HAL
open science

Structure-based Clustering Algorithm for Model Reduction of Large-scale Network Systems

Muhammad Umar B. Niazi, Xiaodong Cheng, Carlos Canudas de Wit, Jacquelien M A Scherpen

To cite this version:

Muhammad Umar B. Niazi, Xiaodong Cheng, Carlos Canudas de Wit, Jacquelien M A Scherpen. Structure-based Clustering Algorithm for Model Reduction of Large-scale Network Systems. CDC 2019 - 58th IEEE Conference on Decision and Control, IEEE, Dec 2019, Nice, France. 10.1109/CDC40024.2019.9029349 . hal-02278884v2

HAL Id: hal-02278884

https://hal.science/hal-02278884v2

Submitted on 9 Sep 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Structure-based Clustering Algorithm for Model Reduction of Large-scale Network Systems

Muhammad Umar B. Niazi†, Xiaodong Cheng§, Carlos Canudas-de-Wit†, and Jacquelien M. A. Scherpen‡

***Abstract*— A model reduction technique is presented that identifies and aggregates clusters in a large-scale network system and yields a reduced model with tractable dimension. The network clustering problem is translated to a graph reduction problem, which is formulated as a minimization of distance from lumpability. The problem is a non-convex, mixed-integer optimization problem and only depends on the graph structure of the system. We provide a heuristic algorithm to identify clusters that are not only suboptimal but are also connected, that is, each cluster forms a connected induced subgraph in the network system.**

***Index Terms*— Large-scale systems, model reduction, lumpability, clustering algorithm.**

I. Introduction

Monitoring and control of large-scale network systems require tremendous amounts of sensing and computational resources. The paradigm of model reduction, [1]–[3], has proven to be quite effective in reducing the complexity of large-scale systems, such as chemical reaction networks [4], building thermal [5], and power networks [6].

For model reduction of network systems, preserving the topological structure is also critical in addition to the dynamical properties of the system. In this regard, clustered model reduction techniques proposed in [7]–[11] have shown promising results. Such techniques not only preserve the topological structure but also provide the technical tools to quantify model approximation error. The main goal is to identify clusters in a network system that can be aggregated to obtain a reduced system, whose state space is of tractable dimension and whose input-output behavior is similar to the input-output behavior of the original network system. This similarity is achieved by minimizing a model approximation error, which is characterized in terms of $\mathcal{H}_2$ or $\mathcal{H}_\infty$ norm of the difference between the frequency responses of both systems.

In this paper, we propose a clustering approach that is based only on the graph structure of the system. Such an approach is called structure-based clustering and it is computationally efficient because the computation of $\mathcal{H}_2$ or $\mathcal{H}_\infty$ norms is not required. The dynamics of a large-scale network system are projected on a lower-dimensional state space, [12], which yields a projected system whose state vector contains the aggregated states of the clusters. The projected system is shown to be influenced by the average deviation vector that is considered to be an unknown input with a certain structure. Neglecting the average deviation vector yields a reduced system, which is an approximation of the projected system. To obtain a better approximation, therefore, we argue that it is sufficient to identify the clusters such that the influence of average deviation vector is minimized. By exploiting the structure of the average deviation vector, we show that the problem boils down to a graph reduction problem, [13], [14].

We revisit the notion of lumpability, [15]–[17], which guarantees that the states of the projected and the reduced systems are equal. We show that lumpability is the ideal situation for model reduction. However, it is difficult to achieve by a clustering algorithm; especially when there are constraints on the number of clusters, the connectivity of clusters, and the placements of dedicated sensors in the network. Therefore, the clustering problem is formulated as a minimization of the distance from lumpability, which is the difference between the states of the projected system and the reduced system. Such an approach is quite reasonable, for instance, in the estimation of the average states of multiple clusters in a network system, [18]–[20].

The clustering problem is a non-convex, mixed integer optimization problem. The initial clustering is generated by an algorithm based on a graph cut method. Then, a heuristic algorithm is proposed that provides a suboptimal clustering solution such that each cluster forms a connected induced subgraph in the network, which is a crucial requirement when dealing with physical network systems, [21].

II. System Definitions

Consider a *network system* with a linear time-invariant state-space representation

$$\mathbf{\Sigma}: \left\{ \begin{array}{rcl} \dot{\mathbf{x}}(t) & = & A\mathbf{x}(t) + B\mathbf{u}(t) \\ \mathbf{y}(t) & = & C\mathbf{x}(t) \end{array} \right. ,$$

where $\mathbf{x}(t) = \begin{bmatrix} x_1(t) & \cdots & x_n(t) \end{bmatrix}^\top \in \mathbb{R}^n$ is the state vector, $\mathbf{u}(t) = \begin{bmatrix} u_1(t) & \cdots & u_p(t) \end{bmatrix}^\top \in \mathbb{R}^p$ is the input vector, $\mathbf{y}(t) \in \mathbb{R}^m$ is the output vector, and $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$ and $C \in \mathbb{R}^{m\times n}$ are the system matrices.

We assume a symmetric state matrix, i.e., $A = A^\top$, which corresponds to an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \ldots, n\}$ is the index set of nodes and $\mathcal{E} = \mathcal{V} \times \mathcal{V}$ the set of edges. That is, if $(i, j) \in \mathcal{E}$, then the ij-entry of

† M.U.B. Niazi and C. Canudas-de-Wit are with GIPSA-Lab, CNRS, Grenoble, France. Email: {muhammad-umar-b.niazi,carlos.canudas-de-wit}@gipsa-lab.fr

§ X. Cheng is with Control Systems Group, Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, Netherlands. Email: x.cheng@tue.nl

‡ J.M.A. Scherpen is with Jan C. Willems Center for Systems and Control, Engineering and Technology Institute, Faculty of Science and Engineering, University of Groningen, Groningen, Netherlands. Email: j.m.a.scherpen@rug.nl

This work is supported by European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme, ERC-AdG no. 694209, Scale-FreeBack (Website: http://scale-freeback.eu/).

A is $a_{ij} = a_{ji} \neq 0$; otherwise, $a_{ij} = a_{ji} = 0$. Without loss of generality, suppose that the output $\mathbf{y}(t)$ consists of the dedicated state measurements at m nodes, which are called *measured nodes*. Let the set of measured nodes be $\mathcal{V}_m \subset \mathcal{V}$. The remaining nodes $\mathcal{V}_u = \mathcal{V} \setminus \mathcal{V}_m$ are called *unmeasured nodes*, where $|\mathcal{V}_u| = l$ and $l = n-m$. Suppose the reordering of nodes such that $\mathcal{V}_u = \{1, \cdots, l\}$ and $\mathcal{V}_m = \{l+1, \cdots, n\}$. Thus the output matrix $C = [\ 0_{m\times l} \quad I_m\ ]$, and the state matrix is partitioned as

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix},$$

where $A_{11} \in \mathbb{R}^{l\times l}$, $A_{12} \in \mathbb{R}^{l\times m}$, $A_{21} \in \mathbb{R}^{m\times l}$, and $A_{22} \in \mathbb{R}^{m\times m}$.

Suppose a *clustering* (or *partition*) of unmeasured nodes be $\mathcal{Q} = \{\mathcal{C}_1, \cdots, \mathcal{C}_k\}$ such that $\mathcal{C}_1 \cup \cdots \cup \mathcal{C}_k = \mathcal{V}_u$ and for any $\alpha \neq \beta$ it holds that $\mathcal{C}_\alpha \cap \mathcal{C}_\beta = \emptyset$, where $\alpha, \beta = 1, \cdots, k$ and each of the set $\mathcal{C}_\alpha$ is called a *cluster*. Let $Q \in \mathbb{R}^{l\times k}$ be the *characteristic matrix* of $\mathcal{Q}$, which is defined as

$$[Q]_{i\alpha} = \begin{cases} 1, & \text{if } i \in \mathcal{C}_\alpha, \\ 0, & \text{otherwise,} \end{cases}$$

for $i \in \mathcal{V}_u$ and $\alpha = 1, \cdots, k$. Note that $Q\mathbf{1}_k = \mathbf{1}_l$, i.e., each node is assigned to at least one (and only one) cluster, and $Q^\top Q = \mathrm{diag}[\,n_1, \cdots, n_k\,]$, where $n_\alpha = |\mathcal{C}_\alpha|$. Throughout the paper, we suppose $\alpha, \beta = 1, \cdots, k$.

The *projection matrix* $P \in \mathbb{R}^{n\times(k+m)}$ is defined as

$$P = \begin{bmatrix} QN & 0_{l\times m} \\ 0_{m\times k} & I_m \end{bmatrix}, \tag{1}$$

where $N = (Q^\top Q)^{-\frac{1}{2}}$. Note that $P^\top P = I_{k+m}$ and there exists $\bar{P} \in \mathbb{R}^{n\times(k+m)}$ such that $PP^\top + \bar{P}\bar{P}^\top = I_n$, where $P^\top \bar{P} = 0$ (see [1], [22]). We project the state vector $\mathbf{x}(t)$ on a lower dimensional state space and obtain

$$\mathbf{z}(t) = P^\top \mathbf{x}(t) = [\,\mathbf{z}_1(t)^\top \ \ \mathbf{y}(t)^\top\,]^\top \in \mathbb{R}^{\hat{n}},$$

where $\hat{n} = k + m$ and $\mathbf{z}_1(t) = [\,z_1(t) \ \cdots \ z_k(t)\,]^\top$ with

$$z_\alpha(t) = \frac{1}{\sqrt{n_\alpha}} \sum_{j\in\mathcal{C}_\alpha} x_j(t).$$

Then, we have $\mathbf{x}(t) = P\mathbf{z}(t) + \bar{P}\bar{P}^\top \mathbf{x}(t)$, where

$$\bar{P}\bar{P}^\top \mathbf{x}(t) = (I - PP^\top)\mathbf{x}(t) = \begin{bmatrix} J \\ 0_{m\times n} \end{bmatrix} \mathbf{x}(t)$$

with $J = [\ I - QN^2Q^\top \quad 0_{l\times m}\ ] \in \mathbb{R}^{l\times n}$. The vector $J\mathbf{x}(t) =: \boldsymbol{\sigma}(t) \in \mathbb{R}^l$ is the *average deviation vector*, whose i-th entry is given by

$$\sigma_i(t) = \begin{cases} x_i(t) - \dfrac{1}{\sqrt{n_\alpha}} z_\alpha(t)\,, & \text{if } i \in \mathcal{C}_\alpha, \\ 0\,, & \text{if } i \in \mathcal{V}_m, \end{cases} \tag{2}$$

where $\left(1/\sqrt{n_\alpha}\right) z_\alpha(t)$ is the average mean of the states of $\mathcal{C}_\alpha$ at time $t \in \mathbb{R}_{\geq 0}$.

The *projected system* corresponding to the network system $\boldsymbol{\Sigma}$ is given as

$$\mathring{\boldsymbol{\Sigma}} : \begin{cases} \dot{\mathbf{z}}(t) &= E\mathbf{z}(t) + F\boldsymbol{\sigma}(t) + G\mathbf{u}(t) \\ \boldsymbol{\sigma}(t) &= J\mathbf{x}(t) \\ \mathbf{y}(t) &= H\mathbf{z}(t) \end{cases},$$

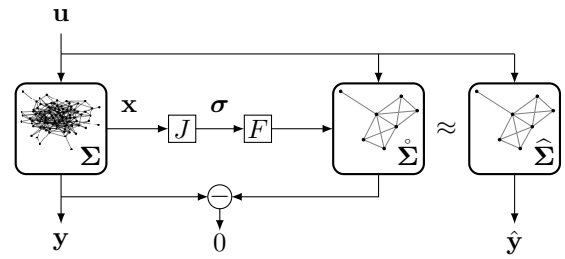


Fig. 1: The reduced system $\widehat{\boldsymbol{\Sigma}}$ is obtained by approximating the projected system $\mathring{\boldsymbol{\Sigma}}$, whose state space is a lower-dimensional projection of the state space of large-scale network system $\boldsymbol{\Sigma}$. The output of $\mathring{\boldsymbol{\Sigma}}$ is equal to the output of $\boldsymbol{\Sigma}$, whereas the output of $\widehat{\boldsymbol{\Sigma}}$ approximates the output of $\boldsymbol{\Sigma}$.

where $E = P^\top AP$, $G = P^\top B$, $H = CP$, and $F\boldsymbol{\sigma}(t) = P^\top A\bar{P}\bar{P}^\top \mathbf{x}(t)$ with

$$F = \begin{bmatrix} NQ^\top A_{11} \\ A_{21} \end{bmatrix}.$$

Note that the output $\mathbf{y}(t)$ of the original system $\boldsymbol{\Sigma}$ and the projected system $\mathring{\boldsymbol{\Sigma}}$ is equal because

$$\mathbf{y}(t) = C\mathbf{x}(t) = C\left(P\mathbf{z}(t) + \bar{P}\bar{P}^\top \mathbf{x}(t)\right) = CP\mathbf{z}(t),$$

where $C\bar{P}\bar{P}^\top = 0$. Also, note that $\boldsymbol{\sigma}(t)$ enters $\mathring{\boldsymbol{\Sigma}}$ as an unknown input since it corresponds to the states of unmeasured nodes $\mathcal{V}_u$. However, we remark that it satisfies the following structural property

$$Q^\top \boldsymbol{\sigma}(t) = 0, \quad \forall\, t \in \mathbb{R}_{\geq 0}. \tag{3}$$

By approximating $\mathring{\boldsymbol{\Sigma}}$, i.e., assuming $F\boldsymbol{\sigma}(t) \approx 0$, we obtain the *reduced system*

$$\widehat{\boldsymbol{\Sigma}} : \begin{cases} \dot{\hat{\mathbf{z}}}(t) &= E\hat{\mathbf{z}}(t) + G\mathbf{u}(t) \\ \hat{\mathbf{y}}(t) &= H\hat{\mathbf{z}}(t) \end{cases},$$

where $\hat{\mathbf{z}}(t) \in \mathbb{R}^{\hat{n}}$ and $\hat{\mathbf{y}}(t) \in \mathbb{R}^m$. The relation between $\boldsymbol{\Sigma}$, $\mathring{\boldsymbol{\Sigma}}$, and $\widehat{\boldsymbol{\Sigma}}$ is illustrated by Figure 1.

III. Lumpability

Lumpability is a notion that ensures that the state trajectories of the reduced-system $\widehat{\boldsymbol{\Sigma}}$ and the projected system $\mathring{\boldsymbol{\Sigma}}$ remain equal for all $t \in \mathbb{R}_{\geq 0}$. This, in turn, ensures that $\mathbf{y}(t) = \hat{\mathbf{y}}(t)$ for all $t \in \mathbb{R}_{\geq 0}$, i.e., the output trajectories of $\widehat{\boldsymbol{\Sigma}}$ and $\boldsymbol{\Sigma}$ remain equal. Although lumpability is already defined in [15]–[17], we define it from the perspective of the projected system.

Definition 1. Let $\mathring{\boldsymbol{\Sigma}}$ be the projected system of the network system $\boldsymbol{\Sigma}$ and let $\widehat{\boldsymbol{\Sigma}}$ be the reduced system obtained by approximating $\mathring{\boldsymbol{\Sigma}}$. Then, we say that $\boldsymbol{\Sigma}$ is *lumpable* to $\widehat{\boldsymbol{\Sigma}}$ if $\mathbf{z}(0) = \hat{\mathbf{z}}(0)$ implies $\mathbf{z}(t) = \hat{\mathbf{z}}(t)$ for all $t \in \mathbb{R}_{>0}$, where $\mathbf{z}(t)$ and $\hat{\mathbf{z}}(t)$ are the states of $\mathring{\boldsymbol{\Sigma}}$ and $\widehat{\boldsymbol{\Sigma}}$, respectively. $\diamondsuit$

To show that the above definition is equivalent to the definition given in [15]–[17], we present the following necessary and sufficient condition of lumpability. We associate this condition to [12], which to our knowledge is the earliest to provide it.

Theorem 1 (Aoki [12])**.** Consider a network system $\boldsymbol{\Sigma}$ with a set $\mathcal{V}_m$ of m measured nodes and a clustering (or partition)

$\mathcal{Q} = \{\mathcal{C}_1, \cdots, \mathcal{C}_k\}$ of l unmeasured nodes. Furthermore, let $\mathring{\mathbf{\Sigma}}$ to be the projected system of $\mathbf{\Sigma}$, and let $\widehat{\mathbf{\Sigma}}$ to be the reduced system. Then, $\mathbf{\Sigma}$ is lumpable to $\widehat{\mathbf{\Sigma}}$ if and only if

$$P^\top A = E P^\top, \tag{4}$$

where $P \in \mathbb{R}^{n \times \hat{n}}$ is the projection matrix given in (1).

Proof. Consider the difference between the state trajectories of $\mathring{\mathbf{\Sigma}}$ and $\widehat{\mathbf{\Sigma}}$

$$\mathbf{z}(t) - \hat{\mathbf{z}}(t) = \int_0^t \exp[E(t-\tau)] F \boldsymbol{\sigma}(\tau) d\tau, \tag{5}$$

where we assume that $\mathbf{z}(0) = \hat{\mathbf{z}}(0)$.

Sufficiency: Assume (4) holds, then

$$F\boldsymbol{\sigma}(t) = P^\top A \bar{P} \bar{P}^\top \mathbf{x}(t) = E P^\top \bar{P} \bar{P}^\top \mathbf{x}(t) = 0,$$

for all $t \in \mathbb{R}_{\geq 0}$, because $P^\top \bar{P} = 0$. Thus $\mathbf{z}(t) - \hat{\mathbf{z}}(t) = 0$ for all $t \in \mathbb{R}_{\geq 0}$.

Necessity: Let $\mathbf{z}(t) - \hat{\mathbf{z}}(t) = 0$, for all $t \in \mathbb{R}_{\geq 0}$, then

$$\int_0^t \exp[P^\top A P (t-\tau)]\, P^\top A \bar{P} \bar{P}^\top \mathbf{x}(\tau) d\tau = 0.$$

The above integral is zero if, and only if, $P^\top A \bar{P} \bar{P}^\top \mathbf{x}(t) = 0$ for all $t \in \mathbb{R}_{\geq 0}$. Since $\mathbf{x}(t)$ evolves in $\mathbb{R}^n$ and the columns of $\bar{P}$ span $\ker(P^\top)$, therefore $P^\top A \bar{P} \bar{P}^\top \mathbf{x}(t) = 0$ is equivalent to the existence of $X \in \mathbb{R}^{\hat{n} \times \hat{n}}$ such that $P^\top A = X P^\top$. By multiplying P from right, we obtain $X = P^\top A P$, which proves the necessity of (4). ■

The condition (4) of lumpability can be interpreted in many ways. Algebraically, (4) is equivalent to saying that $\ker(P^\top)$ is A-invariant or that $\ker(P^\top) \subseteq \ker(P^\top A)$, [17]. However, in terms of graph theory, (4) is satisfied only if the partition $\mathcal{N} = \{\mathcal{C}_1, \cdots, \mathcal{C}_k, \mathcal{V}_m\} \supset \mathcal{Q}$ is almost equitable, [7], [23]. This is a necessary condition, and not sufficient, because almost equitable partition only concerns with the inter-cluster topology of the graph and excludes the induced subgraph formed by each cluster. Also, it excludes the diagonal values of A, which play a crucial role in lumpability. However, when A is assumed to be the Adjacency matrix of $\mathcal{G}$, we have a necessary and sufficient condition, that is, (4) holds if and only if $\mathcal{N}$ is an equitable partition, [24], [25].

The goal of model reduction is to minimize $\|\mathbf{y}(t) - \hat{\mathbf{y}}(t)\|$ for all $t \in \mathbb{R}_{\geq 0}$ by choosing the suitable clustering $\mathcal{Q}$. For this purpose, we remark that lumpability is the ideal situation. That is, since lumpability ensures that $\|\mathbf{z}(t) - \hat{\mathbf{z}}(t)\| = 0$, it also ensures $\|\mathbf{y}(t) - \hat{\mathbf{y}}(t)\| = 0$, for all $t \in \mathbb{R}_{\geq 0}$, because

$$\|\mathbf{y}(t) - \hat{\mathbf{y}}(t)\| \leq |||H|||\, \|\mathbf{z}(t) - \hat{\mathbf{z}}(t)\|,$$

where $|||H||| = 1$ with $|||\cdot|||$ induced by the Euclidean norm. However, the number of clusters, k, is already specified and dedicated sensors already placed at m measured nodes in the network system. In addition to that, we require that the induced subgraph formed by each cluster in the network is connected. In the presence of such constraints, lumpability is almost impossible to achieve by clustering algorithms. Nevertheless, we can achieve 'almost' lumpability by postulating the clustering problem that aims to minimize $\|\mathbf{z}(t) - \hat{\mathbf{z}}(t)\|$ for all $t \in \mathbb{R}_{\geq 0}$.

IV. Clustering Problem

The foundation of the clustering problem has been laid above. That is, the clustering $\mathcal{Q}$ of unmeasured nodes must be such that $\mathbf{\Sigma}$ is 'almost' lumpable to $\widehat{\mathbf{\Sigma}}$. Therefore, we formulate the clustering problem as a minimization of $\delta(t) := \|\mathbf{z}(t) - \hat{\mathbf{z}}(t)\|$, for all $t \in \mathbb{R}_{\geq 0}$, which is the distance from lumpability and can be considered as a model approximation error.

From (5), we have

$$\begin{aligned} \delta(t) &\leq \int_0^t |||\exp[E(t-\tau)]|||\, \|F\boldsymbol{\sigma}(\tau)\| d\tau \\ &\leq \left(\sup_{0 \leq \gamma \leq t} \|F\boldsymbol{\sigma}(\gamma)\| \right) \int_0^t |||\exp(E\tau)||| d\tau. \end{aligned}$$

Due to the interlacing property, [24], i.e., the eigenvalues of E interlace the eigenvalues of A, we note that the value of $\int_0^t |||\exp(E\tau)||| d\tau$ depends on the eigenvalues of A. However, we can minimize $\|F\boldsymbol{\sigma}(t)\|$ for all $t \in \mathbb{R}_{\geq 0}$. Note that $\|F\boldsymbol{\sigma}(t)\| = \|FJ\mathbf{x}(t)\| \leq |||FJ|||\, \|\mathbf{x}(t)\|$, where $|||FJ||| = |||\Psi(Q)|||$ with

$$\Psi(Q) = \begin{bmatrix} N Q^\top A_{11} (I - Q N^2 Q^\top) \\ A_{21} (I - Q N^2 Q^\top) \end{bmatrix}. \tag{6}$$

Since $\|\mathbf{x}(t)\|$ depends on the dynamical properties of $\mathbf{\Sigma}$, we aim to minimize $|||\Psi(Q)|||$ by choosing a suitable clustering $\mathcal{Q}$ of unmeasured nodes with a characteristic matrix Q.

Suppose we require k clusters of unmeasured nodes, i.e., $\mathcal{Q} = \{\mathcal{C}_1, \cdots, \mathcal{C}_k\}$ with its characteristic matrix $Q \in \mathbb{R}^{l \times k}$ and $N = (Q^\top Q)^{-\frac{1}{2}}$. Then, the clustering problem is formulated as

$$\begin{aligned} \min_{Q \in \mathbb{R}^{l \times k}} \quad & |||\Psi(Q)||| \\ \text{subject to} \quad & Q \mathbf{1}_k = \mathbf{1}_l, \quad [Q]_{i\alpha} \in \{0, 1\}; \end{aligned} \tag{7}$$

where $i = 1, \cdots, l$ and $\alpha = 1, \cdots, k$. Note that the constraint $Q\mathbf{1}_k = \mathbf{1}_l$ guarantees that each unmeasured node is assigned to at least one and only one cluster.

It is worth mentioning that (7) is a non-convex, mixed integer optimization problem. That is to say that solving (7) only yields a suboptimal solution, which is a local minimum.

V. Structure-based Clustering Algorithm

We present a heuristic algorithm to identify clusters corresponding to a suboptimal solution Q^* of (7). Note that (7) can be solved by the existing algorithms, [26]; however, the connectedness of each cluster may not be guaranteed. That is, we may obtain a solution $\mathcal{Q}^* = \{\mathcal{C}_1, \cdots, \mathcal{C}_k\}$ corresponding to the characteristic matrix Q^* such that the induced subgraphs $\mathcal{G}_\alpha = (\mathcal{C}_\alpha, \mathcal{E}_\alpha)$, where $\mathcal{E}_\alpha = \mathcal{E} \cap (\mathcal{C}_\alpha \times \mathcal{C}_\alpha)$, are not connected. Such a clustering is not desirable since it complicates the interpretation of the reduced graph in terms of the original graph. The algorithm presented in this section guarantees the connectedness of the obtained clusters.

A. Notations

The connectedness of the clusters means that, for every pair of nodes $i, j \in \mathcal{C}_\alpha$, we require that there exists a path (a sequence of edges in $\mathcal{E}_\alpha$) that connect i and j. If there

exists such a path, then we say that i is adjacent to $\mathcal{C}_\alpha$ and use a notation $i \leftrightarrow \mathcal{V}_\alpha$. If $i \in \mathcal{C}_\alpha$, where $\mathcal{G}_\alpha$ is the subgraph formed by $\mathcal{C}_\alpha$, then removing i from $\mathcal{C}_\alpha$ gives a graph $\mathcal{G}_\alpha^{-i}$. If the graph $\mathcal{G}_\alpha^{-i}$ is connected, which is equivalent to verifying $\operatorname{rank} L(\mathcal{G}_\alpha^{-i}) = |\mathcal{C}_\alpha| - 2$, then we say that removing i doesn't disconnect the graph $\mathcal{G}_\alpha$ and use a notation $i \hookleftarrow \mathcal{C}_\alpha$.

B. Clustering Initialization

Consider the induced subgraph $\mathcal{G}_u = (\mathcal{V}_u, \mathcal{E}_u)$, where $\mathcal{V}_u = \{1, \cdots, l\}$ is the set of unmeasured nodes, $l = n - m$, and $\mathcal{E}_u = \mathcal{E} \cap (\mathcal{V}_u \times \mathcal{V}_u)$ are the edges among $\mathcal{V}_u$. We associate a symmetric, weighted adjacency matrix $\Lambda \in \mathbb{R}^{l \times l}$ to $\mathcal{G}_u$ as

$$[\Lambda]_{ij} = [\Lambda]_{ji} = \begin{cases} |||\Psi(R_{ij})|||, & \text{if } (i,j) \in \mathcal{E}_u; \\ 0, & \text{otherwise;} \end{cases} \tag{8}$$

where $\Psi(\cdot)$ is given in (6) and $R_{ij} \in \mathbb{R}^{l \times (l-1)}$ is the characteristic matrix that puts the nodes i and j in one cluster while keeping other $l-2$ nodes in distinct $l-2$ clusters. Note that the matrix N in (6) is computed according to R_{ij} at this step, i.e., $N(R_{ij}) = (R_{ij}^\top R_{ij})^{-\frac{1}{2}}$. To illustrate further, if there exists an edge $(i,j) \in \mathcal{E}_u$, we associate a weight on the edge which is the cost of assigning nodes i and j into one cluster while keeping all other nodes unassigned. The Laplacian matrix of $\mathcal{G}_u$ is defined as

$$\Gamma = \operatorname{diag}(\Lambda \mathbf{1}_l) - \Lambda, \tag{9}$$

where $\operatorname{diag}(\Lambda \mathbf{1}_l) \in \mathbb{R}^{l \times l}$ is a diagonal matrix with its ii-entry equal to $[\Lambda \mathbf{1}_l]_i$.

Algorithm 1 Clustering initialization

Input: Number of unmeasured nodes l, number of clusters k, and the state matrix A.
Output: Clustering $\mathcal{Q}_0 = \{\mathcal{C}_1, \mathcal{C}_2, \cdots, \mathcal{C}_k\}$.

1: Obtain $\mathcal{G}_u = (\mathcal{V}_u, \mathcal{E}_u)$ and construct Λ as in (8).
2: **repeat**
3: Find an edge $(\mu, \nu) := \arg\max_{(i,j) \in \mathcal{E}_u} [\Lambda]_{ij}$.
4: Remove (μ, ν) from $\mathcal{G}_u$ and let $[\Lambda]_{\mu\nu} = [\Lambda]_{\nu\mu} = 0$.
5: Compute Γ as in (9).
6: **until** $\operatorname{rank} \Gamma = l - k$
7: Let $c_{\max} := \arg\max_\alpha |\mathcal{C}_\alpha|$ and $c_{\min} := \arg\min_\alpha |\mathcal{C}_\alpha|$.
8: **repeat**
9: **for** each node $i \in \mathcal{C}_{c_{\max}}$ **do**
10: **if** $i \hookleftarrow \mathcal{C}_{c_{\max}}$ and $i \leftrightarrow \mathcal{C}_{c_{\min}}$ **then**
11: Move i from $\mathcal{C}_{c_{\max}}$ to $\mathcal{C}_{c_{\min}}$.
12: **end if**
13: **end for**
14: $c_{\max} := \arg\max_\alpha |\mathcal{C}_\alpha|$ and $c_{\min} := \arg\min_\alpha |\mathcal{C}_\alpha|$.
15: **until** ($c_{\max} \leq \frac{l}{k}$ and $c_{\min} > 1$) or the maximum number of iterations.
16: **return** $\mathcal{Q}_0 = \{\mathcal{C}_1, \mathcal{C}_2, \cdots, \mathcal{C}_k\}$.

The lines 1–5 of Algorithm 1 remove the edge between a pair of nodes i and j that yield a higher cost $|||\Psi(R_{ij})|||$. The process iterates until we are left with k connected components of $\mathcal{G}_u$. Note that $\mathcal{G}_u$ has $k \leq l$ connected components if and only if $\operatorname{rank} \Gamma = l - k$. However, this process may yield clusters that consist of just one node. To obtain a better clustering initialization, the lines 6–14 of the algorithm transfer nodes from the biggest cluster to the smallest one in an iterative fashion. A node is transferred only if it is adjacent to the smallest cluster and removing it doesn't make the induced subgraph formed by the biggest cluster disconnected.

C. Heuristic Algorithm

We obtain an initial clustering $\mathcal{Q}_0$ from Algorithm 1 and compute its characteristic matrix $Q_0 \in \mathbb{R}^{l \times k}$.

Algorithm 2 Structure-based clustering

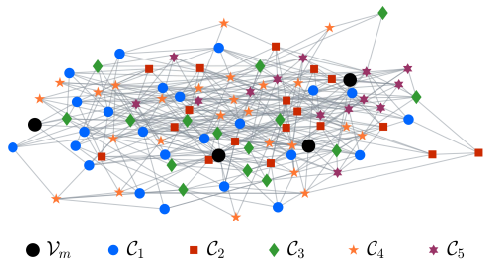
Input: Number of unmeasured nodes l, state matrix A, and the initial clustering $\mathcal{Q}_0 = \{\mathcal{C}_1, \cdots, \mathcal{C}_k\}$.
Output: Suboptimal graph clustering $\mathcal{Q}^* = \{\mathcal{C}_1^*, \cdots, \mathcal{C}_k^*\}$

Compute $\psi_0 = |||\Psi(Q_0)|||$.
Assign $\psi_M \leftarrow \psi_0$.
repeat
  for each node $i = 1, 2, \cdots, l$ **do**
    Let ζ be such that $i \in \mathcal{C}_\zeta$.
    Assign $\psi \leftarrow \psi_M$ and $M \leftarrow \zeta$.
    if $|\mathcal{C}_\zeta| > 1$ and $i \hookleftarrow \mathcal{C}_\zeta$ **then**
      for $\theta = 1 : k$, $\theta \neq \zeta$ and $i \leftrightarrow \mathcal{C}_\theta$ **do**
        Move node i from $\mathcal{C}_\zeta$ into $\mathcal{C}_\theta$.
        Compute the characteristic matrix Q.
        Compute $\psi_\theta = |||\Psi(Q)|||$.
        if $\psi_\theta < \psi$ **then**
          $\psi \leftarrow \psi_\theta$, $M \leftarrow \theta$
        end if
        Move node i back to the cluster $\mathcal{C}_\zeta$.
      end for
      Move node i from $\mathcal{C}_\zeta$ to $\mathcal{C}_M$, and $\psi_M \leftarrow \psi$.
    end if
  end for
until convergence or the maximum number of iterations.
return $\mathcal{Q}^* = \{\mathcal{V}_1^*, \cdots, \mathcal{V}_k^*\}$

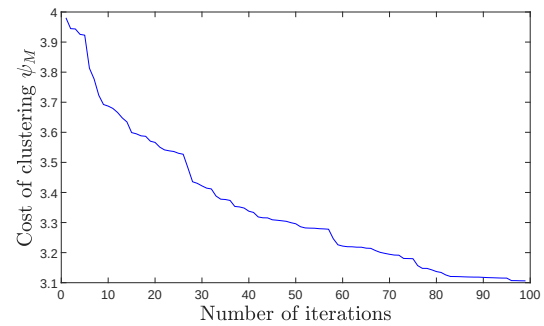
At each iteration, Algorithm 2 moves each node $i \in \mathcal{C}_\zeta$ to a cluster $\mathcal{C}_\theta$ such that $i \hookleftarrow \mathcal{C}_\zeta$, $i \leftrightarrow \mathcal{C}_\theta$, and the cost in (7) is minimized, where $\beta, \gamma \in \{1, \cdots, k\}$. Note that this algorithm converges to a suboptimal solution in a finite number of iterations. Moreover, the constraints $i \hookleftarrow \mathcal{C}_\zeta$ and $i \leftrightarrow \mathcal{C}_\theta$ yield $\mathcal{Q}^*$ with connected clusters.

VI. Simulation Example

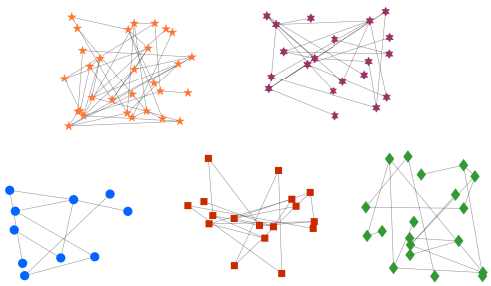
Suppose an undirected random graph $\mathcal{G}$ representing $\mathbf{\Sigma}$ as shown in Fig. 2(a) with 100 nodes. We assume 4 measured nodes $\mathcal{V}_m$ shown as black and find a suboptimal clustering with 5 clusters $\mathcal{C}_1, \cdots, \mathcal{C}_5$ for 96 unmeasured nodes. The state matrix $A = -L(\mathcal{G})$, the input matrix $B \in \{0,1\}^{100 \times 4}$ is generated randomly, and the input vector $\mathbf{u}(t) = [\sin t \quad \sin 5t \quad \sin 10t \quad \sin 50t]^\top$. We initialize the clusters by using Algorithm 1, where the connected subgraphs formed by each cluster are shown in Fig. 2(c). Then, Algorithm 2 finds a suboptimal clustering as shown in Fig.2(a), where each cluster forms a connected induced



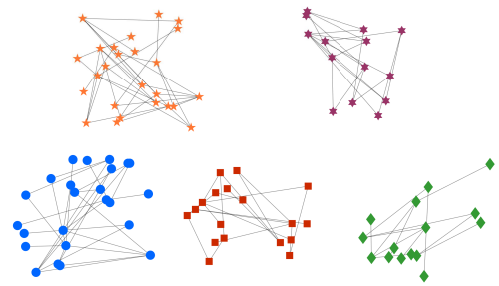
(a) Random network with 100 nodes, $\mathcal{V}_m$ are the measured nodes, and $\mathcal{C}_1, \cdots, \mathcal{C}_5$ are the obtained suboptimal clusters.



(b) Cost minimization at each iteration.



(c) Connected induced subgraphs formed by initial clusters obtained from Algorithm 1.



(d) Connected induced subgraphs formed by suboptimal clusters in (a) obtained from Algorithm 2 with initial clusters as in (c).

Fig. 2: Illustration of the clustering algorithm

subgraph as shown in Fig.2(d). The cost minimization with respect to iterations is shown in Fig. 2(b).

The reduced system $\widehat{\Sigma}$ is obtained by aggregating the clusters using the projection matrix P in (1). The norm of the output of $\mathring{\Sigma}$, i.e., $\|\mathbf{y}(t)\|$; and the norm of the output of $\widehat{\Sigma}$ with initial clustering, i.e., $\|\mathbf{y}^0(t)\|$, and with suboptimal clustering, i.e., $\|\mathbf{y}^*(t)\|$, are shown in Fig. 3(a). Likewise, the norm of states are shown in Fig. 3(b).

Fig. 3(c) and (d) show the comparison between the errors for initial clustering and the suboptimal clustering. An interesting thing to note is that suboptimal clustering reduces the error $\|\mathbf{z}(t)-\hat{\mathbf{z}}(t)\|$ by around 95% and the error $\|\mathbf{y}(t)-\hat{\mathbf{y}}(t)\|$ by around 80%. This suggests that structure-based clustering algorithm, on the one hand, is well-suited for control and estimation of aggregated states; and, on the other hand, it yields a reduced system whose input-output behavior is very similar to the input-output behavior of the original network system.

VII. Conclusions

Clustered model reduction is an effective tool to reduce the complexity of control and estimation problem for large-scale network systems. We presented a structure-based clustering approach for model reduction, in contrast to the input-output behavior approaches already presented in the literature. That is, we translated the model reduction problem to a static graph reduction problem to attain computational feasibility. The proposed clustering algorithm is heuristic by nature and achieves a suboptimal clustering that minimizes the distance from lumpability – a notion that is ideal for model reduction. The algorithm is novel in a sense that it ensures the connectivity of clusters, which is a crucial requirement in several applications.

Structure-based clustering provides a computationally tractable way to study the problem of estimation and regulation of aggregated states of clusters in a large-scale network system. The tractability results from the fact that the observer and the controller can be designed based on the reduced system with tractable dimensions. Such a methodology is efficient because the states of the reduced model approximate the aggregated states of the original network system.

Acknowledgement

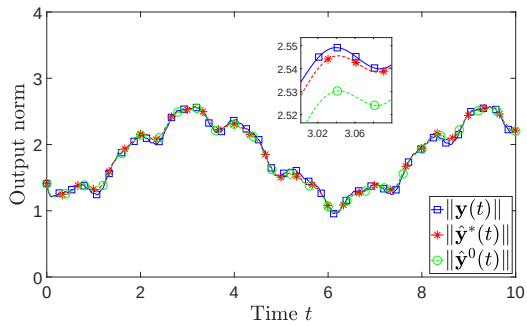
This work is supported by European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme, ERC-AdG no. 694209, Scale-FreeBack (Website: http://scale-freeback.eu/).

References

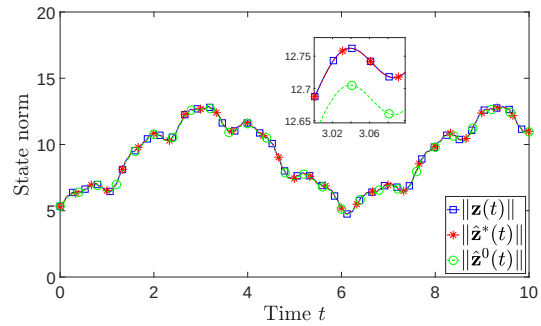
[1] A. C. Antoulas, *Approximation of large-scale dynamical systems*. Philadelphia, PA, USA: SIAM, 2005.

[2] S. Gugercin, A. C. Antoulas, and C. Beattie, "$\mathcal{H}_2$ model reduction for large-scale linear dynamical systems," *SIAM journal on matrix analysis and applications*, vol. 30, no. 2, pp. 609–638, 2008.

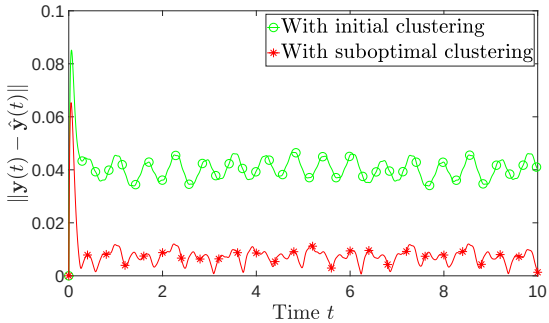
[3] A. C. Antoulas, C. A. Beattie, and S. Gugercin, "Interpolatory model reduction of large-scale dynamical systems," in *Efficient modeling and control of large-scale systems*. Springer, 2010, pp. 3–58.



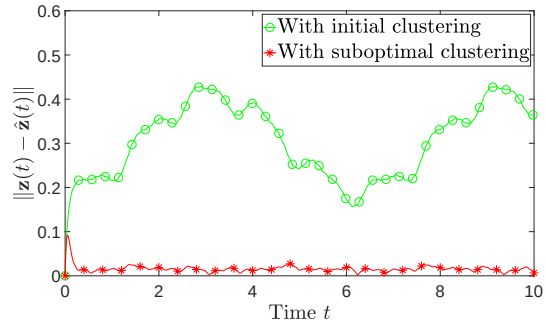
(a) Norm of the network system's output $\mathbf{y}(t)$ and the reduced system's output: $\hat{\mathbf{y}}^*(t)$ with suboptimal clustering and $\hat{\mathbf{y}}^0(t)$ with initial clustering.



(b) Norm of the projected system's state $\mathbf{z}(t)$ and the reduced system's state: $\hat{\mathbf{z}}^*(t)$ with suboptimal clustering and $\hat{\mathbf{z}}^0(t)$ with initial clustering.



(c) The norm of the output error with initial clustering and final (suboptimal) clustering.



(d) Model approximation error with initial clustering and final (suboptimal) clustering.

Fig. 3: Model approximation error minimization by the structure-based clustering algorithm.

[4] A. van der Schaft, S. Rao, and B. Jayawardhana, "On the mathematical structure of balanced chemical reaction networks governed by mass action kinetics," *SIAM Journal on Applied Mathematics*, vol. 73, no. 2, pp. 953–973, 2013.

[5] K. Deng, P. Barooah, P. G. Mehta, and S. P. Meyn, "Building thermal model reduction via aggregation of states," in *American Control Conference (ACC)*, 2010, pp. 5118–5123.

[6] X. Cheng and J. M. Scherpen, "Clustering approach to model order reduction of power networks with distributed controllers," *Advances in Computational Mathematics*, vol. 44, no. 6, pp. 1917–1939, 2018.

[7] N. Monshizadeh, H. L. Trentelman, and M. K. Camlibel, "Projection-based model reduction of multi-agent systems using graph partitions," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 2, pp. 145–154, 2014.

[8] T. Ishizaki, K. Kashima, J.-i. Imura, and K. Aihara, "Model reduction and clusterization of large-scale bidirectional networks." *IEEE Transactions on Automatic Control*, vol. 59, no. 1, pp. 48–63, 2014.

[9] T. Ishizaki, K. Kashima, A. Girard, J.-i. Imura, L. Chen, and K. Aihara, "Clustered model reduction of positive directed networks," *Automatica*, vol. 59, pp. 238–247, 2015.

[10] X. Cheng, Y. Kawano, and J. M. Scherpen, "Reduction of second-order network systems with structure preservation," *IEEE Transactions on Automatic Control*, vol. 62, no. 10, pp. 5026–5038, 2017.

[11] ——, "Model reduction of multiagent systems using dissimilarity-based clustering," *IEEE Transactions on Automatic Control*, vol. 64, no. 4, pp. 1663–1670, 2018.

[12] M. Aoki, "Control of large-scale dynamic systems by aggregation," *IEEE Transactions on Automatic Control*, vol. 13, no. 3, pp. 246–253, 1968.

[13] F. Dörfler and F. Bullo, "Kron reduction of graphs with applications to electrical networks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 60, no. 1, pp. 150–163, 2012.

[14] N. Martin, P. Frasca, and C. Canudas-De-Wit, "Large-scale network reduction towards scale-free structure," *IEEE Transactions on Network Science and Engineering*, 2018, doi: 10.1109/TNSE.2018.2871348.

[15] J. Wei and J. C. W. Kou, "A lumping analysis in monomolecular reaction systems," *Industrial & Engineering Chemistry Fundamentals*, vol. 8, pp. 114–123, 1969.

[16] P. G. Coxson, "Lumpability and observability of linear systems," *Journal of Mathematical Analysis and Applications*, vol. 99, no. 2, pp. 435–446, 1984.

[17] F. M. Atay and L. Roncoroni, "Lumpability of linear evolution equations in banach spaces," *Evolution Equations and Control Theory*, vol. 6, no. 1, pp. 15–34, 2017.

[18] T. Sadamoto, T. Ishizaki, and J.-i. Imura, "Average state observers for large-scale network systems," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 4, pp. 761–769, 2017.

[19] M. U. B. Niazi, C. Canudas-de-Wit, and A. Y. Kibangou, "Average observability of large-scale network systems," in *2019 18th European Control Conference (ECC)*, June 2019, pp. 1506–1511.

[20] M. U. B. Niazi, D. Deplano, C. Canudas-de-Wit, and A. Y. Kibangou, "Scale-free estimation of the average state in large-scale systems," *IEEE Control Systems Letters*, vol. 4, no. 1, pp. 211–216, Jan 2020.

[21] N. Martin, P. Frasca, T. Ishizaki, J.-i. Imura, and C. Canudas-de Wit, "The price of connectedness in graph partitioning problems," in *2019 18th European Control Conference (ECC)*, June 2019, pp. 2313–2318.

[22] S. L. Campbell and C. D. Meyer, *Generalized inverses of linear transformations*. SIAM, 2009.

[23] C. O. Aguilar and B. Gharesifard, "Almost equitable partitions and new necessary conditions for network controllability," *Automatica*, vol. 80, pp. 25–31, 2017.

[24] C. Godsil and G. Royle, *Algebraic Graph Theory*. New York: Springer-Verlag, 2001.

[25] M. Ji and M. Egerstedt, "Observability and estimation in distributed sensor networks," in *46th IEEE Conference on Decision and Control*, 2007, pp. 4221–4226.

[26] S. Burer and A. N. Letchford, "Non-convex mixed-integer nonlinear programming: A survey," *Surveys in Operations Research and Management Science*, vol. 17, no. 2, pp. 97–106, 2012.