Model Reduction Methods for Complex Network Systems

Cheng, X.; Scherpen, J. M. A.

Published in:
Annual review of control, robots, and Aauthonomous systems, VOL 4, 2021

DOI:
10.1146/annurev-control-061820-083817

ANNUAL REVIEWS

Annual Review of Control, Robotics, and Autonomous Systems

Model Reduction Methods for Complex Network Systems

X. Cheng[1] and J.M.A. Scherpen[2]

[1]Control Systems Group, Department of Electrical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands; email: x.cheng@tue.nl

[2]Jan C. Willems Center for Systems and Control, Engineering and Technology Institute Groningen (ENTEG), University of Groningen, 9747 AG Groningen, The Netherlands; email: j.m.a.scherpen@rug.nl

Annu. Rev. Control Robot. Auton. Syst. 2021. 4:425–53

First published as a Review in Advance on November 3, 2020

The *Annual Review of Control, Robotics, and Autonomous Systems* is online at control.annualreviews.org

https://doi.org/10.1146/annurev-control-061820-083817

Keywords

reduced-order modeling, network systems, interconnected systems, multiagent systems, graph clustering, synchronization, semistability, structure-preserving

Abstract

Network systems consist of subsystems and their interconnections and provide a powerful framework for the analysis, modeling, and control of complex systems. However, subsystems may have high-dimensional dynamics and a large number of complex interconnections, and it is therefore relevant to study reduction methods for network systems. Here, we provide an overview of reduction methods for both the topological (interconnection) structure of a network and the dynamics of the nodes while preserving structural properties of the network. We first review topological complexity reduction methods based on graph clustering and aggregation, producing a reduced-order network model. Next, we consider reduction of the nodal dynamics using extensions of classical methods while preserving the stability and synchronization properties. Finally, we present a structure-preserving generalized balancing method for simultaneously simplifying the topological structure and the order of the nodal dynamics.

1. INTRODUCTION

The backbone of many modern technological systems is a network system (a system of systems), which bonds diverse multiphysics components together. Many large-scale systems can be modeled as network systems that are composed of multiple subsystems interacting with each other via certain coupling protocols. Such systems are becoming increasingly prevalent in various domains; chemical reaction chains, cellular and metabolic networks, social networks, multirobot coordination, and large-scale power grids are only a few examples (1–8).

However, with the increasing complexity of network scales and subsystem dynamics, the models describing the behavior of network systems can be of extremely high dimension. This will lead to serious scalability issues in simulation, optimization, transient analysis, and control synthesis due to limited computational capability and storage capacity. These issues spur the development of methodologies on complexity reduction for large-scale network systems, aiming to acquire pertinent information on the systems in a computationally manageable fashion.

In the past few decades, a variety of theories and techniques for model reduction have been developed for generic dynamical systems, including Krylov subspace methods (also known as moment matching), balanced truncation, and Hankel norm approximation (see 9–14 and the references therein). These conventional methodologies can generate reduced-order models that well approximate the input–output mapping of a high-dimensional system. However, when addressing the model reduction problem of large-scale network system, we must rethink how to implement those methods in a structure-preserving manner, because the analysis, control, and monitoring of complex networks rely heavily on their interconnection structure (15–18). Actually, preserving essential network configurations in the approximation of network systems presents the most challenging problem. Early work on controller reduction can be viewed as a predecessor of structure-preserving reduction of interconnected systems, which takes into account the coupling structure between plants and controllers (19, 20). However, recent developments in large-scale networked systems have gone far beyond the simple closed-loop structure.

In this article, we provide an overview of recent advances in dimension reduction of complex network systems. The complexity we consider consists of two aspects—large-scale topology and high-dimensional subsystems (nodal dynamics)—which lead to two types of model reduction problems in the context of network systems.

The first of these problems is focused on how to simplify a complicated network structure by reducing the number of nodes. Inspired by the classification and pattern recognition in data science and computer graphics (21, 22), reduction methods based on clustering and aggregation are mainstream for reducing topological complexity. We cover most of the relevant work (23–31) in this article and briefly review other topological methods, including the well-known singular perturbation approximation.

The second problem considers how to reduce the dimension of individual subsystems in a network. The relevant approximation approaches for interconnected systems or coupled systems based on subsystem structuring have been of interest for a long time (32–34). More recent works by Monshizadeh et al. (35) and Cheng et al. (36) further discuss diffusively coupled linear/nonlinear systems, where the reduction is performed on each subsystem in a way that retains certain properties of the entire network, such as synchronization and stability. Furthermore, techniques developed by Ishizaki et al. (37) and Cheng et al. (38) combine the complexity reduction of network structures and subsystem dynamics while also providing an attractive way to simplify the complexity of entire network systems.

2. FROM GRAPHS TO NETWORK SYSTEMS

In this section, we cover some preliminaries regarding algebraic graph theory along with key concepts for modeling, analyzing, and designing network systems, and then introduce the graph-based modeling of network systems. Bullo (39) and Wu (40) provide further details on these subjects.

2.1. Algebraic Graph Theory

The language of graphs is essential in the modeling and control of networked systems and provides a natural tool for characterizing the interconnection structure of a network. Any finite graph $\mathcal{G}$ can be featured by a finite and nonempty node set $\mathcal{V} := \{1, 2, \ldots, n\}$ and an edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. Depending on whether the edges have specific orientations, we have two basic categories of graphs: directed and undirected.

2.1.1. Directed graphs. Directed graph (digraph) structures are found in various applications, including biochemical reactions and social networks (see, e.g., 5, 6), where the transmission of information or energy among network nodes is directional. This directionality can be encoded in the edges, which are ordered pairs of elements of $\mathcal{V}$, and we say that there is an edge directed from node i to node j if $(i, j) \in \mathcal{E}$. A digraph $\mathcal{G}$ is called simple if it does not contain self-loops [i.e., $\mathcal{E}$ does not contain edges of the form (i, i), $\forall i$] and there exists exactly one edge directed from i to j if $(i, j) \in \mathcal{E}$. In a simple digraph $\mathcal{G}$, a node i_n is reachable from another node i_0 if there is a directed path from i_0 to i_n. Here, this path is defined as a sequence of edges of the form (i_{k-1}, i_k), $k = 1, \ldots, n$, which joins a sequence of distinct nodes $i_0, i_1, \ldots, i_n$.

Next, we present the connectivity notions for a digraph $\mathcal{G}$: (*a*) $\mathcal{G}$ is strongly connected if any two nodes are reachable from each other, (*b*) $\mathcal{G}$ is quasi-strongly connected if all the nodes are reachable from a common node, and (*c*) $\mathcal{G}$ is weakly connected if its undirected version $\mathcal{G}_u := (\mathcal{V}, \mathcal{E}_u)$ is strongly connected, where the set $\mathcal{E}_u \supseteq \mathcal{E}$ includes both (i, j) and (j, i) if there is an edge $(i, j) \in \mathcal{E}$. Note that any simple digraph $\mathcal{G}$ can be decomposed into a unique set of maximal strongly connected components (SCCs), which are the largest strongly connected subgraphs of $\mathcal{G}$. If an SCC has outflows only to other SCCs, it is called a root SCC (RSCC). A weakly connected digraph may contain multiple RSCCs, while a quasi-strongly connected digraph has only one RSCC.

Three matrices are commonly used to characterize the topology of a digraph. The incidence matrix $\mathcal{B} \in \mathbb{R}^{n \times |\mathcal{E}|}$ of $\mathcal{G}$ is defined such that $[\mathcal{B}]_{ij} = 1$ if edge $(i, j) \in \mathcal{E}$, $[\mathcal{B}]_{ij} = -1$ if edge $(j, i) \in \mathcal{E}$, and $[\mathcal{B}]_{ij} = 0$ otherwise, where each column indicates a directed edge. When this edge is assigned a positive value (weight)—that is, $\mathcal{G}$ is weighted—we define a weighted adjacency matrix $\mathcal{A}$, where $[\mathcal{A}]_{ij}$ is equal to the weight of the edge (j, i) if $(j, i) \in \mathcal{E}$, and $[\mathcal{A}]_{ij} = 0$ otherwise. Moreover, the weighted out-degree and in-degree matrices of $\mathcal{G}$ are the diagonal matrices defined by $D_{out} := \mathrm{Diag}(\mathcal{A}\mathbb{1})$ and $D_{in} := \mathrm{Diag}(\mathbb{1}^\top \mathcal{A})$, respectively. A strongly connected digraph is called balanced if $D_{out} = D_{in}$. The Laplacian matrix of a digraph $\mathcal{G}$ is defined as $L := D_{out} - \mathcal{A}$, and the elements of L are given by

$$[L]_{ij} = \begin{cases} \sum_{j=1, j\neq i}^{n} \mathcal{A}_{ij}, & i = j, \\ -\mathcal{A}_{ij}, & \text{otherwise.} \end{cases} \quad 1.$$

Laplacian matrices are instrumental in modeling various diffusion processes (e.g., 4, 41, 42). A Laplacian matrix enjoys two fundamental properties: (*a*) $L\mathbb{1} = 0$ and (*b*) $[L]_{ii} \geq 0$, $\forall i \in \mathcal{V}$, and $[L]_{ij} \leq 0$, $\forall i \neq j$. Conversely, a real square matrix satisfying the two properties can also be interpreted as a Laplacian matrix that represents a weighted simple digraph. Note that Laplacian matrices are singular. If a weakly connected digraph has m RSCCs, then its Laplacian matrix also

has semisimple zero eigenvalues with multiplicity m, while all the other nonzero eigenvalues have positive real parts.

2.1.2. Undirected graphs. Undirected graphs are commonly used to characterize the interconnection structure of physical systems, such as power grids, RC circuits, and mass-damper systems (23, 28, 43, 44). An undirected graph can be viewed as a special digraph whose weighted adjacency matrix $\mathcal{A}$ (or Laplacian matrix L) is symmetric. In this case, we can define a Laplacian matrix using an alternative formula:

$$L = \mathcal{B}W\mathcal{B}^{\mathrm{T}}, \qquad 2.$$

where $\mathcal{B}$ is the incidence matrix obtained by assigning an arbitrary orientation to each edge of $\mathcal{G}$, and $W := \mathrm{Diag}(w_1, w_2, \ldots, w_{|\mathcal{E}|})$, with w_k the weight associated to the edge k for each $k = 1, 2, \ldots, |\mathcal{E}|$. If $\mathcal{G}$ is an undirected connected graph, then the Laplacian matrix L has the properties (*a*) $L^{\top} = L$ and $\ker L = \mathrm{span}\{\mathbb{1}\}$ and (*b*) $[L]_{ij} \leq 0$ if $i \neq j$ and $[L]_{ii} > 0$.

2.2. Modeling of Network Systems

In the field of network science, the evolution of network topology over time is often specified as the dynamics of networks (45). For control systems, by contrast, the dynamics over networks is of interest; here, nodes represent individual dynamical systems that are coupled through edges (46). We use the latter notion when referring to a network system. Network systems have a clear interconnection structure, either physically or virtually; examples include chemical reaction networks, power grids, and robotic networks. Additionally, network modeling is applicable to spatially discretized systems that are originally described by partial differential equations, such as simple beam models or fluid dynamical systems.

2.2.1. Networks of single integrators. The simplest network systems consider all the nodes to be single integrators—that is, $\dot{x}_i(t) = v_i(t)$, where $x_i(t), v_i(t) \in \mathbb{R}$ are the state and input of node i, respectively. A digraph $\mathcal{G}$ then captures the interconnection topology of n single integrators, where the coupling rule is

$$v_i(t) = -d_i x_i(t) + \sum_{j=1, j\neq i}^{n} [\mathcal{A}]_{ij} x_j(t). \qquad 3.$$

In Equation 3, $d_i \in \mathbb{R}$ represents the state feedback gain, and $\mathcal{A}$ is the weighted adjacency matrix of $\mathcal{G}$, whose entry $[\mathcal{A}]_{ij}$ indicates the strength of the coupling between nodes i and j. Taking into account the external control signals $u(t) \in \mathbb{R}^p$ and measurements $y(t) \in \mathbb{R}^q$ of the network, we derive a compact form for the network system as

$$\dot{x}(t) = \Gamma x(t) + Fu(t), \quad y(t) = Hx(t), \qquad 4.$$

where $\Gamma := \mathcal{A} - D$, with $D = \mathrm{Diag}(d_1, \ldots, d_n)$, and $F \in \mathbb{R}^{n\times p}$ and $H \in \mathbb{R}^{q\times n}$ are the input and output matrices, respectively.

Equation 4 is regarded as a rather general representation for single-integrator networks, whose stability depends on the values in D and $\mathcal{A}$. If $D \geq 0$, then Γ becomes a Metzler matrix, leading to the concept of monotone systems or positive systems (47–49). In particular, if we choose $D > D_{\mathrm{out}}$ or $D > D_{\mathrm{in}}$, then the Metzler matrix Γ is strictly row or column diagonally dominant, respectively. Following the Gershgorin circle theorem (50), Γ is a Hurwitz matrix, leading to the asymptotic stability of the network system. If $D = D_{\mathrm{in}}$—or, equivalently, $\Gamma = -L^{\top}$, with L the Laplacian matrix

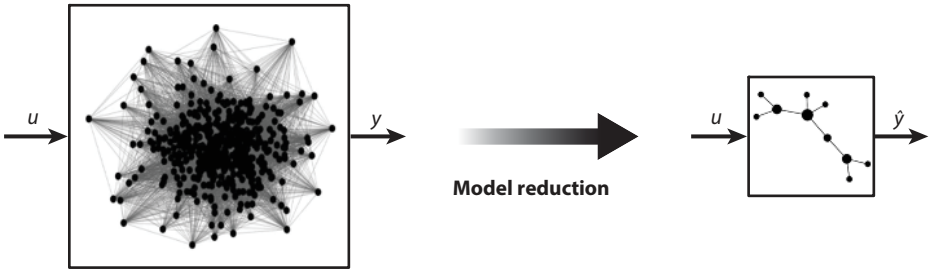

Figure 1

Model reduction with preservation of network structure.

of $\mathcal{G}$—then we have a network flow model (5, 42). Furthermore, if $D = D_{\text{out}}$—that is , $\Gamma = -L$—then Equation 4 becomes a consensus network, or a continuous-time averaging system (39). The coupling rule in Equation 3 becomes

$$v_i(t) = -\sum_{j=1, j\neq i}^{n} [\mathcal{A}]_{ij}\big[x_i(t) - x_j(t)\big], \qquad 5.$$

which is known as the diffusive coupling rule. For both $\Gamma = -L^\top$ and $\Gamma = -L$, the system in Equation 4 is semistable (or semiconvergent)—that is, $\lim_{t\to\infty} e^{\Gamma t}$ exists for any initial condition $x(0)$. Specifically, when $\mathcal{G}$ is strongly connected, $\lim_{t\to\infty} e^{-Lt}$ is equal to $\mathbb{1}\omega^\top$, with ω, satisfying $\mathbb{1}^\top\omega = 1$, the left eigenvector of L for eigenvalue 0.

Definition 1. A network system $\dot{x}(t) = \Gamma x(t)$ achieves synchronization if

$$\lim_{t\to\infty}\big[x_i(t) - x_j(t)\big] = 0,\ \forall i, j \in \mathcal{V}, \qquad 6.$$

holds for all initial conditions $x(0)$.

The approximation of the network system represented by Equation 4 aims for a reduced network consisting of a smaller number of nodes that captures essential properties of the original network. Specifically, a model reduction problem (**Figure 1**) is formulated to find a reduced-order model

$$\dot{\hat{x}}(t) = \hat{\Gamma}\hat{x}(t) + \hat{F}u(t), \quad \hat{y}(t) = \hat{H}\hat{x}(t), \qquad 7.$$

where $\hat{x} \in \mathbb{R}^r$ $(r < n)$, $\hat{y} \in \mathbb{R}^q$, such that (*a*) $\hat{\Gamma} \in \mathbb{R}^{r\times r}$ is interpretable as a reduced graph and (*b*) the approximation error is minimized between the original and the reduced-order models. The approximation error is usually evaluated by the $\mathcal{H}_\infty$ or $\mathcal{H}_2$ norms of $\eta(s) - \hat{\eta}(s)$,

$$\eta(s) := H(sI_n - \Gamma)^{-1}F, \quad \hat{\eta}(s) := \hat{H}(sI_r - \hat{\Gamma})^{-1}\hat{F}. \qquad 8.$$

2.2.2. Networked linear systems. The network model in Equation 4 can be extended beyond single integrators to consider each node as a high-order linear subsystem:

$$\dot{x}_i(t) = A_i x_i(t) + B_i v_i(t), \quad y_i(t) = C_i x_i(t), \qquad 9.$$

where $x_i \in \mathbb{R}^{\ell_i}$, $v_i \in \mathbb{R}^{m_i}$, and $y_i \in \mathbb{R}^{\mu_i}$ are internal states, inputs, and outputs, respectively. Suppose that n subsystems are interconnected through the relations $v_i(t) = \sum_{j=1}^n K_{ij}y_j(t) + F_i u(t)$, $y(t) = \sum_{i=1}^n H_i y_i(t)$, with $K_{ij} \in \mathbb{R}^{m_i\times\mu_j}$ the coupling coefficient between nodes i and j, where $K_{ij} = 0$ if and only if there are no signals passing from j to i. The vectors $u(t)$ and $y(t)$ are denoted as

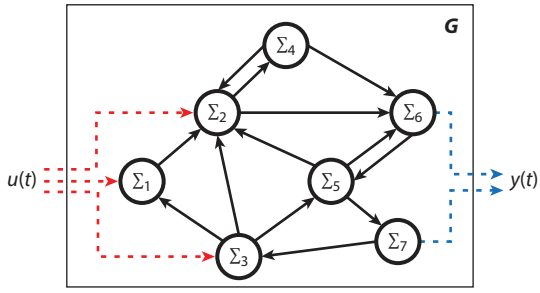

Figure 2

An example of networked linear systems.

external inputs and outputs. Combining this with Equation 9, we obtain a compact representation of the overall network system (32, 34):

$$\dot{x}(t) = (A_{\rm n} + B_{\rm n}\Gamma C_{\rm n})\,x(t) + B_{\rm n}Fu(t), \quad y(t) = HC_{\rm n}x(t), \tag{10.}$$

where $A_{\rm n} := \text{blkdiag}(A_1, \ldots, A_n)$, $B_{\rm n} := \text{blkdiag}(B_1, \ldots, B_n)$, $C_{\rm n} := \text{blkdiag}(C_1, \ldots, C_n)$, and

$$\Gamma = \begin{bmatrix} K_{11} & \cdots & K_{1n} \\ \vdots & \ddots & \vdots \\ K_{n1} & \cdots & K_{nn} \end{bmatrix}, \quad F = \begin{bmatrix} F_1 \\ \vdots \\ F_n \end{bmatrix}, \quad H = \begin{bmatrix} H_1 & \cdots & H_n \end{bmatrix}.$$

An example of a networked linear system containing six subsystems is shown in **Figure 2**. Networked linear systems of the form shown in Equation 10 are also known as interconnected or coupled systems (32, 34). The subsystems in Equation 9 could have different dynamics, in which case the network is called heterogeneous.

Homogeneous networks are defined when each node has the same dynamics:

$$\dot{x}_i(t) = Ax_i(t) + Bv_i(t), \quad y_i(t) = Cx_i(t), \tag{11.}$$

where $x_i \in \mathbb{R}^\ell$, $v_i \in \mathbb{R}^m$, and $y_i \in \mathbb{R}^m$ are the internal state, input, and output of node i, respectively. Under a simple static output feedback interconnection similar to Equation 3, the dynamics of networked homogeneous linear systems is presented in compact form as

$$\boldsymbol{\Sigma}: \begin{cases} \dot{x}(t) = (I \otimes A + \Gamma \otimes BC)\,x(t) + (F \otimes B)u(t), \\ y(t) = (H \otimes C)\,x(t), \end{cases} \tag{12.}$$

with joint state vector $x(t) \in \mathbb{R}^{n\ell}$, external control inputs $u(t) \in \mathbb{R}^{pm}$, and external outputs $y \in \mathbb{R}^{qm}$. The matrix $\Gamma \in \mathbb{R}^{n\times n}$ indicates how the subsystems are interconnected.

One commonly studied type of network system in the form of Equation 12 is diffusively coupled linear systems, where $\Gamma = -L$ is the Laplacian matrix of the underlying graph. Thus, the coupling rule among the nodes becomes $v_i(t) = -\sum_{j=1, j\neq i}^{n} [\mathcal{A}]_{ij}\,[y_i(t) - y_j(t)]$. In this setting, the synchronization problem of the system $\boldsymbol{\Sigma}$ has been intensively studied (see, e.g., 3, 39, 40, 51, 52).

Theorem 1. Consider a network of diffusively coupled homogeneous linear systems described by Equation 12 with the symmetric Laplacian L. Let $0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_n$ denote the eigenvalues of L. Then, the network system represented by Equation 12 achieves synchronization if and only if $A - \lambda_i BC$ is a Hurwitz matrix for all $i \in \{2, 3, \ldots, n\}$.

A sufficient condition for synchronization is provided by assuming that the subsystem (A, B, C) is passive—that is, there exists a symmetric positive definite matrix $K > 0$ that verifies

$$A^\top K + KA \leq 0, \quad \text{and} \quad C^\top = BK. \tag{13.}$$

Passivity is a natural property of physical systems, including mechanical systems, electrical networks, and thermodynamical systems (53). With passivity, we obtain a synchronization condition that is independent from the spectrum of the graph Laplacian (51, 54, 55).

Theorem 2. Let $\Gamma = -L$ represent any connected undirected graph or any strongly connected digraph. If the subsystem (A, B, C) in Equation 11 is passive and observable, then the network system represented by Equation 12 achieves synchronization.

The model complexity of networked linear systems comes from two aspects: the dimension of the subsystems and the topological scale of the network. The first reduction problem is thus to reduce each subsystem (or a subset of them) by taking into account the coupling structure in order to approximate the entire network system. For the reduction of the heterogeneous network system represented by Equation 10, the objective is to construct a network model composed of reduced-order subsystems $(\hat{A}_i, \hat{B}_i, \hat{C}_i)$, yielding an approximation of the entire system with the same form as Equation 10, where $\hat{A}_{\rm n} := {\rm blkdiag}(\hat{A}_1, \ldots, \hat{A}_n)$, $\hat{B}_{\rm n} := {\rm blkdiag}(\hat{B}_1, \ldots, \hat{B}_n)$, and $\hat{C}_{\rm n} := {\rm blkdiag}(\hat{C}_1, \ldots, \hat{C}_n)$. The matrices Γ, H, and F remain the same as the original ones. Homogeneous network systems can be reduced in a similar manner such that each original subsystem (A, B, C) is replaced by a lower-order approximation $(\hat{A}, \hat{B}, \hat{C})$:

$$\begin{cases} \dot{\hat{x}}(t) = \left(I_n \otimes \hat{A} - \Gamma \otimes \hat{B}\hat{C}\right)\hat{x}(t) + (F \otimes \hat{B})u(t), \\ \hat{y}(t) = (H \otimes \hat{C})\hat{x}(t). \end{cases} \quad 14.$$

The second reduction problem is focused on a simplification of the graph structure, as illustrated in **Figure 1**. A resulting reduced-order model for networked homogeneous linear systems is in the form

$$\begin{cases} \dot{z}(t) = \left(I_{\rm r} \otimes A - \hat{\Gamma} \otimes BC\right) z(t) + (\hat{F} \otimes B)u(t), \\ \hat{y}(t) = (\hat{H} \otimes C)z(t), \end{cases} \quad 15.$$

where $\hat{\Gamma} \in \mathbb{R}^{r\times r}$ represents a reduced graph consisting of r nodes. Let $G(s)$ and $\hat{G}(s)$ be the transfer matrices of the models represented by Equations 12 and 15, respectively. The objective now is to minimize the reduction error $G(s) - \hat{G}(s)$ with respect to certain norms.

3. REDUCTION OF TOPOLOGICAL STRUCTURES

One powerful paradigm for simplifying a large-scale network is graph clustering—a process that divides a set of nodes into nonempty and disjoint subsets, where nodes in each subset are considered related by some similarity measure. Different names are used in different fields, including community detection in the context of social networks and classification in the context of data science. Graph detection is closely related to unsupervised learning in pattern recognition systems (21, 22). Generally, well-established clustering algorithms (such as hierarchical clustering, spectral clustering, and K-means clustering) were developed for static graphs or measured data. In this section, we present a series of clustering-based model reduction techniques for dynamic networks.

3.1. Clustering-Based Projection

Consider a linear time-invariant system with triplet (A, B, C). The Petrov–Galerkin framework (9) projects the state space onto a lower-dimensional subspace, resulting in a reduced-order model $(V^\dagger AV, V^\dagger B, CV)$, where V is full-column rank representing the basis of the subspace, and $V^\dagger$ is

a left inverse of V (i.e., $V^\dagger V = I$). Clearly, the choice of V is essential for obtaining the reduced-order model. For structure-preserving model reduction of network systems, V can be constructed by considering an aggregation of node states.

Definition 2. Consider a graph $\mathcal{G}$ with node set $|\mathcal{V}| = n$. Graph clustering of $\mathcal{G}$ is a process that divides $\mathcal{V}$ into r nonempty and disjoint subsets, denoted by $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_r$, where $\mathcal{C}_i$ is called a cluster (or a cell of $\mathcal{G}$). The characteristic matrix of the clustering $\{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_r\}$ is a binary matrix $\Pi \in \mathbb{R}^{n\times r}$ with

$$[\Pi]_{ij} := \begin{cases} 1, & \text{if node } i \in \mathcal{C}_j, \\ 0, & \text{otherwise.} \end{cases} \qquad 16.$$

Note that each row of Π has exactly one nonzero element, indicating that each node is assigned to a unique cluster. The number of nonzero elements in each column is the cardinality of the corresponding cluster. Specifically, $\Pi \mathbb{1}_r$ is equal to $\mathbb{1}_n$, and $\mathbb{1}_n^\top \Pi$ is equal to $[|\mathcal{C}_1|, |\mathcal{C}_2|, \ldots, |\mathcal{C}_r|]$. For any given undirected graph Laplacian L, the matrix $\Pi^\top L \Pi$ is a Laplacian matrix representing an undirected graph of smaller size. This important property allows for structure-preserving model reduction of network systems using Π for the Petrov–Galerkin projection. To construct a reduced-order network system with r nodes, as in Equation 4 or Equation 12, we must first find a clustering that partitions the nodes of a network into r clusters.

Consider the network system represented by Equation 4, which is assumed to be semistable (see the sidebar titled Linear Semistable Systems and Pseudo-Gramians). The projection matrix is then defined as $VV^\dagger \in \mathbb{R}^{n\times n}$,

$$V = N\Pi \in \mathbb{R}^{n\times r}, \quad V^\dagger := (\Pi^\top M N \Pi)^{-1} \Pi^\top M \in \mathbb{R}^{r\times n}, \qquad 17.$$

where M and N are nonsingular diagonal weighting matrices (58). A reduced-order model is thereby obtained in the form of Equation 7, where $\hat{\Gamma} = V^\dagger \Gamma V$, $\hat{F} = V^\dagger F$, and $\hat{H} = HV$.

LINEAR SEMISTABLE SYSTEMS AND PSEUDO-GRAMIANS

Semistability is a more general concept than asymptotic stability, as it allows for multiple poles that are zero. The systems' trajectories thus may converge to a nonzero Lyapunov stable equilibrium (56, 57). Specifically, a linear system $\dot{x}(t) = Ax(t)$ is semistable if $\lim_{t\to\infty} e^{At}$ is nonzero and exists for all initial states $x(0)$, or, equivalently, the zero eigenvalues of A are semisimple and all the other eigenvalues have negative real parts.

It is well known that the standard controllability and observability Gramians (9) are not well defined for a semistable system, and Cheng & Scherpen (58) therefore presented the definition of pseudo-Gramians. Consider a linear semistable system (A, B, C). The pseudo-controllability and observability Gramians are defined as

$$\mathcal{P} = \int_0^\infty (e^{At} - \mathcal{J})BB^\top (e^{A^\top t} - \mathcal{J}^\top)\mathrm{d}t \quad \text{and} \quad \mathcal{Q} = \int_0^\infty (e^{A^\top t} - \mathcal{J}^\top)C^\top C(e^{At} - \mathcal{J})\mathrm{d}t, \qquad 18.$$

respectively, where $\mathcal{J} := \lim_{t\to\infty} e^{At}$ is a constant matrix. The pseudo-Gramians $\mathcal{P}$ and $\mathcal{Q}$ in Equation 18 are well defined for semistable systems. The pseudo-Gramians can be computed as $\mathcal{P} = \tilde{\mathcal{P}} - \mathcal{J}\tilde{\mathcal{P}}\mathcal{J}^\top$ and $\mathcal{Q} = \tilde{\mathcal{Q}} - \mathcal{J}^\top\tilde{\mathcal{Q}}\mathcal{J}$, where $\tilde{\mathcal{P}}$ and $\tilde{\mathcal{Q}}$ are arbitrary symmetric solutions of the Lyapunov equations

$$A\tilde{\mathcal{P}} + \tilde{\mathcal{P}}A^\top + (I - \mathcal{J})BB^\top(I - \mathcal{J}^\top) = 0 \quad \text{and} \quad A^\top\tilde{\mathcal{Q}} + \tilde{\mathcal{Q}}A + (I - \mathcal{J}^\top)C^\top C(I - \mathcal{J}) = 0, \qquad 19.$$

respectively. The pseudo-Gramians are useful for computing the $\mathcal{H}_2$ norm of a semistable system. The transfer matrix $G(s)$ is an element in $\mathcal{H}_2$ if and only if $C\mathcal{J}B = 0$. Furthermore, $\|G(s)\|_{\mathcal{H}_2}^2$ is equal to $\mathrm{Tr}(C\mathcal{P}C^\top)$ and $\mathrm{Tr}(B^\top\mathcal{Q}B)$.

Theorem 3 (from Reference 58). Let $\eta(s)$ and $\hat{\eta}(s)$ be the transfer matrices of the original and reduced-order models in Equations 4 and 7. A bounded $\mathcal{H}_2$ reduction error is then guaranteed [i.e., $\eta(s) - \hat{\eta}(s) \in \mathcal{H}_2$] if there exist diagonal and positive definite matrices M and N such that

$$(\mathbf{e}_i - \mathbf{e}_j)^\top N^{-1} \mathcal{J} = 0 \text{ and } \mathcal{J} M^{-1} (\mathbf{e}_i - \mathbf{e}_j) = 0 \qquad 20.$$

hold for each pair $i, j \in \mathcal{C}_k$, with any $k \in \{1, 2, \ldots, r\}$.

If Γ is a Hurwitz matrix, or $\Gamma = -L$, with L the Laplacian of a connected undirected graph, we simply choose $M = N = I_n$, which always guarantees $\eta(s) - \hat{\eta}(s) \in \mathcal{H}_2$. A Hurwitz Γ implies $\mathcal{J} = 0$, while a connected undirected graph yields $\mathcal{J} = \frac{1}{n} \mathbb{1}_n \mathbb{1}_n^\top$. Ishizaki et al. (49), Cheng & Scherpen (59), and Cheng et al. (60) treated dynamic networks with a strongly connected topology—that is, those where Γ is irreducible and has only one zero eigenvalue, with corresponding left and right eigenvectors μ_l and μ_r the so-called Frobenius eigenvectors with all real and positive entries (61). In this case, $M = \mathrm{Diag}(\mu_\mathrm{l})$ and $N = \mathrm{Diag}(\mu_\mathrm{r})$ satisfy Equation 20 for any clustering. Furthermore, we have $\eta(s) - \hat{\eta}(s) \in \mathcal{H}_2$ and $N\Gamma^\top M + M\Gamma N \leq 0$. Following Cheng & Scherpen (58), we can obtain an a posteriori bound on the reduction error as $\|\eta(s) - \hat{\eta}(s)\|_{\mathcal{H}_2} \leq \gamma_s \sqrt{\mathrm{Tr}(I - VV^\dagger)\mathcal{P}(I - VV^\dagger)^\top}$, where $\mathcal{P}$ is the pseudo-controllability Gramian of the system represented by Equation 4, and $\gamma_s \in \mathbb{R}_{>0}$ satisfies

$$\begin{bmatrix} N\Gamma^\top M + M\Gamma N & M\Gamma & (I - \mathcal{J}^\top)H^\top \\ \Gamma^\top M & -\gamma_s I & H^\top \\ H(I - \mathcal{J}) & H & -\gamma_s I \end{bmatrix} \leq 0. \qquad 21.$$

There is a balanced graph representation of the digraph system represented by Equation 4 as follows:

$$MN\dot{\xi}(t) = L_\mathrm{b}\xi(t) + MFu(t), \quad y(t) = HN\xi(t), \qquad 22.$$

where $L_\mathrm{b} := M\Gamma N$ is the Laplacian matrix of the balanced digraph, and the resulting reduced-order model in Equation 7 becomes

$$\Pi^\top MN\Pi\dot{\hat{\xi}}(t) = \Pi^\top L_\mathrm{b}\Pi\hat{\xi}(t) + \Pi^\top MFu(t), \quad \hat{y}(t) = HN\Pi\hat{\xi}(t),$$

with $\Pi^\top L_\mathrm{b}\Pi$ representing a reduced balanced digraph. Cheng & Scherpen (62) defined a generalized balanced digraph as a weakly connected digraph in which each RSCC is balanced and all non-RSCC nodes are removed, resulting in a generalized balanced graph representation similar to Equation 22. For networks with a weakly connected topology, the error system $\eta(s) - \hat{\eta}(s)$ is generally not in the $\mathcal{H}_2$ space. Clusterability is then defined between two nodes i, j if they satisfy Equation 20. The clusterability of all nodes in each cluster then guarantees the stability of the error $\eta(s) - \hat{\eta}(s)$ (62).

Example 1. Consider the mass-damper system in **Figure 3*a***, where the masses are interconnected via linear dampers. $u_1(t)$ and $u_2(t)$ represent external forces, and $y_1(t)$ and $y_2(t)$ are measured velocities. Suppose that all the masses are identical. The network system in the form of Equation 4 is obtained as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \end{bmatrix} = \underbrace{-\begin{bmatrix} 6 & -3 & 0 & -2 & -1 \\ -3 & 4 & -1 & 0 & 0 \\ 0 & -1 & 6 & -2 & -3 \\ -2 & 0 & -2 & 5 & -1 \\ -1 & 0 & -3 & -1 & 5 \end{bmatrix}}_{\Gamma} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} + \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}}_{F} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}}_{H} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix},$$

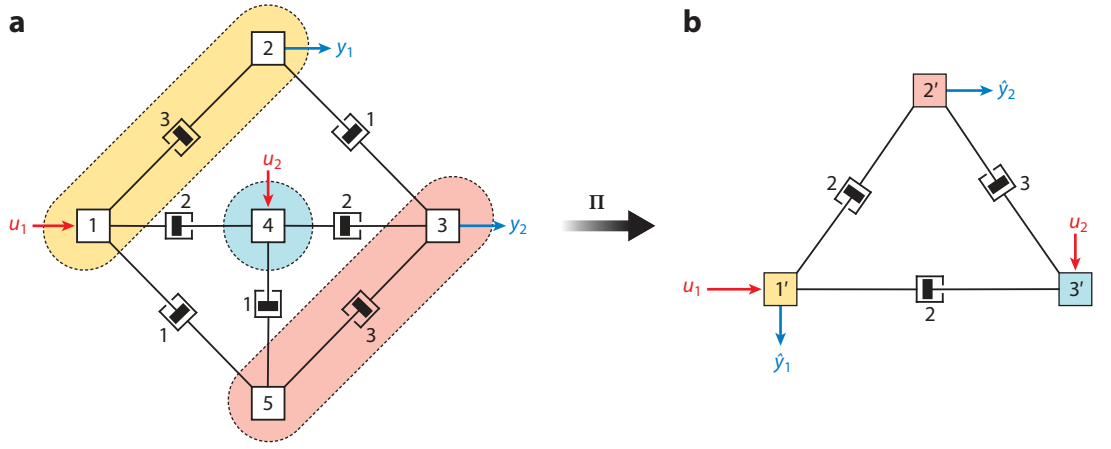

Figure 3

An illustrative example of clustering-based model reduction of a mass-damper network system. (*a*) The original network, divided into three clusters. (*b*) The network representation of the reduced-order model.

where $-\Gamma$ is an undirected graph Laplacian, and the off-diagonal entry $[\Gamma]_{ij}$ represents the damping coefficient of the edge (i, j). Consider $\{\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3\} = \{\{1, 2\}, \{3, 5\}, \{4\}\}$ to be the clustering of the graph, which leads to the following characteristic matrix:

$$\Pi = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}^\top.$$

Therefore, a reduced-order network model is obtained as

$$\underbrace{\begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\Pi^\top \Pi} \begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = -\underbrace{\begin{bmatrix} 4 & -2 & -2 \\ -2 & 5 & -3 \\ -2 & -3 & 5 \end{bmatrix}}_{\Pi^\top \Gamma \Pi} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} + \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}}_{\Pi^\top F} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{H\Pi} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix},$$

where $-\Pi^\top \Gamma \Pi$ is again an undirected graph Laplacian. To bring it into the form of Equation 7, we can define $\hat{\Gamma} := (\Pi^\top \Pi)^{-1} \Pi^\top \Gamma \Pi$, $\hat{F} := (\Pi^\top \Pi)^{-1} \Pi^\top F$, and $\hat{H} := H\Pi$. However, from $-\Pi^\top \Gamma \Pi$, it follows that this model allows for a physical interpretation, as shown in **Figure 3*b***: The nodes in each cluster are aggregated into a single node in the reduced network, while all edges connecting nodes from two distinct clusters are merged to a single edge that links the corresponding nodes in the reduced network.

Analogously, and beyond the single-integrator case, a reduced-order model of networked homogeneous linear systems in Equation 12 can be formed using the Petrov–Galerkin projection framework of Equation 17, which gives a reduced-order model in the form of Equation 15, where $\hat{\Gamma} := \Pi^\top L \Pi$, $\hat{F} = \Pi^\top F$, and $\hat{H} = H\Pi$ are the same as in Equation 7. The new state vector $z^\top(t) := \left[z_1^\top(t)\, z_2^\top(t)\, \ldots\, z_r^\top(t)\right] \in \mathbb{R}^{r\ell}$, $z_i(t) \in \mathbb{R}^\ell$, $i = 1, \ldots r$, represents an estimate of the state vector of the dynamics of all the nodes in the ith cluster. Note that the extension of clustering-based approaches toward networks of heterogeneous subsystems in Equation 10 remains an open problem. A major challenge lies in the representation of a cluster of nonidentical subsystems.

Denote $G(s)$ and $\hat{G}(s)$ as the transfer matrices of the models represented by Equations 12 and 15, respectively. The analysis of the reduction error $G(s) - \hat{G}(s)$ is more complicated here than it

is in the single-integrator case, and for general subsystems, the reduction error $G(s) - \hat{G}(s)$ may not be stable. However, there is a theoretical guarantee if the subsystem (A, B, C) in Equation 11 is observable and passive. With Theorem 2, we can verify that if the original network is undirected or strongly connected, the reduced-order network system in Equation 15 achieves synchronization, and $G(s) - \hat{G}(s) \in \mathcal{H}_2$, for any clustering Π (27, 29).

In the framework of clustering-based projection, the approximation error $\|G(s) - \hat{G}(s)\|_{\mathcal{H}_2}$ depends only on the choice of graph clustering. Thus, the most crucial problem in this framework is how to determine clusters of nodes to minimize the approximation error. In the following, we review several specific cluster selection approaches.

3.1.1. Almost equitable partitions. Almost equitable partitions provide a graph clustering where nodes in the same cluster are connected to other clusters in a similar fashion.

Definition 3. Consider a weighted undirected graph $\mathcal{G}$ with adjacency matrix $\mathcal{A}$. A clustering $\{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_r\}$ is called an almost equitable partition if for any indexes $\mu, \nu \in \{1, 2, \ldots, r\}$ with $\mu \neq \nu$, it holds that $\sum_{k \in \mathcal{C}_\nu} w_{ik} = \sum_{k \in \mathcal{C}_\nu} w_{jk}, \forall i, j \in \mathcal{C}_\mu$, where $w_{ij} := [\mathcal{A}]_{ij}$.

Figure 4 shows an example of the almost equitable partition of an undirected graph (25). The nodes in a cluster have the same total edge weight for their connection to another cluster. An almost equitable partition of an undirected graph has the key property that $\mathrm{Im}(\Pi)$ is L invariant—that is, $L\,\mathrm{Im}(\Pi) \subseteq \mathrm{Im}(\Pi)$, where $L = L^\top$ is the Laplacian of an undirected graph (see, e.g., 25, 63). Furthermore, we have $L\Pi = \Pi\hat{L}$ with $\hat{L} := (\Pi^\top\Pi)^{-1}\Pi^\top L\Pi$. Aguilar & Gharesifard (64) considered a generalization of almost equitable partitions to digraphs, where nodes in the same cluster should have identical weighted out-degrees. $\mathrm{Im}(\Pi)$ is then still L invariant. Now consider the following error system:

$$\Delta(s) = \eta(s) - \hat{\eta}(s) = \begin{bmatrix} H & -H\Pi \end{bmatrix} \begin{bmatrix} sI_n + L & 0 \\ 0 & sI_r + \Pi^\dagger L\Pi \end{bmatrix}^{-1} \begin{bmatrix} F \\ \Pi^\dagger F \end{bmatrix}, \qquad 23.$$

where $\Pi^\dagger = (\Pi^\top\Pi)^{-1}\Pi^\top$ and $\eta(s)$, $\hat{\eta}(s)$ are the transfer matrices of Equation 8. From the L invariance, it is verified that $\hat{\eta}(-s)^\top\Delta(s) = 0$, and thus $\|\Delta(s)\|_{\mathcal{H}_2}$ is equal to $\|\eta(s)\|_{\mathcal{H}_2} - \|\hat{\eta}(s)\|_{\mathcal{H}_2}$.

Furthermore, for a special output of the system represented by Equation 4, Monshizadeh et al. (25) and Jongsma et al. (30) provided explicit expressions for the reduction error $\|\Delta(s)\|_{\mathcal{H}_2}$ and

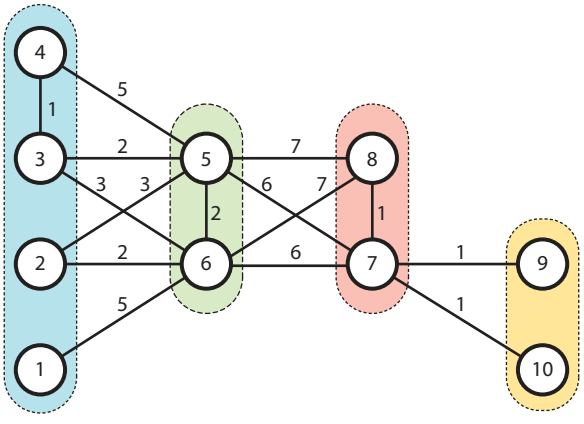

Figure 4

Example of the almost equitable partition of a 10-node graph.

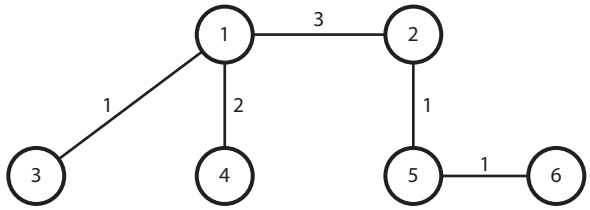

Figure 5

A tree with six nodes.

$\|\Delta(s)\|_{\mathcal{H}_\infty}$; Jongsma et al. (30) also further discussed model reduction of networked symmetric linear systems based on almost equitable partitions. Although an almost equitable partition as a particular clustering offers us analytical expression for the reduction error, it does not necessary lead to a small error. In fact, the methods described by Mlinarić et al. (65) and Cheng et al. (66) provide significantly lower errors via alternative choices of clustering for some examples. Moreover, how to find all almost equitable partitions for a large-scale graph is generally a rather difficult and computationally expensive problem (25).

3.1.2. Tree networks. Next, we focus on a particular class of undirected networks with a tree topology. In graph theory, a tree is a connected undirected graph in which there is only one path between any two nodes. An example of an undirected tree is shown in **Figure 5**.

Clearly, a tree $\mathcal{T}$ with n nodes has exactly $n-1$ edges. Let $\mathcal{B} \in \mathbb{R}^{n\times(n-1)}$ be the incidence matrix of $\mathcal{T}$. Relevant to the expression of the graph Laplacian L in Equation 2, we define an edge Laplacian as $L_e = \mathcal{B}^\top \mathcal{B} W \in \mathbb{R}^{(n-1)\times(n-1)}$, where W is the diagonal edge weight matrix. Observe that L_e is full rank and has all eigenvalues real and positive. The eigenvalues of L_e coincide with the nonzero eigenvalues of L, the Laplacian matrix of $\mathcal{T}$.

Consider the Laplacian dynamics in Equation 4, where $\Gamma = -L$ is an undirected graph Laplacian in Equation 2. Applying the transformation $x_e = \mathcal{B}^\top x$ then leads to the so-called edge agreement protocol (1, 67): $\dot{x}_e(t) = -L_e x_e(t) + \mathcal{B}^\top F u(t)$, which is asymptotically stable and minimal. Network reduction approaches can be developed based on edge operations. For example, Leiter & Zelazo (68) provided a greedy algorithm for edge-based contraction to simplify the graph topology. A more general form of the edge agreement protocol is derived when subsystems are taken into account. Besselink et al. (27) defined the edge system of a network system with $\Gamma = -L$ representing an undirected tree graph as

$$\boldsymbol{\Sigma}_e : \begin{cases} \dot{x}_e(t) = (I_{n-1} \otimes A - L_e \otimes BC)x_e(t) + (\mathcal{B}^\top F \otimes B)u(t), \\ y_e(t) = (H\mathcal{B}W \otimes C)x_e(t), \end{cases} \qquad 24.$$

where $x_e = (\mathcal{B}^\top \otimes I)x \in \mathbb{R}^{(n-1)\ell}$. Assuming that the subsystem (A, B, C) in Equation 11 is passive and minimal, we have the synchronization property of the network system $\boldsymbol{\Sigma}$ from Theorem 2. It then follows that the edge system $\boldsymbol{\Sigma}_e$ is asymptotically stable. Furthermore, we can define a pair of generalized controllability and observability Gramians of the edge system $\boldsymbol{\Sigma}_e$ as follows (27): $P_e := X \otimes K^{-1}$, $\quad Q_e := Y \otimes K$, where $K > 0$ satisfies Equation 13 for the passive subsystem and $X > 0$ and $Y > 0$ are solutions of the inequalities

$$-L_e X - X L_e^\top + \mathcal{B}^\top F F^\top \mathcal{B} \le 0, \quad -L_e^\top Y - Y L_e + W\mathcal{B}^\top H^\top H \mathcal{B} W \le 0.$$

The matrices X and Y admit a diagonal structure: $X = \mathrm{Diag}(\xi_1, \xi_2, \ldots, \xi_{n-1})$, $Y = \mathrm{Diag}(\eta_1, \eta_2, \ldots, \eta_{n-1})$, where the ordering $\xi_i \eta_i \ge \xi_{i+1}\eta_{i+1}$ is imposed. The value of $\xi_i \eta_i$ can be roughly viewed as an indication of the importance of the ith edge, since, similar to

balanced realization theory, ξ_i and η_i are related to controllability and observability properties of the edges. Reduction by truncation methods is then equivalent to aggregating nodes connected by the truncated edges; moreover, Besselink et al. (27) provided an a priori upper bound for the approximation error in terms of the $\mathcal{H}_\infty$ norm $\|G(s) - \hat{G}(s)\|_{\mathcal{H}_\infty} \leq 2(\sum_{i=r}^{n-1} [L_e^{-1}]_{ii} \sqrt{\xi_i \eta_i})$, where r is the number of nodes in the reduced network.

For an extension beyond tree graphs, a major challenge lies in the characterization of the edge system (27).

3.1.3. Dissimilarity-based clustering. For generic network systems, we may resort to a dissimilarity-based clustering approach presented in, e.g., References 28, 29, 62, and 69. In line with data classification or pattern recognition in the other domains, dissimilarity-based clustering for dynamic networks starts with a proper metric that quantifies the difference between any pair of nodes (subsystems) in a network. For static graphs, dissimilarity (or distance) between two nodes or data points can be computed in a vector space using some sort of metric (21, 22). Considering dynamic networks with external inputs, the dissimilarity metric is then featured in a function space (28, 29, 62).

Definition 4. Consider the network system of Equation 12. The dissimilarity between nodes i and j is defined as

$$\mathcal{D}_{ij} = \mathcal{D}_{ji} := \|\eta_i(s) - \eta_j(s)\|_{\mathcal{H}_2}, \tag{25.}$$

where $\eta_i(s) := (\mathbf{e}_i^\top \otimes C)(sI_{n\ell} - I_n \otimes A + \Gamma \otimes BC)^{-1}(F \otimes B)$. Now consider a single-integrator network in Equation 4. Node dissimilarity can be defined simply as

$$\mathcal{D}_{ij} = \|(\mathbf{e}_i - \mathbf{e}_j)^\top (sI_n - \Gamma)^{-1} F\|_{\mathcal{H}_2}. \tag{26.}$$

The transfer matrix $\eta_i(s)$ maps the external control signal $u \in \mathbb{R}^p$ to the measured output of the ith node $y_i \in \mathbb{R}^m$. Thus, $\eta_i(s)$ can be interpreted as the behavior of the ith node with respect to the external inputs, while the dissimilarity measure in Equation 25 indicates how different two nodes behave. The locations of inputs and the network topology determine the value of dissimilarity. Dissimilarity can also be defined in terms of other function norms, such as the $\mathcal{H}_\infty$ norm. However, to compute the dissimilarity between every pair of nodes in a large-scale network, the $\mathcal{H}_2$ norm of a stable linear time-invariant system can be characterized by its Gramians (9), whereas for other norms there is no such characterization, making them computationally less feasible.

Note that the dissimilarity in Equation 25 or Equation 26 is well defined only when $\eta_i(s) - \eta_j(s) \in \mathcal{H}_2$. This condition is guaranteed for network systems that are asymptotically stable or achieve synchronization (29). For instance, if Γ in Equation 4 is a Hurwitz matrix, then Equation 26 immediately becomes $\mathcal{D}_{ij} = \sqrt{(\mathbf{e}_i - \mathbf{e}_j)^\top P (\mathbf{e}_i - \mathbf{e}_j)}$, where P is the controllability Gramian of the network system represented by Equations 4, and P is computed as the unique solution of the Lyapunov equation $\Gamma P + P\Gamma^\top + FF^\top = 0$. If the system represented by Equation 4 is semistable, we resort to pseudo-Gramians defined by Equation 18 for the computation of the $\mathcal{H}_2$ norm.

For example, consider a single-integrator network in Equation 4 with $\Gamma = -L$ as the Laplacian matrix of an undirected graph. Following Cheng & Scherpen (58, 70), the pseudo-controllability Gramian of the system is computed as $\mathcal{P} = \mathcal{J}\tilde{\mathcal{P}}\mathcal{J}^\top$, where $\tilde{\mathcal{P}}$ is a solution of

$$-L\tilde{\mathcal{P}} - \tilde{\mathcal{P}}L + (I - \mathcal{J})FF^\top(I - \mathcal{J}) = 0, \quad \mathcal{J} := \frac{1}{n}\mathbb{1}\mathbb{1}^\top. \tag{27.}$$

The dissimilarity in Equation 26 is thereby obtained as

$$\mathcal{D}_{ij} = \sqrt{(\mathbf{e}_i - \mathbf{e}_j)^\top \mathcal{P} (\mathbf{e}_i - \mathbf{e}_j)}. \tag{28.}$$

Further consider the network system represented by Equation 12 with $\Gamma = -L$ a symmetric Laplacian matrix. Assume that the network achieves synchronization. In this case, Cheng et al. (29) provided the expression for Equation 25 as

$$\mathcal{D}_{ij} = \sqrt{\mathrm{Tr}(\Psi_{ij}\bar{\mathcal{P}}\Psi_{ij}^{\top})}, \qquad 29.$$

where Ψ_{ij} is defined with the help of the output matrix C, and $\bar{\mathcal{P}} \in \mathbb{R}^{(n-1)\ell\times(n-1)\ell}$ is the unique solution of a Lyapunov equation with matrices built from the system matrices.

The dissimilarity in Definition 4 is a pairwise measure that shows how close the behavior of two subsystems is, thus taking dynamics into account. This is significantly different from conventional node dissimilarity in data science or computer graphics (21, 22). Nevertheless, we can still follow similar clustering procedures or algorithms for data sets or static graphs.

Formally, given a network, the goal of clustering is to divide the nodes into clusters such that the elements assigned to a particular cluster are similar in a predefined metric. However, clustering with respect to a distance metric is generally an NP-hard combinatorial optimization problem, which is commonly solved by approximation algorithms. Here, we review two such algorithms and their adaptation to the clustering of network systems.

Agglomerative hierarchical clustering is a method that produces multilevel clusters. The key to this method is to define the proximity between two clusters $\mathcal{C}_i$ and $\mathcal{C}_j$. There are several alternatives to such a definition, such as considering the minimum, maximum, or average dissimilarity between any node in $\mathcal{C}_i$ and any node in $\mathcal{C}_j$. The proximity of two clusters allows us to identify a pair of clusters with the smallest proximity and merge them into a single cluster. This operation is executed iteratively to generate clusters in a hierarchy structure, which is visualized as a dendrogram—a tree-like diagram that records the sequences of cluster merges, such as the one shown in **Figure 6**. A graph clustering is obtained by cutting the dendrogram at the desired level, after which each connected component forms a cluster. Applications of hierarchical clustering to the model reduction of network systems can be found in, e.g., References 28, 59, 71, and 72.

Example 2. Consider the networked mass-damper system in Example 1. The dissimilarity matrix can be computed as in Equation 28, which yields

$$\mathcal{D} = \begin{bmatrix} 0 & 0.2494 & 0.3154 & 0.3919 & 0.4142 \\ 0.2494 & 0 & 0.2119 & 0.3688 & 0.3842 \\ 0.3154 & 0.2119 & 0 & 0.2410 & 0.2394 \\ 0.3919 & 0.3688 & 0.2410 & 0 & 0.0396 \\ 0.4142 & 0.3842 & 0.2394 & 0.0396 & 0 \end{bmatrix}.$$

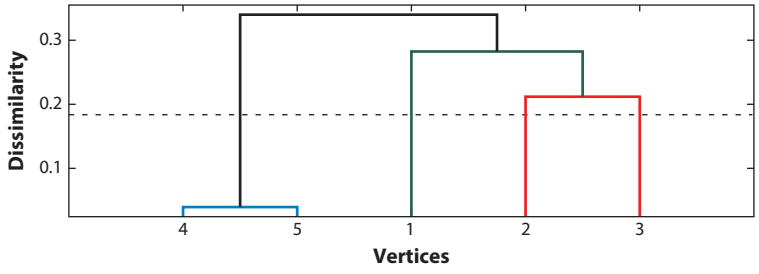

Figure 6

Dendrogram illustrating the hierarchical clustering of a networked mass-damper system. The horizontal axis is labeled by node numberings, and the vertical axis represents the proximity of clusters. The level at which branches merge indicates the proximity between two clusters.

We use the average link to define the cluster proximity, and a dendrogram is generated as depicted in **Figure 6**, showing how clusters are merged hierarchically. The dashed line cuts the dendrogram at a chosen level such that three clusters are formed: {1}, {2, 3}, and {4, 5}.

K-means clustering is a typical centroid-based partitioning method (21, 73) in which a cluster is constructed such that all the nodes within the cluster are more similar to the centroid of this cluster than to the centroids of any other clusters. For a network system of Equation 12, the centroid of a cluster $\mathcal{C}_k$ can be defined as

$$\mu(\mathcal{C}_k) = \frac{1}{|\mathcal{C}_k|} \sum_{i \in \mathcal{C}_k} \eta_i(s), \qquad 30.$$

with $\eta_i(s)$ defined in Equation 25. Given a network of n nodes, *K*-means clustering aims to partition the nodes into r subsets so as to minimize the following objective function: $\arg\min \sum_{k=1}^{r} \sum_{i \in \mathcal{C}_k} \|\eta_i(s) - \mu(\mathcal{C}_k)\|_{\mathcal{H}_2}^2$, in which $\eta_i(s) - \mu(\mathcal{C}_k) \in \mathcal{H}_2$ holds for synchronized networks. This problem can be solved using a simple iterative algorithm. First, take an initial group of r clusters of a given network and specify the centroid for each cluster as in Equation 30. Then, compute the dissimilarity between every node i and the r centroids, and assign node i to the cluster whose centroid is the closest to i. Finally, update the cluster centroids accordingly and repeat the steps until convergence.

Note that the formation of clusters in hierarchical clustering or *K*-means clustering relies solely on the dissimilarity measures and thus does not take into account the connectedness of nodes within the same cluster. It is worth noting that both methods can be modified to produce clusters of a graph, where each cluster forms a connected subgraph (29, 62, 74).

Generally, an upper bound on the reduction error $G(s) - \hat{G}(s)$, with $G(s)$ and $\hat{G}(s)$ the transfer matrices of systems represented by Equations 12 and 15, is not easy to obtain. We thus impose extra assumptions on the network system represented by Equation 12: $\Gamma = -L$ represents a connected undirected graph, and A in Equation 11 satisfies $A + A^\top < 0$. The a posteriori error bound is then given as $\|G(s) - \hat{G}(s)\|_{\mathcal{H}_2} < \gamma \sum_{k=1}^{r} |\mathcal{C}_k| \cdot \max_{i,j \in \mathcal{C}_k} \mathcal{D}_{ij}$, where $\gamma \in \mathbb{R}_{>0}$ depends only on the original system represented by Equation 12 and satisfies a linear matrix inequality (29, 37).

The most crucial part in dissimilarity-based clustering is to properly define the dissimilarity of nodes and clusters. For linear time-invariant network systems, dissimilarity can be defined using transfer matrices, which is applicable to different types of dynamical networks (for more generalizations to second-order networks, directed networks, and controlled power networks, see, e.g., 28, 62, 71). However, how to extend the dissimilarity-based clustering to network systems containing nonlinearities still needs further exploration. One potential solution resorts to the DC gain of monotone systems, introduced by Kawano et al. (72), where the DC gain can be regarded as an indicator of the node importance.

3.2. Clustering Meets Optimization

In the previous section, we reviewed how to select clusters and construct a reduced-order network model using clustering-based projection. In this section, we formulate the model reduction problem from the perspective of optimization, constructing a lower-order network model that minimizes a certain reduction error.

3.2.1. Reducibility and an $\mathcal{H}_2$ error bound. The pioneering work by Ishizaki et al. (24, 49) on clustering-based model reduction of dynamic networks introduced the notion of cluster reducibility, which is relevant to the classic notions of exact aggregation and approximate aggregation from the control and model reduction literature (75, 76).

Consider the network system in Equation 4 with Γ a Hurwitz, Metzler, and symmetric matrix and $F \in \mathbb{R}^n$. A cluster $\mathcal{C}_k$ is said to be reducible if there exist a scalar rational function $g_\star(s)$ and a vector $p_k \in \mathbb{R}^{|\mathcal{C}_k|}$ such that $I(\mathcal{C}_k)g(s) = p_k g_\star(s)$, where $g(s) := (sI - \Gamma)^{-1}F$ and $I(\mathcal{C}_k)$ denotes the matrix composed of the column vectors of I_n compatible with the set $\mathcal{C}_k$. Reducibility reflects the uncontrollability of node states in a cluster and can be further characterized in an algebraic manner. Ishizaki et al. (24) showed that the pair (Γ, F) can be converted into a positive tridiagonal realization by a unitary matrix T. Define $\Phi := -T\Gamma^{-1}F \in \mathbb{R}^{n\times n}$. A cluster $\mathcal{C}_k$ is then reducible if and only if there exists a vector $\phi_k \in \mathbb{R}^{|\mathcal{C}_k|}$ such that $I(\mathcal{C}_k)\Phi = p_k\phi_k^\top$, where $p_k := -I(\mathcal{C}_k)\Gamma^{-1}F$. With the vector p_k, an aggregation matrix can be defined as

$$\bar{\Pi} := \Big[I(\mathcal{C}_1)\ \cdots\ I(\mathcal{C}_r)\Big]\mathrm{blkdiag}\left(\frac{p_1}{\|p_1\|}, \ldots, \frac{p_r}{\|p_r\|}\right), \tag{31.}$$

which can be viewed as a weighted characteristic matrix Π in Definition 2. Note that $\bar{\Pi}^\top\bar{\Pi} = I_r$, and the reduced-order network model then becomes $\hat{g}(s) = \bar{\Pi}(sI_r - \bar{\Pi}^\top\Gamma\bar{\Pi})^{-1}\bar{\Pi}^\top F$.

If all the clusters are reducible, then the obtained reduced-order network model in Equation 7 has exactly the same input–output behavior as the original network, and thus $\|g(s) - \hat{g}(s)\|_{\mathcal{H}_2} = 0$. To further reduce the network model, the so-called θ reducibility is defined for a cluster $\mathcal{C}_k$ as $\|I(\mathcal{C}_k)\Phi - p_k\phi_k^\top\|_{\mathrm{F}} \le \theta$ for vectors $p_k, \phi_k \in \mathbb{R}^{|\mathcal{C}_k|}$. If all the clusters are θ reducible, an a posteriori upper bound on the reduction error can be formed:

$$\|g(s) - \hat{g}(s)\|_{\mathcal{H}_2} \le \gamma\sqrt{\sum_{k=1}^{r}|\mathcal{C}_k|(|\mathcal{C}_k| - 1)\theta}, \tag{32.}$$

where γ is an upper bound of $\|\bar{\Pi}(sI_r - \bar{\Pi}^\top\Gamma\bar{\Pi})^{-1}\bar{\Pi}^\top\Gamma + I_n\|_{\mathcal{H}_\infty}$ and characterized by a Riccati inequality.

Ishizaki et al. (49) provided a generalization of the above method that considers semistable directed networks. In this case, the Frobenius eigenvector of Γ is used to construct the aggregation matrix in Equation 31 in order to preserve both semistability and positivity in the reduced-order network model. A so-called projected controllability Gramian, which can be viewed as a special pseudo-controllability Gramian, is used for the characterization. The a posteriori error bound in Equation 32 is also extended to directed networks. However, this extension turns out to be questionable, since the relevant Riccati inequality in general does not have a solution for semistable systems. Cheng & Scherpen (58) provided a correct formulation in terms of the $\mathcal{H}_\infty$ norm of a semistable system, which has the form of Equation 21.

The notion of reducibility and the error bound are essential for the clustering procedure described by Ishizaki et al. (24, 49). The core step in the clustering algorithm is to produce a set of θ-reducible clusters, where the value of θ is adjusted in relation to the approximation error bound. Ishizaki & Imura (77) and Ishizaki et al. (37) extended this approach to reduce stable second-order network systems and to reduce networked dissipative systems in the form of Equation 12.

3.2.2. $\mathcal{H}_2$ suboptimal methods. Model reduction of a network system can be formulated as a nonconvex optimization problem in which the objective function is the $\mathcal{H}_2$ reduction error. The characteristic matrix Π is the optimization variable and is subject to the constraint

$$\Pi \in C := \big\{\Pi\mathbb{1}_r = \mathbb{1}_n,\ \text{and}\ [\Pi]_{ij} = \{0, 1\},\ \forall i = 1, \ldots, n,\ j = 1, \ldots, r\big\}. \tag{33.}$$

The optimization problem itself is nonconvex due to the nonlinear objective function in terms of the $\mathcal{H}_2$ norm and the binary variable Π. To solve such a nonconvex problem, a relaxation of the binary constraints can be taken, leading to suboptimal approaches.

When we drop the constraint $\Pi \in C$, we obtain an $\mathcal{H}_2$ optimal model reduction problem for a generic linear time-invariant system that can be solved by using the so-called iterative rational Krylov algorithm to seek a (locally) optimal solution (65). The algorithm gives a subspace of dimension r with the basis $V_r \in \mathbb{R}^{n\times r}$. In contrast to the V defined in Equation 17, this V_r is not a feasible solution, since it does not preserve the network structure. Therefore, the idea is to find a Π matrix in the feasible set C such that the image of Π is approximately equal to the image of V_r, that is, $\mathrm{Im}(\Pi) \approx \mathrm{Im}(V_r)$. To this end, Mlinarić et al. (65) adopted an approach based on a QR decomposition with column pivoting, which was originated by Zha et al. (73) for solving K-means clustering problems.

An alternative method further studies this nonconvex optimization problem and specifies the objective function using the controllability and observability Gramians (78). Consider the single-integrator network in Equation 4 with $\Gamma = -L$ representing a connected undirected graph. We aim for the following optimization problem:

$$\min_{\Pi\in\mathbb{R}^{n\times r}} J(\Pi) := \mathrm{Tr}(B_{\mathrm{e}}^{\top} P_{\mathrm{e}} B_{\mathrm{e}}) \text{ subject to Equation 33 and } A_{\mathrm{e}}^{\top} P_{\mathrm{e}} + P_{\mathrm{e}} A_{\mathrm{e}} + C_{\mathrm{e}}^{\top} C_{\mathrm{e}} = 0, \qquad 34.$$

where P_{e} is the observability Gramian of the stable system $(A_{\mathrm{e}}, B_{\mathrm{e}}, C_{\mathrm{e}})$, defined as

$$A_{\mathrm{e}} = -\begin{bmatrix} \mathcal{S}_n^{\dagger} L \mathcal{S}_n & 0 \\ 0 & \mathcal{S}_r^{\dagger} \Pi^{\top} L \Pi \mathcal{S}_r \end{bmatrix},\ B_{\mathrm{e}} = \begin{bmatrix} \mathcal{S}_n^{\dagger} F \\ \mathcal{S}_r^{\dagger} \Pi^{\top} F \end{bmatrix},\ C_{\mathrm{e}} = \begin{bmatrix} H \mathcal{S}_n & -\frac{r}{n} H \Pi \mathcal{S}_r \end{bmatrix}.$$

The matrices $\mathcal{S}_n$ and $\mathcal{S}_r$ are given by $\mathcal{S}_k = [I_{k-1}\ \mathbb{1}_{k-1}]^{\top}$ for $k = r, n$. The latter matrices are used to filter out the subspace corresponding to the zero eigenvalues, so that A_{e} is a Hurwitz matrix. From the objective function $J(\Pi)$, we can derive an explicit expression for its gradient $\nabla J(\Pi)$ so that gradient-based algorithms—including projected gradient descent, Frank–Wolfe optimization, and conditional gradient methods—can be applied to solve the optimization problem in Equation 34 (for more details, see 78, 79, and the references therein).

3.2.3. Edge weighting. Instead of seeking a way to do the graph clustering, the $\mathcal{H}_2$ optimization scheme can also be applied to construct a "good" reduced-order model from a given clustering. To achieve this, we must go beyond the framework of Petrov–Galerkin projection. Given a certain clustering, the topology of a reduced-order network is known, while the coupling strengths (edge weights) are considered free parameters to be determined (60, 66).

Consider a network system in Equation 4 with a connected undirected graph $\mathcal{G}$. Let $\{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_r\}$ be a given graph clustering of $\mathcal{G}$. A quotient graph $\hat{\mathcal{G}}$ is then an r-node digraph obtained by aggregating all the nodes in each cluster as a single node while retaining connections between clusters and ignoring the edges within each cluster. The incidence matrix $\hat{\mathcal{B}}$ of the quotient graph $\hat{\mathcal{G}}$ can be obtained by removing all the zero columns of $\Pi^{\top}\mathcal{B}$, where $\mathcal{B}$ is the incidence matrix of $\mathcal{G}$ and Π is the characteristic matrix of the clustering. Denote $\hat{W} = \mathrm{Diag}(\hat{w})$, with $\hat{w} = [\hat{w}_1\ \hat{w}_2\ \ldots\ \hat{w}_\kappa]^{\top}$, as the edge-weighting matrix of $\hat{\mathcal{G}}$, with $\hat{w}_k \in \mathbb{R}_{>0}$ and κ the number of edges in $\hat{\mathcal{G}}$. The parameterized model of a reduced-order network is then obtained as

$$(\Pi^{\top}\Pi)\dot{z}(t) = -\hat{\mathcal{B}}\hat{W}\hat{\mathcal{B}}^{\top} z(t) + \Pi^{\top} F u(t), \quad \hat{y}(t) = H\Pi z(t), \qquad 35.$$

which has as a transfer matrix $\hat{\eta}_p(s, \hat{W}) = H\Pi(s\Pi^{\top}\Pi + \hat{\mathcal{B}}\hat{W}\hat{\mathcal{B}}^{\top})^{-1}\Pi^{-1}F$. In the reduced-order model, the edge weight matrix $\hat{W}$ is the only unknown and can be determined via an optimization procedure.

Example 3. Consider an undirected graph composed of six nodes, as shown in **Figure 7*a***. Given a clustering with $\mathcal{C}_1 = \{1, 2\}$, $\mathcal{C}_2 = \{3\}$, $\mathcal{C}_3 = \{4\}$, and $\mathcal{C}_4 = \{5, 6\}$, the quotient graph

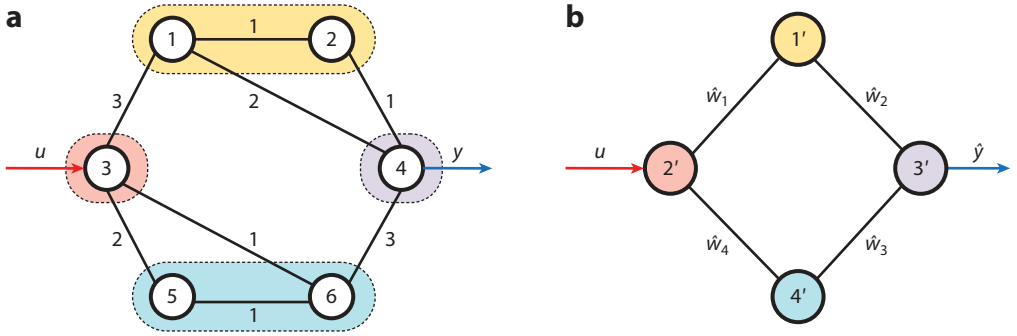

Figure 7

(*a*) An undirected network consisting of six nodes, in which node 3 is the leader and node 4 is measured. Four clusters are indicated by different colors. (*b*) A quotient graph obtained by clustering.

is obtained in **Figure 7*b*** with edge weight matrix $\hat{W} = \mathrm{Diag}(\hat{w}_1, \hat{w}_2, \hat{w}_3, \hat{w}_4)$. The parameterized model of the reduced network is then constructed as

$$\underbrace{\begin{bmatrix} 2&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&0&0&2 \end{bmatrix}}_{\Pi^\top\Pi} \begin{bmatrix} \dot{z}_1\\ \dot{z}_2\\ \dot{z}_3\\ \dot{z}_4 \end{bmatrix} = -\underbrace{\begin{bmatrix} \hat{w}_1+\hat{w}_2 & -\hat{w}_1 & -\hat{w}_2 & 0\\ -\hat{w}_1 & \hat{w}_1+\hat{w}_4 & 0 & -\hat{w}_4\\ -\hat{w}_2 & 0 & \hat{w}_2+\hat{w}_3 & -\hat{w}_3\\ 0 & -\hat{w}_4 & -\hat{w}_3 & \hat{w}_3+\hat{w}_4 \end{bmatrix}}_{\hat{\mathcal{B}}\hat{W}\hat{\mathcal{B}}^\top} \begin{bmatrix} z_1\\ z_2\\ z_3\\ z_4 \end{bmatrix} + \underbrace{\begin{bmatrix} 0\\ 1\\ 0\\ 0 \end{bmatrix}}_{\Pi^\top F} u,$$

and $y = z_3$. In this model, the diagonal elements of $\hat{W}$ are the parameters to be determined.

Similar to the process in Section 3.2.2, an optimization problem can be formulated to minimize the reduction error between the original and reduced-order network systems by tuning the edge weights. Specifically, the objective function is $\|\eta(s) - \hat{\eta}_p(s, \hat{W})\|_{\mathcal{H}_2}$, in which a diagonal and positive definite $\hat{W}$ is the optimization variable.

There are multiple algorithms for solving such a problem. Cheng et al. (66) characterized the upper bound of the reduction error—that is, the expression $\|\eta(s) - \hat{\eta}_p(s, \hat{W})\|_{\mathcal{H}_2} < \gamma$—by a set of linear matrix inequalities and applied a cross-iteration algorithm such that the upper bound γ decreases via iterations. An alternative approach uses convex–concave decomposition (60); this approach was inspired by the work of Dinh et al. (80) and is based on linearization of the optimization problem at a given point, so that the problem becomes convex and can be solved efficiently. The overall problem can then be solved in an iterative fashion, and in each iteration a convex optimization problem needs to be solved.

It is worth noting that the edge-weighting approach can be implemented as a subsequent procedure after the clustering-based methods described earlier—that is, we can first apply one of the above-mentioned algorithms to find a graph clustering, and then use the edge weights of that clustering to initialize $\hat{W}$ in the edge-weighting approach. Then, through iterations, we can generate a more accurate reduced-order network model.

3.3. Other Topological Reduction Methods

There is a vast amount of literature about the problem of topological reduction. In this section, we briefly summarize several other representative methods.

3.3.1. Singular perturbation approximation and Kron reduction. Along with graph clustering, the other mainstream methodology for simplifying the topological complexity of a network

is based on timescale separation analysis, particularly singular perturbation approximation (81). This method has been extensively investigated in the applications of biochemical reaction systems (4, 82, 83) and electric power networks (8, 84–86). A key feature of those systems is that there is usually an explicit or nonexplicit separation of timescales in the states of networks. For example, slow coherency theory implies that power networks are naturally decomposed into areas, where power generators within each area synchronize on a faster timescale, while network-wide interactions between the areas are captured by slower motions. Singular perturbation methods then help to separate these dynamics so that they can be analyzed separately. This produces a reduced-order network model that retains, for example, the low-frequency behavior of a large-scale network. Identifying and separating fast and slow states are crucial steps in this method, and its use is highly dependent on the specific application.

A relevant concept is Kron reduction of graphs, which is a term commonly used in classic circuit theory and related fields (such as electrical impedance tomography) and in transient stability assessment in power networks (see, e.g., 8, 87, 88). This idea may also be used for exact reduction—for example, to go from a differential algebraic description of a network system to a differential description with structure preservation.

3.3.2. Kullback–Leibler aggregation. Networked dynamical systems derived from the discretization of thermodynamics and fluid dynamics are usually modeled as regular discrete-time Markov chains without control inputs. For this type of system, the notion of Kullback–Leibler divergence rate can be adopted to measure the difference between two Markov chains defined on the same state space. Deng et al. (89) further extended this notion to measure the Kullback–Leibler divergence rate between the original and reduced-order models defined on different state spaces. With the new Kullback–Leibler divergence rate, an optimization problem is formulated that aims to find an optimal partition of the states, which are aggregated to form a reduced-order model. An application of this method has been explored in reducing complex building thermal systems (26).

3.3.3. Network reduction toward scale-free structure. Graph clustering and aggregation have also been studied to retain more relevant properties, such as connectivity and scale freeness (31, 90). The scale-free networks typically contain a few nodes with a large degree (the so-called hubs) and a large number of nodes with a small degree, and the distribution of the node degree follows a certain power law (31).

The model reduction problem of networks preserving the scale-free property can be formulated as an optimization problem that finds a clustering of a given large-scale network such that the aggregated network has a degree distribution closest to a desired scale-free distribution. The preservation of the scale-free structure is particularly important for applications of flow networks, such as traffic networks, power networks, and packet flow networks.

3.3.4. Indirect network reduction methods. In contrast to the mechanisms of clustering and aggregation that directly produce a reduced network, indirect methods seek a structure-preserving reduced-order model using a two-step procedure: reduction and transformation (38, 91–93). In the reduction step, a lower-dimensional model of a given large-scale network system is constructed by using conventional model reduction methods, such as generalized balanced truncation (38, 91) or moment matching (92). Generally, the reduced-order model generated in this step does not allow for a network interpretation. The transformation step is then implemented, which converts the reduced-order model obtained in the previous step into a network model. This method is also relevant for the combined nodal and topological reduction procedure in Section 4.2.

Cheng & Scherpen (91) and Cheng et al. (38) presented a key part of the transformation: A matrix is similar to a Laplacian matrix of a connected undirected graph if and only if it is diagonalizable and has exactly one zero eigenvalue, while all the other eigenvalues are real positive. This result guides the reduction step, in which a certain spectral constraint must be imposed. The second step then turns out to be an eigenvalue matching problem.

Similarly, an eigenvalue assignment approach (93) directly selects a subset of the Laplacian spectrum of the original network to be the eigenvalues of the Laplacian matrix for the reduced network. This method allows certain properties of the original network, such as stability and synchronization, to be preserved through the reduction process.

4. REDUCTION OF THE FULL SYSTEM DYNAMICS

In the previous section, we considered the reduction of the topological structure using clustering methods. The input–output structure is considered when looking at, for example, the $\mathcal{H}_2$ norm, but for control systems it is undoubtedly important that the model preserves certain input–output or control structures beyond considering only the $\mathcal{H}_2$ norm. Various methods have been developed for general linear and nonlinear systems (e.g., 9, 94, 95; for a brief introduction, see the sidebar titled Model Reduction for Linear Systems). Here, we first consider reduction of the nodal dynamics while preserving certain graph properties, and then present a combined nodal and topological method.

4.1. Reduction of the Nodal Dynamics

For the nodal dynamics, it is useful to consider reduction methods for interconnected systems. In the corresponding literature, the network perspective is not the primary focus, but the methods are nevertheless relevant for network systems and treated in this subsection. In addition, we discuss reduction methods for nodal dynamics that do explicitly take the network perspective and preserve properties such as synchronization.

4.1.1. Reduction methods for interconnected linear systems. Perhaps one of the first relevant papers to consider reduction methods for interconnected systems was by Sandberg & Murray (34), who considered linear fractional transformations (see **Figure 8**). The paper presented two methods for reduction, both based on balanced truncation principles. The first method considers only the diagonal blocks of the observability and controllability Gramians, with each block corresponding to a subsystem, hence neglecting the off-diagonal blocks. This method can provide an expression for the a posteriori error bound, but unfortunately, it cannot obtain an a priori error bound. The second method uses generalized Gramians; that is, instead of considering the observability and controllability Gramians that are the unique solutions to corresponding Lyapunov equations, it considers the (nonunique) solutions to Lyapunov inequalities. The freedom in choosing solutions to these Lyapunov inequalities makes it possible to pick block-diagonal solutions and consequently results in an a priori error bound. Ishizaki et al. (97) provided an extension toward a singular perturbation perspective, and Jaoude & Farhood (98) provided balanced truncation based on generalized Gramians in the discrete-time setting for interconnected systems. More generally, Reis & Stykel (32) and Vandendorpe & Van Dooren (33) presented an overview of the various methods until 2008 and reduction methods for coupled systems, respectively.

A more recent result deals with interconnected systems in a graph setting (e.g., 99). In particular, the subsystems are of the form $\dot{x}_i = \sum_{j\in\mathcal{N}_i} A_{ij}x_j + B_i u$, where $\mathcal{N}_i$ is the index set of the connections of the ith subsystem with other subsystems. With the help of generalized Gramians,

MODEL REDUCTION FOR LINEAR SYSTEMS

It is generally accepted that model reduction approaches for linear control systems can be roughly divided into two types of approaches: those based on singular values and those based on moment matching (e.g., 9). A very brief review follows. Let (A, B, C) be a state space realization of Σ with dimension n, input $u \in \mathbb{R}^m$, state $x \in \mathbb{R}^n$, and output $y \in \mathbb{R}^p$. We assume that the system is asymptotically stable and minimal (i.e., controllable and observable). The corresponding transfer matrix is given by $G(s) = C(sI - A)^{-1}B$.

Balanced Realizations

Theorem 4 (from Reference 12). Consider the system Σ. Take P as the controllability Gramian and Q as the observability Gramian. The eigenvalues of QP are similarity invariants—that is, they do not depend on the choice of the state space coordinates. Furthermore, there exists a state space representation where $\Sigma := Q = P = \mathrm{Diag}\{\sigma_1, \ldots, \sigma_n\}$, with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0$ the square roots of the eigenvalues of QP. Such representations are called balanced, and the system is in balanced form. Furthermore, the σ_i's, $i = 1, \ldots, n$, equal the Hankel singular values (i.e., the singular values of the Hankel operator).

The Hankel singular values form a measure for the contribution to the minimality of a state component, which provides the basis for model reduction methods based on balanced realizations. Model reduction based on balancing is possible with a priori error bounds in various norms, such as the $\mathcal{H}_\infty$ and Hankel norms. In particular, it can be shown that balanced truncation results in an error bound corresponding to the sum of the truncated Hankel singular values—that is, $\|G(s) - G_r(s)\|_\infty \leq 2\sum_{k=r}^{n} \sigma_k$, where $G_r(s)$ represents the transfer matrix of the reduced-order system (13). Antoulas (9) and Scherpen (95) provided a more detailed overview.

Moment Matching

The principle of moment matching for a linear system is based on the series representation of the transfer matrix of the system (for more detailed expositions, see, e.g., 9, 96). Without loss of generality, it is assumed that $m = p = 1$.

Definition 5. The zero moment at $s_1 \in \mathbb{C}$ of Σ is the complex number $\eta_0(s_1) = C(s_1 I - A)^{-1}B$. The k moment at s_1 is given by the complex number

$$\eta_k(s_1) = \frac{(-1)^k}{k!} \frac{d^k \left\{C(sI - A)^{-1}B\right\}}{ds^k}\Big|_{s = s_1}, \quad k = 1, 2, 3, \ldots.$$

The point $s_1 \in \mathbb{C}$ is called an interpolation point. The approximation problem for system (A, B, C) at $s_1 \in \mathbb{C}$ is to find a system $(\hat{A}, \hat{B}, \hat{C})$ of order $\nu < n$, with transfer function $\hat{G}(s) = \hat{C}(sI - \hat{A})^{-1}\hat{B}$, such that $\eta_k(s_1) = \hat{\eta}_k(s_1)$, with $\hat{\eta}_k(s_1)$ the moments of $\hat{G}(s)$, $k = 1, \ldots, \nu$. Various types of moment-matching methods have been developed. Generally, it is not possible to provide an a priori error bound. These methods are, however, computationally very interesting if one handles systems with millions of states.

optimal $\mathcal{H}_2$ moment matching–based model reduction is performed while preserving the network structure. This is done in a convex optimization setting, and a projected gradient method is additionally applied for the nonconvex case.

4.1.2. Nodal reduction while preserving synchronization properties. The above methods consider interconnected systems but do not yet take into account the preservation of properties that are typically relevant for (controlled) network systems, such as consensus or synchronization properties, which requires additional steps. To the best of our knowledge, the first work to consider the preservation of synchronization properties in network systems was by Monshizadeh

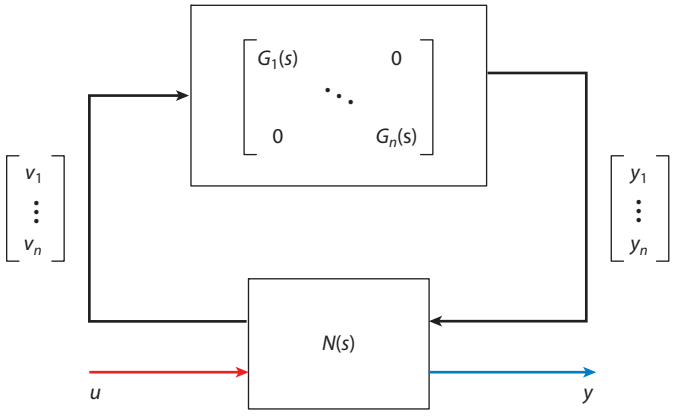

Figure 8

An interconnected system with subsystems to be reduced, $G_1(s), \ldots, G_n(s)$, stored in a block-diagonal transfer matrix $G(s)$ and connected to a transfer matrix $N(s)$ that models the interconnection topology and excitation and measurement dynamics (34).

et al. (35). Consider a diffusively coupled linear network system as in Equation 12 with a symmetric Laplacian $\Gamma = -L$. A bounded real balancing method can then be used to preserve the stability of the network. In addition, synchronization can be preserved by considering the positive definite solution K of the following Riccati equation:

$$(A - \lambda BC)^{\mathrm{T}}K + K(A - \lambda BC) + C^{\mathrm{T}}C + \delta^2 KBB^{\mathrm{T}}K = 0,$$

where δ is a scalar that must fulfill some additional conditions, and λ is an eigenvalue of the Laplacian L for which $A - \lambda BC$ is a Hurwitz matrix. The maximum and minimum solutions of the equation can now be balanced (i.e., simultaneously diagonalized), as is done for standard balancing. Truncating the system based on these diagonal values results in reduced-order dynamics of the agents and a network system that is still synchronized. Furthermore, an a priori error bound is provided based on the truncated diagonal values, δ, and the largest eigenvalue of the Laplacian.

The above balancing method based on the minimum and maximum solution of a Riccati equation can be generalized to finding a solution to the inequality, and as such, it can be related to solutions of linear matrix inequalities. Also, so far, only the linear dynamics in the nodes have been considered, but in practice nonlinearities also play an important role. Because the error dynamics is more difficult to handle in the case of nonlinear systems, it is useful to consider nodal dynamics represented by Lur'e systems, that is, systems with a static nonlinearity in the feedback loop. Cheng et al. (36) and Cheng & Scherpen (100) considered robust synchronization-preserving reduction methods for nodal Lur'e systems. In this context, consider the following nodal dynamics:

$$\boldsymbol{\Sigma_i}: \begin{cases} \dot{x}_i(t) = Ax_i(t) + Bu_i(t) + Ez_i(t), \\ y_i(t) = Cx_i(t), \\ z_i(t) = -\phi(y_i(t)), \end{cases} , \qquad 36.$$

where $x_i \in \mathbb{R}^n$; $u_i, y_i, z_i \in \mathbb{R}^m$; and $\phi : \mathbb{R}^m \to \mathbb{R}^m$, fulfilling some sector bound condition. Without loss of generality, we take $m = 1$ (100). For a diffusively coupled network that is robustly synchronized and has nodal dynamics as in Equation 36, we now take the minimum and maximum

solutions $\mathcal{K}_{\min}$ and $\mathcal{K}_{\max}$ of the following linear matrix inequality:

$$\begin{bmatrix} A^{\mathrm{T}}\mathcal{K} + \mathcal{K}A + C^{\mathrm{T}}C & \lambda_n \mathcal{K}B & \mathcal{K}E - C^{\mathrm{T}} \\ \lambda_n B^{\mathrm{T}}\mathcal{K} & -I & 0 \\ E^{\mathrm{T}}\mathcal{K} - C & 0 & 0 \end{bmatrix} \preccurlyeq 0,$$

with λ_n the largest eigenvalue of the Laplacian. $\mathcal{K}_{\min}$ and $\mathcal{K}_{\max}$ can be balanced, and reduction based on them results in a robustly synchronized network of Lur'e systems with a priori determined error bounds. Variations on this method can obtain even better error bounds, and Cheng et al. (36) provided an extension to the multi-input, multi-output case.

4.2. Combined Topological and Nodal Reduction

As shown by the previous sections, the techniques for topological simplification and subsystem reduction in network systems derive from rather different perspectives. The methods for reducing subsystem dynamics are commonly adapted from classic model reduction techniques, such as balanced truncation or Krylov subsystem methods, while in the structure-preserving simplification of network structures, clustering-based approaches have demonstrated superior performance. In this section, we discuss the combination of the two approximation problems in a unified framework. This is particularly needed when dealing with a network with both a complex topology and high-order subsystems.

However, performing a simultaneous reduction of topological complexity and subsystem dynamics is not straightforward. Naively, we may apply the methods of Sections 3 and 4.1 separately to achieve the two approximation goals one by one. Nevertheless, which reduction sequence gives a better approximation is still unclear, and moreover, there is hardly a guarantee on the approximation error. Relevant results in the literature on combined topological and nodal reduction are rare. The existing ones are developed generally for networked homogeneous linear systems, as in Equation 12 under specific assumptions (37, 38). For example, Ishizaki et al. (37) imposed the regularity and dissipativity of the entire system matrix $I \otimes A + \Gamma \otimes BC$, which admits a block-diagonal Lyapunov function. This is essential to preserve the stability of the reduced-order model and to derive an upper bound on the approximation error caused by the reduction of both network structure and subsystems, where the topological reduction is done by graph clustering.

Cheng et al. (38) considered a network system that is synchronized but not necessarily stable. The synchronization property is based on the assumption that each subsystem (A, B, C) is minimal and passive. By extending the results for networked single integrators of Section 3.3.4, we can reduce the topological complexity in an indirect manner. A generalized balanced truncation method then results in a unified framework to simplify the network structure and subsystem dynamics simultaneously. The reduction scheme is illustrated in **Figure 9**, which contains two core steps.

The first step is to decompose the considered network system into two parts, which correspond to the average of all subsystems and the discrepancy among the subsystem states, respectively. Due to the synchronization property, the latter part, which captures the main dynamics of the entire network, is represented by an asymptotically stable system $\mathbf{\Sigma}_{\mathrm{s}}$ with a Hurwitz system matrix $(I_{n-1} \otimes A - \bar{\Lambda} \otimes BC)$, where $\bar{\Lambda} \in \mathbb{R}^{(n-1)\times(n-1)}$ shares all the nonzero eigenvalues with the Laplacian matrix $L \in \mathbb{R}^{n\times n}$ of the original network. A pair of generalized Gramians with a Kronecker product structure is then selected, which is key to decouple the balanced truncation procedures of the network structure and subsystem dynamics. In the second step, the simplified stable system and average system are integrated, resulting in a reduced-order model of the original $\mathbf{\Sigma}$. However, such a reduced-order model provides only an approximation of the input–output mapping of $\mathbf{\Sigma}$,

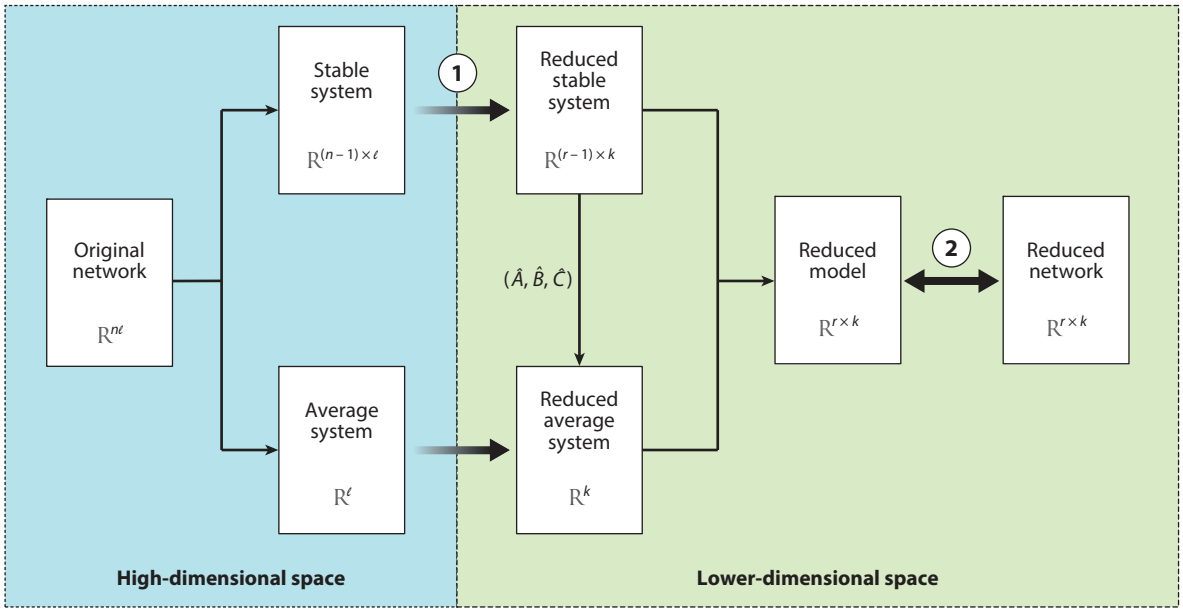

Figure 9

The model reduction procedure for a networked passive system, which contains two key techniques: (①) balanced truncation based on generalized Gramians and (②) a coordinate transformation that reconstructs a network.

rather than the network structure. To restore the network interpretation in that reduced-order model, a coordinate transformation is required to recover a reduced Laplacian matrix. Cheng & Scherpen (91) presented the following theoretical foundation for such transformation.

Theorem 5. A real square matrix $\mathcal{N}$ is similar to the Laplacian matrix L associated with a weighted undirected connected graph if and only if $\mathcal{N}$ is diagonalizable and has an eigenvalue at 0 with multiplicity 1 while all the other eigenvalues are real and positive.

Cheng et al. (38) provided a detailed proof of the above theorem and a method for reconstructing an undirected network from given eigenvalues $0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_n$. Forrow et al. (101) presented an alternative graph reconstruction approach. The reconstructed graphs are usually complete—that is, there is an edge between any pair of nodes. A subsequent procedure can be used to sparsify the interconnection structure (see, e.g., 101, 102).

SUMMARY POINTS

1. Graph-theoretical analysis plays a paramount role in reduced-order modeling of complex network systems.
2. Clustering methods provide a structure-preserving way to reduce the topology of network systems. Dissimilarity-based clustering provides a rather general framework for simplifying the topological complexity of a network system, where the key is to properly define a metric to characterize the dissimilarity between nodes or clusters.

3. Network systems that achieve synchronization are naturally semistable. Novel pseudo-Gramian notions are introduced for semistable systems as an extension of Gramian matrices for asymptotically stable systems, providing a useful tool to characterize dissimilarity and compute the reduction error.
4. Generalized balanced truncation allows more freedom in constructing a reduced-order model with some desired structures or properties. Thus, it is widely used in the approximation of network systems, particularly in dealing with subsystem reduction.
5. Graph reconstruction can realize a network representation from a reduced-order model satisfying certain spectral constraints. This makes it possible to apply more classical model order reduction methods from the control systems literature to structure-preserving reduction of network systems.

FUTURE ISSUES

1. The approximation of complex network systems with nonlinear couplings and nonlinear subsystems is still challenging and requires further investigation.
2. How to reduce the topological complexity of dynamic networks composed of heterogeneous subsystems is not yet clear.
3. The application of reduced-order network models for designing controllers and observers for large-scale networks is appealing, and obtaining provable guarantees on the functionality of the controllers or observers based on reduced-order models should be further explored.

DISCLOSURE STATEMENT

The authors are not aware of any affiliations, memberships, funding, or financial holdings that might be perceived as affecting the objectivity of this review.

LITERATURE CITED

1. Mesbahi M, Egerstedt M. 2010. *Graph Theoretic Methods in Multiagent Networks*. Princeton, NJ: Princeton Univ. Press
2. Newman ME. 2003. The structure and function of complex networks. *SIAM Rev*. 45:167–256
3. Ren W, Beard RW, Atkins EM. 2005. A survey of consensus problems in multi-agent coordination. In *Proceedings of the 2005 American Control Conference*, pp. 1859–64. Piscataway, NJ: IEEE
4. Rao S, van der Schaft AJ, Jayawardhana B. 2013. A graph-theoretical approach for the analysis and model reduction of complex-balanced chemical reaction networks. *J. Math. Chem.* 51:2401–22
5. Ahsendorf T, Wong F, Eils R, Gunawardena J. 2014. A framework for modelling gene regulation which accommodates non-equilibrium mechanisms. *BMC Biol.* 12:102
6. Proskurnikov AV, Tempo R. 2017. A tutorial on modeling and analysis of dynamic social networks. Part I. *Annu. Rev. Control* 43:65–79
7. Dörfler F, Bullo F. 2012. Synchronization and transient stability in power networks and nonuniform kuramoto oscillators. *SIAM J. Control Optim.* 50:1616–42
8. Chow JH, ed. 2013. *Power System Coherency and Model Reduction*. New York: Springer

9. Antoulas AC. 2005. *Approximation of Large-Scale Dynamical Systems*. Philadelphia: Soc. Ind. Appl. Math.
10. Bai Z. 2002. Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems. *Appl. Numer. Math.* 43:9–44
11. Astolfi A. 2010. Model reduction by moment matching for linear and nonlinear systems. *IEEE Trans. Autom. Control* 55:2321–36
12. Moore BC. 1981. Principal component analysis in linear systems: controllability, observability, and model reduction. *IEEE Trans. Autom. Control* 26:17–32
13. Glover K. 1984. All optimal Hankel-norm approximations of linear multi-variable systems and their L_∞-error bounds. *Int. J. Control* 39:1115–93
14. Besselink B, Tabak U, Lutowska A, Van de Wouw N, Nijmeijer H, et al. 2013. A comparison of model reduction techniques from structural dynamics, numerical mathematics and systems and control. *J. Sound Vib.* 332:4403–22
15. Summers TH, Lygeros J. 2014. Optimal sensor and actuator placement in complex dynamical networks. *IFAC Proc. Vol.* 47(3):3784–89
16. Gates AJ, Rocha LM. 2016. Control of complex networks requires both structure and dynamics. *Sci. Rep.* 6:24456
17. Kim JZ, Soffer JM, Kahn AE, Vettel JM, Pasqualetti F, Bassett DS. 2018. Role of graph architecture in controlling dynamical networks with applications to neural systems. *Nat. Phys.* 14:91–98
18. Ishizaki T, Chakrabortty A, Imura JI. 2018. Graph-theoretic analysis of power systems. *Proc. IEEE* 106:931–52
19. Obinata G, Anderson BDO. 2012. *Model Reduction for Control System Design*. London: Springer
20. Mustafa D, Glover K. 1991. Controller reduction by H_∞ balanced truncation. *IEEE Trans. Autom. Control* 36:668–82
21. Jain AK, Murty MN, Flynn PJ. 1999. Data clustering: a review. *ACM Comput. Surv.* 31:264–323
22. Schaeffer SE. 2007. Graph clustering. *Comput. Sci. Rev.* 1:27–64
23. van der Schaft AJ. 2014. On model reduction of physical network systems. In *Proceedings of the 21st International Symposium on Mathematical Theory of Networks and Systems*, pp. 1419–25. Groningen, Neth.: Univ. Groningen
24. Ishizaki T, Kashima K, Imura JI, Aihara K. 2014. Model reduction and clusterization of large-scale bidirectional networks. *IEEE Trans. Autom. Control* 59:48–63
25. Monshizadeh N, Trentelman HL, Camlibel MK. 2014. Projection-based model reduction of multi-agent systems using graph partitions. *IEEE Trans. Control Netw. Syst.* 1:145–54
26. Deng K, Goyal S, Barooah P, Mehta PG. 2014. Structure-preserving model reduction of nonlinear building thermal models. *Automatica* 50:1188–95
27. Besselink B, Sandberg H, Johansson KH. 2016. Clustering-based model reduction of networked passive systems. *IEEE Trans. Autom. Control* 61:2958–73
28. Cheng X, Kawano Y, Scherpen JMA. 2017. Reduction of second-order network systems with structure preservation. *IEEE Trans. Autom. Control* 62:5026–38
29. Cheng X, Kawano Y, Scherpen JMA. 2019. Model reduction of multi-agent systems using dissimilarity-based clustering. *IEEE Trans. Autom. Control* 64:1663–70
30. Jongsma HJ, Mlinarić P, Grundel S, Benner P, Trentelman HL. 2018. Model reduction of linear multi-agent systems by clustering with h_2 and H_∞ error bounds. *Math. Control Signals Syst.* 30:6
31. Martin N, Frasca P, Canudas-de Wit C. 2018. Large-scale network reduction towards scale-free structure. *IEEE Trans. Netw. Sci. Eng.* 6:711–23
32. Reis T, Stykel T. 2008. A survey on model reduction of coupled systems. In *Model Order Reduction: Theory, Research Aspects and Applications*, ed. WHA Schilders, HA van der Vorst, J Rommes, pp. 133–55. Berlin: Springer
33. Vandendorpe A, Van Dooren P. 2008. Model reduction of interconnected systems. In *Model Order Reduction: Theory, Research Aspects and Applications*, ed. WHA Schilders, HA van der Vorst, J Rommes, pp. 305–21. Berlin: Springer
34. Sandberg H, Murray RM. 2009. Model reduction of interconnected linear systems. *Optim. Control Appl. Methods* 30:225–45

35. Monshizadeh N, Trentelman HL, Camlibel MK. 2013. Stability and synchronization preserving model reduction of multi-agent systems. *Syst. Control Lett.* 62:1–10
36. Cheng X, Scherpen JMA, Zhang F. 2019. Model reduction of synchronized homogeneous Lur'e networks with incrementally sector-bounded nonlinearities. *Eur. J. Control* 50:11–19
37. Ishizaki T, Ku R, Imura Ji. 2016. Clustered model reduction of networked dissipative systems. In *Proceedings of the 2016 American Control Conference*, pp. 3662–67. Piscataway, NJ: IEEE
38. Cheng X, Scherpen JMA, Besselink B. 2019. Balanced truncation of networked linear passive systems. *Automatica* 104:17–25
39. Bullo F. 2019. *Lectures on Network Systems*. Seattle, WA: Kindle Direct Publ.
40. Wu CW. 2007. *Synchronization in Complex Networks of Nonlinear Dynamical Systems*. Singapore: World Sci.
41. Fax JA, Murray RM. 2001. Graph Laplacians and stabilization of vehicle formations. *IFAC Proc. Vol.* 35(1):55–60
42. Mirzaev I, Gunawardena J. 2013. Laplacian dynamics on general graphs. *Bull. Math. Biol.* 75:2118–49
43. Dörfler F, Simpson-Porco JW, Bullo F. 2018. Electrical networks and algebraic graph theory: models, properties, and applications. *Proc. IEEE* 106:977–1005
44. van der Schaft AJ. 2017. Modeling of physical network systems. *Syst. Control Lett.* 101:21–27
45. Newman ME, Barabási ALE, Watts DJ. 2006. *The Structure and Dynamics of Networks*. Princeton, NJ: Princeton Univ. Press
46. Fagnani F, Frasca P. 2017. *Introduction to Averaging Dynamics over Networks*. Cham, Switz.: Springer
47. Rantzer A. 2015. Scalable control of positive systems. *Eur. J. Control* 24:72–80
48. Rantzer A, Valcher ME. 2018. A tutorial on positive systems and large scale control. In *2018 IEEE Conference on Decision and Control*, pp. 3686–97. Piscataway, NJ: IEEE
49. Ishizaki T, Kashima K, Girard A, Imura J, Chen L, Aihara K. 2015. Clustered model reduction of positive directed networks. *Automatica* 59:238–47
50. Johnson CR. 1990. *Matrix Theory and Applications*. Providence, RI: Am. Math. Soc.
51. Scardovi L, Sepulchre R. 2008. Synchronization in networks of identical linear systems. In *Proceedings of the 47th IEEE Conference on Decision and Control*, pp. 546–51. Piscataway, NJ: IEEE
52. Li Z, Duan Z, Chen G, Huang L. 2010. Consensus of multiagent systems and synchronization of complex networks: a unified viewpoint. *IEEE Trans. Circuits Syst. I* 57:213–24
53. Willems JC. 1976. Realization of systems with internal passivity and symmetry constraints. *J. Franklin Inst.* 301:605–21
54. Arcak M. 2007. Passivity as a design tool for group coordination. *IEEE Trans. Autom. Control* 52:1380–90
55. Chopra N. 2012. Output synchronization on strongly connected graphs. *IEEE Trans. Autom. Control* 57:2896–901
56. Bhat SP, Bernstein DS. 1999. Lyapunov analysis of semistability. In *Proceedings of the 1999 American Control Conference*, Vol. 3, pp. 1608–12. Piscataway, NJ: IEEE
57. Hui Q, Haddad WM, Bhat SP. 2009. Semistability, finite-time stability, differential inclusions, and discontinuous dynamical systems having a continuum of equilibria. *IEEE Trans. Autom. Control* 54:2465–70
58. Cheng X, Scherpen JMA. 2020. Novel Gramians for linear semistable systems. *Automatica* 115:108911
59. Cheng X, Scherpen JMA. 2017. A new controllability Gramian for semistable systems and its application to approximation of directed networks. In *Proceedings of the 56th IEEE Conference on Decision and Control*, pp. 3823–28. Piscataway, NJ: IEEE
60. Cheng X, Yu L, Ren D, Scherpen JMA. 2020. Reduced order modeling of diffusively coupled network systems: an optimal edge weighting approach. arXiv:2003.03559 [math.OC]
61. Farina L, Rinaldi S. 2011. *Positive Linear Systems: Theory and Applications*. New York: Wiley & Sons
62. Cheng X, Scherpen JMA. 2020. Clustering-based model reduction of Laplacian dynamics with weakly connected topology. *IEEE Trans. Autom. Control* 65:4393–99
63. Zhang S, Cao M, Camlibel MK. 2013. Upper and lower bounds for controllable subspaces of networks of diffusively coupled agents. *IEEE Trans. Autom. Control* 59:745–50
64. Aguilar CO, Gharesifard B. 2017. Almost equitable partitions and new necessary conditions for network controllability. *Automatica* 80:25–31

65. Mlinarić P, Grundel S, Benner P. 2015. Efficient model order reduction for multi-agent systems using QR decomposition-based clustering. In *Proceedings of the 54th IEEE Conference on Decision and Control*, pp. 4794–99. Piscataway, NJ: IEEE
66. Cheng X, Yu L, Scherpen JMA. 2019. Reduced order modeling of linear consensus networks using weight assignments. In *Proceedings of the 17th European Control Conference*, pp. 2005–10. Piscataway, NJ: IEEE
67. Zelazo D, Schuler S, Allgöwer F. 2013. Performance and design of cycles in consensus networks. *Syst. Control Lett.* 62:85–96
68. Leiter N, Zelazo D. 2017. Graph-based model reduction of the controlled consensus protocol. *IFAC-PapersOnLine* 50(1):9456–61
69. Cheng X, Kawano Y, Scherpen JMA. 2016. Graph structure-preserving model reduction of linear network systems. In *Proceedings of the 15th European Control Conference*, pp. 1970–75. Piscataway, NJ: IEEE
70. Cheng X, Scherpen JMA. 2016. Introducing network Gramians to undirected network systems for structure-preserving model reduction. In *Proceedings of the 55th IEEE Conference on Decision and Control*, pp. 5756–61. Piscataway, NJ: IEEE
71. Cheng X, Scherpen JMA. 2018. Clustering approach to model order reduction of power networks with distributed controllers. *Adv. Comput. Math.* 44:1917–39
72. Kawano Y, Besselink B, Scherpen JM, Cao M. 2020. Data-driven model reduction of monotone systems by nonlinear DC gains. *IEEE Trans. Autom. Control* 65:2094–106
73. Zha H, He X, Ding C, Gu M, Simon HD. 2002. Spectral relaxation for K-means clustering. In *Advances in Neural Information Processing Systems 14*, ed. TG Dietterich, S Becker, Z Ghahramani, pp. 1057–64. Cambridge, MA: MIT Press
74. Niazi MUB, Cheng X, Canudas-de-Wit C, Scherpen JMA. 2019. Structure-based clustering for model reduction of large-scale networks. In *Proceedings of the 58th IEEE Conference on Decision and Control*, pp. 5038–43. Piscataway, NJ: IEEE
75. Aoki M. 1968. Control of large-scale dynamic systems by aggregation. *IEEE Trans. Autom. Control* 13:246–53
76. Feliachi A, Bhurtun C. 1987. Model reduction of large-scale interconnected systems. *Int. J. Syst. Sci.* 18:2249–59
77. Ishizaki T, Imura JI. 2015. Clustered model reduction of interconnected second-order systems. *Nonlinear Theory Appl.* 6:26–37
78. Cheng X, Necoara I, Lupu D. 2020. A suboptimal h_2 clustering-based model reduction approach for linear network systems. In *Proceedings of the 17th European Control Conference*, pp. 1961–66. Piscataway, NJ: IEEE
79. Lacoste-Julien S. 2016. Convergence rate of Frank-Wolfe for nonconvex objectives. arXiv:1607.00345 [math.OC]
80. Dinh QT, Gumussoy S, Michiels W, Diehl M. 2011. Combining convex-concave decompositions and linearization approaches for solving BMIs, with application to static output feedback. *IEEE Trans. Autom. Control* 57:1377–90
81. Kokotovic PV, O'Malley RE Jr., Sannuti P. 1976. Singular perturbations and order reduction in control theory—an overview. *Automatica* 12:123–32
82. Anderson J, Chang YC, Papachristodoulou A. 2011. Model decomposition and reduction tools for large-scale networks in systems biology. *Automatica* 47:1165–74
83. Hancock EJ, Stan GB, Arpino JA, Papachristodoulou A. 2015. Simplified mechanistic models of gene regulation for analysis and design. *J. R. Soc. Interface* 12:20150312
84. Chow JH, Galarza R, Accari P, Price WW. 1995. Inertial and slow coherency aggregation algorithms for power system dynamic model reduction. *IEEE Trans. Power Syst.* 10:680–85
85. Bıyık E, Arcak M. 2008. Area aggregation and time-scale modeling for sparse nonlinear networks. *Syst. Control Lett.* 57:142–49
86. Romeres D, Dörfler F, Bullo F. 2013. Novel results on slow coherency in consensus and power networks. In *Proceedings of the 2013 European Control Conference*, pp. 742–47. Piscataway, NJ: IEEE
87. Dörfler F, Bullo F. 2013. Kron reduction of graphs with applications to electrical networks. *IEEE Trans. Circuits Syst. I* 60:150–63

88. Monshizadeh N, De Persis C, van der Schaft AJ, Scherpen JMA. 2017. A novel reduced model for electrical networks with constant power loads. *IEEE Trans. Autom. Control* 63:1288–99
89. Deng K, Mehta PG, Meyn SP. 2011. Optimal Kullback-Leibler aggregation via spectral theory of Markov chains. *IEEE Trans. Autom. Control* 56:2793–808
90. Martin N, Frasca P, Canudas-de-Wit C. 2018. MergeToCure: a new strategy to allocate cure in an epidemic over a grid-like network using a scale-free abstraction. *IFAC-PapersOnLine* 51(23):34–39
91. Cheng X, Scherpen JMA. 2017. Balanced truncation approach to linear network system model order reduction. *IFAC-PapersOnLine* 50(1):2451–56
92. Yu L, Cheng X, Scherpen JMA. 2019. h_2 suboptimal model reduction for second-order network systems. In *Proceedings of the 58th IEEE Conference on Decision and Control*, pp. 5062–67. Piscataway, NJ: IEEE
93. Yu L, Cheng X, Scherpen JMA. 2019. Synchronization preserving model reduction of multi-agent network systems by eigenvalue assignments. In *Proceedings of the 58th IEEE Conference on Decision and Control*, pp. 7794–99. Piscataway, NJ: IEEE
94. Fujimoto K, Scherpen JMA. 2010. Balanced realization and model order reduction for nonlinear systems based on singular value analysis. *SIAM J. Control Optim.* 48:4591–623
95. Scherpen JMA. 2011. Balanced realization, balanced truncation and the Hankel operator. In *The Control Handbook: Control System Advanced Methods*, ed. W Levine, pp. 4-1–24. Boca Raton, FL: CRC
96. Scarciotti G, Astolfi A. 2005. Nonlinear model reduction by moment matching. *Found. Trends Syst. Control* 4:224–409
97. Ishizaki T, Sandberg H, Johansson KH, Kashima K, Imura J, Aihara K. 2013. Structured model reduction of interconnected linear systems based on singular perturbation. In *Proceedings of the 2013 American Control Conference*, pp. 5524–29. Piscataway, NJ: IEEE
98. Jaoude DA, Farhood M. 2017. Balanced truncation model reduction of nonstationary systems interconnected over arbitrary graphs. *Automatica* 85:405–11
99. Necoara I, Ionescu TC. 2020. H_2 model reduction of linear network systems by moment matching and optimization. *IEEE Trans. Autom. Control* 65:5328–35
100. Cheng X, Scherpen JMA. 2018. Robust synchronization preserving model reduction of Lur'e networks. In *Proceedings of the 16th European Control Conference*, pp. 2254–59. Piscataway, NJ: IEEE
101. Forrow A, Woodhouse FG, Dunkel J. 2018. Functional control of network dynamics using designed Laplacian spectra. *Phys. Rev. X* 8:041043
102. Jongsma HJ, Trentelman HL, Camlibel KM. 2017. Model reduction of networked multiagent systems by cycle removal. *IEEE Trans. Autom. Control* 63:657–71

Annual Review of Control, Robotics, and Autonomous Systems

Volume 4, 2021

Contents

Errata

An online log of corrections to *Annual Review of Control, Robotics, and Autonomous Systems* articles may be found at http://www.annualreviews.org/errata/control