

Clustering-Based Model Order Reduction for Nonlinear Network Systems

Peter Benner* Sara Grundel* Petar Mlinarić*

March 17, 2020

Abstract

Clustering by projection has been proposed as a way to preserve network structure in linear multi-agent systems. Here, we extend this approach to a class of nonlinear network systems. Additionally, we generalize our clustering method which restores the network structure in an arbitrary reduced-order model obtained by projection. We demonstrate this method on a number of examples.

1 Introduction

Nonlinear network systems appear in various application areas, including energy distribution networks, water networks, multi-robot networks, and chemical reaction networks. Model order reduction (MOR) enables faster simulation, optimization, and control of large-scale network systems. However, standard methods generally do not preserve the network structure. Preserving the network structure is necessary, e.g., if an optimization method assumes this structure.

Clustering was proposed in the literature as a way to preserve the multi-agent structure. Methods based on equitable partitions were described in [17, 25, 4] with an extension to almost equitable partitions in [18]. Based on this, a priori error expressions were developed in [24] with generalizations in [15]. Ishizaki et al. [13] developed a clustering-based \mathcal{H}_∞ -MOR method based on positive tridiagonalization and reducible clusters, applicable to linear time-invariant systems with asymptotically stable and symmetric dynamics matrices. In [11], they presented an efficient clustering-based method also based on reducible clusters for \mathcal{H}_2 -MOR of linear positive networks, which include systems with Laplacian-based dynamics. Besselink et al. [3] studied networks of identical passive systems over weighted and directed graphs with tree structures.

In this work, we extend the clustering-based approach for linear time-invariant multi-agent systems from [21, 22]. There, we proposed a method combining the iterative rational Krylov algorithm (IRKA) [1] and QR decomposition-based clustering [27]. We generalize this approach to be able to combine any projection-based MOR method and clustering algorithm. In particular, this allows applying the method to nonlinear network systems. We show that for a class of nonlinear multi-agent systems, clustering by Galerkin projection preserves network structure, which additionally avoids the need for hyper-reduction to simplify the nonlinear part.

*Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany, {benner,grundel,mlinearic}@mpi-magdeburg.mpg.de

The outline of this paper is as follows. First, we provide some background information on linear multi-agent systems in Section 2. In Section 3, we recall our clustering-based MOR method for linear multi-agent systems and generalize it to a framework which allows combining any projection-based MOR method and clustering algorithm. In Section 4, we extend clustering by projection to a class of nonlinear multi-agent systems, which also permits the applicability of our framework. We demonstrate the approach numerically in Section 5 and conclude with Section 6.

2 Preliminaries

We present some basic concepts from graph theory in Section 2.1, graph partitions in Section 2.2, before moving on to linear multi-agent systems in Section 2.3 and clustering-based MOR in Section 2.4. Additionally, we give remarks on MOR for non-asymptotically stable linear multi-agent systems in Section 2.5.

2.1 Graph theory

The notation in this section is based on [20] and [7].

A *graph* G consists of a *vertex set* V and an *edge set* E encoding the relation between vertices. *Undirected* graphs are those for which the edge set is a subset of the set of all unordered pairs of vertices, i.e., $E \subseteq \{\{i, j\} : i, j \in V, i \neq j\}$. On the other hand, a graph is *directed* if $E \subseteq \{(i, j) : i, j \in V, i \neq j\}$. We think of an edge (i, j) as an arrow starting from vertex i and ending at j . We only consider *simple* graphs, i.e., graphs without self-loops or multiple copies of the same edge. Additionally, we only consider *finite graphs*, i.e., graphs with a finite number of vertices $n := |V|$. Without loss of generality, let $V = \{1, 2, \dots, n\}$.

For an undirected graph, a *path* of length ℓ is a sequence of distinct vertices i_0, i_1, \dots, i_ℓ such that $\{i_k, i_{k+1}\} \in E$ for $k = 0, 1, \dots, \ell - 1$. For a directed graph, a *directed path* of length ℓ is a sequence of distinct vertices i_0, i_1, \dots, i_ℓ such that $(i_k, i_{k+1}) \in E$ for $k = 0, 1, \dots, \ell - 1$. An undirected graph is *connected* if there is a path between any two distinct vertices $i, j \in V$. A directed graph is *strongly connected* if there is a directed path between any two distinct vertices $i, j \in V$.

We can associate weights to edges of a graph by a *weight function* $w: E \rightarrow \mathbb{R}$. If $w(e) > 0$ for all $e \in E$, the tuple $G = (V, E, w)$ is called a *weighted graph*. In the following, we will focus on weighted graphs. In particular, we will directly generalize concepts for unweighted graphs from [20, 7], as was done in [24].

The *adjacency matrix* $A = [a_{ij}]_{i,j \in V} \in \mathbb{R}^{n \times n}$ of an undirected weighted graph is defined component-wise by

$$a_{ij} := \begin{cases} w(\{i, j\}), & \text{if } \{i, j\} \in E, \\ 0, & \text{otherwise,} \end{cases}$$

and for a directed weighted graph as

$$a_{ij} := \begin{cases} w((j, i)), & \text{if } (j, i) \in E, \\ 0, & \text{otherwise.} \end{cases}$$

For every vertex $i \in V$, its *in-degree* is $\delta_i := \sum_{j=1}^n a_{ij}$. The diagonal matrix $D := \text{diag}(\delta_1, \delta_2, \dots, \delta_n)$ is called the *in-degree matrix*. Notice that $D = \text{diag}(A\mathbf{1})$, where $\mathbf{1}$ is the vector of all ones.

Let $e_1, e_2, \dots, e_{|E|}$ be all the edges of G in some order. The *incidence matrix* $R \in \mathbb{R}^{n \times |E|}$ of a directed graph G is defined component-wise

$$[R]_{ik} := \begin{cases} -1, & \text{if } e_k = (i, j) \text{ for some } j \in V, \\ 1, & \text{if } e_k = (j, i) \text{ for some } j \in V, \\ 0, & \text{otherwise.} \end{cases}$$

If G is undirected, we assign some orientation to every edge to define a directed graph G° , and define the incidence matrix of G to be the incidence matrix of G° . The *weight matrix* is defined as $W := \text{diag}(w(e_1), w(e_2), \dots, w(e_{|E|}))$.

The (*in-degree*) *Laplacian matrix* L is defined by $L := D - A$. For undirected graphs, it can be checked that $L = RWR^T$, using

$$RWR^T = \sum_{\{i,j\} \in E} a_{ij}(e_i - e_j)(e_i - e_j)^T,$$

which is independent of the order of edges defining R and W or the orientation of edges in G° . From the definition of L , it directly follows that the sum of each row in L is zero, i.e., $L\mathbf{1} = 0$. From $L = RWR^T$, we immediately see that, for undirected weighted graphs, the Laplacian matrix L is symmetric positive semidefinite.

The following theorem, based on Theorem 2.8 in [20], states how connectedness of a graph is related to the spectral properties of L .

Theorem 2.1. *Let $G = (V, E, w)$ be an undirected weighted graph, L its Laplacian matrix, and $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ the eigenvalues of L . Then the following statements are equivalent:*

1. G is connected,
2. $\lambda_2 > 0$,
3. $\ker(L) = \text{im}(\mathbf{1})$.

2.2 Graph partitions

A nonempty subset $C \subseteq V$ is called a *cluster* of V . A *graph partition* π is a partition of the vertex set V . The *characteristic vector* of a cluster $C \subseteq V$ is the vector $p(C) \in \mathbb{R}^n$ defined with

$$[p(C)]_i := \begin{cases} 1 & \text{if } i \in C, \\ 0 & \text{otherwise.} \end{cases}$$

The *characteristic matrix of a partition* $\pi = \{C_1, C_2, \dots, C_r\}$ of the graph G is the matrix $P \in \mathbb{R}^{n \times r}$ defined by

$$P := [p(C_1) \quad p(C_2) \quad \dots \quad p(C_r)].$$

2.3 Linear multi-agent systems

Here, we focus on linear time-invariant multi-agent systems (cf. [3, 5, 6, 13, 12, 14, 23, 24]). Additionally, we restrict ourselves to multi-agent systems defined over an undirected, weighted, and connected graph $G = (V, E, w)$.

The dynamics of the i th agent, for $i \in \mathcal{V} = \{1, 2, \dots, n\}$, is

$$\begin{aligned} E\dot{x}_i(t) &= Ax_i(t) + Bv_i(t), \\ z_i(t) &= Cx_i(t), \end{aligned}$$

with system matrices $E, A \in \mathbb{R}^{n \times n}$, input matrix $B \in \mathbb{R}^{n \times m}$, output matrix $C \in \mathbb{R}^{p \times n}$, state $x_i(t) \in \mathbb{R}^n$, input $v_i(t) \in \mathbb{R}^m$, and output $z_i(t) \in \mathbb{R}^p$. We assume the matrix E to be invertible. The interconnections are

$$m_i v_i(t) = \sum_{j=1}^n a_{ij} K(z_j(t) - z_i(t)) + \sum_{k=1}^m b_{ik} u_k(t),$$

for $i = 1, 2, \dots, n$, with inertias $m_i > 0$, coupling matrix $K \in \mathbb{R}^{m \times p}$, external inputs $u_k(t) \in \mathbb{R}^m$, $k = 1, 2, \dots, m$, where $A = [a_{ij}]$ is the adjacency matrix of the graph G . The outputs are

$$y_\ell(t) = \sum_{j=1}^n c_{\ell j} z_j(t)$$

for $\ell = 1, 2, \dots, p$. Define

$$\begin{aligned} M &:= \text{diag}(m_i) \in \mathbb{R}^{n \times n}, \quad B := [b_{ik}] \in \mathbb{R}^{n \times m}, \quad C := [c_{\ell j}] \in \mathbb{R}^{p \times n}, \\ x(t) &:= \text{col}(x_i(t)) \in \mathbb{R}^{nn}, \quad v(t) := \text{col}(v_i(t)) \in \mathbb{R}^{nm}, \quad z(t) := \text{col}(z_i(t)) \in \mathbb{R}^{np}, \\ u(t) &:= \text{col}(u_k(t)) \in \mathbb{R}^{mm}, \quad \text{and } y(t) := \text{col}(y_\ell(t)) \in \mathbb{R}^{pp}. \end{aligned}$$

Then the agent dynamics can be rewritten as

$$\begin{aligned} (I_n \otimes E)\dot{x}(t) &= (I_n \otimes A)x(t) + (I_n \otimes B)v(t), \\ z(t) &= (I_n \otimes C)x(t), \end{aligned}$$

interconnection as

$$(M \otimes I_n)v(t) = (-L \otimes K)z(t) + (B \otimes I_m)u(t),$$

and output as

$$y(t) = (C \otimes I_p)z(t).$$

Therefore, we have

$$\begin{aligned} (M \otimes E)\dot{x}(t) &= (M \otimes A - L \otimes BKC)x(t) + (B \otimes B)u(t), \\ y(t) &= (C \otimes C)x(t). \end{aligned} \tag{1}$$

Of particular interest are *leader-follower multi-agent systems* where only some agents (*leaders*) receive external input, while other agents (*followers*) receive no inputs. Let $m \in \{1, 2, \dots, n\}$ be the number of leaders, $\mathcal{V}_L = \{v_1, v_2, \dots, v_m\} \subseteq \mathcal{V}$ the set of leaders, and $\mathcal{V}_F = \mathcal{V} \setminus \mathcal{V}_L$ the set of followers. Then, with B defined by

$$b_{ik} := \begin{cases} 1, & \text{if } i = v_k, \\ 0, & \text{otherwise,} \end{cases}$$

the system (1) becomes a leader-follower multi-agent system. One important class are multi-agent systems with *single-integrator agents*, i.e., with $n = 1$, $A = 0$, and $B = C = K = E = 1$. Thus, system (1) becomes

$$\begin{aligned} \mathbf{M}\dot{x}(t) &= -\mathbf{L}x(t) + \mathbf{B}u(t), \\ y(t) &= \mathbf{C}x(t). \end{aligned} \tag{2}$$

The property of interest for multi-agent systems is *synchronization*.

Definition 2.2. *The system $(\mathbf{M} \otimes E)\dot{x}(t) = (\mathbf{M} \otimes A - \mathbf{L} \otimes BKC)x(t)$ is synchronized if*

$$\lim_{t \rightarrow \infty} (x_i(t) - x_j(t)) = 0,$$

for all $i, j \in \mathcal{V}$ and all initial conditions $x(0) = x_0$.

For system (1), this is equivalent to $(\mathbf{L} \otimes I_n)x(t) \rightarrow 0$, because the multi-agent system is defined on a connected graph. The following results gives another equivalent condition ([16, Theorem 1], [23, Lemma 4.2]).

Proposition 2.3. *The system $(\mathbf{M} \otimes E)\dot{x}(t) = (\mathbf{M} \otimes A - \mathbf{L} \otimes BKC)x(t)$ is synchronized if and only if $(A - \lambda BKC, E)$ is Hurwitz for all nonzero eigenvalues λ of (\mathbf{L}, \mathbf{M}) .*

Note that linear multi-agent systems with single integrator agents, as in (2), are synchronized.

2.4 Clustering-based model order reduction

By choosing some matrices $V, W \in \mathbb{R}^{n \times r}$, we get the reduced model for (2)

$$\begin{aligned} W^T \mathbf{M} V \dot{\hat{x}}(t) &= -W^T \mathbf{L} V \hat{x}(t) + W^T \mathbf{B} u(t), \\ \hat{y}(t) &= \mathbf{C} V \hat{x}(t), \end{aligned} \tag{3}$$

or, for (1),

$$\begin{aligned} W^T (\mathbf{M} \otimes E) V \dot{\hat{x}}(t) &= W^T (\mathbf{M} \otimes A - \mathbf{L} \otimes BKC) V \hat{x}(t) + W^T (\mathbf{B} \otimes B) u(t), \\ \hat{y}(t) &= (\mathbf{C} \otimes C) V \hat{x}(t), \end{aligned} \tag{4}$$

which is not necessarily a multi-agent system. As suggested in [5] (similar to [24, 13]), using

$$V = W = \mathbf{P}, \tag{5}$$

in (3), or in general

$$V = W = \mathbf{P} \otimes I_n, \tag{6}$$

in (4), preserves the structure, where \mathbf{P} is a characteristic matrix of a partition π of the vertex set \mathcal{V} . In particular, $\mathbf{P}^T \mathbf{M} \mathbf{P}$ is a positive definite diagonal matrix and $\mathbf{P}^T \mathbf{L} \mathbf{P}$ is the Laplacian matrix of the reduced graph.

2.5 Model reduction for non-asymptotically stable systems

Note that the system (2) is not (internally) asymptotically stable since L has a zero eigenvalue. We consider decomposition into asymptotically and non-asymptotically stable part for model reduction and computation of the systems norms. Therefore, we want to find an invertible matrix T such that

$$T^T M T = \begin{bmatrix} M_- & 0 \\ 0 & m_+ \end{bmatrix} \text{ and } T^T L T = \begin{bmatrix} L_- & 0 \\ 0 & 0 \end{bmatrix},$$

where $\sigma(-L_-, M_-) \subset \mathbb{C}_-$. We see that if

$$T = [T_- \quad \mathbf{1}_n],$$

then

$$T^T M T = \begin{bmatrix} T_-^T M T_- & T_-^T M \mathbf{1}_n \\ \mathbf{1}_n^T M T_- & \mathbf{1}_n^T M \mathbf{1}_n \end{bmatrix} \text{ and } T^T L T = \begin{bmatrix} T_-^T L T_- & 0 \\ 0 & 0 \end{bmatrix}.$$

To have $T_-^T M \mathbf{1}_n = 0$, we need for the columns of T_- to be orthogonal to $M \mathbf{1}_n$. This will also ensure that $\sigma(-L_-, M_-) = \sigma(-T_-^T L T_-, T_-^T M T_-) \subset \mathbb{C}_-$. Additionally, T_- should be such that both $T_-^T M T_-$ and $T_-^T L T_-$ are sparse. We chose

$$T_- = \begin{bmatrix} \alpha_1 & & & & & \\ -\beta_1 & \alpha_2 & & & & \\ & -\beta_2 & \ddots & & & \\ & & & \ddots & & \\ & & & & \alpha_{n-1} & \\ & & & & -\beta_{n-1} & \end{bmatrix}$$

with some $\alpha_i, \beta_i > 0$, $i = 1, 2, \dots, n-1$, which we determine next. From $e_i^T T_-^T M \mathbf{1}_n = 0$, we find $\alpha_i m_i = \beta_i m_{i+1}$. If we additionally set $\alpha_i^2 + \beta_i^2 = 1$, we get

$$\alpha_i = \frac{m_{i+1}}{\sqrt{m_i^2 + m_{i+1}^2}} \quad \text{and} \quad \beta_i = \frac{m_i}{\sqrt{m_i^2 + m_{i+1}^2}}.$$

To find the asymptotically stable part of the system (1), let T and S be invertible matrices such that

$$S^T E T = \begin{bmatrix} E_- & 0 \\ 0 & E_+ \end{bmatrix} \text{ and } S^T A T = \begin{bmatrix} A_- & 0 \\ 0 & A_+ \end{bmatrix},$$

where $\sigma(A_-, E_-) \subset \mathbb{C}_-$ and $\sigma(A_+, E_+) \subset \overline{\mathbb{C}_+}$. Then

$$\begin{aligned} & \begin{bmatrix} I_{(n-1)n} & 0 \\ 0 & S \end{bmatrix}^T (T \otimes I_n)^T (M \otimes E) (T \otimes I_n) \begin{bmatrix} I_{(n-1)n} & 0 \\ 0 & T \end{bmatrix} \\ &= \begin{bmatrix} I_{(n-1)n} & 0 \\ 0 & S \end{bmatrix}^T \begin{bmatrix} M_- \otimes E & 0 \\ 0 & m_+ E \end{bmatrix} \begin{bmatrix} I_{(n-1)n} & 0 \\ 0 & T \end{bmatrix} \\ &= \begin{bmatrix} M_- \otimes E & 0 & 0 \\ 0 & m_+ E_- & 0 \\ 0 & 0 & m_+ E_+ \end{bmatrix} \end{aligned}$$

and

$$\begin{aligned}
& \begin{bmatrix} I_{(n-1)n} & 0 \\ 0 & S \end{bmatrix}^T (\mathbb{T} \otimes I_n)^T (\mathbb{M} \otimes A - \mathbb{L} \otimes BKC) (\mathbb{T} \otimes I_n) \begin{bmatrix} I_{(n-1)n} & 0 \\ 0 & T \end{bmatrix} \\
&= \begin{bmatrix} I_{(n-1)n} & 0 \\ 0 & S \end{bmatrix}^T \begin{bmatrix} \mathbb{M}_- \otimes A - \mathbb{L}_- \otimes BKC & 0 \\ 0 & \mathbb{m}_+ A \end{bmatrix} \begin{bmatrix} I_{(n-1)n} & 0 \\ 0 & T \end{bmatrix} \\
&= \begin{bmatrix} \mathbb{M}_- \otimes A - \mathbb{L}_- \otimes BKC & 0 & 0 \\ 0 & \mathbb{m}_+ A_- & 0 \\ 0 & 0 & \mathbb{m}_+ A_+ \end{bmatrix}.
\end{aligned}$$

Since the original system is assumed to be synchronized, we have that $\sigma(\mathbb{M}_- \otimes A - \mathbb{L}_- \otimes BKC, \mathbb{M}_- \otimes E) \subset \mathbb{C}_-$.

3 Clustering for linear multi-agent systems

In this section, we motivate our general approach for clustering-based linear multi-agent systems. Since clustering is generally a difficult combinatorial problem (see, e.g., [26]), we propose a heuristic approach for finding suboptimal partitions.

In Section 3.1, we motivate and describe our MOR method on the case of linear multi-agent system with single-integrator agents and extend it in Section 3.2 to higher-order agents.

3.1 Single-integrator agents

For simplicity, we first consider multi-agent systems with single-integrator agents as in (2). Let

$$\begin{aligned}
H(s) &= C(sM + L)^{-1}B, \\
\widehat{H}(s) &= CV(sW^T M V + W^T L V)^{-1}W^T B
\end{aligned}$$

be the transfer functions of systems (2) and (3), respectively, where $V, W \in \mathbb{R}^{n \times r_P}$ are obtained using a projection-based method such as balanced truncation or IRKA.

In [21], motivated by (5) and the properties of clustering using QR decomposition with column pivoting (see [27, Section 3], [21, Lemma 1]), we proposed applying it to the set of rows of V or W to recover the partition. Here, we want to emphasize that the approach is not restricted to this choice of clustering algorithm. In particular, the following result on the forward error in the Petrov-Galerkin projection ([2, Theorem 3.3]) motivates using the k-means clustering [9].

Theorem 3.1. *Let $V_1, V_2, W_1, W_2 \in \mathbb{R}^{n \times r_P}$ be full-rank matrices and*

$$\mathcal{V}_i = \text{im}(V_i), \quad \mathcal{W}_i = \text{im}(W_i), \quad \widehat{H}_i(s) = CV_i(sW_i^T E V_i - W_i^T A V_i)^{-1}W_i^T B,$$

for $i = 1, 2$. Then

$$\frac{\|\widehat{H}_1 - \widehat{H}_2\|_{\mathcal{H}_\infty}}{\frac{1}{2} \left(\|\widehat{H}_1\|_{\mathcal{H}_\infty} + \|\widehat{H}_2\|_{\mathcal{H}_\infty} \right)} \leq M \max(\sin \Theta(\mathcal{V}_1, \mathcal{V}_2), \sin \Theta(\mathcal{W}_1, \mathcal{W}_2)),$$

where

$$\begin{aligned}
M &= 2 \max(M_1, M_2), \\
M_1 &= \frac{\max_{\omega \in \mathbb{R}} \|C\|_2 \left\| V_1 (\boldsymbol{w} W_1^T E V_1 - W_1^T A V_1)^{-1} W_1^T B \right\|_2 \left\| \widehat{H}_1(\boldsymbol{w}) \right\|_2^{-1}}{\min_{\omega \in \mathbb{R}} \cos \Theta \left(\ker \left(W_2^T (\boldsymbol{w} E - A)^{-1} \right)^\perp, \mathcal{V}_2 \right)}, \\
M_2 &= \frac{\max_{\omega \in \mathbb{R}} \left\| C V_2 (\boldsymbol{w} W_2^T E V_2 - W_2^T A V_2)^{-1} W_2^T \right\|_2 \|B\|_2 \left\| \widehat{H}_2(\boldsymbol{w}) \right\|_2^{-1}}{\min_{\omega \in \mathbb{R}} \cos \Theta \left(\text{im} \left((\boldsymbol{w} E - A)^{-1} V_1 \right), \mathcal{W}_1 \right)},
\end{aligned}$$

and $\Theta(\mathcal{M}, \mathcal{N})$ is the largest principal angle between subspaces $\mathcal{M}, \mathcal{N} \subseteq \mathbb{R}^n$.

The angle between two subspaces $\mathcal{V}_1, \mathcal{V}_2 \subseteq \mathbb{R}^n$ is defined by (see [2, Section 3.1])

$$\sin \Theta(\mathcal{V}_1, \mathcal{V}_2) := \sup_{v_1 \in \mathcal{V}_1} \inf_{v_2 \in \mathcal{V}_2} \frac{\|v_2 - v_1\|_2}{\|v_1\|_2}.$$

If \mathcal{V}_1 and \mathcal{V}_2 are of equal dimension, then we have

$$\sin \Theta(\mathcal{V}_1, \mathcal{V}_2) = \sin \Theta(\mathcal{V}_2, \mathcal{V}_1) = \|(I - V_1 V_1^T) V_2\|_2,$$

where $\mathcal{V}_1 = \text{im}(V_1)$, $\mathcal{V}_2 = \text{im}(V_2)$, with V_1 and V_2 having orthonormal columns. If additionally $V_1 = P(P^T P)^{-1/2} \in \mathbb{R}^{n \times r}$ and V_2 is the $V \in \mathbb{R}^{n \times r_P}$ from the projection-based method, then

$$\begin{aligned}
(\sin \Theta(\mathcal{V}_1, \mathcal{V}_2))^2 &\leq \left\| (I - P(P^T P)^{-1} P^T) V \right\|_{\mathbb{F}}^2 \\
&= \left\| \left(I - \begin{bmatrix} p(\mathbf{C}_1) & \cdots & p(\mathbf{C}_r) \end{bmatrix} \begin{bmatrix} |\mathbf{C}_1|^{-1} & & \\ & \ddots & \\ & & |\mathbf{C}_r|^{-1} \end{bmatrix} \begin{bmatrix} p(\mathbf{C}_1)^T \\ \vdots \\ p(\mathbf{C}_r)^T \end{bmatrix} \right) V \right\|_{\mathbb{F}}^2 \\
&= \left\| \left(I - \sum_{i=1}^r \frac{1}{|\mathbf{C}_i|} p(\mathbf{C}_i) p(\mathbf{C}_i)^T \right) V \right\|_{\mathbb{F}}^2 \\
&= \sum_{i=1}^r \left\| V_{\mathbf{C}_i, :} - \frac{1}{|\mathbf{C}_i|} \mathbf{1}_{|\mathbf{C}_i|} \mathbf{1}_{|\mathbf{C}_i|}^T V_{\mathbf{C}_i, :} \right\|_{\mathbb{F}}^2 \\
&= \sum_{i=1}^r \sum_{p \in \mathbf{C}_i} \left\| V_{p, :} - \frac{1}{|\mathbf{C}_i|} \sum_{q \in \mathbf{C}_i} V_{q, :} \right\|_{\mathbb{F}}^2,
\end{aligned}$$

which is equal to the k-means cost functional for the set of rows of V , where $V_{\mathbf{C}_i, :}$ is the submatrix of rows in V corresponding to cluster \mathbf{C}_i and $V_{p, :}$ is the p th row of V (and similarly for $V_{q, :}$). Therefore, applying the k-means algorithm to the rows of V will minimize an upper bound on the largest principal angle between $\text{im}(V)$ and $\text{im}(P)$.

The advantage of using k-means compared to QR decomposition-based clustering is in that the latter can only, given $V \in \mathbb{R}^{n \times r_P}$, return a partition with r_P clusters. On the other hand, k-means clustering can return a partition with any number of clusters r . This makes it more efficient when $r_P \ll r$ and projection-based MOR method already generates a good subspace $\text{im}(V)$.

3.2 Extension to higher-order agents

For multi-agent systems (1) with agents of order n , we have the matrices V and W as in (6). QR decomposition-based clustering can then be extended as in Algorithm 2 from [22] by clustering the block-columns of V^T (or W^T). For the k-means algorithm, we can show in a similar way as in the single-integrator case that clustering the block-rows leads to minimizing an upper bound of the largest principal angle. Therefore, k-means can be directly applied to the set of block-rows of V or W .

Additionally, we need to consider synchronization preservation. In the single-integrator case, clustering using any partition preserves synchronization. In the general case, using Theorem 2.3, we need that $(A - \hat{\lambda}BKC, E)$ is Hurwitz for all nonzero eigenvalues $\hat{\lambda}$ of (\hat{L}, \hat{M}) . Therefore, if $(A - \lambda BKC, E)$ is Hurwitz for all $\lambda \in [\lambda_2, \lambda_n]$, where $0 = \lambda_1 < \lambda_2 \leq \dots \leq \lambda_n$ are the eigenvalues of (L, M) , then from the interlacing property [8], we get that every partition preserves synchronization.

4 Clustering for nonlinear multi-agent systems

In this section, we extend the approach from the previous section to a class of nonlinear multi-agent systems. We describe the class of multi-agent systems in Section 4.1. Next, in Section 4.2, we show that clustering by projection preserves structure for this class of systems.

4.1 Nonlinear multi-agent systems

Here, we consider a class of nonlinear multi-agent systems. In particular, let the dynamics of the i th agent, for $i = 1, 2, \dots, n$, be defined by the control-affine system

$$\dot{x}_i(t) = A(x_i(t)) + B(x_i(t))v_i(t), \quad (7a)$$

$$z_i(t) = C(x_i(t)), \quad (7b)$$

with functions $A: \mathbb{R}^n \rightarrow \mathbb{R}^n$, $B: \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$, $C: \mathbb{R}^n \rightarrow \mathbb{R}^p$, state $x_i(t) \in \mathbb{R}^n$, input $v_i(t) \in \mathbb{R}^m$, and output $z_i(t) \in \mathbb{R}^p$. Furthermore, let the interconnections be

$$m_i v_i(t) = \sum_{j=1}^n a_{ij} K(z_i(t), z_j(t)) + \sum_{k=1}^m b_{ik} u_k(t), \quad (7c)$$

for $i = 1, 2, \dots, n$, with inertias $m_i > 0$ and $M = \text{diag}(m_i)$, coupling $K: \mathbb{R}^p \times \mathbb{R}^p \rightarrow \mathbb{R}^m$, external input $u_k(t) \in \mathbb{R}^m$, $k = 1, 2, \dots, m$, where $A = [a_{ij}]$ is the adjacency matrix of the graph G , and $B = [b_{ik}]$. Additionally, let the external output be

$$y_\ell(t) = \sum_{j=1}^n c_{\ell j} z_j(t), \quad (7d)$$

with $C = [c_{\ell j}]$. We assume functions A, B, C, K are continuous and that there is a unique global solution $x(t) = \text{col}(x_1(t), x_2(t), \dots, x_n(t))$ for any admissible $u(t)$.

4.2 Clustering by projection

We want to find the form of the reduced order model obtained from Galerkin projection with $V = P \otimes I_n$. We can rewrite (7) to

$$\begin{aligned} (M \otimes I_n)\dot{x}(t) &= f(x(t), u(t)), \\ y(t) &= g(x(t)), \end{aligned}$$

for some functions f and g . The reduced model is

$$\begin{aligned} (P^T M P \otimes I_n)\dot{\hat{x}}(t) &= (P^T \otimes I_n)f((P \otimes I_n)\hat{x}(t), u(t)), \\ \hat{y}(t) &= g((P \otimes I_n)\hat{x}(t)), \end{aligned} \quad (8)$$

with $\hat{x}(t) = \text{col}(\hat{x}_1(t), \hat{x}_2(t), \dots, \hat{x}_r(t))$ and $\hat{x}_i(t) \in \mathbb{R}^n$. Let $\pi(j) \in \{1, 2, \dots, r\}$ be such that $j \in C_{\pi(j)}$, for $j \in \{1, 2, \dots, n\}$. Premultiplying (8) with $e_i^T \otimes I_n$ for some $i \in \{1, 2, \dots, r\}$, we find

$$\begin{aligned} &\hat{m}_i \dot{\hat{x}}_i(t) \\ &= \sum_{i \in C_i} \left(m_i A(\hat{x}_i(t)) + B(\hat{x}_i(t)) \left(\sum_{j=1}^n a_{ij} K(C(\hat{x}_i(t)), C(\hat{x}_{\pi(j)}(t))) + \sum_{k=1}^m b_{ik} u_k(t) \right) \right) \\ &= \hat{m}_i A(\hat{x}_i(t)) \\ &\quad + B(\hat{x}_i(t)) \left(\sum_{i \in C_i} \sum_{j=1}^n a_{ij} K(C(\hat{x}_i(t)), C(\hat{x}_{\pi(j)}(t))) + \sum_{i \in C_i} \sum_{k=1}^m b_{ik} u_k(t) \right) \\ &= \hat{m}_i A(\hat{x}_i(t)) \\ &\quad + B(\hat{x}_i(t)) \left(\sum_{j=1}^r \sum_{i \in C_i} \sum_{j \in C_j} a_{ij} K(C(\hat{x}_i(t)), C(\hat{x}_j(t))) + \sum_{k=1}^m \sum_{i \in C_i} b_{ik} u_k(t) \right) \\ &= \hat{m}_i A(\hat{x}_i(t)) + B(\hat{x}_i(t)) \left(\sum_{j=1}^r \hat{a}_{ij} K(C(\hat{x}_i(t)), C(\hat{x}_j(t))) + \sum_{k=1}^m \hat{b}_{ik} u_k(t) \right), \end{aligned}$$

for

$$\hat{m}_i = \sum_{i \in C_i} m_i, \quad \hat{a}_{ij} = \sum_{i \in C_i} \sum_{j \in C_j} a_{ij}, \quad \hat{b}_{ik} = \sum_{i \in C_i} b_{ik}.$$

Defining $\hat{M} := \text{diag}(\hat{m}_i)$, $\hat{A} := [\hat{a}_{ij}]$, and $\hat{B} := [\hat{b}_{ik}]$, we see that $\hat{M} = P^T M P$, $\hat{A} = P^T A P$, and $\hat{B} = P^T B$. For the output, we have

$$\hat{y}_\ell(t) = \sum_{j=1}^n c_{\ell j} C(\hat{x}_{\pi(j)}(t)) = \sum_{j=1}^r \sum_{j \in C_j} c_{\ell j} C(\hat{x}_j(t)) = \sum_{j=1}^r \hat{c}_{\ell j} C(\hat{x}_j(t)),$$

where

$$\hat{c}_{\ell j} = \sum_{j \in C_j} c_{\ell j}.$$

Thus, for $\widehat{\mathbf{C}} := [\widehat{\mathbf{c}}_{\ell_j}]$, we have $\widehat{\mathbf{C}} = \mathbf{C}\mathbf{P}$. Therefore, we showed how to construct a reduced model of the same structure as the original multi-agent system. Based on this, to find a good partition, we can apply any projection-based MOR method for nonlinear systems (e.g., proper orthogonal decomposition [10]) and cluster the block-rows of the matrix used to project the system.

5 Numerical examples

Here, we demonstrate our approach for different network examples, beginning with a small linear multi-agent system in Section 5.1. Next, in Section 5.2, we use the Van der Pol oscillator network.

The source code of the implementations used to compute the presented results can be obtained from

<https://doi.org/10.5281/zenodo.3700182>

and is authored by Petar Mlinarić.

5.1 Small network example

To illustrate distance to optimality, we use a small example from [24] with 10 single-integrator agents, where we can compute the \mathcal{H}_2 and \mathcal{H}_∞ errors for all possible partitions. The Laplacian and input matrices are

$$\mathbf{L} = \begin{bmatrix} 5 & 0 & 0 & 0 & 0 & -5 & 0 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 & -3 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 6 & -1 & -2 & -3 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 6 & -5 & 0 & 0 & 0 & 0 & 0 \\ 0 & -3 & -2 & -5 & 25 & -2 & -6 & -7 & 0 & 0 \\ -5 & -2 & -3 & 0 & -2 & 25 & -6 & -7 & 0 & 0 \\ 0 & 0 & 0 & 0 & -6 & -6 & 15 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & -7 & -7 & -1 & 15 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix},$$

and we chose the edge ordering and orientation such that the incidence and edge-weights matrices are

$$\mathbf{R} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

and $\mathbf{W} = \text{diag}(5, 3, 2, 1, 2, 3, 5, 2, 6, 7, 6, 7, 1, 1, 1)$, respectively. The output matrix is $\mathbf{C} = \mathbf{W}^{1/2}\mathbf{R}^T$.

For this example, we focus on partitions with five clusters. There are in total 42 525 such partitions. Table 1 shows the 15 best partitions with respect to the \mathcal{H}_2 and \mathcal{H}_∞ errors.

First, we used IRKA to find a reduced model of order $r = 5$. It found a reduced model with relative \mathcal{H}_2 error of 3.30412×10^{-2} , which is 3.88 times better than the best partition. The partition resulting from QR decomposition-based clustering applied to IRKA's V matrix is

$$\{\{1, 3\}, \{2, 4, 9, 10\}, \{5, 8\}, \{6\}, \{7\}\},$$

with the associated relative \mathcal{H}_2 error of 0.150654. It is more than 4 times worse than using IRKA, but note that this partition is the 14th best partition and that the best partition produces about 1.18 times better error.

We notice by (5) that W can also be used to find a good partition. In this example, QR decomposition-based clustering returns the partition

$$\{\{1, 2, 3, 9, 10\}, \{4, 8\}, \{5\}, \{6\}, \{7\}\},$$

with the relative \mathcal{H}_2 error 0.179746, which is worse than using only V from IRKA. Using the first five left singular vectors of $[V \ W]$ gives us

$$\{\{1, 2, 3, 4\}, \{5, 8\}, \{6\}, \{7\}, \{9, 10\}\},$$

which is the second best partition in terms of the \mathcal{H}_2 error and sixth best in terms of the \mathcal{H}_∞ error. Furthermore, using balanced truncation instead of IRKA produces the same partition, using either V or the first five left singular vectors of $[V \ W]$.

Therefore, at least in this example, clustering the rows of V and/or W gives close to optimal partitions. Additionally, k-means clustering performs better than QR decomposition-based clustering.

5.2 Van der Pol oscillators

Here, we use the Van der Pol oscillator network example from [19], where the agents are given by

$$\dot{x}_{i,1}(t) = x_{i,2}(t) + \sigma v_i(t), \quad (9a)$$

$$\dot{x}_{i,2}(t) = \mu(1 - x_{i,1}(t)^2)x_{i,2}(t) - x_{i,1}(t) - cv_i(t), \quad (9b)$$

and interconnections by

$$v_i(t) = \sum_{j=1}^n a_{ij}((x_{i,1}(t) - x_{j,1}(t)) + (x_{i,2}(t) - x_{j,2}(t))) + \sum_{k=1}^m b_{ik}u_k(t), \quad (9c)$$

with $\mu = 0.5$ and $\sigma = 0.1$. Additionally, we chose a larger 10×10 grid graph ($n = 100$), set the input matrix to be $B = e_1$ (i.e., one of the corner agents receives external input) and used $c = 100$ to have synchronization.

Figure 1 shows the state trajectory of the system for zero initial condition and input $u(t) = e^{-t}$, using an adaptive BDF integrator producing 987 snapshots. We used these snapshots to find the POD modes, with associated singular values shown in Figure 2.

Changing the input to $u(t) = e^{-t/10} \sin t$ gives the trajectory in Figure 3. Applying k-means clustering to the first two POD modes to generate 10 clusters produces a reduced model with the error trajectory in Figure 4. We computed the relative \mathcal{L}_2 error for k-means clustering using the first two POD modes with different number of clustering, which can be seen in Figure 5. For this example, we see that the error decays exponentially with the order of the reduced model.

6 Conclusions

We extended clustering by projection to a class of nonlinear multi-agent systems and presented our clustering-based MOR method, combining any projection-based MOR method

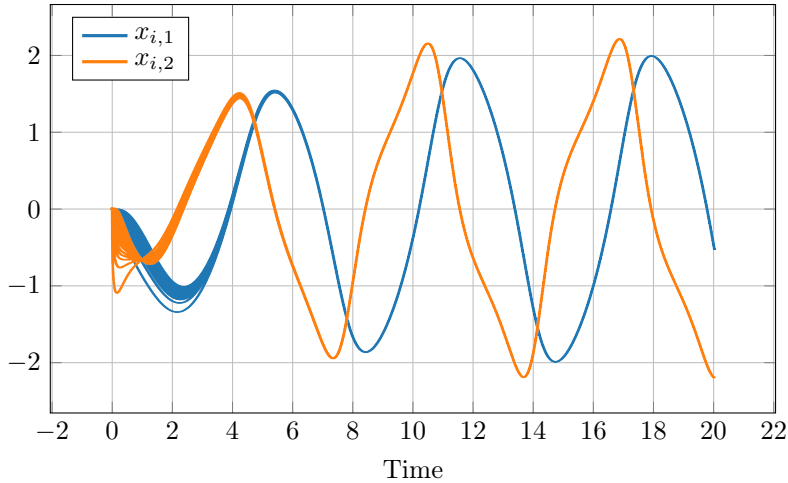


Figure 1: State trajectory of the Van der Pol oscillator network (9) for zero initial condition and input $u(t) = e^{-t}$

and a clustering algorithm, for reduction of multi-agent systems using graph partitions. In particular, we motivated the use of the k-means algorithm.

Our numerical test for a small network shows that our algorithm finds close to optimal partitions. We demonstrated our method on a larger nonlinear oscillator network.

References

- [1] A. C. Antoulas, C. A. Beattie, and S. Gugercin. Interpolatory model reduction of large-scale dynamical systems. In Javad Mohammadpour and Karolos M. Grigoriadis, editors, *Efficient Modeling and Control of Large-Scale Systems*, pages 3–58. Springer US, 2010.
- [2] C. A. Beattie, S. Gugercin, and S. Wyatt. Inexact solves in interpolatory model reduction. *Linear Algebra Appl.*, 436(8):2916–2943, 2012.
- [3] B. Besselink, H. Sandberg, and K. H. Johansson. Clustering-based model reduction of networked passive systems. *IEEE Trans. Autom. Control*, 61(10):2958–2973, 2016.
- [4] A. Chapman and M. Mesbahi. UAV flocking with wind gusts: Adaptive topology and model reduction. In *American Control Conference (ACC)*, pages 1045–1050, June 2011.
- [5] X. Cheng, Y. Kawano, and J. M. A. Scherpen. Graph structure-preserving model reduction of linear network systems. In *European Control Conference (ECC)*, pages 1970–1975, 2016.
- [6] X. Cheng, Y. Kawano, and J. M. A. Scherpen. Model reduction of multi-agent systems using dissimilarity-based clustering. *IEEE Trans. Autom. Control*, 2018.
- [7] C. Godsil and G. Royle. *Algebraic graph theory*, volume 207 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 2001.

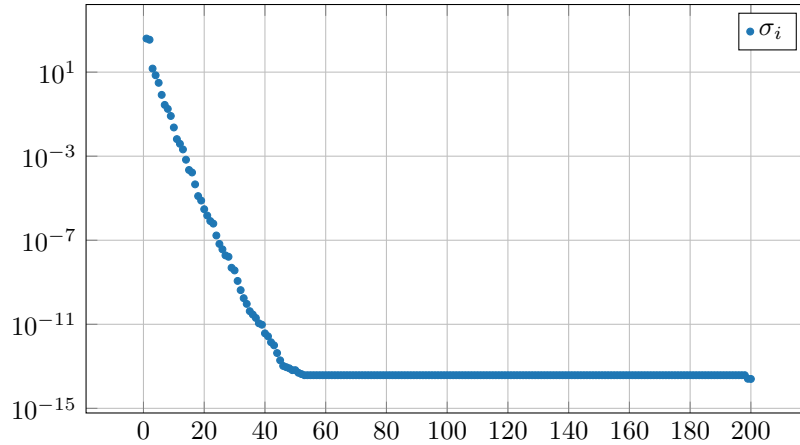


Figure 2: POD singular values based on snapshots from Figure 1

- [8] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, fourth edition, 2013.
- [9] J. A. Hartigan and M. A. Wong. Algorithm AS 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.
- [10] M. Hinze and S. Volkwein. Proper orthogonal decomposition surrogate models for nonlinear dynamical systems: Error estimates and suboptimal control. In P. Benner, V. Mehrmann, and D.C. Sorensen, editors, *Dimension Reduction of Large-Scale Systems*, volume 45 of *Lect. Notes Comput. Sci. Eng.*, pages 261–306. Springer-Verlag, Berlin/Heidelberg, Germany, 2005.
- [11] T. Ishizaki, K. Kashima, A. Girard, J. Imura, L. Chen, and K. Aihara. Clustering-based \mathcal{H}_2 -state aggregation of positive networks and its application to reduction of chemical master equations. In *51st IEEE Conference on Decision and Control (CDC)*, pages 4175–4180, December 2012.
- [12] T. Ishizaki, K. Kashima, A. Girard, J. Imura, L. Chen, and K. Aihara. Clustered model reduction of positive directed networks. *Automatica J. IFAC*, 59:238–247, 2015.
- [13] T. Ishizaki, K. Kashima, J. Imura, and K. Aihara. Model reduction and clusterization of large-scale bidirectional networks. *IEEE Trans. Autom. Control*, 59(1):48–63, January 2014.
- [14] T. Ishizaki, R. Ku, and J. Imura. Clustered model reduction of networked dissipative systems. In *American Control Conference (ACC)*, pages 3662–3667, 2016.
- [15] H.-J. Jongsma, P. Mlinarić, S. Grundel, P. Benner, and H. L. Trentelman. Model reduction of linear multi-agent systems by clustering with \mathcal{H}_2 and \mathcal{H}_∞ error bounds. *Math. Control Signals Systems*, 30(6), April 2018.

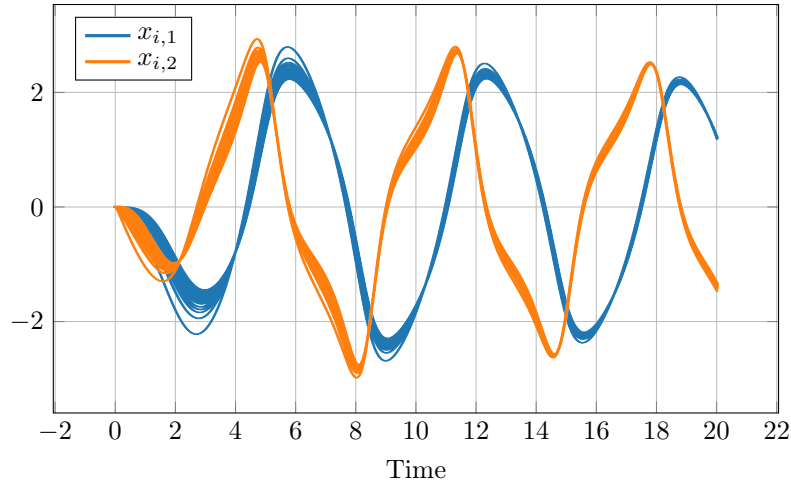


Figure 3: Van der Pol oscillator state trajectory for zero initial condition and input $u(t) = e^{-t/10} \sin t$

- [16] Z. Li, Z. Duan, G. Chen, and L. Huang. Consensus of multiagent systems and synchronization of complex networks: A unified viewpoint. *IEEE Trans. Circuits Syst. I, Regular Papers*, 57(1):213–224, January 2010.
- [17] S. Martini, M. Egerstedt, and A. Bicchi. Controllability decompositions of networked systems through quotient graphs. In *47th IEEE Conference on Decision and Control (CDC)*, pages 5244–5249, December 2008.
- [18] S. Martini, M. Egerstedt, and A. Bicchi. Controllability analysis of multi-agent systems using relaxed equitable partitions. *Int. J. Syst., Control Commun.*, 2(1/2/3):100–121, January 2010.
- [19] P. Massioni and G. Scorletti. Consensus analysis of large-scale nonlinear homogeneous multiagent formations with polynomial dynamics. *International Journal of Robust and Nonlinear Control*, 28(17):5605–5617, 2018.
- [20] M. Mesbahi and M. Egerstedt. *Graph Theoretic Methods in Multiagent Networks*. Princeton Series in Applied Mathematics. Princeton University Press, Princeton, NJ, 2010.
- [21] P. Mlinarić, S. Grundel, and P. Benner. Efficient model order reduction for multi-agent systems using QR decomposition-based clustering. In *54th IEEE Conference on Decision and Control (CDC)*, pages 4794–4799, December 2015.
- [22] P. Mlinarić, S. Grundel, and P. Benner. Clustering-based model order reduction for multi-agent systems with general linear time-invariant agents. In *22nd International Symposium on Mathematical Theory of Networks and Systems (MTNS)*, pages 230–235, Minneapolis, MN, USA, 2016.
- [23] N. Monshizadeh, H. L. Trentelman, and M. K. Camlibel. Stability and synchronization preserving model reduction of multi-agent systems. *Syst. Control Lett.*, 62(1):1–10, 2013.

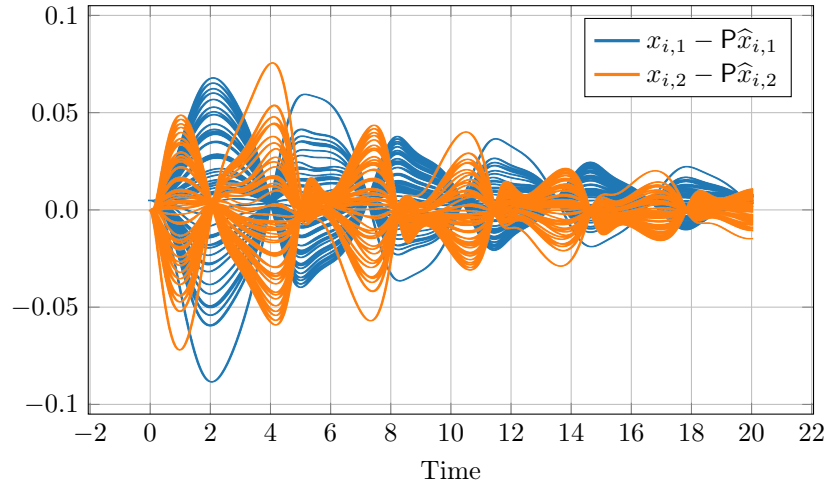


Figure 4: Van der Pol oscillator error when using k-means with the first two POD modes for zero initial condition and input $u(t) = e^{-t/10} \sin t$

- [24] N. Monshizadeh, H. L. Trentelman, and M. K. Camlibel. Projection-based model reduction of multi-agent systems using graph partitions. *IEEE Trans. Control Netw. Syst.*, 1(2):145–154, June 2014.
- [25] A. Rahmani, M. Ji, M. Mesbahi, and M. Egerstedt. Controllability of multi-agent systems from a graph-theoretic perspective. *SIAM J. Control Optim.*, 48(1):162–186, 2009.
- [26] S. E. Schaeffer. Graph clustering. *Comput. Sci. Rev.*, 1(1):27–64, 2007.
- [27] H. Zha, X. He, C. Ding, H. Simon, and M. Gu. Spectral relaxation for k-means clustering. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, pages 1057–1064, 2001.

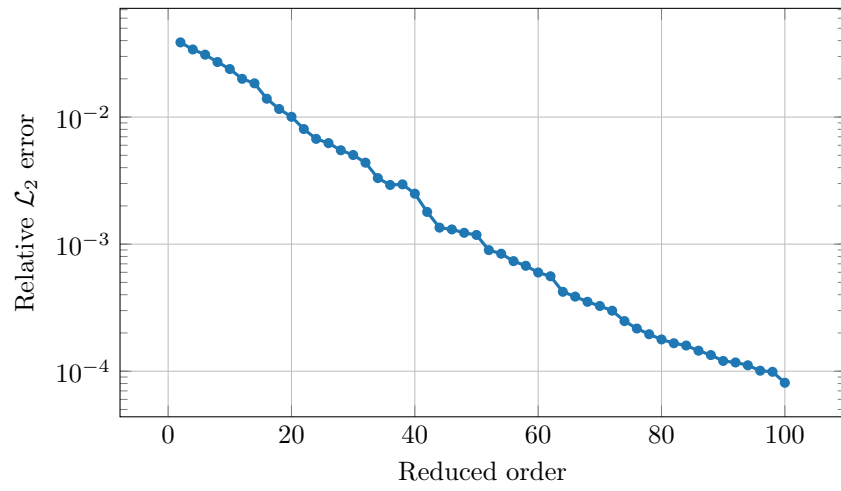


Figure 5: Relative \mathcal{L}_2 error for zero initial condition and test input $u(t) = e^{-t/10} \sin t$ for k-means clustering using the first two POD modes

Table 1: Top 15 partitions with 5 clusters by \mathcal{H}_2 error and \mathcal{H}_∞ error for reducing the multi-agent system in Section 5.1

Rank	Relative \mathcal{H}_2 error	Partition
1	0.128053	$\{\{1, 8\}, \{2, 3, 4, 9, 10\}, \{5\}, \{6\}, \{7\}\}$
2	0.131311	$\{\{1, 2, 3, 4\}, \{5, 8\}, \{6\}, \{7\}, \{9, 10\}\}$
3	0.137466	$\{\{1, 2, 3, 4, 9, 10\}, \{5\}, \{6\}, \{7\}, \{8\}\}$
4	0.137473	$\{\{1, 3, 8\}, \{2, 4, 9, 10\}, \{5\}, \{6\}, \{7\}\}$
5	0.143700	$\{\{1, 5, 8\}, \{2, 3, 4\}, \{6\}, \{7\}, \{9, 10\}\}$
6	0.145900	$\{\{1, 2, 3\}, \{4, 9, 10\}, \{5, 8\}, \{6\}, \{7\}\}$
7	0.146196	$\{\{1, 8\}, \{2, 3, 4, 9\}, \{5, 10\}, \{6\}, \{7\}\}$
8	0.146196	$\{\{1, 8\}, \{2, 3, 4, 10\}, \{5, 9\}, \{6\}, \{7\}\}$
9	0.147022	$\{\{1, 2, 3, 8\}, \{4, 9, 10\}, \{5\}, \{6\}, \{7\}\}$
10	0.149240	$\{\{1, 8, 10\}, \{2, 3, 4, 9\}, \{5\}, \{6\}, \{7\}\}$
11	0.149240	$\{\{1, 8, 9\}, \{2, 3, 4, 10\}, \{5\}, \{6\}, \{7\}\}$
12	0.149654	$\{\{1, 8\}, \{2, 4, 9, 10\}, \{3, 5\}, \{6\}, \{7\}\}$
13	0.150440	$\{\{1, 5\}, \{2, 3, 4, 9, 10\}, \{6\}, \{7\}, \{8\}\}$
14	0.150654	$\{\{1, 3\}, \{2, 4, 9, 10\}, \{5, 8\}, \{6\}, \{7\}\}$
15	0.151684	$\{\{1, 2, 8\}, \{3, 4, 9, 10\}, \{5\}, \{6\}, \{7\}\}$

Rank	Relative \mathcal{H}_∞ error	Partition
1	0.253975	$\{\{1, 3, 5, 8\}, \{2, 4\}, \{6\}, \{7\}, \{9, 10\}\}$
2	0.254376	$\{\{1, 2, 5, 8\}, \{3, 4\}, \{6\}, \{7\}, \{9, 10\}\}$
3	0.254818	$\{\{1, 5, 8\}, \{2, 3, 4\}, \{6\}, \{7\}, \{9, 10\}\}$
4	0.259483	$\{\{1, 2, 3, 5, 8\}, \{4\}, \{6\}, \{7\}, \{9, 10\}\}$
5	0.260859	$\{\{1, 2, 4\}, \{3, 5, 8\}, \{6\}, \{7\}, \{9, 10\}\}$
6	0.262244	$\{\{1, 2, 3, 4\}, \{5, 8\}, \{6\}, \{7\}, \{9, 10\}\}$
7	0.266387	$\{\{1, 3, 4\}, \{2, 5, 8\}, \{6\}, \{7\}, \{9, 10\}\}$
8	0.273663	$\{\{1, 4\}, \{2, 3, 5, 8\}, \{6\}, \{7\}, \{9, 10\}\}$
9	0.276919	$\{\{1, 4, 5, 8\}, \{2, 3\}, \{6\}, \{7\}, \{9, 10\}\}$
10	0.286961	$\{\{1, 3, 4, 5, 8\}, \{2\}, \{6\}, \{7\}, \{9, 10\}\}$
11	0.288414	$\{\{1, 2, 3\}, \{4, 5, 8\}, \{6\}, \{7\}, \{9, 10\}\}$
12	0.293773	$\{\{1, 5\}, \{2, 3, 4, 8\}, \{6\}, \{7\}, \{9, 10\}\}$
13	0.294028	$\{\{1, 2, 3, 4, 8\}, \{5\}, \{6\}, \{7\}, \{9, 10\}\}$
14	0.299845	$\{\{1, 2\}, \{3, 4, 5, 8\}, \{6\}, \{7\}, \{9, 10\}\}$
15	0.305583	$\{\{1, 2, 4, 8\}, \{3, 5\}, \{6\}, \{7\}, \{9, 10\}\}$