\( \mathcal{H}_2 \)-Clustering of Closed-loop Consensus Networks under a Class of LQR Design

Nan Xue\(^1\) and Aranya Chakrabortty\(^2\)

Abstract—In this paper we address the problem of clustering closed-loop consensus networks where the closed-loop controller is designed using a class of Linear Quadratic Regulator (LQR). Given any positive integer \( r \), our objective is to develop a strategy for grouping the states of the \( n \)-node network into \( r \leq n \) distinct non-overlapping groups. The criterion for this partitioning is defined as follows. First, a LQR controller is defined for the original \( n \)-node network. Then, a \( r \)-dimensional reduced-order network is constructed by imposing a projection matrix \( P \) on the \( n \)-node open-loop network, and a reduced-order \( r \)-dimensional LQR controller is constructed for this reduced-order system. The resulting controller is, thereafter, projected back to its original coordinates, and implemented in the \( n \)-node network. The problem, therefore, is to find a grouping strategy or \( P \) that will minimize the difference between the closed-loop transfer matrix of the original network with the full-order controller and that with the projected controller, in the sense of \( \mathcal{H}_2 \) norm. We derive an upper bound on this difference in terms of \( P \), and, thereby propose a design for \( P \) using K-means that tightens the bound while guaranteeing numerical feasibility. We discuss the computational benefits of the method, and illustrate the trade-off between \( r \) and \( \mathcal{H}_2 \)-performance using two network simulations.

Index Terms—Clustering, Consensus, Network Dynamic Systems, Grouping, Controller Order Reduction, LQR, \( \mathcal{H}_2 \)-performance.

I. INTRODUCTION

Designing control systems for stability and performance of consensus networks has been a problem of perennial interest in network science and engineering. Over the past two decades, seminal papers such as [1]-[3], and references therein, have proposed various sets of control designs for stabilization of complex consensus networks, while those such as [4]-[7] have proposed different optimization and adaptation techniques to enhance their closed-loop performance. Translating these designs to networks with very large number of nodes in practice, however, is a rather challenging task. For example, majority of today’s networks, ranging from power system networks to wireless networks to social or biological networks, consist of thousands to tens of thousands of nodes. Designing complicated controllers for such large networks is not only computationally taxing but also severely limited by scalability. Network engineers are, therefore, often interested in exploring model reduction techniques that can help them in designing simpler controllers by exploiting the inherent topological structure of the network.

There exists an extensive literature, developed mostly in the 1980’s for large-scale power system networks, on how network structure can be used for open-loop model reduction via clustering of network nodes. The fundamental tools used for such clustering include singular perturbation theory [8], synchronic modal equivalencing [9], Krylov projections [10] [11], etc. Recent papers such as [12], [13] have presented clustering techniques using \( \mathcal{H}_\infty \) and \( \mathcal{H}_2 \) norm optimization. Almost all of these results, however, pertain to open-loop analysis, and not closed-loop control. There is very limited insight on how model reduction via clustering of large-scale networks can be used for designing simpler controllers. Some preliminary results were presented recently in [14], but the clustering was still imposed on the open-loop network without any insight on how it can be extended to benefit control.

To bridge this gap, in this paper we develop a strategy for grouping the closed-loop states of a \( n \)-node consensus network into \( r \leq n \) distinct non-overlapping groups, where \( r > 0 \) is a desired number of clusters, specified by the network administrator. The criterion for this partitioning is defined as follows. First, we define a LQR controller for the original \( n \)-node network under certain assumptions on \( Q \) and \( R \). Note that this LQR controller only needs to be defined in theory for analyses, it does not need to be ‘designed’ in practice. The closed-loop transfer matrix (TM) of the network output from a disturbance input under this full-order LQR controller is defined as the benchmark TM. Then, a \( r \)-dimensional reduced-order network is created by imposing a projection matrix \( P \) on the \( n \)-node open-loop network, and a reduced-order \( r \)-dimensional LQR controller is constructed for this reduced-order system. The resulting controller is, thereafter, projected back to its original coordinates, and implemented in the \( n \)-node network. The problem, therefore, is to find a grouping strategy or \( P \) that will minimize the difference between the benchmark TM and the closed-loop transfer matrix of the original network with the projected controller in terms of their \( \mathcal{H}_2 \) norms. We derive an upper bound on this difference as a function of \( P \), and, thereby propose a design for \( P \) using K-means that tightens the bound while guaranteeing numerical feasibility. Clustering of networks, in general, is accepted as a NP-hard problem, and hence a lot of research has been devoted by numerical graph theorists on developing heuristic algorithms over different types of graphs. An extensive survey of these numerical clustering algorithms can be found in [15]. Two of such widely used algorithms are K-means [16] and support vector clustering (SVC) [17]. Technically speaking, either of these

N. Xue and A. Chakrabortty are with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC, 27695 USA, e-mail: nxue@ncsu.edu, achakra2@ncsu.edu

The work is supported partly by the US Department of Energy grant DE-OE0000654

Preprint submitted to 2016 American Control Conference.
Received September 28, 2015.
two algorithms, or any other alternate algorithm, can be used for constructing $P$ for our problem. For the sake of simplicity, however, in this paper we stick to only $K$-means. Our approach extends the $\mathcal{H}_\infty$-norm based clustering proposed in [12] to an $\mathcal{H}_2$ equivalent for LQR.

**NOTATION** The following notations will be used throughout this paper: $n$: the dimension of full-order, $r$: the dimension of reduced-order, $I_n$: a column vector of size $n$ with all 1 entries, $I_k$: the unitary matrix of size $k$, $E_{ij}$: indicator matrix with all entries equal to 0 but $(i,j)$ equal to 1, $\|M\|_F$: Frobenius norm of $M$, i.e. $\|M\|_F = \sqrt{tr(MM^T)}$, $|m|$: absolute value of $m$, $|S|_c$: cardinality of a set $S$, $\text{diag}(m)$: diagonal matrix with the vector $m$ on its diagonal, $[M_{i,j}]$: a matrix whose $(i,j)$ element is defined by $M_{i,j}$. A transfer matrix is defined as $g(s) = C(sI_n - A)^{-1}B + D$, with a realization form of $g(s) = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$. $g(s)$ is stable if $A < 0$, or marginally stable if $A \succeq 0$. Furthermore, the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norm of a stable transfer matrix $g(s)$ are defined by $\|g(s)\|_{\mathcal{H}_2} = \sqrt{\int_0^\infty \text{tr}[g^*(t)g(t)]dt} = \sqrt{\int_0^\infty \text{tr}[g^*(\omega)g(\omega)]d\omega}$ and $\|g(s)\|_{\mathcal{H}_\infty} = \sup_{\omega} \sigma[g(\omega)]$, where $\sigma$ represents the largest singular value.

**II. PROBLEM FORMULATION**

Consider a consensus network defined over a graph $\mathcal{G}$ containing $n$ nodes, each defined by a scalar state variable $x_i$ and a scalar control input $u_i$, $i = 1, \ldots, n$. To formulate our problem, we define the following four representations of this model - namely, (1) the full-order open-loop model, (2) the full-order closed-loop model with a full-order LQR controller, (3) the reduced-order open-loop model for designing a reduced-order LQR controller, and finally, (4) the full-order closed-loop model with a projected LQR controller. To begin with, the full-order open-loop system is defined as

$$
\begin{align*}
\dot{x}(t) &= Ax(t) + Bu(t) + Bd(t) \\
y(t) &= Cx(t)
\end{align*}
$$

where $A \in \mathbb{R}^{n \times n}$, $B = I_n$, $b \in \mathbb{R}^n$, $C = I_n$ (for full state feedback), $Q \in \mathbb{R}^{n \times n}$, $R \in \mathbb{R}^{n \times n}$, and $x(t) \in \mathbb{R}^n$ represent the vector of state and control variables. $d(t) \in \mathbb{R}$ is a scalar disturbance input entering any node in the network, while $b \in \mathbb{R}^n$ is a non-zero vector. Since the network is running consensus, $A$ is an edge-weighted graph Laplacian defined by

$$A(i,j) = \begin{cases} w_{i,j} & i \neq j \\
-\sum_{i} w_{i,j} & i = j \end{cases}$$

where $w_{i,j}$ is the weight of the edge connecting the $i^{th}$ and the $j^{th}$ node. By definition, $A \preceq 0$ contains a 0 eigenvalue spanned by $I_n$ [1]. We assume $A$ to be diagonalizable. In addition, we assume $Q = \gamma I_n$ and $R = \nu I_n$ throughout this paper, where $\gamma$ and $\nu$ are positive scalars. The LQR design for this system is then posed as finding a full-state feedback $u(t) = -Kx(t)$, where $K = K^T > 0$ such that the cost function

$$J = \int_0^\infty z^T(t)z(t)dt = \int_0^\infty [x^T(t)Qx(t) + u^T(t)Ru(t)]dt$$

is minimized. To facilitate the analysis, we denote the TMs of the plant and the controller of (1) by

$$G(s) := (sI_n - A)^{-1}, \quad K(s) := \begin{bmatrix} 0 \\ 0 \\ R \end{bmatrix} = K.$$  

Hence, we can write the closed-loop TM from disturbance $d$ to output $y$ as

$$g_c(s) := (sI_n - A + K)^{-1}b = G(s)[I_n + K(s)G(s)]^{-1}b = G_c(s)b.$$  

The model (5) represents the full-order closed-loop system with full-order LQR control.

We next define a state aggregation strategy to repose the control problem using a reduced-order controller as follows. 

**Definition 2.1**: Given an integer $r > 0$, define $r$ non-empty, distinct, and non-overlapping sub-sets of the state variables $\{x_1, \ldots, x_n\}$, denoted as $S_1, \ldots, S_r$ such that $S_1 \cup \ldots \cup S_r = \{x_1, \ldots, x_n\}$, the aggregate state $\tilde{x} = [\tilde{x}_1, \ldots, \tilde{x}_r]^T \in \mathbb{R}^r$ is

$$\tilde{x} = Px,$$

where the projection matrix $P \in \mathbb{R}^{r \times n}$ is defined by

$$P(i,j) := \begin{cases} \frac{1}{\sqrt{|S_i|}} & x_j \in S_i \\
0 & \text{otherwise} \end{cases}$$

with $r \leq n$ and $PP^T = I_r$. For example, $P$ corresponding to a clustering strategy of $S_1 = \{x_1, x_2, x_3\}$, $S_2 = \{x_4, x_5\}$ and $S_3 = \{x_6\}$ can be written as

$$P = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$  

We next apply the projection $P$ on the open-loop model (1) to define an aggregate model:

$$\begin{align*}
\dot{\tilde{x}}(t) &= \tilde{A}\tilde{x}(t) + \tilde{B}\tilde{u}(t) + \tilde{b}d(t) \\
\dot{\tilde{y}}(t) &= \tilde{C}\tilde{x}(t) \\
\dot{\tilde{z}}(t) &= \tilde{Q}\tilde{x}(t) + \tilde{R}\tilde{u}(t)
\end{align*}$$

where $\tilde{A} = PAP^T \in \mathbb{R}^{r \times r}$, $\tilde{B} = PBP^T = I_r$, $\tilde{b} = Pb \in \mathbb{R}^r$, $\tilde{C} = PCP^T = I_r$, $\tilde{Q} = PQP^T = \gamma I_r$ and $\tilde{R} = PRP^T = \nu I_r$. $\tilde{u}(t) \in \mathbb{R}^r$ is similarly projected by $\tilde{u}(t) = Pu(t)$.

Next, we design a reduced-order LQR controller using $\tilde{u} = -\tilde{K}\tilde{x}$, where $\tilde{K} = \tilde{K}^T \in \mathbb{R}^{r \times r}$, and denote the reduced-order plant and controller as

$$\tilde{G}(s) := (sI_r - PAP^T)^{-1}, \quad \tilde{K}(s) := \tilde{K}.$$  

Finally, we project the reduced-order controller to its original coordinates and implement it in the full-order model.
using \( u = P^T \hat{u} = -P^T \hat{X} P \) to the full-order system (1), which implies that in this case the effective feedback matrix is \( P^T \hat{K} \). Therefore, the closed-loop system implemented with the projected controller can be written as

\[
\hat{g}_c(s) := (sI_n - A + P^T \hat{K} P)^{-1} b = G(s)[I_n + P^T \hat{K} P G(s)]^{-1} b = \hat{G}_c(s)b. \tag{11}
\]

Using these definitions, we next state our problem of \( H_2 \) clustering for closed-loop consensus network as follows:

**Problem Statement:** Given system (1) and \( r \), find a clustering strategy \( S \) such that the corresponding \( P \) matrix minimizes \( \|g_c(s) - \hat{g}_c(s)\|_{H_2} \).

However, since the problem itself is nonconvex, as a result of which it becomes impractical to find a global minimum. Hence in this paper, we take a hint on deriving an upper bound for \( \|g_c(s) - \hat{g}_c(s)\|_{H_2} \), from which the \( P \) selected can push the bound to sufficiently small. Once the problem is solved, two main advantages can therefore be brought up:

1. The LQR design problem is projected to a lower dimension \( r \), which simplifies the design of a feedback matrix.
2. The feedback matrix in lower dimension is re-projected to full dimension. The recovered feedback matrix will have same entries corresponding to the nodes assigned to same cluster. That means we only need to compute one control signal for all the nodes inside one cluster, which would simplify the implementation especially when \( n \gg r \). An illustrative example is shown below, where \( n = 6, r = 3 \) and \( P \) is taken from (8).

\[
\begin{pmatrix}
3* & \sqrt{5}\diamond & \sqrt{3}\downarrow \\
\sqrt{6}\circ & 2\downarrow & \sqrt{2}\uparrow \\
\sqrt{3}\uparrow & \sqrt{2}\downarrow & \\
\end{pmatrix} \rightarrow \begin{pmatrix}
\ast & \ast & \ast & \ast & \ast & \ast \\
\ast & \ast & \ast & \ast & \ast & \ast \\
\ast & \ast & \ast & \ast & \ast & \ast \\
\ast & \ast & \ast & \ast & \ast & \ast \\
\ast & \ast & \ast & \ast & \ast & \ast \\
\ast & \ast & \ast & \ast & \ast & \ast \\
\end{pmatrix}
\]

\[
P^T \hat{K} P
\]

**III. Recap of \( H_2 \) Clustering for Open-Loop Consensus Network**

\( H_2 \) clustering on a positive system, namely \( A \succeq 0 \) being a Metzler matrix, is detailed in [13]. Since a Laplacian matrix is also Metzler, we apply the main theorem from [13] here without proof.

**Theorem 3.1:** Given \( A \) and \( b \) in (1), define \( \Phi := \Phi_2 \Phi_2^T \),

\[
\Phi := \int_{0}^{\infty} e^{W T} W W^{T} W b b^{T} W T e^{W^{T} T} W^{T} W d T W,
\]

where \( W \) is the complement of \( I_n^{T} / \sqrt{n} \). The aggregate model \((P A P^T, \Phi)\) associated with \( P \) satisfies

\[
\|g(s) - \hat{g}(s)\|_{H_2} \leq \|\Xi(s)\|_{H_\infty} \theta, \tag{12}
\]

where \( \Xi(s) \) is stable and

\[
\begin{align*}
g(s) := (sI_n - A)^{-1} b = G(s)b, \\
\hat{g}(s) := P^T (sI_n - P A P^T)^{-1} P b = P^T \hat{G}(s)b, \\
\Xi(s) := P^T (sI_n - P A P^T)^{-1} P A + I_n, \\
\theta := \|P^T \Phi_2 - \Phi_2^T\|_F.
\end{align*}
\]

Theorem 3.1 provides a metric for clustering a open-loop system, where the \( H_2 \) norm of the error system \( g(s) - \hat{g}(s) \) is upper bounded by a function of \( \theta, \theta \) represents the \( K \)-means error of \( \Phi_2 \). In this sense, the clustering matrix \( P \) can be found by providing \( \Phi_2 \) as an input to any typical clustering algorithm such as \( K \)-means and SVC. In this paper, we will only utilize \( K \)-means as the clustering algorithm.

Since \( A = A^T \), a straightforward expression on \( g_c(s) - \hat{g}_c(s) \) is derived as

\[
g_c(s) - \hat{g}_c(s) = [P^T \hat{K} P - K(s)]G_c(s)\hat{G}_c(s)b, \tag{13}
\]

which further yields the inequality

\[
\|g_c(s) - \hat{g}_c(s)\|_{H_2} \leq \|P^T \hat{K} P - K\|_F \|G_c(s)\hat{G}_c(s)b\|_{H_\infty}. \tag{14}
\]

Since \( G_c(s)\hat{G}_c(s)b \) is stable or, equivalently, \( \|G_c(s)\hat{G}_c(s)b\|_{H_\infty} \) is bounded, the \( H_2 \) norm of \( g_c(s) - \hat{g}_c(s) \) is linearly dependent on \( \|P^T \hat{K} P - K\|_F \). By Theorem 3.1, a natural choice for \( P \) can follow from minimizing \( \|P^T \hat{K} P - K\|_F \) with respect to \( P \). However, different from (12), the difficulty of this problem lies in the fact that a direct projection \( P \) only applies from (1) to (9) but not from \( K \) to \( \hat{K} \). In other words, Theorem 3.1 cannot quantify the relation between \( P \) and \( \|P^T \hat{K} P - K\|_F \). Hence in the following section, we present an alternative approach for solving the problem by constructing a new bound on \( \|g_c(s) - \hat{g}_c(s)\|_{H_2} \), which will reveal the rule for selecting \( P \).

**IV. \( H_2 \) Clustering for Closed-Loop Consensus Network**

To circumvent the problem of relating \((P^T \hat{K} P - K)\) to \( P \), we introduce an intermediate TM \( \tilde{g}_c \), which is defined as the TM from \( d \) to \( P^T \tilde{y} \) as

\[
\tilde{g}_c(s) := P^T(sI - \tilde{A} + \hat{K})^{-1}Pb = P^T \tilde{G}(s)[I_r + \hat{K}(s)\tilde{G}(s)]^{-1}Pb = \hat{G}_c(s)b, \tag{15}
\]

and rewrite the objective as

\[
\|g_c(s) - \hat{g}_c(s)\|_{H_2} = \|g_c(s) - \hat{g}_c(s) + \hat{g}_c(s) - \hat{g}_c(s)\|_{H_2}
\leq \|g_c(s) - \hat{g}_c(s)\|_{H_2} + \|\hat{g}_c(s) - \hat{g}_c(s)\|_{H_2}. \tag{16}
\]

In (16), the objective is bounded by \( H_2 \) norms of two error systems, namely, \( g_c(s) - \hat{g}_c(s) \), the error between full-order and reduced-order closed-loop systems, and \( \hat{g}_c(s) - \hat{g}_c(s) \), the error between implementing the same controller on reduced-order and full-order open-loop systems. In the following two subsections, we will detail the derivation of these two error terms separately, then combine them together to form the algorithm for selecting \( P \).

**A. Upper Bound on \( \|g_c(s) - \hat{g}_c(s)\|_{H_2} \)**

The bound for \( \|g_c(s) - \hat{g}_c(s)\|_{H_2} \) is given by the following theorem.

**Theorem 4.1:** Given the aggregated system and a LQR feedback controller \( \hat{K}(s) \) in (9), when the recovered controller \( P^T \hat{K}(s)P \) is applied to the open-loop system (1), then the closed-loop transfer matrix \( \hat{g}_c(s) \) will satisfy

\[
\|\hat{g}_c(s) - \hat{g}_c(s)\|_{H_2} \leq \|\Psi(s)\|_{H_\infty} \theta, \tag{17}
\]

CONFIDENTIAL. Limited circulation. For review only.
where
\[ \Psi(s) = -\Xi(s)P^T[I_r + \hat{K}(s)\hat{G}(s)]^{-1}P[I_n + \hat{K}(s)G(s)]^{-1}I_n \]
is stable, with the same \( \theta \) and \( \Xi(s) \) as in Theorem 3.1.

**Proof:** (I): Define \( U \) as the complement of \( P \), say
\[ [P^T \quad U^T]^T \] is unitary, then
\[
\hat{\gamma}_e(s) - \hat{\gamma}_c(s) = P^T\hat{G}(s)PP^T[I_r + \hat{K}(s)\hat{G}(s)]^{-1}Pb
\]
\[ - G(s)[I_n + \hat{K}(s)G(s)]^{-1}b \]
\[ = P^T\hat{G}(s)PT_2^{-1}T_2P^TP_T \]
\[ - G(s)P^TPT_2 \]
\[ = [-\Xi(s)U^TUG(s)b + U^TUG(s)b]P^TPT_2b \]
\[ - U^TUG(s)b \]
\[ = [I_n - \Xi(s)PT_1PT_2 - T_2U^TUG(s)b]. \]

Taking norm on both sides, we have
\[
\|\hat{\gamma}_e(s) - \hat{\gamma}_c(s)\|_{\mathcal{H}_2} \leq \|\Psi(s)\|_{\mathcal{H}_\infty}\|U^TUG(s)b\|_{\mathcal{H}_2}. \]

Since \( \|U^TUG(s)b\|_{\mathcal{H}_2} = \theta \) from [13], this proves the theorem.

(II): The stability of \( \Psi(s) \) is equivalent to the stability of \( T_1 \) and \( T_2 \). Since \( \Xi(s) \) is stable by Theorem 3.1, \( T_1 \) and \( T_2 \) can be written as
\[
T_1(s) = -\hat{K}(sI_r - \hat{A} + \hat{K})^{-1}Pb + I_r, \quad (18)
\]
\[
T_2(s) = -P^T\hat{K}P(sI_n - A + P^T\hat{K}P)^{-1}b + I_n, \quad (19)
\]
where \( \hat{A} - \hat{K} < 0 \). The negativeness of \( A - P^T\hat{K}P < 0 \) holds if and only if any non-zero vector \( x \) satisfies
\[
x(A - P^T\hat{K}P)x^T = xAx^T - xPT\hat{K}Ptx^T < 0,
\]
where \( xAx^T \leq 0 \) and \( -xPT\hat{K}Ptx^T \leq 0 \). Note that
\[
xAx^T = 0 \quad \text{if and only if} \quad x \in \{1\},
\]
while
\[
-xPT\hat{K}Ptx^T = 0 \quad \text{if and only if} \quad x^T \in \text{span}\{W^T\}.
\]
Therefore, it follows that \( A - P^T\hat{K}P < 0 \).

**Remark 4.2:** For the interest of this paper, the controller \( \hat{K}(s) \) is limited to the LQR feedback. However, note that for any arbitrary controller \( \hat{K}(s) \), (17) holds if and only if \( \hat{K}(s) \) satisfies \( \hat{K}(s) = \hat{K}^T(s) \), and \( \hat{\gamma}_e(s) \) and \( \hat{\gamma}_c(s) \) are both internally stable.

### B. Two Supporting Lemmas

Before delving into the LQR theory, we first give the definition of Cauchy matrix, and provide two lemmas which will help us deriving the bound of \( \|\hat{\gamma}_e(s) - \hat{\gamma}_c(s)\|_{\mathcal{H}_2} \).

**Definition 4.3:** Given \( \Lambda = \text{diag}([\lambda_1, \ldots, \lambda_n]) \) and \( \Lambda = \text{diag}([\mu_1, \ldots, \mu_r]) \), where \( \Lambda \) and \( M \) are non-singular, a Cauchy matrix \( C_{\Lambda, M} \) is defined by
\[
C_{\Lambda, M} := \begin{bmatrix}
\frac{1}{\lambda_1 + \mu_1} & \frac{1}{\lambda_2 + \mu_2} & \cdots & \frac{1}{\lambda_n + \mu_n} \\
\frac{1}{\lambda_2 + \mu_1} & \frac{1}{\lambda_2 + \mu_2} & \cdots & \frac{1}{\lambda_n + \mu_n} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{1}{\lambda_n + \mu_1} & \frac{1}{\lambda_n + \mu_2} & \cdots & \frac{1}{\lambda_n + \mu_n}
\end{bmatrix}. \tag{20}
\]

The first lemma guides us in constructing the controllability Gramian.

**Lemma 4.4:** For a stable pair \( (A, b) \), with the eigenvalue decomposition of \( A \) written as
\[
A = TAT^{-1} = [v_1 \cdots v_n] \begin{bmatrix}
\lambda_1 & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & \lambda_n
\end{bmatrix} \begin{bmatrix}
v_1^T \\
\vdots \\
v_n^T
\end{bmatrix}, \tag{21}
\]
the controllability Gramian \( \Phi := \int_0^\infty e^{A\tau}bb^Te^{A^T\tau}d\tau \) can be found through
\[
\Phi = R\mathcal{C}_{\Lambda, \Lambda}R^T \tag{22}
\]
where the residue matrix \( R \) is defined by
\[
R := [v_1w_1 \cdots v_nw_nb] = T \cdot \text{diag}(T^{-1}b). \tag{23}
\]

The proof follows the basic definition of controllability Gramian and is sketched in the appendix. The next lemma states a general expression for the \( \mathcal{H}_2 \) norm of the error between two stable systems:

**Lemma 4.5:** Given \( P \) as defined in (7), and stable TMs \( g_1 = (sI_n - A)^{-1}b \) and \( g_2 = P^T(sI_r - PAP^T)^{-1}Pb \), the \( \mathcal{H}_2 \) norm of the error system \( g_1 - g_2 \) is given by
\[
\|g_1 - g_2\|_2^2 = \text{trace}(R_1C_{\Lambda_1, \Lambda_1}R_1^T - R_1C_{\Lambda_1, \Lambda_2}R_2^T + R_2C_{\Lambda_2, \Lambda_1}R_1^T + R_2C_{\Lambda_2, \Lambda_2}R_2^T), \tag{24}
\]
where \( R_1 = T_1\text{diag}(T_1^{-1}b) \) and \( R_2 = P^T T_2\text{diag}(T_2^{-1}Pb) \), with the eigenvalue decompositions \( A = T_1\Lambda_1T_1^{-1} \) and \( \hat{A} = T_2\Lambda_2T_2^{-1} \).

**Proof:** See Appendix.

#### C. Upper Bound on \( \|\hat{\gamma}_e(s) - \hat{\gamma}_c(s)\|_{\mathcal{H}_2} \)

Since \( \hat{\gamma}_e(s) \) and \( \hat{\gamma}_c(s) \) are both equipped with LQR feedback, we start the discussion on \( \|\hat{\gamma}_e(s) - \hat{\gamma}_c(s)\|_{\mathcal{H}_2} \) from the basic LQR theory. Corresponding to (3), the ARE is written as
\[
A^TX + XA + Q - XBR^{-1}BTX = 0 \tag{25}
\]
where feedback matrix \( K = R^{-1}BTX \) and \( X > 0 \). The Hamiltonian matrix corresponding to (25) can be written as
\[
H = \begin{bmatrix}
A & -BR^{-1}BT \\
-Q & -A^T
\end{bmatrix} = \begin{bmatrix}
A & -\frac{1}{\nu}I_n \\
-\gamma I_n & -A
\end{bmatrix}. \tag{26}
\]
Suppose the columns of the matrix \( \begin{bmatrix} X_1 & X_2 \end{bmatrix}_{2n \times n} \) span the stable invariant subspace of \( H \), i.e.
\[
H \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \Lambda^- \tag{27}
\]
where \( \Lambda^{-} = \text{diag}(\lambda_{1}^{-}, \ldots, \lambda_{n}^{-}) < 0 \). The stabilizing solution of ARE can then be found by \( X = X_{2}X_{1}^{-1} \) [18].

Taking the similarity transformation of \( T = \begin{bmatrix} I & 0 \\ X & I \end{bmatrix} \) on \( H \), we shall have

\[
THT^{-1} = \begin{bmatrix} A - BR^{-1}B^{T}X & -BR^{-1}B^{T} \\ 0 & -(A - BR^{-1}B^{T}X)^{T} \end{bmatrix},
\]

which shows that \( H \) consists of the closed-loop eigenvalues of \( A - BK \) and eigenvalues of its negative \( -(A - BK) \). Another important property of Hamiltonian is that normalized \( X_{1} \), denoted by \( x_{1} \), is the right eigenspace of \( A - BK \). Hence,

\[
A - BK = x_{1} \Lambda^{-} x_{1}^{-1} = X_{1} \Lambda^{-} X_{1}^{-1}.
\]

Recall that for reduced-order system (9), the corresponding LQR design is posed by the following ARE

\[
PA^{T}P^{T}X + \bar{X}PAP^{T} + PQP^{T} - \bar{X}PBBR^{-1}B^{T}P^{T}\bar{X} = 0
\]

and the Hamiltonian matrix

\[
\tilde{H} = \begin{bmatrix} P \\ P \end{bmatrix} \begin{bmatrix} A & -BR^{-1}B^{T} \\ -Q & -A^{T} \end{bmatrix} \begin{bmatrix} P^{T} \\ P^{T} \end{bmatrix}.
\]

\( \tilde{H} \) can be similarly written as

\[
\tilde{H} \tilde{X}_{1} X_{2} = \begin{bmatrix} \tilde{X}_{1} \\ \tilde{X}_{2} \end{bmatrix} \tilde{\Lambda}^{-},
\]

where \( \tilde{\Lambda}^{-} = \text{diag}(\tilde{\lambda}_{1}^{-}, \ldots, \tilde{\lambda}_{n}^{-}) < 0 \) and normalized \( \tilde{X}_{1} \) is denoted by \( \tilde{x}_{1} \). Here we denote the eigenvalue decomposition of \( A \) and \( \tilde{A} \) by

\[
A = x\Lambda^{A} x^{T}, \quad \tilde{A} = \tilde{x}\Lambda^{\tilde{A}} \tilde{x}^{T},
\]

where \( \Lambda^{A} = \text{diag}(\lambda_{1}^{A}, \ldots, \lambda_{n}^{A}) \) and \( \Lambda^{\tilde{A}} = \text{diag}(\tilde{\lambda}_{1}^{A}, \ldots, \tilde{\lambda}_{n}^{A}) \) are both stable. Hence we are able to write the \( \Lambda^{A} \) and \( X_{1} \) from \( H \) in terms of eigenvalues and eigenvectors of \( A \) explicitly as follows.

**Proposition 4.6:** For \( H \) and \( A \) expressed by (26) and (32), the stable eigenvalues of \( H \) satisfies the following two equations:

\[
(\lambda_{i}^{A})^{2} - (\lambda_{i}^{-})^{2} = -\frac{\gamma}{\upsilon}
\]

where \( \lambda_{i}^{A} \) and \( \lambda_{i}^{-} \) follow from equation (27) and (32), and \( \upsilon \) and \( \gamma \) are defined after (1), and

\[
H \begin{bmatrix} X_{1} \\ X_{2} \end{bmatrix} = \begin{bmatrix} A & -\frac{1}{\upsilon}I_{n} \\ -\gamma I_{n} & -A \end{bmatrix} \begin{bmatrix} x\alpha \\ x\beta \end{bmatrix} = \begin{bmatrix} x\alpha \\ x\beta \end{bmatrix} \Lambda^{-}
\]

where \( \alpha = \text{diag}(\alpha_{1}, \ldots, \alpha_{n}) \), \( \beta = \text{diag}(\beta_{1}, \ldots, \beta_{n}) \),

\[
\alpha_{i} = \sqrt{\frac{1}{\upsilon^{2}} + (\lambda_{i}^{A} - \lambda_{i}^{-})^{2}}, \quad \beta_{i} = \sqrt{\frac{1}{\upsilon^{2}} + (\lambda_{i}^{A} - \lambda_{i}^{-})^{2}}
\]

and \( \alpha^{2} + \beta^{2} = I_{n} \).

**Proposition 4.6** also holds for \( \tilde{H} \) and \( \tilde{A} \) with \( \tilde{\alpha} \) and \( \tilde{\beta} \) by substituting \( \lambda_{i}^{A} \) and \( \lambda_{i}^{-} \) with \( \tilde{\lambda}_{i}^{A} \) and \( \tilde{\lambda}_{i}^{-} \) in (35). It shows that
and $O_1$, $O_2$, $O_3$ denote
\begin{align*}
O_1 &= \left[ \begin{array}{c}
\beta^2 \lambda_i + \beta^2 \lambda_j \\
\alpha_i \lambda_i + \alpha_j \lambda_j
\end{array} \right] \left( \frac{\lambda_i + \lambda_j}{\lambda_i + \lambda_j} \right), \\
O_2 &= \left[ \begin{array}{c}
\beta^2 \lambda_i + \beta^2 \lambda_j \\
\alpha_i \lambda_i + \alpha_j \lambda_j
\end{array} \right] \\
O_3 &= \left[ \begin{array}{c}
\beta^2 \lambda_i + \beta^2 \lambda_j \\
\alpha_i \lambda_i + \alpha_j \lambda_j
\end{array} \right].
\end{align*}

Proof: Proof of this part follows directly from Lemma 4.5.

\[ \|g_c(s) - \hat{g}_c(s)\|_{\mathcal{H}_2}^2 = \|g_c(s) - \tilde{g}_s(s)\|_{\mathcal{H}_2}^2 - \text{trace}(\xi), \]  
(40)
where

\[ \xi = \mathcal{R}_1(C_{\lambda A, \lambda A, \lambda A - C_{\lambda, \lambda}})\mathcal{R}_1^T - \mathcal{R}_1(C_{\lambda A, \lambda A, \lambda A - C_{\lambda, \lambda}})\mathcal{R}_2^T - \mathcal{R}_2(C_{\lambda A, \lambda A, \lambda A - C_{\lambda, \lambda}})\mathcal{R}_1^T + \mathcal{R}_2(C_{\lambda A, \lambda A, \lambda A - C_{\lambda, \lambda}})\mathcal{R}_2^T. \]

Apply Theorem 3.1 to $\|g_c(s) - \hat{g}_c(s)\|_{\mathcal{H}_2}$ then expand the right hand side of (40) yields (39).

From Theorem 4.7, the error $\xi$ is governed by $\alpha$, $\alpha$, $\beta$, and $\beta$, which as expressed in (35) are functions of $(\lambda_i^A - \lambda_i)$ or $(\lambda_i^A - \lambda_i)$. However, given the fact that $(\lambda_i^A - \lambda_i)$ and $(\lambda_i^A - \lambda_i)$ are both bounded by $\sqrt{\gamma}$, we can push the entries in $O_1$, $O_2$, and $O_3$ to 0 by choosing $\gamma$ and $\upsilon$ such that $\frac{1}{\gamma} \gg \frac{1}{\upsilon}$ which implies $\gamma \ll 1$, and then guarantee $\text{trace}(\xi) \approx 0$.

D. Design of Clustering Matrix $P$

Combining (17) and (39) yields the final result of this paper:

\[ \|g_c(s) - \hat{g}_c(s)\|_{\mathcal{H}_2} \leq \sqrt{\|\|g_c(s)\|_{\mathcal{H}_2}^2 - \text{trace}(\xi)\| + \|\Psi(s)\|_{\mathcal{H}_\infty} \theta}. \]  
(41)

Since $\text{trace}(\xi) \approx 0$ with $\gamma \ll 1$, we have

\[ \|g_c(s) - \hat{g}_c(s)\|_{\mathcal{H}_2} \leq \|\|g_c(s)\|_{\mathcal{H}_2}^2 \theta + \|\Psi(s)\|_{\mathcal{H}_\infty} \theta. \]  
(42)

The value of right hand side will depend on $\theta$ and $\theta_s$. Hence we can put weights on $\Phi_2^r$ and $\Phi_4^r$, and form

\[ \Phi_{inp} := \left[ (1 - \rho)\Phi_2^r \quad \rho \Phi_4^r \right], \quad 0 \leq \rho \leq 1 \]  
(43)
as the input to K-means algorithm. The corresponding K-means error $\theta_{inp}$ will satisfy

\[ \theta_{inp} = \sqrt{(1 - \rho)^2 \theta^2 + \rho^2 \theta^2}. \]  
(44)

Thus by minimizing $\theta_{inp}$, $\theta$ and $\theta_s$ will both be reduced, but in different proportions. For a standard K-means algorithm, $\Phi^r$ should be provided along with the number of clusters $r$. Hence, $r$ and $\rho$ can be both tuned with respect to the refinement of the clustering results. The clustering algorithm is summarized in Algorithm 1. Note that when $\rho = 1$, Algorithm 1 is equivalent to the open-loop clustering as mentioned before.

<table>
<thead>
<tr>
<th>Algorithm 1: Algorithm for finding $H_2$ closed-loop clustering matrix $P$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input</strong></td>
</tr>
<tr>
<td>1 Compute eigenvalue $\Lambda^A$ and eigenspace $x$ from $A$;</td>
</tr>
<tr>
<td>2 Compute $\Phi$ from Theorem 3.1 and Lemma 4.4;</td>
</tr>
<tr>
<td>3 Compute closed-loop eigenvalue $\Lambda^-$;</td>
</tr>
<tr>
<td>4 Calculate $\alpha$ from (35);</td>
</tr>
<tr>
<td>5 Compute $\Phi^*$ from Theorem 4.7 and Lemma 4.4;</td>
</tr>
<tr>
<td>6 Find $\Phi_2^r$ and $\Phi_4^r$ through LDL decomposition, and form $\Phi_{inp}$ with $\rho$ from (43);</td>
</tr>
<tr>
<td>7 Input $\Phi_{inp}$ with $r$ to K-means algorithm;</td>
</tr>
<tr>
<td>8 With clustering labels from K-means results, construct $P$ based on Definition 2.1.</td>
</tr>
<tr>
<td><strong>Output</strong></td>
</tr>
</tbody>
</table>

V. SIMULATION RESULTS

To compare the approach in this paper with typical open-loop clustering, we first start with a 51-node network as it is easy to visualize the clusterings. We assume the disturbance to enter from node 4, i.e., $b$ is the 4th column of $I_{51}$. As we have discussed in proposition 4.6, the value of $\frac{1}{\upsilon}$ will decide the eigenvalues of the closed-loop system. Hence, considering a fixed ratio of $\frac{1}{\upsilon} = 100$, we choose four pairs of LQR specifications based on the value of $\frac{1}{\upsilon}$ in a descending order as $\{\gamma = 10^{-3}, \upsilon = 10^{-5}\}, \{\gamma = 1, \upsilon = 10^{-2}\}, \{\gamma = 10, \upsilon = 0.1\}$ and $\gamma = 100, \upsilon = 1\}$. The corresponding approximation errors $\text{trace}(\xi)$ are shown in Figure 2. As $r$, specified number of clusters increases to $n$, $\text{trace}(\xi)$ for all the cases move to 0. But it is also clear that for this fixed value of $\frac{1}{\upsilon}$, as $\gamma$ decreases, $\text{trace}(\xi)$ becomes sufficiently small, which implies $\|g_c(s) - \hat{g}_c(s)\|_{\mathcal{H}_2}$ to be a good indicator of $\|g_c(s) - \tilde{g}_s(s)\|_{\mathcal{H}_2}$. Therefore, in the remaining simulations, we apply $\{\gamma = 10^{-3}, \upsilon = 10^{-5}\}$ as the design parameter for LQR.

To cluster this 51-node network, the weight $\rho$ is provided as 0, 0.2, 0.4, 0.6, 0.8 and 1 respectively as an input to Algorithm 1. Shown in Figure 3 are the ratios computed by $\|g_c(s) - \hat{g}_c(s)\|_{\mathcal{H}_2}/\|g_c(s)\|_{\mathcal{H}_2}$ with respect to clustering outputs from each of the $P$. In this example, when $\rho = 0$, namely the input to the K-means is only $\Phi^*$, the resulting clustering strategy outperforms all the other scenarios in approaching the closed-loop performance of $g_c(s)$. In the sense of controller design, that is, with $\rho = 0$ the controller designed from sufficiently small dimension $r$ can already achieve the performance of a full-order LQR. As a contrast, a worst-case scenario given by $\rho = 1$ shows that by applying a LQR controller designed from a model clustered merely by open-loop characteristics, $\hat{g}_c(s)$ is not close to $g_c(s)$ at all especially for a small $r$, e.g. $r \leq 30$. To illustrate the difference of clustering strategies in network, we draw the network graph for $\rho = 0$ and $\rho = 1$ with $r = 5$ clusters as shown in Figure 4 and 5, where nodes with same color are assigned to the same cluster. As can be found from the network structure, the network is naturally partitioned into 5 clusters. However, as it is also clearly shown in
Fig. 2: Values of $\text{trace}(\xi)$ with respect to four LQR specifications for a 51-node network

Fig. 3: Ratios of $\frac{\|g_c(s) - \tilde{g}_c(s)\|_{H_2}}{\|g_c(s)\|_{H_2}}$ with respect to choices of $\rho$ for a 51-node network

Figure 4 and 5, for both strategies yielded by $\rho = 0$ and $\rho = 1$, the similarity of nodes in our problem is not coherent with the geometric characteristics, but is rather related to the controllability spaces, which draws a line between our algorithm and other traditional graph partitioning problems.

The algorithm is also applied to a 1000-node graph Laplacian (disturbance still comes from node 4). The results follow the similar pattern as shown in Figure 6, where $\rho = 0$ still outperforms $\rho = 1$ in a much larger network. Moreover, at $r = 2$, the ratio $\frac{\|g_c(s) - \tilde{g}_c(s)\|_{H_2}}{\|g_c(s)\|_{H_2}}$ for $\rho = 0$ is already under 2%. That is, a LQR controller designed from this clustered model with $r = 2$ is already good enough, which would save a lot of effort in practice.

VI. Conclusion

In this paper, we proposed an algorithm to find a clustering matrix, such that for a class of LQR design, i.e. $Q = \gamma I$ and $R = \nu I$, the LQR controller designed from clustered model can approximate the full-order LQR in closed-loop performance. An error bound for this algorithm is derived in terms of two Gramians as well as an approximation error $\text{trace}(\xi)$. From simulations, we showed that the proposed algorithm produces different clusterings than open-loop clustering, and significantly outperforms in the sense of closed-loop performance. Also by this algorithm, the LQR controller designed from a lower order clustered model can achieve the performance of a full-order design closely.

APPENDIX

Proof: Demonstration of Lemma 4.4: The controllability Gramian can be solved from Lyapunov equation

$$A\Phi + \Phi A^T + bb^T = 0.$$ (45)

Applying (21), we get

$$T\Lambda T^{-1} \Phi + \Phi T^{-T} \Lambda T^T + bb^T = 0,$$

$$\iff \Lambda T^{-1} \Phi T^{-T} + T^{-1} \Phi T^{-T} \Lambda + T^{-1} bb^T T^{-T} = 0.$$
and according to Lemma 4.4,
\[ C_e \Phi_e C_e^T = C_e T_e \cdot \text{diag}(T_e^{-1}b_e) C_{\Lambda_e} \cdot \text{diag}(T_e^{-1}b_e) T_e^{-1} C_e^T \]
\[ = \begin{bmatrix} R_1 & -R_2 \\ C_{\Lambda_1} & C_{\Lambda_2} \end{bmatrix} \begin{bmatrix} R_1 & -R_2 \end{bmatrix}^T. \]

Expanding (50) yields
\[ C_e \Phi_e C_e^T = R_1 C_{\Lambda_1} R_1^T - R_1 C_{\Lambda_2} R_2^T - R_2 C_{\Lambda_2} R_1^T + R_2 C_{\Lambda_2} R_2^T, \]
which completes the proof.

**ACKNOWLEDGMENT**

The author would like to thank Dr. Tom R. Nudell for the insightful discussions on this topic and help in formatting the network graphs in this paper.

**REFERENCES**