8] T. Sadamoto and A. Chakrabortty, "Fast real-time reinforcement learning for partially-observable large-scale systems," *IEEE Trans. Artif. Intell.*, vol. 1, no. 3, pp. 206–218, 2020.

[9] Y. Yang, Z. Guo, H. Xiong, D.-W. Ding, Y. Yin, and D. C. Wunsch, "Data-driven robust control of discrete-time uncertain linear systems via off-policy reinforcement learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 12, pp. 3735–3747, 2019.

[10] F. Borrelli and T. Keviczky, "Distributed LQR design for identical dynamically decoupled systems," *IEEE Trans. Autom. Control*, vol. 53, no. 8, pp. 1901–1912, 2008.

[11] E. E. Vlahakis, L. D. Dritsas, and G. D. Halikias, "Distributed lqr-based suboptimal control for coupled linear systems," *Int. Fed. Autom. Control*, vol. 52, no. 20, pp. 109–114, 2019.

[12] P. J. SCHMID, "Dynamic mode decomposition of numerical and experimental data," *J. Fluid Mech.*, vol. 656, pp. 5–28, 2010.

[13] J. L. Proctor, S. L. Brunton, and J. N. Kutz, "Dynamic mode decomposition with control," *SIAM J. Appl. Dyn. Syst.*, vol. 15, no. 1, pp. 142–161, 2016.

[14] C. Eckart and G. M. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, pp. 211–218, 1936.

[15] D. Donoho, "De-noising by soft-thresholding," *IEEE Transactions on Information Theory*, vol. 41, no. 3, pp. 613–627, 1995.

[16] M. Gavish and D. L. Donoho, "The optimal hard threshold for singular values is $4/\sqrt{3}$," *IEEE Transactions on Information Theory*, vol. 60, no. 8, pp. 5040–5053, 2014.

[17] M. Budišić, R. Mohr, and I. Mezić, "Applied koopmanism," *Chaos: Interdiscip. J. Nonlinear Sci.*, vol. 22, no. 4, p. 047510, 2012.

[18] H. Modares and F. L. Lewis, "Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning," *Automatica*, vol. 50, no. 7, pp. 1780–1792, 2014.

[19] M. Johnson, S. Bhasin, and W. E. Dixon, "Nonlinear two-player zero-sum game approximate solution using a policy iteration algorithm," *IEEE Conf. Decis. Control*, pp. 142–147, 2011.

[20] B. Kiumarsi, H. Modares, F. L. Lewis, and Z.-P. Jiang, "H_∞ optimal control of unknown linear discrete-time systems: An off-policy reinforcement learning approach," in *IEEE Conf. Robot., Autom., Mechatron.*, 2015, pp. 41–46.

[21] J. Tu, C. Rowley, D. Luchtenburg, S. Brunton, and J. Kutz, "On dynamic mode decomposition: Theory and applications," *J. Comput. Dyn.*, vol. 1, no. 2, pp. 391–421, 2014.

[22] B. Moore, "Principal component analysis in linear systems: Controllability, observability, and model reduction," *IEEE Trans. Autom. Control*, vol. 26, no. 1, pp. 17–32, 1981.

[23] J. Tu and C. Rowley, "An improved algorithm for balanced POD through an analytic treatment of impulse response tails," *J. Comput. Phys.*, vol. 231, no. 16, pp. 5317–5333, 2012.

[24] C. Rowley, "Model reduction for fluids, using balanced proper orthogonal decomposition," *Int. J. Bifurcation and Chaos*, vol. 15, no. 03, pp. 997–1013, 2005.

[25] K. Kashima, "Noise response data reveal novel controllability gramian for nonlinear network dynamics," *Sci. Rep.*, vol. 6, pp. 1–8, 2016.

[26] J. Juang and R. Pappa, "An eigensystem realization algorithm for modal parameter identification and model reduction," *J. Guid., Control, Dyn.*, vol. 8, no. 5, pp. 620–627, 1985.

[27] K. Willcox and J. Peraire, "Balanced model reduction via the proper orthogonal decomposition," *AIAA J.*, vol. 40, no. 11, pp. 2323–2330, 2002.

[28] J. Kutz, S. Brunton, B. Brunton, and J. Proctor, *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems*. Philadelphia, PA, USA: SIAM, 2016.

[29] K. Glover, "All optimal hankel-norm approximations of linear multivariable systems and their L^∞-error bounds," *Int. J. Control*, vol. 39, no. 6, pp. 1115–1193, 1984.

[30] C. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. Henningson, "Spectral analysis of nonlinear flows," *J. Fluid Mech.*, vol. 641, pp. 115–127, 2009.

[31] L. Prashanth, N. Korda, and R. Munos, "Fast LSTD using stochastic approximation: Finite time analysis and application to traffic control," *Joint Eur. Conf. Mach. Learn. and Knowl. Discovery in Databases*, pp. 66–81, 2014.

[32] Z. Yan and Y. Xu, "A multi-agent deep reinforcement learning method for cooperative load frequency control of a multi-area power system," *IEEE Trans. Power Syst.*, vol. 35, no. 6, pp. 4599–4608, 2020.

[33] J. Ansari, A. R. Abbasi, and B. B. Firouzi, "Decentralized LMI-based event-triggered integral sliding mode lfc of power systems with disturbance observer," *Int. J. Electr. Power Energy Syst.*, vol. 138, p. 107971, 2022.

[34] K. Vrdoljak, N. Perić, and I. Petrović, "Sliding mode based load-frequency control in power systems," *Electr. Power Syst. Res.*, vol. 80, no. 5, pp. 514–527, 2010.

[35] V. S. Donge, B. Lian, F. L. Lewis, and A. Davoudi, "Multi-agent graphical games with inverse reinforcement learning," *IEEE Trans. Control Netw. Syst. doi: 10.1109/TCNS.2022.3210856*, 2022.

[36] W. Xue, B. Lian, J. Fan, P. Kolaric, T. Chai, and F. L. Lewis, "Inverse reinforcement Q-learning through expert imitation for discrete-time systems," *IEEE Trans. Neural Netw. Learn. Syst. doi: 10.1109/TNNLS.2021.3106635*, 2021.

Vrushabh S. Donge received the B.E. degree in electrical engineering from the Government College of Engineering, Aurangabad, Maharashtra, India, in 2016, and the M.Tech. degree in control systems from the Veermata Jijabai Technological Institute, Mumbai, Maharashtra, India, in 2020. He is currently working toward the Ph.D. degree in electrical engineering with the University of Texas at Arlington, Arlington, TX, USA. His research interests include optimal control, reinforcement learning, multi-agent systems and distributed control.

Bosen Lian received his Ph.D. in the Electrical Engineering from the University of Texas at Arlington, TX, USA, in 2021. He is currently an Adjunct Professor at the Electrical Engineering Department and a Postdoctoral Research Associate at University of Texas at Arlington Research Institute. His research interests focus on reinforcement learning, inverse reinforcement learning, distributed estimation and distributed control.

Frank L. Lewis (S'70-M'81-SM'86-F'94) received the bachelor's degree in physics/electrical engineering and the M.S.E.E. degree from Rice University, Houston, TX, USA, in 1971, the M.S. degree in aeronautical engineering from the University of West Florida, Pensacola, FL, USA, in 1977, and the Ph.D. degree from the Georgia Institute of Technology, Atlanta, GA, USA, in 1988.

He is currently a member National Academy of Inventors, IEEE Fellow, IFAC Fellow, Fellow Inst. Measurement & Control, PE Texas, U.K. Chartered Engineer, UTA Distinguished Scholar Professor, UTA Distinguished Teaching Professor, and Moncrief-O'Donnell Chair at the Uni- versity of Texas at Arlington Research Institute. He is author of 7 U.S. patents, numerous journal special issues, numerous journal papers, and 20 books. He received the Fulbright Research Award, NSF Research Initiation Grant, ASEE Terman Award, Int. Neural Network Soc. Gabor Award, U.K. Inst Measurement & Control Honeywell Field Engineering Medal, IEEE Computational Intelligence Society Neural Networks Pioneer Award, AIAA Intelligent Systems Award. Was listed in Ft. Worth Business Press Top 200 Leaders in Manufacturing. Texas Regents Outstanding Teaching Award 2013. Founding Member of the Board of Governors of the Mediterranean Control Association.

Ali Davoudi (S'04-M'11-SM'15-F'23) received his Ph.D. in Electrical and Computer Engineering from the University of Illinois, Urbana-Champaign, IL, USA, in 2010. He is currently a Professor in the Electrical Engineering Department, University of Texas, Arlington, TX, USA. He is an Associate Editor for the IEEE TRANSACTIONS ON POWER ELECTRONICS, and an Editor for the IEEE TRANSACTIONS ON ENERGY CONVERSION as well as IEEE POWER ENGINEERING LETTERS. His research interests include modeling, control, and optimization of power systems.