Model reduction of networked multi-agent systems by cycle removal

Jongsma, Hidde-Jan; Trentelman, Harry L.; Camlibel, Kanat M.

Published in:
IEEE Transactions on Automatic Control

DOI:
10.1109/TAC.2017.2731900

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2018

Model Reduction of Networked Multiagent Systems by Cycle Removal

Hidde-Jan Jongsma, Harry L. Trentelman, *Fellow, IEEE*, and Kanat M. Camlibel, *Member, IEEE*

***Abstract*—In this paper, we consider the problem of model reduction of consensus networks. We propose a new method of model reduction based on removing edges that close cycles in the network graph. The agent dynamics of the consensus network is given by a symmetric multivariable input-state-output system. In the network, the agents exchange relative output information with their neighbors. We assume that the network graph is connected, unweighted, and undirected. The network used to approximate the original system is defined on the same number of nodes as the original graph, but its edge set is a strict subset of the original edge set. Explicit expressions and upper bounds for the approximation errors are formulated in terms of the signed path vectors of the removed edges and the eigenvalues of the Laplacian matrices of the original and reduced network graphs.**

***Index Terms*—Multi-Agent systems, eigenvalues and eigenfunctions, network theory, reduced order systems, symmetric matrices, control systems.**

I. Introduction

COMPLEX networks, smart grids, distributed systems, and networked multiagent systems are subjects that have seen substantial amounts of attention in the past years. In the case of networked multiagent systems, the network consists of a group of dynamical input–output systems called *agents*. These agents are interconnected through a certain communication topology and exchange relative state or output information with their neighbors in the network. The interaction topology of the network is represented by a graph called the *network graph*. The nodes of the network graph represent the agents of the network and the edges encode the neighbors of each agent. Depending on the specific problem under investigation, the network graph might be weighted or unweighted, and directed or undirected. In this paper, we will only consider unweighted, undirected network graphs. A crucial object in the theory of networked systems is the *Laplacian matrix* of the network graph. Various properties of the network graph are expressed in terms of the spectrum of the Laplacian matrix, see [1] and [2]. Closely related to the Laplacian matrix is the *edge Laplacian matrix*, which plays an important role in this paper.

In the context of networks, the *consensus* problem is one of the most widely studied problems, see [3]–[6]. Results on the consensus problem have had applications in for instance optimization problems, distributed control and estimation. Similarly to consensus, some other well-studied directions are flocking, sensor placement, controllability analysis, and formation control. An overview of the state of the art can be found in, for instance, [7].

Even with the almost exponential increase in computing power that has been achieved in the past years and is predicted for the future, the analysis of the behavior of large-scale networked systems still remains a complicated task. This holds in particular for problems that scale in complexity as a power of the number of nodes or the number of edges in a graph. A helpful approach is to use simplified models to approximate the behavior of networked systems. This can ease some of the difficulties arising in analysis and controller design for these complex networked systems.

Applying existing well-known *model reduction* techniques, such as balanced truncation, Hankel-norm approximation, and Krylov projection, to the dynamical models of networked systems will in general result in the loss of the spatial structure of the network and other important properties. In order to circumvent this problem, in the past, *clustering-based algorithms* have been introduced in [8]–[10]. These clustering-based techniques cluster certain groups of agents and edges in the network graph, thus reducing the dimension of the network state, while retaining some of the original network structure. In [11], clustering techniques based on a specific class of graph partitions, called almost equitable partitions, were introduced.

In this paper, we will study an alternative method for model reduction of networks based on removing edges that close cycles in the network graph. This approach is inspired by results from [12] and [13] on the $\mathcal{H}_2$-performance of consensus networks when adding and removing cycles in the network graph.

By the removal of certain edges from the network graph, the complexity of the interconnection topology of the network is reduced. Thus, removing edges can indeed be considered as a method of model reduction or, more accurately, model simplification for networked multiagent systems.

The idea of removing edges from the graph in order to reduce complexity has been adopted before. For example in [14] and [15], in the theory of graph sparsification, algorithms

Manuscript received December 20, 2016; revised April 21, 2017; accepted June 15, 2017. Date of publication July 25, 2017; date of current version February 26, 2018. Recommended by Associate Editor Jun-ichi Imura. *(Corresponding author: Hidde-Jan Jongsma.)*

The authors are with the Johann Bernoulli Institute for Mathematics and Computer Science, University of Groningen, Groningen 9700AK, The Netherlands (e-mail: h.jongsma@rug.nl; h.l.trentelman@rug.nl; m.k.camlibel@rug.nl).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAC.2017.2731900

have been presented that compute a sparse graph, where the number of edges is of the same order as the number of nodes, thus approximating a full graph with many edges. These approximating graphs are close to the original graphs according to a certain metric, for instance in the sense of cut-similarity [16] or spectral-similarity [15]. While these algorithms provide accurate approximations of the original graph, they do not take dynamics on the nodes of the network into consideration. Consequently, these algorithms are not directly applicable to the problem of model reduction of networked multiagent systems. For resistor networks with scalar agent dynamics, the problem of model reduction by edge-removal was studied in [17] and [18]. In both papers, the nodes in the resistor networks are divided into a set of external nodes and one of internal nodes. In [17], an efficient method (reduceR) was introduced for finding an *equivalent* resistor network with the same number of external nodes, but with a reduced number of internal nodes. In [18], the number of external and internal nodes is kept the same for the reduced network. The approximation error that is considered is the worst case relative error between the steady-state voltages over the nodes of the original and reduced network. The upper bound on the approximation error given in [18] depends on the conductance matrices of both the original and the reduced network.

In the present paper, we will provide expressions for the $\mathcal{H}_2$-approximation error, expressed in terms of the signed path vectors of the removed edges, the nonzero Laplacian eigenvalues, and the associated eigenvectors of the edge Laplacian matrices of the original and reduced network graphs. In [19], preliminary results were achieved for the case that the agent dynamics is a single integrator and the networked system is a standard diffusively coupled network. In the present paper, we extend these results to the case that the agent dynamics is given by a general, symmetric linear input-state-output system.

This paper is organized as follows. In Section II, we review some relevant graph theory and investigate the notion of signed path vectors and the edge Laplacian matrix of a graph. In Section III, we discuss the dynamical model of a network with symmetric agent dynamics and investigate the dynamics of the reduced networked system. Then, Section IV contains the main results of this paper and expressions for the model reduction error when a single, or multiple edge-disjoint cycles are removed from the network graph. In Section V, we elaborate on the special case that the network graph has a star graph as a spanning tree. In Section VI, we briefly investigate the case that the agent dynamics is a single integrator with a drift term and how these results relate to the main results of [19]. In Section VII, we illustrate our results by means of a few numerical examples. Finally, we conclude the paper in Section VIII with concluding remarks.

II. Preliminaries

In this paper, we mostly use standard notation. Let $\mathbb{R}$ denote the field of real numbers, $\mathbb{R}^p$ the p-dimensional Euclidean space, and $\mathbb{R}^{p\times p}$ the space of $p\times p$ real matrices. The field of complex numbers is denoted by $\mathbb{C}$. Let I denote the identity matrix of appropriate dimension. The trace of a square matrix X is denoted by $\mathrm{tr}(X)$ and is the sum of its diagonal elements. Let $|\mathcal{E}|$ denote the cardinality of the set $\mathcal{E}$. The $\mathcal{H}_2$-norm of the transfer matrix of a system Σ is denoted by $\|\Sigma\|_2$. The Kronecker product of matrices A and B is written as $A\otimes B$. The diagonal matrix with $\alpha_1,\alpha_2,\ldots,\alpha_p$ on the diagonal is denoted by $\mathrm{diag}(\alpha_1,\alpha_2,\ldots,\alpha_p)$. If instead we have matrices $A_1,A_2,\ldots,A_p$, then $\mathrm{diag}(A_1,A_2,\ldots,A_p)$ denotes the matrix with $A_1,A_2,\ldots,A_p$ as diagonal blocks. For a full-rank matrix $B\in\mathbb{R}^{n\times m}$ with $n>m$, $B^\perp\in\mathbb{R}^{(n-m)\times n}$ is any full-rank matrix such that $B^\perp B=0$.

This paper considers networks whose interconnection topology is represented by an *undirected graph*. An undirected graph is a pair $G=(V,\mathcal{E})$, which consists of a set of *nodes* $V=\{1,2,\ldots,p\}$ and a set of *edges* $\mathcal{E}$, which is a subset of $\{\{i,j\}\,|\,i,j\in V\}$, the set of all unordered pairs. These edges can, for example, represent the communication channels between unordered pairs of nodes, see [1] and [20]. While the network graph is undirected, in this paper an arbitrary orientation will be assigned to each edge. The specific orientation that is chosen is not important. We call two distinct nodes *adjacent* if they are connected by an edge. A sequence of nodes in which sequential nodes are adjacent is called a *path*. The length of a path is the number of edges that have been traversed while following the path. A path is called a *cycle* if the first and last node in the path coincide, and all other nodes in the path occur only once. In a *connected graph*, there exists a path from every single node to every other node. Throughout the paper, we will only consider networks that have connected network graphs.

A graph $G'=(V',\mathcal{E}')$ is called a *subgraph* of G, if $V'\subset V$ and $\mathcal{E}'\subset\mathcal{E}$. This is denoted by $G'\subset G$. A *spanning tree* $\mathcal{T}=(V,\mathcal{E}_t)$ of the graph $G=(V,\mathcal{E})$ is any connected and cycle-free subgraph of G. Since any spanning tree of G has p nodes and is a tree, we have $|\mathcal{E}_t|=p-1$. Let $\mathcal{E}_t$ contain the edges of a spanning tree $\mathcal{T}$. The set containing the edges in G that are not in $\mathcal{T}$ is then $\mathcal{E}_c=\mathcal{E}\backslash\mathcal{E}_t$. Therefore, these edges close the cycles in G. Of course, which edges are in $\mathcal{E}_c$ depends on the specific choice of $\mathcal{T}$. The *complement* of a given graph G is denoted by $\overline{G}=(V,\overline{\mathcal{E}})$ and is the graph defined on the same node set V with the complementary edge set $\overline{\mathcal{E}}=\{\{i,j\}\,|\,i,j\in V\}\backslash\mathcal{E}$.

After fixing the orientation of the edges in G, the *incidence matrix* of G is the $|V|\times|\mathcal{E}|$-dimensional matrix E, where $E_{ik}=1$ if node i is the initial node of the edge k, $E_{ik}=-1$ if it is the terminal node, and $E_{ik}=0$ otherwise, see, e.g., [20]. The incidence matrix depends on the numbering and the orientation of the edges in $\mathcal{E}$.

For a specific spanning tree $\mathcal{T}$ of G, let E_t denote the columns of the incidence matrix E that correspond to the edges in $\mathcal{E}_t$. Let E_c contain the columns that correspond to the edges in $\mathcal{E}_c$, that close the cycles of G. After a suitable renumbering of the edges in $\mathcal{E}$, we then have

$$E=(E_t\ E_c).$$

Next, we discuss *signed path vectors*, which are representations for paths in G, see, e.g., [13]. Recall that, we have assigned a specific but otherwise arbitrary orientation to each edge in the undirected graph G. A signed path vector $v\in\mathbb{R}^{|\mathcal{E}|}$ then encodes how a given path traverses the oriented edges in $\mathcal{E}$. The ith element of v is 1 if the path follows the edge $e_i\in\mathcal{E}$ in

the positive direction, i.e., from the initial to the terminal node. The ith element is -1 if edge e_i is traversed in the opposite direction, and it is 0 if the path does not follow the edge at all. It is easily seen that the length of the path, denoted $l(v)$, is given by $l(v) = v^T v$. The next lemma plays an important role in this paper, and follows from [13, Th. 1]. The lemma implies that the columns of E_c are linear combinations of the columns of E_t.

Lemma 1: For a connected graph G and spanning tree $\mathcal{T}$, let E_t correspond to the edges in $\mathcal{E}_t$ and E_c to the edges in $\mathcal{E}_c$. Define $T = (E_t^T E_t)^{-1} E_t^T E_c$. Then T satisfies $E_t T = E_c$.

The proof of this lemma is omitted here. We can interpret the columns of $T = (c_1 \ \cdots \ c_{|\mathcal{E}_c|})$ as signed path vectors which express the edges in $\mathcal{E}_c$ in terms of paths in the spanning tree $\mathcal{T}$. Indeed, let the ith column $E_t c_i$ of E_c represent the (cycle closing) edge from node u to node v. Then c_i is the signed path vector of the (alternative) path from u to v in the graph $\mathcal{T}$. An element of c_i is 1 if that path traverses the corresponding edge in $\mathcal{T}$ in the chosen positive direction, -1 if in the opposite direction, and 0 if the edge is not part of the path. In this paper, we often refer to c_i as the ith cycle of the graph G. Obviously, the length of this cycle is equal to $c_i^T c_i + 1$.

Following [13], we have the following definitions for *edge-disjoint* and *correlated* cycles.

Definition 2: Two cycles are called *edge-disjoint*, or *independent* if they have no edges in common. Two cycles are called *correlated*, or *dependent* if they are not edge-disjoint.

The matrix T encodes the above mentioned features of the cycles in the graph. This is captured by the following lemma, for which the proof can be derived from that of [13, Proposition 1].

Lemma 3: The matrix $T^T T$ encodes the following information of the cycles in G:

1) $(T^T T)_{ii} = c_i^T c_i = l(c_i) - 1$.
2) $(T^T T)_{ij} = c_i^T c_j = 0$ if and only if c_i and c_j are edge-disjoint.

We will further use the fact that the matrix $I + TT^T$ is invertible.

Example 4: As an example, consider the graph in Fig. 1. The graph is defined on seven nodes and contains the star graph S_6 as a subgraph. If we choose as a spanning tree $\mathcal{T} = S_6$, we see that the matrices E_t, E_c, T, and $T^T T$ are given by

$$E_t = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix}, E_c = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ -1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & -1 \\ 0 & 1 \end{pmatrix}$$

$$T = \begin{pmatrix} -1 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & -1 \end{pmatrix}, \quad T^T T = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}.$$

Indeed, it is easily verified that $E_c = E_t T$. Note that the length of the cycles and the fact that they are edge-disjoint can be obtained from the columns of T. Each column has two nonzero elements, so the length of each cycle is 3. Furthermore, the columns of T are orthogonal to each other, so the cycles are edge-disjoint.

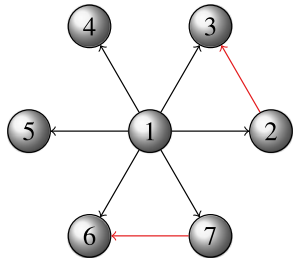


Fig. 1. Graph on seven nodes with star graph S_6 as spanning tree.

An important object in graph theory is the *Laplacian matrix*. The Laplacian matrix L of an undirected graph G is a positive semidefinite matrix in $\mathbb{R}^{p\times p}$. It can be defined in terms of the incidence matrix E as

$$L = EE^T.$$

The Laplacian matrix does not depend on the chosen numbering and orientation of the edges. It has rank $p-1$ if and only if the graph G is connected. In that case, L has a simple eigenvalue at 0. Obviously, the remaining eigenvalues of L are real and positive. They can therefore be ordered in increasing fashion as

$$0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_p.$$

Closely related to the Laplacian matrix is the *edge Laplacian matrix*, see [12], which is defined as

$$L_e = E^T E.$$

The edge Laplacian is a symmetric positive semidefinite matrix in $\mathbb{R}^{|\mathcal{E}|\times|\mathcal{E}|}$. It has the same nonzero eigenvalues as the Laplacian. For a tree graph on p nodes, the edge Laplacian L_e is in $\mathbb{R}^{(p-1)\times(p-1)}$, and is a positive definite matrix. Since it is symmetric, there exists an orthogonal transformation U that brings L_e to diagonal form:

$$U^T L_e U = \Lambda = \operatorname{diag}(\lambda_2, \lambda_3, \ldots, \lambda_p). \tag{1}$$

Here, the columns of $U = (x_2 \ x_3 \ \ldots \ x_p)$ are orthonormal eigenvectors of L_e.

III. Consensus Model

In the standard consensus model, the agents of the network are single integrators that exchange their *relative state* with their neighbors in the network graph. In [19], model reduction for the standard consensus model was investigated. In the single integrator case, the network dynamics is given by

$$\dot{x} = -Lx$$

where $x \in \mathbb{R}^p$ contains the states of all the agents in the network. Here, L is the Laplacian matrix of the network graph G. In [19], the behavior of the network is approximated by a reduced network with network graph $\hat{G}$, with $\hat{G}$ a connected subgraph of G. After reduction, the network dynamics is

given by

$$\dot{\hat{x}} = -\hat{L}\hat{x}$$

with $\hat{L}$ the Laplacian of $\hat{G}$ and $\hat{x}$ again in $\mathbb{R}^p$. In [19], we also derived expressions and bounds for the approximation error.

In the present paper, we will consider the case that the agent dynamics no longer consists of a single integrator system, but is given by the symmetric input-state-output system

$$\dot{x}_i = Ax_i + Bu_i, \quad y_i = B^T x_i \tag{2}$$

where $x_i \in \mathbb{R}^n$, $u_i, y_i \in \mathbb{R}^m$, and $A = A^T$. Symmetric input-state-output systems appear, for example, in electrical and power networks, and chemical reactions networks. In particular, they appear in mechanical systems with only potential energy or only kinetic energy, and in electrical systems with only electrical energy or only magnetic energy (e.g., *RL* or *RC* circuits), see, e.g., [21] or [22].

The agents are connected through the diffusive coupling

$$u_i = -\sum_{j\in\mathcal{N}_i} y_i - y_j = -B^T \sum_{j\in\mathcal{N}_i} x_i - x_j \tag{3}$$

where $\mathcal{N}_i$ is the neighboring set of agent i. The network dynamics is then represented by the autonomous system

$$\dot{x} = (I \otimes A - L \otimes BB^T)x. \tag{4}$$

The vector $x \in \mathbb{R}^{pn}$ contains the states of all agents and L is the Laplacian matrix of the network graph $G = (V, E)$. It will be a standing assumption in this paper that the network (4) reaches consensus, i.e., $x_i - x_j \to 0$ as $t \to \infty$ for all i, j. It is a well-known fact, see, e.g., [5], that (4) reaches consensus if and only if the matrices $A - \lambda_i BB^T$ are Hurwitz for all nonzero eigenvalues $\lambda_2, \lambda_3, \ldots, \lambda_p$ of the Laplacian matrix L.

The idea of the model reduction procedure proposed in this paper is now to remove selected edges from the underlying network graph. This then yields a reduced network, and the corresponding *reduced graph* $G' = (V, \mathcal{E}') \subset G$ is the subgraph of G constructed by removing these edges in the original graph. It has the same nodes as G, and the edge set $\mathcal{E}'$ is a strict subset of $\mathcal{E}$. The dynamics of the reduced network is given by

$$\dot{x}' = (I \otimes A - L' \otimes BB^T)x' \tag{5}$$

where L' denotes the Laplacian of G'. Since the reduced network has the same number of agents, the state-space dimension remains equal to pn. The reduced network will reach consensus if and only if the nonzero eigenvalues $\omega_2, \omega_3, \ldots, \omega_p$ of the reduced Laplacian L' satisfy $A - \omega_i BB^T$ is Hurwitz for $i = 2, 3, \ldots, p$. In the sequel, we will formulate necessary and sufficient conditions for consensus of both the original and reduced network. For this, we will use Finsler's Lemma, see, e.g., [23, Th. 2.3.10], which, for a given symmetric A, gives necessary and sufficient conditions for the existence of $\mu \in \mathbb{R}$ such that $A - \mu BB^T$ is Hurwitz, and a lower bound for such μ.

First note that the algebraic connectivity does not increase with the removal of edges from a graph, see [13] and [20]. This implies that for every reduced network, we have $\omega_2 \leq \lambda_2$. Combining this fact with Finsler's Lemma, we immediately obtain the following necessary and sufficient conditions for consensus of both the original and reduced network.

Lemma 5: Assume that $m < n$ and that B has full column rank. Then the original and reduced network reach consensus if and only if $B^\perp AB^{\perp T} < 0$ and

$$\omega_2 > \lambda_{\max}(B^+(A - AB^{\perp T}(B^\perp AB^{\perp T})^{-1}B^\perp A)B^{+T}). \tag{6}$$

Here, B^+ is any left inverse of B. If B is square nonsingular, then consensus is reached for both networks if and only if

$$\omega_2 > \lambda_{\max}(B^{-1}AB^{-T}).$$

For the case that B does not have full column rank, analogous conditions can be obtained, see [23]. The details are omitted here. In the remainder of this paper, we will assume that both the original and the reduced network reach consensus.

Remark 6: Lemma 5 gives a *necessary and sufficient condition* for the reduced network to reach consensus: the lemma expresses this as a lower bound condition on ω_2. If the original network reaches consensus, then already the condition $B^\perp AB^{\perp T} < 0$ holds. The second inequality $\omega_2 > \lambda_{\max}(B^+(A - AB^{\perp T}(B^\perp AB^{\perp T})^{-1}B^\perp A)B^{+T})$ then holds if and only if the reduced network reaches consensus. If, after removing certain edges, ω_2 would turn out not to satisfy this inequality, then removing these particular edges should not be done, because consensus is lost. In other words, the inequality in Lemma 5 gives a criterion to check whether certain edges can be removed without losing consensus.

We will now make the idea of reducing the network by removing edges explicit. First, we assign a virtual input d and an output z to the original network (4). This is done as follows: given the network graph $G = (V, \mathcal{E})$, we first choose a particular spanning tree $\mathcal{T} = (V, \mathcal{E}_t)$, with incidence matrix E_t, and as output we consider the *disagreement over the spanning tree*, i.e., $z = (E_t^T \otimes I)x$. Thus, we associate with the original network (4), the two-port consensus model

$$\begin{aligned} \dot{x} &= (I \otimes A - L \otimes BB^T)x + (I \otimes I)d \\ z &= (E_t^T \otimes I)x. \end{aligned} \tag{7}$$

Remark 7: A usual choice of output in consensus models is $z = (E^T \otimes I)x$, with E the incidence matrix of G. This output represents the disagreements between all neighboring nodes. Note from the previous that, after choosing a spanning tree $\mathcal{T}$, there exists a matrix T such that $E = (E_t \; E_t T)$. Thus, actually

$$z = \begin{pmatrix} (E_t^T \otimes I)x \\ (T^T E_t^T \otimes I)x \end{pmatrix}$$

which shows that already $(E_t^T \otimes I)x$ contains all relevant information on the disagreements; the second part of the output vector is just a linear transformation of the first part. This observation is the motivation for us to choose as output $z = (E_t^T \otimes I)x$ in (7). A similar approach was taken in [12] and [13].

The impulse response of the system (7) quantifies the initial condition response of (4) to the vector of disagreements over the spanning tree $\mathcal{T}$. While the state space representation (7) is controllable, it is not a *minimal representation* since it is not

observable. Indeed, let $\mathbb{1}$ be the vector of ones, and let y and λ be such that y is nonzero and $Ay = \lambda y$, then

$$(I \otimes A - L \otimes BB^T)\mathbb{1} \otimes y = \lambda(\mathbb{1} \otimes y), \quad (E_t^T \otimes I)\mathbb{1} \otimes y = 0.$$

Utilizing ideas from [13], we construct a minimal representation for (7) by taking $\xi = (E_t^T \otimes I)x$ as the new state. Indeed, using the fact that $E_c = E_t T$ [where $T = (c_1 \ \cdots \ c_r)$ encodes the cycles in the graph] we obtain the following minimal representation Σ of (7):

$$\begin{aligned} \dot{\xi} &= (I \otimes A - L_e(I + TT^T) \otimes BB^T)\xi + (E_t^T \otimes I)d \\ z &= \xi. \end{aligned} \tag{8}$$

The matrix $L_e(I + TT^T)$ appearing in (8) is called the *essential edge Laplacian* of G, see [13]. The matrix L_e appearing in (8) is the edge Laplacian of the spanning tree $\mathcal{T}$, satisfying

$$L_e = E_t^T E_t.$$

Now, the system (8) is a minimal state space representation of the original network (7). It is straightforward to verify that the eigenvalues of $I \otimes A - L_e(I + TT^T) \otimes BB^T$ coincide with the eigenvalues of the matrices $A - \lambda_i BB^T$ for $i = 2, 3, \ldots, p$. Since it is assumed that the system reaches consensus, the matrix $I \otimes A - L_e(I + TT^T) \otimes BB^T$ is Hurwitz, and for d equal to zero, $\xi \to 0$ as $t \to \infty$. The procedure of model reduction of the original network (7) that we propose is now to remove selected cycles from the network graph. This is exactly achieved by removing the columns corresponding to these cycles from the matrix T in the minimal realization (8) of the network. In this way we obtain a reduced order network having a T matrix, say $\hat{T}$, with a reduced number of columns. This reduced order network $\hat{\Sigma}$ is represented by

$$\begin{aligned} \dot{w} &= (I \otimes A - L_e(I + \hat{T}\hat{T}^T) \otimes BB^T)w + (E_t^T \otimes I)d \\ \hat{z} &= w. \end{aligned} \tag{9}$$

Again, by our standing assumption we have that the reduced network achieves consensus and that the matrix $I \otimes A - L_e(I + \hat{T}\hat{T}^T) \otimes BB^T$ is also Hurwitz. Obviously, the extreme case is to remove *all* cycles from the network, which is achieved by setting $\hat{T} = 0$.

We summarize the above in the following reduction procedure:

Input: Network with graph G, incidence matrix E, Laplacian L,
Step 1: Choose a spanning tree $\mathcal{T}$ with incidence matrix E_t,
Step 2: Choose a list of to be removed cycle-closing edges,
Step 3: Renumber the edges in G so that its incidence matrix is $(E_t \ E_c)$ and $E_c = (\hat{E}_c \ \bar{E}_c)$, where the columns of $\bar{E}_c$ correspond to the to-be-removed edges,
Output: Network with reduced graph $\hat{G}$, defined by the incidence matrix $\hat{E} = (E_t \ \hat{E}_c)$, Laplacian $\hat{L} = E_t E_t^T + \hat{E}_c \hat{E}_c^T$.

As error, we take the $\mathcal{H}_2$-norm of the difference of the transfer matrices of the original two-port consensus network (7) and the reduced network, which is represented by

$$\begin{aligned} \dot{\hat{x}} &= (I \otimes A - \hat{L} \otimes BB^T)\hat{x} + (I \otimes I)d \\ \hat{z} &= (E_t^T \otimes I)\hat{x}. \end{aligned} \tag{10}$$

Clearly, these transfer matrices coincide with those of Σ and $\hat{\Sigma}$ given by (8) and (9), respectively, and this will be exploited heavily in the sequel.

Remark 8: We note that a different choice of orientations of the edges in G will in general lead to different representations (7) and (10) of the original and reduced network. Obviously, in both representations the state equations remain the same (by the way the Laplacians are formed), but the components of z and $\hat{z}$ may differ in sign. However, if a component of z changes sign, the same will happen with the corresponding component in $\hat{z}$. Thus, the $\mathcal{H}_2$-norm of the system mapping d to $z - \hat{z}$ will be independent of the orientation chosen.

In the following section, we will derive explicit expressions and upper bounds for the error between Σ and $\hat{\Sigma}$.

IV. Model Reduction Error

In this section, we will first establish expressions for the error in the case that from the original network graph we remove *all* edges that close cycles. The reduced network is then given by (9), with $\hat{T} = 0$, equivalently by (10) with $\hat{L} = E_t E_t^T$, the Laplacian of the chosen spanning tree. As a measure for the accuracy of the approximation (9), we take the $\mathcal{H}_2$-norm of the error system. The error system maps the input d to the error $e = \hat{z} - z$. The error dynamics is given by

$$\begin{pmatrix} \dot{w} \\ \dot{\xi} \end{pmatrix} = A_e \begin{pmatrix} w \\ \xi \end{pmatrix} + B_e d, \quad e = C_e \begin{pmatrix} w \\ \xi \end{pmatrix} \tag{11}$$

where the system matrices in (11) are given by

$$A_e := \begin{pmatrix} I \otimes A - L_e \otimes BB^T & 0 \\ 0 & I \otimes A - L_e(I + TT^T) \otimes BB^T \end{pmatrix}$$

$$B_e := \begin{pmatrix} E_t^T \otimes I \\ E_t^T \otimes I \end{pmatrix}, \ C_e := \begin{pmatrix} I \otimes I & -I \otimes I \end{pmatrix}.$$

For the system to have a well-defined $\mathcal{H}_2$-norm, we need the matrix A_e to be Hurwitz. Since, by assumption, the matrices $I \otimes A - L_e(I + TT^T) \otimes BB^T$ and $I \otimes A - L_e \otimes BB^T$ are both Hurwitz, this is clearly the case. It is well-known, see, e.g., [24], that the squared $\mathcal{H}_2$-norm $\|\Sigma - \hat{\Sigma}\|_2^2$ of the error system equals

$$\|\Sigma - \hat{\Sigma}\|_2^2 = \mathrm{tr}(C_e X C_e^T) \tag{12}$$

where the symmetric positive semidefinite matrix X is the unique solution to the Lyapunov equation

$$A_e X + X A_e^T + B_e B_e^T = 0. \tag{13}$$

By partitioning X as

$$X = \begin{pmatrix} X_1 & X_{12} \\ X_{12}^T & X_2 \end{pmatrix}$$

with $X_1, X_{12}, X_2 \in \mathbb{R}^{(p-1)n\times(p-1)n}$, it follows that (12) can be rewritten as

$$\|\Sigma - \hat{\Sigma}\|_2^2 = \operatorname{tr}(X_1) + \operatorname{tr}(X_2) - 2\operatorname{tr}(X_{12}). \tag{14}$$

It is easily seen that X solves Lyapunov equation (13) if and only if the matrices X_1, X_{12}, and X_2 solve the following equations:

$$\begin{aligned} 0 &= (I\otimes A - L_e\otimes BB^T)X_1 \\ &\quad + X_1(I\otimes A - L_e\otimes BB^T) \\ &\quad + L_e\otimes I \end{aligned} \tag{15a}$$

$$\begin{aligned} 0 &= (I\otimes A - L_e\otimes BB^T)X_{12} \\ &\quad + X_{12}(I\otimes A - (I+TT^T)L_e\otimes BB^T) \\ &\quad + L_e\otimes I \end{aligned} \tag{15b}$$

$$\begin{aligned} 0 &= (I\otimes A - L_e(I+TT^T)\otimes BB^T)X_2 \\ &\quad + X_2(I\otimes A - (I+TT^T)L_e\otimes BB^T) \\ &\quad + L_e\otimes I. \end{aligned} \tag{15c}$$

We will now first compute X_1. Let U be the orthogonal matrix that diagonalizes the edge Laplacian L_e of the spanning tree $\mathcal{T}$. Then, we have $U^T L_e U = \Omega = \operatorname{diag}(\omega_2, \omega_3, \ldots, \omega_p)$, where $\omega_2, \omega_3, \ldots, \omega_p$ are the nonzero eigenvalues of the Laplacian matrix of the spanning tree. We premultiply (15a) by $U^T\otimes I$ and postmultiply by $U\otimes I$. We then substitute $Y_1 = (U^T\otimes I)X_1(U\otimes I)$ into the resulting expression to obtain

$$\begin{aligned} &(I\otimes A - \Omega\otimes BB^T)Y_1 \\ &\quad + Y_1(I\otimes A - \Omega\otimes BB^T) + \Omega\otimes I = 0. \end{aligned} \tag{16}$$

This equation is solved by taking

$$Y_1 = \operatorname{diag}(Z_2, Z_3, \ldots, Z_p)$$

where Z_i for $i = 2, 3, \ldots, p$ solves

$$Z_i(A - \omega_i BB^T) + (A - \omega_i BB^T)Z_i + \omega_i I = 0.$$

Hence Z_i is the observability Gramian of the system

$$\dot{x} = (A - \omega_i BB^T)x + E^T d, \quad z = \sqrt{\omega_i}x$$

and we can compute

$$Z_i = \frac{\omega_i}{2}(\omega_i BB^T - A)^{-1}.$$

Note that, by assumption, the reduced network reaches consensus, and equivalently, $A - \omega_i BB^T$ is Hurwitz, so its inverse exists. Finally, since $X_1 = (U\otimes I)Y_1(U^T\otimes I)$, we obtain the following expression for X_1:

$$X_1 = (U\otimes I)\operatorname{diag}(Z_2, Z_3, \ldots, Z_p)(U^T\otimes I). \tag{17}$$

Next, we will compute X_2. First, we symmetrize (15c) by pre- and postmultiplying (15c) by $(I+TT^T)^{\frac{1}{2}}\otimes I$. We then substitute $Y_2 = ((I+TT^T)^{\frac{1}{2}}\otimes I)X_2((I+TT^T)^{\frac{1}{2}}\otimes I)$ to obtain

$$\begin{aligned} &[I\otimes A - (I+TT^T)^{\frac{1}{2}}L_2(I+TT^T)^{\frac{1}{2}}\otimes BB^T]Y_2 \\ &+ Y_2[I\otimes A - (I+TT^T)^{\frac{1}{2}}L_e(I+TT^T)^{\frac{1}{2}}\otimes BB^T] \\ &+ (I+TT^T)^{\frac{1}{2}}L_e(I+TT^T)^{\frac{1}{2}}\otimes I = 0. \end{aligned}$$

The expression above can be brought to block diagonal form by the transformation $V\otimes I$, where V is an orthogonal matrix that brings the matrix

$$\bar{L}_e = (I+TT^T)^{\frac{1}{2}}L_e(I+TT^T)^{\frac{1}{2}} \tag{18}$$

to diagonal form:

$$\begin{aligned} \Lambda := V^T\bar{L}_e V &= V^T(I+TT^T)^{\frac{1}{2}}L_e(I+TT^T)^{\frac{1}{2}}V \\ &= \operatorname{diag}(\lambda_2, \lambda_3, \ldots, \lambda_p). \end{aligned}$$

The matrix $\bar{L}_e$ is similar to the matrix $L_e(I+TT^T)$, hence its eigenvalues coincide with the nonzero eigenvalues $\lambda_2, \lambda_3, \ldots, \lambda_p$ of the Laplacian matrix L of our original network graph G. Applying the transformation $V\otimes I$, we obtain

$$(I\otimes A - \Lambda\otimes BB^T)\tilde{Y}_2 + \tilde{Y}_2(I\otimes A - \Lambda\otimes BB^T) + \Lambda\otimes I = 0 \tag{19}$$

where $\tilde{Y}_2 = (V^T\otimes I)Y_2(V\otimes I)$. Solving for $\tilde{Y}_2$ gives

$$\tilde{Y}_2 = \operatorname{diag}(Q_2, Q_3, \ldots, Q_p)$$

where Q_i for $i = 2, 3, \ldots, p$ solves

$$Q_i(A - \lambda_i BB^T) + (A - \lambda_i BB^T)Q_i + \lambda_i I = 0.$$

Hence, Q_i is the observability Gramian of the system

$$\dot{x} = (A - \lambda_i BB^T)x + E^T d, \quad z = \sqrt{\lambda_i}x.$$

An explicit solution for Q_i is then given by

$$Q_i = \frac{\lambda_i}{2}(\lambda_i BB^T - A)^{-1}.$$

Here, again, the inverse exists since we assume that the network reaches consensus, equivalently, $A - \lambda_i BB^T$ is Hurwitz. Finally, we obtain the following expression for X_2:

$$\begin{aligned} X_2 = \left((I+TT^T)^{-\frac{1}{2}}V\otimes I\right)&\operatorname{diag}(Q_2, Q_3, \ldots, Q_p) \\ &\times \left(V^T(I+TT^T)^{-\frac{1}{2}}\otimes I\right). \end{aligned} \tag{20}$$

The solution X_{12} to the Sylvester equation (15b) can be given in terms of the eigenvectors and eigenvalues of $I\otimes A - L_e\otimes BB^T$. From [25, Th. 6.5, p. 178], we derive the following lemma:

Lemma 9: Let $M = M^T$, N, and C be square matrices in $\mathbb{R}^{q\times q}$. Furthermore, let x_i be a normalized eigenvector of M corresponding to eigenvalue μ_i $(i = 1, 2, \ldots, q)$. If $\sigma(M)$ and $\sigma(-N)$ are disjoint, then the Sylvester equation

$$MX + XN = C$$

has a unique solution given by

$$X = \sum_{i=1}^{q} x_i x_i^T C(\mu_i I + N)^{-1}.$$

A proof follows immediately from the proof of the more general case in [25]. Using Lemma 9, we have the following expression for X_{12}:

$$X_{12} = \sum_{i,j} (x_i \otimes y_{ij})(x_i \otimes y_{ij})^T (-L_e \otimes I) \times \left(\sigma_{ij} I \otimes I + I \otimes A - (I + TT^T) L_e \otimes BB^T\right)^{-1} \quad (21)$$

where $i = 2, 3, \ldots, p$ and $j = 1, 2, \ldots, n$. The vectors x_i are normalized eigenvectors of L_e, σ_{ij} is the jth eigenvalue of the matrix $A - \omega_i BB^T$, and y_{ij} a corresponding normalized eigenvector. To see that the vectors $x_i \otimes y_{ij}$ are indeed eigenvectors of the matrix $I \otimes A - L_e \otimes BB^T$, first apply the transformation $U \otimes I$ to the matrix, where $U = (x_2 \ x_3 \ \ldots \ x_p)$ is the orthogonal matrix that diagonalizes L_e. We obtain

$$(U^T \otimes I)(I \otimes A - L_e \otimes BB^T)(U \otimes I) = I \otimes A - \Omega \otimes BB^T.$$

This yields a block-diagonal matrix with $A - \omega_i BB^T$ for $i = 2, 3, \ldots p$ on the diagonal. Since A is symmetric, the matrices $A - \omega_i BB^T$ are also symmetric. Thus, there exists an orthogonal matrix $Y_i = (y_{i1} \ y_{i2} \ \ldots \ y_{in})$ such that

$$Y_i^T (A - \omega_i BB^T) Y_i = \operatorname{diag}(\sigma_{i1}, \sigma_{i2}, \ldots, \sigma_{in}).$$

It is easily seen that the vectors $x_i \otimes y_{ij}$ are eigenvectors and the σ_{ij}s are the eigenvalues of $I \otimes A - L_e \otimes BB^T$. Next, we will prove a lemma that will be instrumental in proving one of our main results.

Lemma 10: Let x_i be an eigenvector of L_e with corresponding eigenvalue ω_i and let y_{ij} be an eigenvector of the matrix $A - \omega_i BB^T$ corresponding to eigenvalue σ_{ij}. Denote

$$M_{ij} = -\sigma_{ij} L_e^{-1} \otimes I - L_e^{-1} \otimes A + I \otimes BB^T.$$

We then have

$$M_{ij}^{-1}(x_i \otimes y_{ij}) = -\frac{\omega_i}{2\sigma_{ij}}(x_i \otimes y_{ij}).$$

Proof: First, we have

$$M_{ij}^{-1} = (U \otimes I)\left(-\sigma_{ij}\Omega^{-1} \otimes I - \Omega^{-1} \otimes A + I \otimes BB^T\right)^{-1} \times (U^T \otimes I). \quad (22)$$

The matrix $-\sigma_{ij}\Omega^{-1} \otimes I - \Omega^{-1} \otimes A + I \otimes BB^T$ is a block-diagonal matrix, whose diagonal blocks are given by $\frac{1}{\omega_i}(-\sigma_{ij} I - A + \omega_i BB^T)$, $i = 2, 3, \ldots, p$. It is easily checked that multiplying $x_i \otimes y_{ij}$ with M_{ij}^{-1} yields

$$M_{ij}^{-1}(x_i \otimes y_{ij}) = x_i \otimes \left(\frac{1}{\omega_i}(-\sigma_{ij} I - A + \omega_i BB^T)\right)^{-1} y_{ij}.$$

Furthermore, since y_{ij} is an eigenvector of the matrix $A - \omega_i BB^T$ corresponding to eigenvalue σ_{ij}, we have

$$(\sigma_{ij} I + A - \omega_i BB^T)^{-1} y_{ij} = \frac{1}{2\sigma_{ij}} y_{ij}.$$

Combining the two expressions above completes the proof. ■

The next theorem now gives an explicit expression for the approximation error in the special case that the network consists of a tree combined with one single extra edge. The approximation error is given in terms of eigenvectors and the eigenvalues of matrices related to the original and reduced systems and the signed path vector of the removed cycle.

Theorem 11: Let Σ be a network consisting of the tree $\mathcal{T}$ combined with a single extra edge $e \in \overline{\mathcal{E}}_t$, with graph $G = (V, \mathcal{E}_t \cup e)$, such that the graph has exactly one cycle c with length $l(c)$. Let $\hat{\Sigma}$ be the network obtained by removing the edge e. Then, the approximation error is given by

$$\|\Sigma - \hat{\Sigma}\|_2^2 = \frac{1}{2}\sum_{i,j} (x_i^T c)^2 \frac{\omega_i^2}{\sigma_{ij}^2} y_{ij}^T B \left(I + \sum_{k=2}^{p} \omega_k (x_k^T c)^2 G_k(-\sigma_{ij})\right)^{-1} B^T y_{ij} + \frac{1}{2}\sum_{i=2}^{p} \left(1 - \frac{(v_i^T c)^2}{l(c)}\right) \lambda_i \operatorname{tr}(\lambda_i BB^T - A)^{-1} - \frac{1}{2}\sum_{i=2}^{p} \omega_i \operatorname{tr}(\omega_i BB^T - A)^{-1}. \quad (23)$$

Here, i runs from $2, 3, \ldots, p$ and j from $1, 2, \ldots, n$. In the above expression:

1) ω_i are the nonzero Laplacian eigenvalues of $\mathcal{T}$ and x_i denote corresponding normalized eigenvectors of L_e.
2) λ_i are the nonzero Laplacian eigenvalues of G and v_i denote corresponding normalized eigenvectors of $\bar{L}_e = (I + cc^T)^{\frac{1}{2}} L_e (I + cc^T)^{\frac{1}{2}}$.
3) σ_{ij} and y_{ij} denote the jth eigenvalue and corresponding normalized eigenvector of the matrix $A - \omega_i BB^T$.
4) $G_k(s)$ denotes the transfer matrix of the system

$$\dot{x} = (A - \omega_k BB^T)x + Bu, \ y = B^T x. \quad (24)$$

Remark 12: To compute (23), we first need to compute the eigenvalues and corresponding eigenvectors of both L_e and $\bar{L}_e$. These can be computed using, for instance, the Jacobi eigenvalue algorithm, which is a fast iterative algorithm that computes all eigenpairs of a symmetric $p \times p$ matrix Q in $O(p^3)$ time, see, e.g., [26, Ch. 9]. Next, for $i = 2, 3, \ldots, p$, we need to compute the eigenpairs (σ_{ij}, y_{ij}) of $A - \omega_i BB^T$ and eigenvalues of $A - \lambda_i BB^T$, which carries a total time cost of $O(pn^3)$. We then have all information needed to compute the last two terms in (23). Next, we can compute $B^T y_{ij}$ for $i = 2, 3, \ldots, p$, $j = 1, 2, \ldots n$ in $O(pn^2 m)$ time. Reusing the eigenpairs of $A - \omega_i BB^T$ and our previously computed $B^T y_{ij}$'s, we can compute the $m \times m$ matrix $N_{ij} = I + \sum_{k=2}^{p} \omega_k (x_k^T c)^2 G_k(-\sigma_{ij})$, where $G_k(-\sigma_{ij}) = B^T Y_k \operatorname{diag}(\frac{1}{-\sigma_{ij} - \sigma_{k1}}, \ldots, \frac{1}{-\sigma_{ij} - \sigma_{kn}})(B^T Y_k)^T$, using only matrix multiplication and addition in $O(pnm^2)$ time. Finally, we can compute $y_{ij}^T B N_{ij}^{-1} B^T y_{ij}$ in $O(m^3)$ time. To compute the first term in (23), this calculation needs to be performed for $i = 2, 3, \ldots, p$, $j = 1, 2, \ldots, n$. If $m \leq n$, then the first term

can be computed in $O(p^2n^2m^2)$ time. Computing (23) by solving (15) combined with (14) in general takes $O(p^3n^3)$ time, see [27]; however, iterative algorithms might be able to capitalize on the sparsity of (15) to more efficiently compute the approximation error.

Remark 13: Note that in the above theorem, ideally we would like to express the nonzero Laplacian eigenvalues ω_i of the reduced network in terms of the nonzero Laplacian eigenvalues λ_i of the original network. For general graphs this is however a hard problem. Only for special types of graphs it is possible to explicitly compute these eigenvalues and associated eigenvectors of the edge Laplacians, thus obtaining a simplified expression for the error. We will elaborate on this for *star graphs* in Section V of this paper. Another, different, approach could be to use results on graph sparsification, see [14] and [15]. One might be able to obtain a specific spanning tree for which the nonzero Laplacian eigenvalues of the reduced network graph are close to those of the original one, together with bounds relating the eigenvalues and eigenvectors of the reduced edge Laplacian to those of the original edge Laplacian. Combining these bounds with the expression in Theorem 11 could lead to upper bounds on the approximation error that are cheaper to compute.

Proof: Since the graph only contains a single cycle, we have that $T = c$ and the essential edge Laplacian is given by $L_e(I + cc^T)$. From (17) and (20), we have expressions for X_1 and X_2. We can now compute the trace of both matrices. For X_1, we have

$$\begin{aligned}\operatorname{tr} X_1 &= \operatorname{tr}(U \otimes I)\operatorname{diag}(Z_2, Z_3, \ldots, Z_p)(U^T \otimes I)\\ &= \sum_{i=2}^{p} \frac{\omega_i}{2} \operatorname{tr}(\omega_i BB^T - A)^{-1}. \end{aligned} \tag{25}$$

Similarly for X_2, we have

$$\begin{aligned}\operatorname{tr} X_2 &= \operatorname{tr}((I + cc^T)^{-\frac{1}{2}} V \otimes I)\operatorname{diag}(Q_2, Q_3, \ldots, Q_p)\\ &\quad \times (V^T (I + cc^T)^{-\frac{1}{2}} \otimes I)\\ &= \operatorname{tr}\left(I - \frac{V^T cc^T V}{l(c)} \otimes I\right)\operatorname{diag}(Q_2, Q_3, \ldots, Q_p)\\ &= \sum_{i=2}^{p} \left(1 - \frac{(v_i^T c)^2}{l(c)}\right) \frac{\lambda_i}{2} \operatorname{tr}(\lambda_i BB^T - A)^{-1} \end{aligned} \tag{26}$$

where the second equality follows from the Sherman–Morrison inversion formula, with $l(c) = c^T c + 1$, and the fact that V is an orthogonal matrix. Next, we consider the trace of X_{12}. Using (21), we have the following expression for X_{12}:

$$\begin{aligned}X_{12} &= \sum_{i,j} (x_i \otimes y_{ij})(x_i \otimes y_{ij})^T (-L_e \otimes I)\\ &\quad \times \left(\sigma_{ij} I \otimes I + I \otimes A - (I + cc^T) L_e \otimes BB^T\right)^{-1}. \end{aligned} \tag{27}$$

Next, take M_{ij} as in Lemma 10. Then (27) can be rewritten as

$$\begin{aligned}X_{12} &= \sum_{i,j} (x_i \otimes y_{ij})(x_i \otimes y_{ij})^T\\ &\quad \times \left(-\sigma_{ij} L_e^{-1} \otimes I - L_e^{-1} \otimes A + (I + cc^T) \otimes BB^T\right)^{-1}\\ &= \sum_{i,j} (x_i \otimes y_{ij})(x_i \otimes y_{ij})^T \left(M_{ij} + (c \otimes B)(c \otimes B)^T\right)^{-1}. \end{aligned}$$

Using the Sherman–Morrison–Woodbury matrix inversion formula, see, e.g., [28], we find that

$$\begin{aligned}\left(M_{ij} + (c \otimes B)(c \otimes B)^T\right)^{-1} &= M_{ij}^{-1} - M_{ij}^{-1}(c \otimes B)\\ &\quad \times \left(I + (c \otimes B)^T M_{ij}^{-1} (c \otimes B)^T\right)^{-1} (c \otimes B)^T M_{ij}^{-1}. \end{aligned}$$

We obtain

$$\begin{aligned}X_{12} &= \sum_{i,j} (x_i \otimes y_{ij})(x_i \otimes y_{ij})^T M_{ij}^{-1}\\ &\quad - \sum_{i,j} (x_i \otimes y_{ij})(x_i \otimes y_{ij})^T M_{ij}^{-1}(c \otimes B)\\ &\quad \times \left(I + (c \otimes B)^T M_{ij}^{-1}(c \otimes B)^T\right)^{-1} (c \otimes B)^T M_{ij}^{-1}\\ &= X_1 - \sum_{i,j} (x_i \otimes y_{ij})(x_i \otimes y_{ij})^T M_{ij}^{-1}(c \otimes B)\\ &\quad \times \left(I + (c \otimes B)^T M_{ij}^{-1}(c \otimes B)^T\right)^{-1} (c \otimes B)^T M_{ij}^{-1} \end{aligned} \tag{28}$$

where the fact that

$$X_1 = \sum_{i,j} (x_i \otimes y_{ij})(x_i \otimes y_{ij})^T M_{ij}^{-1}$$

follows immediately by interpreting (15a) as a Sylvester equation and taking X_1 as the explicit solution, similar to (21). Next, denote

$$\begin{aligned}E &= \sum_{i,j} (x_i \otimes y_{ij})(x_i \otimes y_{ij})^T M_{ij}^{-1}(c \otimes B)\\ &\quad \times \left(I + (c \otimes B)^T M_{ij}^{-1}(c \otimes B)^T\right)^{-1} (c \otimes B)^T M_{ij}^{-1}. \end{aligned}$$

Combining (14) with (28), we obtain that the approximation error is then given by

$$\|S - \hat{S}\|_2^2 = \operatorname{tr} X_2 - \operatorname{tr} X_1 + 2 \operatorname{tr} E. \tag{29}$$

We can now write the trace of E as

$$\begin{aligned}\operatorname{tr} E &= \sum_{i,j} (x_i \otimes y_{ij})^T M_{ij}^{-1}(c \otimes B)\\ &\quad \times \left(I + (c \otimes B)^T M_{ij}^{-1}(c \otimes B)^T\right)^{-1}\\ &\quad \times (c \otimes B)^T M_{ij}^{-1}(x_i \otimes y_{ij}). \end{aligned}$$

Using Lemma 10, this expression can be rewritten as

$$\begin{aligned}\operatorname{tr} E &= \frac{1}{4}\sum_{i,j}\frac{\omega_i^2}{\sigma_{ij}^2}(x_i^T c\otimes y_{ij}^T B)\\ &\quad\times\Big(I+(c\otimes B)^T M_{ij}^{-1}(c\otimes B)\Big)^{-1}(c^T x_i\otimes B^T y_{ij})\\ &= \frac{1}{4}\sum_{i,j}\left(x_i^T c\right)^2\frac{\omega_i^2}{\sigma_{ij}^2}y_{ij}^T B\Big(I+(c\otimes B)^T M_{ij}^{-1}(c\otimes B)\Big)^{-1}\\ &\quad\times B^T y_{ij}.\end{aligned}$$

Using (22) for the inverse of M_{ij}, we can compute that

$$\begin{aligned}&(c\otimes B)^T M_{ij}^{-1}(c\otimes B)\\ &\quad=\sum_{k=2}^{p}\omega_k(x_k^T c)^2 B^T\left(-\sigma_{ij}I-A+\omega_k BB^T\right)^{-1}B.\end{aligned}$$

Using the two expressions above, we arrive at the following expression for the trace of E:

$$\begin{aligned}\operatorname{tr} E &= \frac{1}{4}\sum_{i,j}\left(x_i^T c\right)^2\frac{\omega_i^2}{\sigma_{ij}^2}y_{ij}^T B\\ &\quad\times\left(I+\sum_{k=2}^{p}\omega_k(x_k^T c)^2 G_k(-\sigma_{ij})\right)^{-1}B^T y_{ij}\end{aligned}$$

where we observe that $G_k(s)=B^T(sI-A+\omega_k BB^T)^{-1}B$ is indeed the transfer matrix of system (24), which together with (25) and (26) proves the last equality in expression (23). ■

Theorem 11 gives an explicit expression for the approximation error if a single cycle is removed from the graph. A similar result holds if multiple edge-disjoint cycles are removed. An expression for the approximation error in the case that the network contains precisely r edge-disjoint cycles is given in the next theorem.

Theorem 14: Consider a network Σ consisting of a tree $\mathcal{T}$ together with r edges $e_1,e_2,\ldots,e_r\in\overline{\mathcal{E}}_t$, with graph $G=(V,\mathcal{E}_t\cup\bigcup_{j=1}^{r}\{e_j\})$, such that the associated cycles $c_1,c_2,\ldots,c_r$ are edge-disjoint. Let $l(c_k)$ denote the length of the cycle c_k. Let $\hat{\Sigma}$ denote the reduced network obtained by removing all r edges e_j. Then the approximation error is given by

$$\begin{aligned}&\|\Sigma-\hat{\Sigma}\|_2^2=\\ &\quad\frac{1}{2}\sum_{i,j}\frac{\omega_i^2}{\sigma_{ij}^2}(x_i^T T\otimes y_{ij}^T B)\Big(I+(T\otimes B)^T M_{ij}^{-1}(T\otimes B)\Big)^{-1}\\ &\qquad\times(T^T x_i\otimes B^T y_{ij})\\ &\quad+\frac{1}{2}\sum_{i=2}^{p}\left(1-\sum_{k=1}^{r}\frac{(v_i^T c_k)^2}{l(c_k)}\right)\lambda_i\operatorname{tr}(\lambda_i BB^T-A)^{-1}\\ &\quad-\frac{1}{2}\sum_{i=2}^{p}\omega_i\operatorname{tr}(\omega_i BB^T-A)^{-1}\end{aligned}\tag{30}$$

where $i=2,3,\ldots,p$ and $j=1,2,\ldots,n$. In the above expression:

1) $T=(c_1\,c_2\ldots c_r)$,
2) ω_i are the nonzero Laplacian eigenvalues of $\mathcal{T}$ and x_i corresponding normalized eigenvectors of L_e,
3) λ_i are the nonzero Laplacian eigenvalues of G and v_i corresponding normalized eigenvectors of $\bar{L}_e=(I+TT^T)^{\frac{1}{2}}L_e(I+TT^T)^{\frac{1}{2}}$,
4) σ_{ij} denotes the jth eigenvalue of the matrix $A-\omega_i BB^T$.

Proof: In the case of multiple cycles, (25) for the trace of X_1 still holds. The matrix X_2 is again given by (20), and similarly to (26), the trace of X_2 satisfies

$$\operatorname{tr} X_2=\operatorname{tr}(V^T(I+TT^T)^{-1}V\otimes I)\operatorname{diag}(Q_2,Q_3,\ldots,Q_p)\tag{31}$$

where V is an orthogonal matrix that diagonalizes the matrix $\bar{L}_e$ in (18). Again using the Woodbury inversion formula, we find that the inverse of $I+TT^T$ is given by $I-T(I+T^TT)^{-1}T^T$. Since the cycles $c_1,\ldots,c_r$ are independent, the matrix T^TT is diagonal with $c_i^T c_i=l(c_i)-1$ on the diagonal. It then follows that

$$(I+TT^T)^{-1}=I-\sum_{k=1}^{r}\frac{c_k c_k^T}{l(c_k)}.$$

It is immediate that the trace of X_2 is then given by

$$\operatorname{tr} X_2=\sum_{i=2}^{p}\left(1-\sum_{k=1}^{r}\frac{(v_i^T c_k)^2}{l(c_k)}\right)\frac{\lambda_i}{2}\operatorname{tr}(\lambda_i BB^T-A)^{-1}.$$

Analogous to (28), we have the following expression for X_{12}:

$$\begin{aligned}X_{12}&=X_1-\sum_{i,j}(x_i\otimes y_{ij})(x_i\otimes y_{ij})^T M_{ij}^{-1}(T\otimes B)\\ &\quad\times\Big(I+(T\otimes B)^T M_{ij}^{-1}(T\otimes B)\Big)^{-1}(T\otimes B)^T M_{ij}^{-1}.\end{aligned}\tag{32}$$

In general, the matrix $I+(T\otimes B)^T M_{ij}^{-1}(T\otimes B)$ is not easily simplified. It is a block matrix consisting of $r\times r$ blocks K_{mq}^{ij}, where the

$$K_{mq}^{ij}=\begin{cases}I+\sum_{k=2}^{p}\omega_k(x_k^T c_m)^2 G_k(-\sigma_{ij}) & \text{if } m=q,\\ \sum_{k=2}^{p}\omega_k(x_k^T c_m)(x_k^T c_q)G_k(-\sigma_{ij}) & \text{otherwise.}\end{cases}\tag{33}$$

Combining (14) with (32) and the result of Lemma 10 completes the proof. ■

The result in Theorem 14 is not a straightforward generalization of (23), mainly because the inverse of the matrix $I+(T\otimes B)^T M_{ij}^{-1}(T\otimes B)$ is not easily expressed in simpler terms. For some specific graph topologies, expression (30) can however be further simplified. This will be illustrated in the following section, in which we will study the case that our network graph has a star graph as a spanning tree.

Finally, we will take a look at the case that only a subset of the cycle closing edges is removed from the network graph. In other words, we consider a network Σ consisting of a tree $\mathcal{T}$ together with r edges $e_1,e_2,\ldots,e_r\in\overline{\mathcal{E}}_t$, with graph $G=(V,\mathcal{E}_t\cup\bigcup_{j=1}^{r}\{e_j\})$. We number the edges in such a way that the last $r-k$ edges $e_{k+1},\ldots,e_r$ are removed, resulting in the reduced network $\hat{\Sigma}$. In this case, we can find an *upper bound* to

the error by comparing both Σ and $\hat{\Sigma}$ with the network obtained by removing *all* edges $e_1, e_2, \ldots, e_r$, resulting in the spanning tree $\mathcal{T}$. Denote this network by $\Sigma_{\mathcal{T}}$. Then we obtain:

$$\|\Sigma - \hat{\Sigma}\|_2 \leq \|\Sigma - \Sigma_{\mathcal{T}}\|_2 + \|\Sigma_{\mathcal{T}} - \hat{\Sigma}\|_2. \tag{34}$$

Next, an upper bound can be obtained by applying Theorem 14 to both errors on the righthand side of (34).

Remark 15: For the case of scalar single integrator dynamics, in [19] explicit expressions were obtained also for the case that the reduced network is obtained by removing multiple *dependent* cycles. Due to the complexity of the error expressions, in the present paper, we will not treat the extension of these results to symmetric input-state-output systems.

V. Star Graphs

In this section, we will zoom in on the special case that the chosen spanning tree is given by a star graph: a tree graph where all the nodes are connected only to a single root node. It turns out that in that case the expression for the approximation error can be greatly simplified, because we can explicitly compute the Laplacian eigenvalues of the original graph and its spanning tree, and also corresponding eigenvectors of L_e and $\bar{L}_e$. In addition, if the original network reaches consensus, then automatically also the reduced network does.

First, we will formulate and prove a lemma that will be instrumental in the sequel:

Lemma 16: Let $c_1, c_2, \ldots, c_r$ be edge-disjoint cycles. Then for the matrix $T = (c_1\ c_2\ \ldots\ c_r)$, we have

$$(I + TT^T)^{\frac{1}{2}} = I + TDT^T$$

where $D = \operatorname{diag}(\alpha_1, \alpha_2, \ldots, \alpha_r)$ and $\alpha_i = \frac{\sqrt{1+|c_i|^2}-1}{|c_i|^2}$.

Proof: The claim is easily checked by squaring $I + TDT^T$ and verifying that $I + 2TDT^T + TDT^TTDT^T$ equals $I + TT^T$, if and only if $|c_i|^2\alpha_i^2 + 2\alpha_i = 1$. By taking α_i as the positive solution of $|c_i|^2\alpha_i^2 + 2\alpha_i = 1$, obviously $I + TDT^T$ is positive semidefinite. ■

The following result deals with the case that after removal of multiple edge-disjoint cycles, we obtain a star graph as spanning tree. The result gives an expression for the approximation error between the network on the original graph and the reduced network on the star graph.

Theorem 17: Consider a network Σ consisting of a star graph S_{p-1} together with r edges $e_1, e_2, \ldots, e_r \in \mathcal{E}(S_{p-1})$, with graph $G = (V, \mathcal{E}(S_{p-1}) \cup \bigcup_{j=1}^{r}\{e_j\})$, such that the associated cycles $c_1, c_2, \ldots, c_r$ are edge-disjoint. Let $\hat{\Sigma}$ be the reduced network obtained by removing all r edges e_j. Then Σ and $\hat{\Sigma}$ reach consensus if and only if $A - BB^T$ is Hurwitz. The approximation error is then given by

$$\|\Sigma - \hat{\Sigma}\|_2^2 = r\sum_{j=1}^{n}\frac{1}{\sigma_j^2}y_j^T B(I + 2G(-\sigma_j))^{-1}B^T y_j + \frac{r}{2}\left(\operatorname{tr}(3BB^T - A)^{-1} - \operatorname{tr}(BB^T - A)^{-1}\right) \tag{35}$$

where σ_j is the jth eigenvalue with corresponding normalized eigenvector y_j of the matrix $A - BB^T$, and $G(s) := B^T(sI - A + BB^T)^{-1}B$.

Remark 18: Note that the approximation error (35) is independent of the number of agents in the network. The error is proportional to the number of cycles that have been removed from the network.

Proof: The proof follows easily by computing the eigenvalues and eigenvectors of the edge Laplacian of the star graph S_{p-1} and constructing the unitary matrix U such that $U^T L_e U = \Omega$. It is well known that the eigenvalues of the edge Laplacian L_e of the star graph S_{p-1} are given by

$$\omega_2 = \omega_3 = \cdots = \omega_{p-1} = 1, \quad \omega_p = p \tag{36}$$

see, e.g., [1] and [2]. The edge Laplacian L_e has a $p-2$ dimensional eigenspace associated with the eigenvalue 1 and a one-dimensional eigenspace associated with the eigenvalue $\omega_p = p$. The eigenspace associated with ω_p is the span of the vector $\mathbb{1} \in \mathbb{R}^{p-1}$, the vector of ones, so we can take $x_p = \frac{1}{\sqrt{p-1}}\mathbb{1}$. Since L_e is symmetric, these two eigenspaces are each other's orthogonal complement.

If we number and orient the edges in the star graph such that edge i is from node 1 to node $i+1$, then without loss of generality we can assume that $c_1 = (-1\ 1\ 0\ \cdots\ 0)^T$, $c_2 = (0\ 0\ -1\ 1\ 0\ \cdots\ 0)^T$, etc. The interpretation is that the first removed edge points from node 2 to node 3, the second edge points from node 4 to node 5, etc. It is easily seen that indeed all the cycles are independent. For each cycle c_i, we have that $c_i^T\mathbb{1} = 0$, from which it follows that they all lie in the eigenspace of L_e associated with the eigenvalue 1. Now, for $j = 1, 2, \ldots, r$, choose

$$x_{j+1} = \frac{1}{|c_j|}c_j = \frac{1}{\sqrt{2}}c_j. \tag{37}$$

Since the cycles are edge disjoint, we have that $x_j^T x_i = 0$ if $i \neq j (i, j < r+2)$. Finally, choose x_i for $i = r+2, r+3, \ldots, p-1$ such that the matrix

$$U = (x_2\ \cdots\ x_{r+1}\ x_{r+2}\ \cdots\ x_{p-1}\ x_p)$$

is orthogonal. We then have that the x_is are eigenvectors of L_e and that $U^T L_e U = \Omega$. Furthermore, it holds that

$$x_i^T c_j = \begin{cases} \sqrt{2} & \text{if } i = j+1 \\ 0 & \text{otherwise.} \end{cases} \tag{38}$$

This follows from the fact that for $i = j+1$, we have $x_i^T c_j = \frac{c_j^T c_j}{|c_j|} = \sqrt{2}$, and from the orthogonality of the eigenvectors. Next, we compute the eigenvalues λ_i and eigenvectors v_i of $\bar{L}_e = (I + TT^T)^{\frac{1}{2}} L_e (I + TT^T)^{\frac{1}{2}}$. For all cycles, we have that $|c_j|^2 = 2$. From Lemma 16, it then follows that

$$(I + TT^T)^{\frac{1}{2}} = I + \alpha TT^T$$

where $\alpha = \frac{1}{2}(\sqrt{3}-1)$. For $i = 2, 3, \ldots, r$, we obtain

$$\begin{aligned}(I &+\alpha TT^T)L_e(I+\alpha TT^T)x_i \\ &= (I+\alpha TT^T)L_e(x_i+\alpha|c_{i-1}|^2x_i)v \\ &= (I+\alpha TT^T)(x_i+\alpha|c_{i-1}|^2x_i) \\ &= x_i+\alpha|c_{i-1}|^2x_i+\alpha|c_{i-1}|^2+\alpha^2|c_{i-1}|^4 \\ &= x_i+|c_{i-1}|^2(2\alpha+\alpha^2|c_{i-1}|^2)x_i \\ &= 3x_i\end{aligned}$$

which follows from (37), the fact that $|c_j|^2 = 2$ for all j and from $|c_j|^2\alpha^2 + 2\alpha = 1$. From (38), it follows that for $i = r+2, r+3, \ldots, p$, we have $T^T x_i = 0$ and consequently

$$\begin{aligned}(I+\alpha TT^T)L_e(I+\alpha TT^T)x_i &= (I+\alpha TT^T)L_e x_i \\ &= (I+\alpha TT^T)\omega_i x_i \\ &= \omega_i x_i.\end{aligned}$$

We see that the eigenvectors of L_e are also the eigenvectors of $\bar{L}_e$ and that $\lambda_2 = \cdots = \lambda_{r+1} = 3$, $\lambda_{r+2} = \cdots = \lambda_{p-1} = 1$, and $\lambda_p = p$. Recall that we have assumed that $A - BB^T$ is Hurwitz. Thus, also $A - 3BB^T$ and $A - pBB^T$ are Hurwitz, so both the original network as well as the reduced network reach consensus.

We can now compute

$$\begin{aligned}&\operatorname{tr} X_2 - \operatorname{tr} X_1 \\ &= \frac{1}{2}\left(\sum_{i=2}^{p}\left(1-\sum_{k=1}^{r}\frac{(v_i^T c_k)^2}{l(c_k)}\right)\lambda_i \operatorname{tr}(\lambda_i BB^T - A)^{-1}\right. \\ &\qquad \left. -\sum_{i=2}^{p}\omega_i \operatorname{tr}(\omega_i BB^T - A)^{-1}\right) \\ &= \frac{1}{2}\Big((p-2-r)\operatorname{tr}(BB^T-A)^{-1} \\ &\qquad + r\operatorname{tr}(3BB^T-A)^{-1} + p\operatorname{tr}(pBB^T-A)^{-1}\Big) \\ &\quad - \frac{1}{2}\Big((p-2)\operatorname{tr}(BB^T-A)^{-1} + p\operatorname{tr}(pBB^T-A)^{-1}\Big) \\ &= \frac{r}{2}\Big(\operatorname{tr}(3BB^T-A)^{-1} - \operatorname{tr}(BB^T-A)^{-1}\Big). \qquad (39)\end{aligned}$$

Next, we consider the following trace:

$$\begin{aligned}\operatorname{tr} E = \frac{1}{4}\sum_{i,j}\frac{\omega_i^2}{\sigma_{ij}^2}(x_i^T T \otimes y_{ij}^T B) \qquad (40)\\ \times\Big(I+(T\otimes B)^T M_{ij}^{-1}(T\otimes B)\Big)^{-1}(T^T x_i \otimes B^T y_{ij}).\end{aligned}$$

As mentioned in the proof of Theorem 14, the matrix $I + (T\otimes B)^T M_{ij}^{-1}(T\otimes B)$ is in general not easily simplified. However, by combining (33) and (38) and the fact that $x_m^T c_j$ and $x_q^T c_j$ are never simultaneously nonzero for $m \neq q$, we see that in the case of a star graph, the matrix is block diagonal. The diagonal blocks are given by

$$K_{mm}^{ij} = I + \omega_{m+1}(x_{m+1}^T c_m)^2 G_{m+1}(-\sigma_{ij})$$

for $m = 1, 2, \ldots, r$. From (36) and (38), we obtain that

$$K_{mm}^{ij} = I + 2G_2(-\sigma_{ij})$$

where G_2 is the transfer matrix of system (24) with $\omega_2 = 1$. Equation (40) then reduces to

$$\operatorname{tr} E = \frac{1}{4}\sum_{i,j}\frac{\omega_i^2}{\sigma_{ij}^2}(x_i^T TT^T x_i)y_{ij}^T B(I+2G_2(-\sigma_{ij}))^{-1}B^T y_{ij}.$$

Recall that for $i > r+1$, we have $T^T x_i = 0$, so only the first r terms of the sum over i are nonzero. From (36), we have that for $i = 2, 3, \ldots, r+1$ the eigenvalues ω_i and σ_{ij} do not depend on i. Combined with (38), we finally obtain

$$\operatorname{tr} E = \frac{r}{2}\sum_{j=1}^{n}\frac{1}{\sigma_j^2}y_j^T B(I+2G_2(-\sigma_j))^{-1}B^T y_j \qquad (41)$$

with σ_j and y_j as in Theorem 17. We conclude the proof by combining (29), (39), and (41). ■

VI. Single Integrator Dynamics

In this section, we will return to the case that we have a general (undirected and connected) graph, and zoom in on the special case of simple first-order agent dynamics. In [19], the agent dynamics was given by a single integrator, which resulted in a standard diffusively coupled network. In this section, we take another look at the single integrator case; however, we include a drift term a in the agent dynamics, in other words, we assume that the dynamics of each agent is given by $\dot{x}_i = ax_i + u_i$ with $a \in \mathbb{R}$. We will show that the results in [19] indeed follow from the more general results presented in this paper. For the single integrator case, in the agent dynamics (2), we now have $A = a$, $B = 1$. Thus, the state of each agent is scalar and $n = 1$. The original model dynamics (8) now reduces to

$$\dot{\xi} = (aI - L_e(I+TT^T))\xi + E_t^T d,\ z = \xi \qquad (42)$$

and, in the case that all cycles are removed, the reduced model dynamics (9) is given by

$$\dot{w} = (aI - L_e)w + E_t^T d,\ \hat{z} = w.$$

The eigenvalues σ_{ij} in Theorem 11 are now given by $a - \omega_i$ and we have $y_{ij} = 1$ for $j = 1$. To guarantee consensus, we assume that the drift term a satisfies $a < \omega_2 \leq \lambda_2$, where ω_2 and λ_2 are the algebraic connectivities of, respectively, the reduced network graph and the original network graph. The transfer function $G_k(s)$ is now given by $\frac{1}{s-a+\omega_k}$. Finally, we observe that now $s\operatorname{tr}(sBB^T - A)^{-1} = \frac{s}{s-a}$. Using these observations, we arrive at the following corollary of Theorem 11:

Corollary 19: Let Σ be a network consisting of the tree $\mathcal{T}$ combined with a single extra edge $e \in \overline{\mathcal{E}}_t$, with graph $G = (V, \mathcal{E}_t \cup e)$, such that the graph has exactly one cycle c. Let the agent dynamics be given by coupled integrators with drift term a, i.e., by $\dot{x}_i = ax_i + u_i$ with $a \in \mathbb{R}$. Let $\hat{\Sigma}$ the reduced network obtained by removing the edge e. The approximation error is

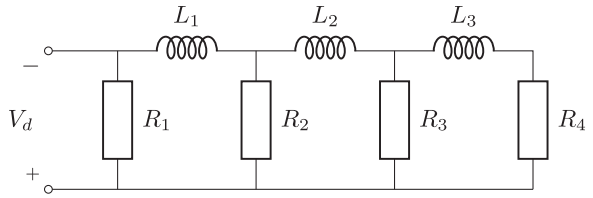


Fig. 2. *RL* circuit with three inductors and four resistors.

then given by

$$\|\Sigma-\hat{\Sigma}\|_2^2=\frac{1}{2}\left(\sum_{i=2}^{p}\frac{\omega_i^2}{(\omega_i-a)^2}\frac{\left(x_i^Tc\right)^2}{1+\sum_{j=2}^{p}\frac{\omega_j}{\omega_j+\omega_i-2a}\left(x_j^Tc\right)^2}+\sum_{i=2}^{p}\left(1-\frac{\left(v_i^Tc\right)^2}{l(c)}\right)\frac{\lambda_i}{\lambda_i-a}-\sum_{i=2}^{p}\frac{\omega_i}{\omega_i-a}\right). \tag{43}$$

■

Remark 20: Of course, by applying Theorem 14, the above can be extended to the case that multiple edges that close edge-disjoint cycles are removed. Also note that by taking $a=0$ in (43), we immediately reobtain the result on the single integrator case in [19, Th. 5]:

$$\|\Sigma-\hat{\Sigma}\|_2^2=\frac{1}{2}\left(\sum_{i=2}^{p}\frac{\left(x_i^Tc\right)^2}{1+\sum_{j=2}^{p}\frac{\omega_j}{\omega_j+\omega_i}\left(x_j^Tc\right)^2}+\frac{1}{l(c)}-1\right).$$

Remark 21: If, as in Section V, we consider the special case that the network graph contains a star graph as a spanning tree, we can take $A=a$, $B=1$ in (35). For a star graph, we need to assume that $a<1$, in order to guarantee that the original and reduced networks reach consensus. Thus, we obtain that, after removing r edges closing edge-disjoint cycles, the approximation error is given by

$$\|\Sigma-\hat{\Sigma}\|_2^2=\frac{r}{(1-a)(2-a)}+\frac{r}{2}\left(\frac{1}{3-a}-\frac{1}{1-a}\right).$$

Computing the limit for $a\to-\infty$, we see that the approximation error approaches zero. Thus, the approximation error decreases with increasing stability of the agent dynamics. For $a\to1$ from the left, we see that the approximation error tends to infinity instead. This means that if the original and reduced network come closer to losing consensus, the error grows unboundedly. If we assume that the agent dynamics is a simple integrator without a drift term, i.e., $a=0$, then we obtain that the error is exactly $\frac{r}{6}$. If we remove one edge, the error is always $\frac{1}{6}$, for any number of nodes in the star graph.

VII. Numerical Examples

In this section, we consider a number of examples to illustrate the theory developed in this paper.

Example 22: Consider a network in which each agent is the *RL* circuit depicted in Fig. 2 . The circuit is taken from [22]. As the network graph, we take the graph of Fig. 1. This graph consists of $p=7$ nodes, has two edge-disjoint cycles ($r=2$), and contains S_6 as a spanning tree.

As the state of each agent, we take the currents through the inductors L_1, L_2, and L_3, so $n=3$. The input each agent receives is the voltage V_d. Finally, as output, we take the current through the first inductor L_1. This results in the following agent dynamics:

$$\dot{x}_i=\begin{pmatrix}\frac{-R_2}{L_1}&\frac{R_2}{L_1}&0\\ \frac{R_2}{L_2}&\frac{-(R_2+R_3)}{L_2}&\frac{R_3}{L_2}\\ 0&\frac{R_3}{L_3}&\frac{-(R_3+R_4)}{L_3}\end{pmatrix}x_i+\begin{pmatrix}\frac{1}{L_1}\\0\\0\end{pmatrix}u_i$$
$$y_i=\begin{pmatrix}1&0&0\end{pmatrix}x_i. \tag{44}$$

If we take $L_1=L_2=L_3=1H$, then (44) is a symmetric system. As in [22], for the resistors, we take $R_1=0.5\ \Omega$, $R_2=8\ \Omega$, $R_3=5\ \Omega$, $R_4=1\ \Omega$. The matrices A, B, and C are now given by

$$A=\begin{pmatrix}-8&8&0\\8&-13&5\\0&5&-6\end{pmatrix},\quad B=C^T=\begin{pmatrix}1\\0\\0\end{pmatrix}.$$

Using expression (35) in Theorem 17, we will now compute the approximation error when removing all cycles from the network. We first compute the eigenvalues σ_j and normalized eigenvectors y_j of $A-BB^T$. We obtain $\sigma_1=-20.3691$, $\sigma_2=-6.9764$, and $\sigma_3=-0.6545$ with corresponding eigenvectors

$$Y=\begin{pmatrix}y_1&y_2&y_3\end{pmatrix}=\begin{pmatrix}0.5535&0.6039&0.5735\\-0.7866&0.1528&0.5983\\0.2737&-0.7823&0.5596\end{pmatrix}.$$

Thus, $A-BB^T$ is Hurwitz, so both the original network as well as its approximation reach consensus. We then compute the first term in (35) as

$$\sum_{j=1}^{3}\frac{1}{\sigma_j^2}y_j^TB(I+2G(-\sigma_j))^{-1}B^Ty_j=0.4790.$$

Next, we compute the trace of $(BB^T-A)^{-1}$ and $(3BB^T-A)^{-1}$, and obtain $\mathrm{tr}(BB^T-A)^{-1}=1.7204$, $\mathrm{tr}(3BB^T-A)^{-1}=0.9950$. Combining the above, we find as approximation error

$$\|\Sigma-\hat{\Sigma}\|_2^2=0.4790\cdot2+\frac{2}{2}\cdot(0.9950-1.7204)=0.2326. \tag{45}$$

Note that, by Theorem 17, this error is independent of the number of nodes in the resulting star graph. If the original graph would be the union of S_{p-1} with r edges that close edge-disjoint cycles, then removing these r edges yield an approximation error $0.4790\cdot r+\frac{r}{2}\cdot(0.9950-1.7204)$.

While this approximation error is independent of the number of nodes in the graph and is always proportional to the number of removed edges, it heavily depends on the eigenvalues of the matrices $A-BB^T$ and $A-3BB^T$. In the previous section, it was shown that in the single integrator case the approximation error decreases as the agent dynamics becomes more stable. We again consider the system (44) and investigate what happens to

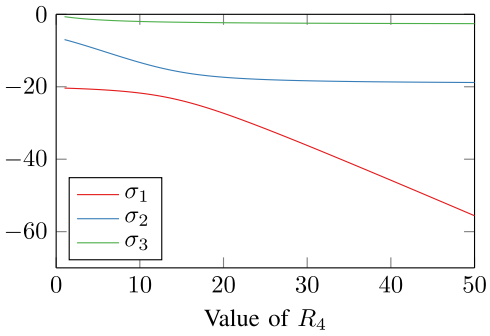


Fig. 3. Eigenvalues $\sigma_1, \sigma_2, \sigma_3$ of $A - BB^T$ for increasing R_4.

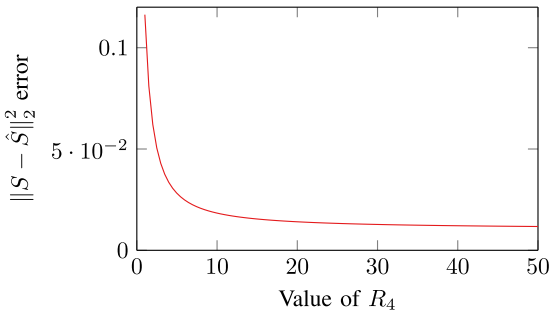


Fig. 4. Approximation error $\|S - \hat{S}\|_2^2$ per removed edge.

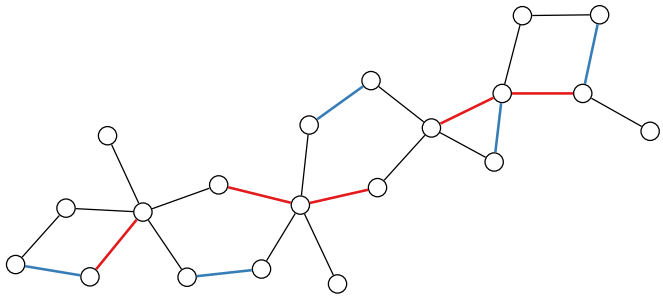


Fig. 5. Graph on $p = 20$ nodes with five edge-disjoint cycles.

the approximation error if we vary the resistance of R_4 from 1 to 50 Ω, thereby moving the eigenvalues of $A - BB^T$ closer to $-\infty$. As seen in Fig. 3, the eigenvalue σ_1 of $A - BB^T$ remains strictly monotonically decreasing, while both σ_2 and σ_3 decrease slightly before converging to a constant lower bound. The corresponding approximation error per removed edge is shown in Fig. 4. Fig. 4 shows that the approximation error rapidly decreases as the resistance of R_4 increases. Indeed, increasing R_4 from 1 to 10 reduces the approximation error per removed edge from 0.1163 to 0.0184.

Example 23: As a final example, we investigate network (42) with network graph G shown in Fig. 5. In this network, the agent dynamics is given by a single integrator system with drift term $a = 0.03$. This network has $p = 20$ agents and five edge-disjoint cycles. Computing λ_2, we obtain $\lambda_2 = 0.0910$, so the original network reaches consensus. It is easily verified that the graph has 1200 spanning trees: in each cycle, we need to remove a single edge, so the number of different spanning trees can be computed by taking the product of the cycle lengths. After selecting a specific spanning tree, Theorem 14 can be used to compute the approximation error. However, how to select the spanning tree that minimizes the approximation error remains an open question. In [29], an upper bound on the approximation error was obtained that decreases with the increase of the algebraic connectivity ω_2 of the reduced network graph. Thus, a sensible approach could be to use the spanning tree with the largest algebraic connectivity ω_2. Unfortunately no algorithm currently exists that efficiently computes this spanning tree. However, while there is no direct relationship between the algebraic connectivity $\omega_2(\mathcal{T})$ of the spanning tree $\mathcal{T}$ and its diameter $\mathrm{diam}(\mathcal{T})$, the diameter can be used to obtain an upper bound on ω_2, see, e.g., [30, Th. 4.1]. This bound is given by

$$\omega_2(\mathcal{T}) \leq 2\left(1 - \cos\left(\frac{\pi}{\mathrm{diam}(\mathcal{T}) + 1}\right)\right).$$

The upper bound increases as the diameter of the spanning tree decreases. The problem of finding a spanning tree with minimum diameter for a given graph has been widely studied and can be solved in $O(qp + p^2 \log p)$ time, where q is the number of edges in the graph, see, e.g., [31].

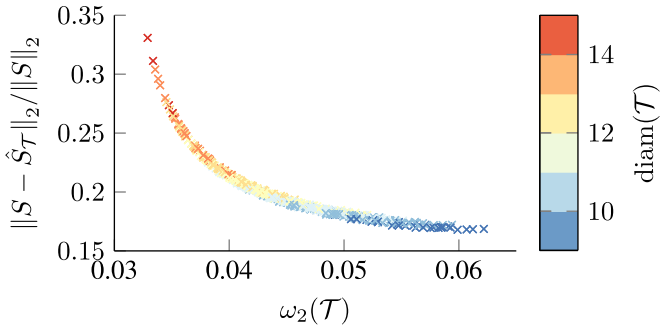


Fig. 6. Relative approximation error versus $\omega_2(\mathcal{T})$ of all possible spanning trees.

For each possible spanning tree $\mathcal{T}$ of the graph G in Fig. 5, the value of $\omega_2(\mathcal{T})$ and the relative approximation error $\|S - \hat{S}_{\mathcal{T}}\|_2 / \|S\|_2$, with $\|S\|_2 = 2.7598$, have been plotted in Fig. 6. Each point in the scatter plot has been colored according to the diameter $\mathrm{diam}(\mathcal{T})$ of the spanning tree, with small diameters colored blue and large diameters colored red. The spanning tree $\mathcal{T}_1$ with the minimum value of ω_2, obtained at $\omega_2(\mathcal{T}_1) = 0.0329$, also achieves the worst relative approximation error: $\|S - \hat{S}_{\mathcal{T}_1}\|_2 / \|S\|_2 = 0.3308$. This tree is obtained from G by removing the red edges in Fig. 5. Furthermore, $\mathrm{diam}(\mathcal{T}_1) = 15$, so it is among the spanning trees with the maximum possible diameter. The tree $\mathcal{T}_2$, which maximizes $\omega_2 = 0.0622$, is obtained by removing the blue edges from G. For $\mathcal{T}_2$, we have $\mathrm{diam}(\mathcal{T}_2) = \mathrm{diam}(G) = 9$ and $\|S - \hat{S}_{\mathcal{T}_2}\|_2 / \|S\|_2 = 0.1687$, which is very close to the minimal relative approximation error obtained at $\min_{\mathcal{T}} \|S - \hat{S}_{\mathcal{T}}\|_2 / \|S\|_2 = 0.1679$.

VIII. Concluding Remarks

This paper proposes a new method for model reduction of linear multiagent systems. Instead of a clustering-based approach,

this new method is based on removing cycles from the network graph. We provided explicit expressions and upper bounds for the approximation error when approximating a network that has a symmetric system as its agent dynamics. The approximation error is expressed in terms of the signed path vectors of the removed cycles and the eigenvalues and eigenvectors of the edge Laplacian matrices of the original and reduced network graphs. Results have been presented for the cases that either a single cycle or multiple uncorrelated cycles are being removed from the network graph. If the reduced network topology is given by a star graph, these expressions can be further simplified. We have shown that previously obtained results for single integrator agent dynamics are easily obtained from the more general results presented in this paper.

While the paper concentrates on the case that the reduced network graph is a tree graph, one can use the present results to obtain upper bounds on the approximation error for general reduced graphs, such as the approximation error when removing an arbitrary cycle, by using the expressions presented in the paper and the triangle inequality. Future research could try to improve these bounds or could try to find explicit expressions for the approximation error in the more general case.

The expressions presented in this paper depend on the Laplacian eigenvalues and eigenvectors of the original and reduced networks. The computation of these values might prove expensive for networks with a large number of agents. Combining the expressions in this paper with results from graph sparsification could produce upper bounds on the approximation error that are cheaper to compute.

In this paper, we have assumed that the chosen spanning tree and to be removed cycles are given. A natural question to ask is how to optimally choose such a spanning tree in such a way that the approximation error in minimized. From the expressions for the approximation error that have been provided in this paper, the answer to this question is not immediately apparent. Earlier results on networks with single integrator dynamics have shown that the approximation error can be bounded by an upper bound that increases with the length of the to be removed cycles and decreases with the increase of the algebraic connectivity of the spanning tree. Future research should investigate whether it is possible to obtain similar upper bounds in the multivariable case.

Other possible extensions could be to find expressions for or bounds on the $\mathcal{H}_\infty$-norm of the error system instead of the $\mathcal{H}_2$-norm used in the present paper. Furthermore, it could be possible to combine the results in this paper with clustering-based techniques. For example, in the past, clustering-based techniques based on almost equitable partitions have been proposed. While finding almost equitable partitions of graphs is a difficult combinatorial problem in general, it might be more feasible for tree graphs. Combining results from both techniques could provide low-dimensional but accurate approximating systems. This paper only deals with the case that the agents are represented by a symmetric input-state-output system. In our paper, this assumption is made for technical reasons. Future research could aim at extending our results to the case that the agent dynamics is a general input-state-output system. Finally, another important topic for the future research is to investigate in what sense the proposed model reduction method leads to faster simulations for the reduced network as compared to the original one. Also, it would be of interest to investigate whether controllers that are designed on the basis of the reduced network lead to satisfactory behavior of the original network.

REFERENCES

[1] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks (Ser. Princeton Series in Applied Mathematics)*. Princeton, NJ, USA: Princeton Univ. Press, 2010.

[2] P. Van Mieghem, *Graph Spectra for Complex Networks*. Cambridge, U.K.: Cambridge Univ. Press, 2011.

[3] L. Moreau, "Stability of multiagent systems with time-dependent communication links," *IEEE Trans. Autom. Control*, vol. 50, no. 2, pp. 169–182, Feb. 2005.

[4] R. Olfati-Saber and R. M. Murray, "Consensus protocols for networks of dynamic agents," in *Proc. Amer. Controls Conf.*, Jun. 2003, vol. 2, pp. 951–956.

[5] Z. Li, Z. Duan, G. Chen, and L. Huang, "Consensus of multiagent systems and synchronization of complex networks: A unified viewpoint," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 1, pp. 213–224, Jan. 2010.

[6] C.-Q. Ma and J.-F. Zhang, "Necessary and sufficient conditions for consensusability of linear multi-agent systems," *IEEE Trans. Autom. Control*, vol. 55, no. 5, pp. 1263–1268, May 2010.

[7] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[8] J.-I. Imura, "Clustered model reduction of large-scale complex networks," in *Proc. 20th Int. Symp. Math. Theory Netw. Syst.*. 2012 [Online]. Available: http://www.mtns2012.conference.net.au/Full%20Paper/MTNS2012_0311_paper.pdf

[9] T. Ishizaki, K. Kashima, J.-I. Imura, and K. Aihara, "Model reduction of multi-input dynamical networks based on clusterwise controllability," in *Proc. Amer. Control Conf.*, Jun. 2012, pp. 2301–2306.

[10] B. Besselink, H. Sandberg, and K. H. Johansson, "Model reduction of networked passive systems through clustering," in *Proc. Eur. Control Conf.*, Jun. 2014, pp. 1069–1074.

[11] N. Monshizadeh, H. L. Trentelman, and M. K. Çamlibel, "Projection-based model reduction of multi-agent systems using graph partitions," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 2, pp. 145–154, Jun. 2014.

[12] D. Zelazo and M. Mesbahi, "Edge agreement: Graph-theoretic performance bounds and passivity analysis," *IEEE Trans. Autom. Control*, vol. 56, no. 3, pp. 544–555, Mar. 2011.

[13] D. Zelazo, S. Schuler, and F. Allgöwer, "Performance and design of cycles in consensus networks," *Syst. Control Lett.*, vol. 62, no. 1, pp. 85–96, 2013.

[14] D. A. Spielman and N. Srivastava, "Graph sparsification by effective resistances," *SIAM J. Comput.*, vol. 40, no. 6, pp. 1913–1926, 2011.

[15] D. A. Spielman and S.-H. Teng, "Spectral sparsification of graphs," *SIAM J. Comput.*, vol. 40, no. 4, pp. 981–1025, 2011.

[16] A. A. Benczúr and D. R. Karger, "Approximating s-t minimum cuts in $\tilde{O}(n^2)$ time," in *Proc. 28th Annu. ACM Symp. Theory Comput.*, 1996, pp. 47–55.

[17] J. Rommes and W. H. A. Schilders, "Efficient methods for large resistor networks," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 29, no. 1, pp. 28–39, Jan. 2010.

[18] M. V. Ugryumova, J. Rommes, and W. H. A. Schilders, "Error bounds for reduction of multi-port resistor networks," *Int. J. Numer. Modelling, Electron. Netw. Devices Fields*, vol. 26, no. 5, pp. 464–477, 2013.

[19] H. J. Jongsma, H. L. Trentelman, and M. K. Çamlibel, "Model reduction of consensus networks by graph simplification," in *Proc. 54th IEEE Conf. Decis. Control*, 2015, pp. 5340–5345.

[20] C. Godsil and G. Royle, *Algebraic Graph Theory (ser. Graduate Texts in Mathematics)*, vol. 207. New York, NY, USA: Springer-Verlag, 2001.

[21] J. C. Willems, "Realization of systems with internal passivity and symmetry constraints," *J. Franklin Inst.*, vol. 301, no. 6, pp. 606–621, 1976.

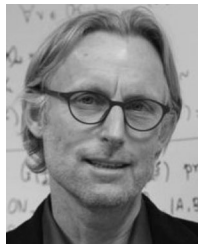
[22] G. I. Bara, "On the dissipative analysis and control of state-space symmetric systems," in *Proc. 50th IEEE Conf. Decis. Control Eur. Control Conf.*, 2011, pp. 459–464.

[23] R. E. Skelton, T. Iwasaki, and K. Grigoriadis, *A Unified Algebraic Approach to Linear Control Design*. London, U.K.: Taylor and Francis, 1998.

[24] H. L. Trentelman, A. A. Stoorvogel, and M. Hautus, *Control Theory for Linear Systems*. New York, NY, USA: Springer-Verlag, 2001.
[25] A. C. Antoulas, *Approximation of Large-Scale Dynamical Systems (Ser. Advances in Design and Control)*, vol. 6. Philadelphia, PA, USA: SIAM, 2005.
[26] B. N. Parlett, *The Symmetric Eigenvalue Problem*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1980.
[27] P. Benner, J.-R. Li, and T. Penzl, "Numerical solution of large-scale Lyapunov equations, Riccati equations, and linear-quadratic optimal control problems," *Numer. Linear Algebra Appl.*, vol. 15, no. 9, pp. 755–777, 2008.
[28] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1985.
[29] H. J. Jongsma, H. L. Trentelman, and M. K. Çamlibel, "Model reduction of consensus networks by graph simplification," in *Proc. 54th IEEE Conf. Decis. Control*, 2015, pp. 5340–5345.
[30] N. M. M. De Abreu, "Old and new results on algebraic connectivity of graphs," *Linear Algebra Appl.*, vol. 423, no. 1, pp. 53–73, 2007.
[31] R. Hassin and A. Tamir, "On the minimum diameter spanning tree problem," *Inf. Process. Lett.*, vol. 53, no. 2, pp. 109–111, 1995.

Hidde-Jan Jongsma was born in 1989 in Heerenveen, The Netherlands. He received the B.Sc. and M.Sc. degrees in applied mathematics both from The University of Groningen, Groningen, The Netherlands, in 2010 and 2012, respectively, and the Ph.D. degree in Robust Synchronization and Model Reduction of Multi-Agent Systems from the Johann Bernoulli Institute for Mathematics and Computer Science, University of Groningen, in 2017.

His research interests include robust control, distributed control, graph theory, networks and multiagent systems, and model reduction.



Harry L. Trentelman (F'15) received the Ph.D. degree in mathematics from the University of Groningen, Groningen, The Netherlands, in 1985 under the supervision of Jan C. Willems.

He is a Professor of systems and control at the Johann Bernoulli Institute for Mathematics and Computer Science, University of Groningen, Groningen, The Netherlands. From 1991 to 2008, he served as an Associate Professor and later as an Adjoint Professor at the same institute. From 1985 to 1991, he was an Assistant Professor, and later, an Associate Professor at the Mathematics Department, University of Technology at Eindhoven, Eindhoven, The Netherlands. His research interests include the behavioral approach to systems and control, robust control, model reduction, multidimensional linear systems, hybrid systems, analysis, control, and model reduction of networked systems, and the geometric theory of linear systems. He is a co-author of the textbook *Control Theory for Linear Systems* (Springer, 2001).

Dr. Trentelman is a Senior Editor of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, and is Past Associate Editor of *Automatica* and the *SIAM Journal on Control and Optimization and Systems and Control Letters.*.

Kanat M. Camlibel (M'05) received the Ph.D. degree in mathematical theory of systems and control from Tilburg University, Tilburg, The Netherlands, in 2001.

He is an Associate Professor with Johann Bernoulli Institute for Mathematics and Computer Science, University of Groningen, Groningen, The Netherlands, where he served as an Assistant Professor between 2007 and 2013. From 2001 to 2007, he held Postdoctoral/Assistant Professor positions with the University of Groningen, Tilburg University, and Eindhoven Technical University, Eindhoven, The Netherlands. His research interests include differential variational inequalities, complementarity problems, optimization, piecewise affine dynamical systems, switched linear systems, constrained linear systems, multiagent systems, model reduction, and geometric theory of linear systems.