university of groningen
faculty of mathematics and natural sciences

Clustering-based approximation of multi-agent systems defined on weighted directed graphs

Bachelor Project Applied Mathematics

July 2016

Student: J.J. Koning

First supervisor: prof. dr. H.L. Trentelman

Second supervisor: dr. K. Efstathiou

Abstract

This thesis deals with multi-agent systems where the information flow between the agents is represented by a weighted directed graph. We consider a model reduction technique for multi-agent systems, introduced by Monshizadeh *et al.* [4], that preserves the network structure of the original model. We extend the results from [4] to the case where the underlying graph is directed. We will see that the reduced-order model again represents a multi-agent system defined on a weighted directed graph and that consensus is preserved. Furthermore, we will give an explicit expression for the relative approximation error that is involved in the model reduction.

Contents

1 Introduction

This thesis deals with model reduction of multi-agent systems. A multi-agent system is a dynamical system composed of multiple interacting agents within an environment. Each agent receives relative information with respect to its so-called neighbors, and the dynamics of the agent is then influenced by this information and by external inputs. An important research topic in multi-agent systems theory is *consensus*, which roughly means reaching an agreement among the agents on certain quantities of interest. We will briefly discuss consensus in this thesis. Topics where the research on multi-agent systems applies include formation control and placing of mobile sensors, see e.g. [7], [8].

The information flow between the agents of multi-agent systems can be represented by a graph. The agents are then represented by the nodes of the graph and the information flow from one agent to another is represented by an arc between the corresponding nodes. The strength of an information flow is represented by a weight on the corresponding arc. The input the system receives only affects some of the agents, called the *leaders* of the system. The dynamics of the remaining agents, called the *followers*, are then solely influenced by the information they receive from their neighbors. Multi-agent systems are sometimes referred to as leader-follower multi-agent systems.

In many cases, the analysis of a multi-agent system requires unmanageable storage capacity and computational requirements due to high dimensionality of the system. Reduced models that approximate the original model are essential in such cases. Many model reduction methods have been developed, such as balanced truncation, Hankel-norm approximation and Krylov projection see e.g. [2]. However, for most of the classical model reduction methods the network structure of the original model is not preserved in the reduced-order model. Techniques that do preserve the network structure are available in the literature, see e.g. [4], [10]. In this thesis we will generalize the clustering-based model reduction method that was introduced by Monshizadeh *et al.* in [4]. As noted, this method has the key property that the reduction procedure preserves the network structure, in the sense that the reduced-order model again represents a multi-agent system. The idea behind this method is to form clusters of agents of the original multi-agent system and then to project each cluster to a single agent of the reduced multi-agent system. The clustering of the agents can be interpreted as a partitioning of the underlying graph. The projection that is used in the model reduction method is then formulated in terms of the characteristic matrix of this partition.

For multi-agent systems defined on weighted undirected graphs, the proposed model reduction method has the nice property that the reduction procedure preserves the property of reaching consensus [4]. That is, if in time an agreement is reached among the agents of the original model, then the reduced-order model reaches an agreement among its agents as well. Another advantage of this method is the preservation of the network structure. Therefore, analysis methods that have been developed for systems with this network structure can be applied to both the original and the reduced-order multi-agent system, and the outcomes can be compared. Furthermore, we might be able to predict certain kinds of behaviour of the original multi-agent system by analysis of the reduced-order multi-agent system and in that way get around very involved analysis of the original model.

Clearly, if one performs a model reduction, it is of interest how the behaviour of the reduced-order model compares to the behaviour of the original model. In order to make a comparison, we assign certain outputs to the systems. For these, we choose quantities that represent differences between the states of the agents. This is a reasonable choice, since these differences are crucial values in distributed control. Then we compare the input-output behaviour of both models, that is, we compare the transfer matrices of the obtained input-output systems. We will consider as approximation error the H_2-norm of the difference between the transfer matrices of the systems. For multi-agent systems defined on weighted undirected graphs, an explicit expression for the relative approximation error in this sense was already provided in [4].

Monshizadeh *et al.* only consider multi-agent systems defined on weighted *undirected* graphs, and therefore most of the results from [4] no longer apply in the case of multi-agents defined on weighted *directed* graphs. The objective of this thesis is to extend the results of [4] to the case of weighted directed graphs. We will investigate what happens if the model reduction method of [4] is applied to a multi-agent system defined on a weighted directed graph. We will see that also in this case, the reduction procedure preserves both the network structure of the system and the property of reaching consensus. Finally, our main result is an explicit expression for the relative approximation error for multi-agent systems defined on weighted directed graphs.

The outline of this thesis is as follows. First we provide some preliminaries on graph theory, graph partitions and multi-agent systems. In Chapter 3 we discuss the model reduction method and the projection it uses. Then we work out an example of this projection. In Chapter 4 we investigate the spanning trees of graphs. After that, we compare the input-output behaviour of the original and the obtained reduced-order model in Chapter 5. In this chapter, we first assign an output to both the original and the reduced-order model. Then we work towards the main result of this thesis, presented in Section 5.2, where we provide an explicit expression for the approximation error. We will then give an example of the approximation of a multi-agent system. In Chapter 7 we conclude the thesis by discussing our main results and making some suggestions for further research. Finally, there is an Appendix containing the proofs of two lemmas.

2 Preliminaries

In this chapter we provide some preliminaries that are needed to state the results of this thesis. We will discuss some graph theory and then we will give a definition of a multi-agent system defined on a graph.

2.1 Graph theory

In this thesis we deal with multi-agent systems defined on weighted directed graphs. Here we will give the definition of both weighted directed graphs and weighted undirected graphs. The multi-agent systems we consider are defined on simple graphs, that is, the graph does not contain any edges or arcs that connect a node to itself and multiple edges or directed arcs in the same direction between two nodes are not permitted. For a weighted directed graph, let $V = \{1, 2, \ldots, n\}$ denote the set of nodes and let $E \subset V \times V$ denote the set of arcs of the graph. For $i, j \in V$ we say $(i, j) \in E$ if there is an arc directed from i to j in the graph. In that case we denote the corresponding weight by a_{ji}, and if there is no arc going from i to j we have $a_{ji} = 0$. Note that for all $i \in V$ we have $a_{ii} = 0$, since the graph is simple. Then a weighted directed graph is defined by the triple $G = (V, E, A)$, where $A = (a_{ij})$ denotes the $n \times n$ adjacency matrix. A graph is called *undirected* if $a_{ij} = a_{ji}$ for all $i \neq j$. In that case the arc from i to j carries the same weight as the arc from j to i. The two arcs are then identified with a single object, called an edge, carrying the weigth $a_{ij} = a_{ji}$. Clearly, the adjacency matrix is symmetric for undirected graphs.

For both weighted directed graphs and weighted undirected graphs, we define the *indegree* of node $i \in V$ as

$$d_i = \sum_{j=1}^{n} a_{ij}. \tag{1}$$

Then the degree matrix of G is given by the diagonal matrix $\Delta = \mathrm{diag}(d_1, d_2, \ldots, d_n)$. The Laplacian matrix of G is defined as $L(G) = \Delta - A$, denoted shortly by L. By definition we have $L\mathbb{1} = 0$, where $\mathbb{1}$ denotes the vector of ones of dimension n. In other words, $\mathbb{1}$ is a right eigenvector of L associated with the zero eigenvalue.

A graph is called *balanced* if $\sum_{i=1}^{n} a_{ij} = \sum_{i=1}^{n} a_{ji}$ for all nodes i. In a balanced graph, each node satisfies the property that the total weight of incoming arcs is equal to the total weight of outgoing arcs. The most important property of balanced graphs is that $\mathbb{1}$ is also a left eigenvector of L associated with the zero eigenvalue, that is, $\mathbb{1}^{\mathrm{T}} L = 0$. Note that an undirected graph is automatically balanced.

In a directed graph, a *directed path* from i_1 to i_k is a sequence of arcs $\{(i_1, i_2), (i_2, i_3), \ldots, (i_{k-1}, i_k)\} \subset E$. The graph G is said to be *strongly connected* if for any two nodes $i, j \in V$ there exists a directed path from i to j. A node i is called a *root* of G if there exists a directed path to every other node of the graph. Note that in a strongly connected graph every node is a root node. We define a *cycle* as a sequence of distinct nodes, where there is an arc between each two consecutive nodes in the sequence and between the last and the first node in the sequence, regardless of the direction of the arc.

A graph is said to be a subgraph of $G = (V, E, A)$ if it has node set V and the arc set is a subset of E. A subgraph T of G is said to be a *spanning tree* of G if T has a root and contains no cycles. If node i is the root of a spanning tree T, we say that T is rooted at i. Note that for a spanning tree, the root has indegree 0 and every other node has exactly one incoming arc. For a spanning tree T of $G = (V, E, A)$, we define the *weight* $w(T)$ of T as the product of the weights corresponding to the arcs in T. We define the *spanning tree vector* of G as the vector $\mu = \begin{pmatrix}\mu_1 & \mu_2 & \cdots \mu_n\end{pmatrix}^{\mathrm{T}}$ where μ_i is equal to the sum of the weights of all spanning trees rooted at node i. If there are no spanning trees rooted at node i, we have $\mu_i = 0$. We define the *spanning tree matrix* of G as $D = \operatorname{diag}(\mu_1, \mu_2, \ldots, \mu_n)$.

Let $G = (V, E, A)$ be a weighted directed graph. We call a nonempty subset of V a *cell* of V. A collection of cells, given by $\pi = \{C_1, \ldots, C_r\}$ is called a *partition* of V, if $\cup_i C_i = V$ and $C_i \cap C_j = \emptyset$ for $i \neq j$. With a little abuse of notation, we will say π is a partition of G. The *characteristic vector* of a cell C is defined as the n-dimensional column vector $p(C)$, given by

$$p_i(C) = \begin{cases} 1 & \text{if } i \in C, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

We define the *characteristic matrix* of the partition $\pi = \{C_1, C_2, \ldots, C_r\}$ as the $n \times r$ matrix

$$P(\pi) = \begin{pmatrix} p(C_1) & p(C_2) & \cdots & p(C_r) \end{pmatrix}, \tag{3}$$

which we will denote shortly by P.

2.2 Multi-agent systems

By a multi-agent system, we mean a dynamical system composed of a group of identical input-output systems, called the agents, that interact by exchanging information with their neighbors. The interaction can be represented by a graph, where the nodes represent the agents and the arcs between the nodes represent the interaction between the agents. As mentioned earlier, in this thesis we will discuss multi-agent systems for which the underlying graph is directed. Now, let $G = (V, E, A)$ be a weighted directed graph and let x_i denote the state of agent $i \in V$. Then a multi-agent system with underlying graph G is given by the following dynamical system:

$$\dot{x}_i = \sum_{j=1}^{n} a_{ij}(x_j - x_i), \tag{4}$$

where a_{ij} denotes the (i, j)-th element of the adjacency matrix.

A multi-agent system is said to reach consensus if the difference between the states of each pair of agents converges to zero, i.e.,

$$\lim_{t\to\infty} x_i - x_j = 0 \quad \text{for all } i, j \in V.$$

In a leader-follower multi-agent system defined on a graph $G = (V, E, A)$, we apply an external input to some of the agents of the graph, which we call the *leaders* of G. The agents that do not receive any input are called *followers*. Let $V_L = \{v_1, v_2, \ldots, v_m\}$ be a subset of V denoting the set of leaders of G and let $V_F = V \setminus V_L$ denote the set of followers of G. For each $l = 1 \ldots, m$ we apply a certain input function u_l to agent $v_l \in V_L$. Then the leader-follower multi-agent system with leader set V_L and follower set V_F is defined as the following dynamical system:

$$\dot{x}_i = \begin{cases} z_i & \text{if } i \in V_F \\ z_i + u_l & \text{if } i = v_l, \end{cases} \tag{5}$$

where $z_i \in \mathbb{R}$ is given by

$$z_i = \sum_{j=1}^{n} a_{ij}(x_j - x_i).$$

Recall that a_{ij} denotes the (i, j)-th element of the adjacency matrix of the underlying graph. Let $x = \mathrm{col}(x_1, x_2, \dots, x_n)$, $u = \mathrm{col}(u_1, u_2, \dots, u_m)$, and define the matrix $M \in \mathbb{R}^{n\times m}$ as

$$M_{il} = \begin{cases} 1 & \text{if } i = v_l, \\ 0 & \text{otherwise.} \end{cases}$$

Recalling that L is the Laplacian matrix of the graph, the leader-follower multi-agent system as defined in (5) can then be written as

$$\dot{x} = -Lx + Mu. \tag{6}$$

3 Projection by graph partitions

In this chapter we will extend the model reduction technique as introduced in [4] to *directed* graphs. First we will briefly introduce the method and show that the obtained reduced-order model again represents a multi-agent system defined on a graph. Then we will discuss the preservation of consensus.

The projection that is used is a Petrov-Galerkin projection $\Gamma = VW^{\mathrm{T}}$, expressed in terms of the characteristic matrix $P = P(\pi)$ of a graph partition π of the graph G of our own choice. The matrices W and V are given by

$$\begin{aligned} W &= P(P^{\mathrm{T}}P)^{-1} \\ V &= P. \end{aligned}$$

Then the following reduced-order model is obtained:

$$\dot{\hat{x}} = -\hat{L}\hat{x} + \hat{M}u, \tag{7}$$

where $\hat{x} \in \mathbb{R}^r$ denotes the state of the reduced-order model, and the matrices $\hat{L}$ and $\hat{M}$ are given by

$$\hat{L} = (P^{\mathrm{T}}P)^{-1}P^{\mathrm{T}}LP, \tag{8}$$

$$\hat{M} = (P^{\mathrm{T}}P)^{-1}P^{\mathrm{T}}M. \tag{9}$$

In the following, we show that the system (7) is associated with a leader-follower multi-agent system defined on a graph. First, we show that $\hat{L}$ is equal to the Laplacian matrix of a weighted directed graph. To see this, let $\hat{G} = (\hat{V}, \hat{E}, \hat{A})$ be the weighted directed graph with r nodes and $r \times r$ adjacency matrix $\hat{A} = (\hat{a}_{pq})$, defined as

$$\hat{a}_{pq} = \frac{1}{|C_p|} \sum_{i\in C_p, j\in C_q} a_{ij} \quad \text{for } p, q \in \{1, 2, \dots, r\}. \tag{10}$$

Observe by (8) that for $p \neq q$, the (p, q)-th element of $\hat{L}$ is given by $\hat{l}_{pq} = -\hat{a}_{pq}$. Furthermore, note that $P\mathbb{1} = \mathbb{1}$, and therefore $\hat{L}\mathbb{1} = 0$, where $\mathbb{1}$ denotes the vector of ones of appropriate dimension. Hence, for $p = 1, 2, \dots, r$ we have $\sum_{q=1}^{r} \hat{l}_{pq} = 0$, and it follows that

$$\hat{l}_{pp} = -\sum_{q\neq p} \hat{l}_{pq} = \sum_{q=1}^{r} \hat{a}_{pq}.$$

Note that this is exactly the indegree of node p. Thus we can write the degree matrix of $\hat{G}$ as $\hat{\Delta} = \mathrm{diag}(\hat{l}_{11}, \hat{l}_{22}, \ldots, \hat{l}_{rr})$. Now, observe that $\hat{L} = \hat{\Delta} - \hat{A}$, so by definition, $\hat{L}$ is equal to the Laplacian matrix of the graph $\hat{G}$.

Furthermore, observe that each column of $\hat{M}$ contains exactly one nonzero element. If for some column of $\hat{M}$ the, say, p^{th} element is nonzero, then by (7) it is clear that we apply a certain input to agent p. In fact, as one easily sees, agent p is a leader of $\hat{G}$ if and only if the cell C_p contains a leader of G. The corresponding input we apply to node p is then equal to the average of all inputs we apply to the nodes in C_p. It is now clear that (7) is associated with a leader-follower multi-agent system defined on a weighted directed graph.

The idea behind the projection we use is that the partition π clusters the nodes of graph G together in cells, and the nodes in each cell are then mapped to a single node in $\hat{G}$. As for the communication between the nodes, observe that by equation (10) we have $\hat{a}_{pq} \neq 0$ if and only if $a_{ij} \neq 0$ for some $i \in C_p$, $j \in C_q$. I.e., there is an arc from a node q to another node p in $\hat{G}$ if and only if there is an arc from a node in cell C_q to a node in cell C_p in the original graph G. Furthermore, note that the associated weight $\hat{a}_{pq}$ is equal to the sum of the weights of all arcs going from a node in C_q to a node in C_p, divided by the number of nodes in cell C_p.

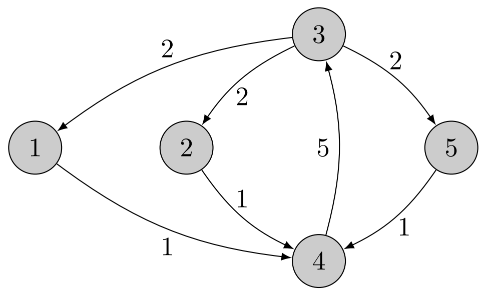

Figure 1: Directed graph.

Example 1. In this example we consider the multi-agent system with underlying graph G depicted in Figure 1, where the agents 1 and 3 are the leaders of the system. The multi-agent system is given by

$$\dot{x} = -Lx + Mu, \tag{11}$$

where

$$L = \begin{pmatrix} 2 & 0 & -2 & 0 & 0 \\ 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 5 & -5 & 0 \\ -1 & -1 & 0 & 3 & -1 \\ 0 & 0 & -2 & 0 & 2 \end{pmatrix}, \qquad M = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}. \tag{12}$$

We partition the nodes of the graph as

$$\pi = \{C_1, C_2, C_3\} = \{\{1, 2, 5\}, \{3\}, \{4\}\}. \tag{13}$$

Observe that the corresponding characteristic matrix P is given by

$$P = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}^{\mathrm{T}}.$$

By applying the model reduction to (11) using the partition π, we obtain the following reduced-order model

$$\dot{\hat{x}} = -\hat{L}\hat{x} + \hat{M}u, \tag{14}$$

where the matrices $\hat{L}$ and $\hat{M}$ are given by

$$\hat{L} = (P^{\mathrm{T}}P)^{-1}P^{\mathrm{T}}LP = \begin{pmatrix} 2 & -2 & 0 \\ 0 & 5 & -5 \\ -3 & 0 & 3 \end{pmatrix}, \qquad \hat{M} = (P^{\mathrm{T}}P)^{-1}P^{\mathrm{T}}M = \begin{pmatrix} \frac{1}{3} & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}.$$

The underlying graph $\hat{G}$ of the reduced-order multi-agent system is depicted in Figure 2.

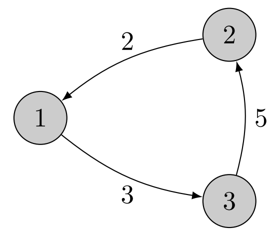

Figure 2: Reduced directed graph.

As mentioned before, the idea behind this model reduction is to project each cluster of agents to a single agent of the reduced multi-agent system. Observe that the number of agents in $\hat{G}$ is indeed equal to the number of cells in π. The arcs in $\hat{G}$ represent the information flow between the cells of G. For instance, the arc $(2,1)$ in $\hat{G}$ represents the information flow from agents in cell 2 to agents in cell 1 in G. In fact, note that the associated weight of this arc is 2, which is equal to the sum of the weights of all arcs in G going from a node in cell $C_2 = \{3\}$ to a node in cell $C_1 = \{1,2,5\}$, divided by $|C_1| = 3$. Observe that this is in accordance with (10). As for the input we apply to the reduced multi-agent system, note that $\hat{M}$ indicates input weights, which depend on the cardinality of the corresponding cell in G. $\square$

The following theorem states that the property of reaching consensus is preserved if we perform the model reduction.

Theorem 1. *Let the multi-agent system* (6) *be defined on a weighted directed graph G with n nodes. Suppose that with $u = 0$ the system reaches consensus. Then for any partition π the reduced-order system*

$$\dot{\hat{x}} = -\hat{L}\hat{x} \tag{15}$$

also reaches consensus, where $\hat{L}$ is given by (8) *and $\hat{x} \in R^r$ denotes the state of the reduced multi-agent system.*

Proof. Since (6) reaches consensus for $u = 0$, we have $Lx = 0$ if and only if $x = \alpha\mathbb{1}$ for some nonzero $\alpha \in \mathbb{R}$. Hence, $\operatorname{rank} L = n-1$ and it follows by [6, Proposition 1] that G has a spanning tree. Denote the graph that defines the communication of the reduced multi-agent system by $\hat{G}$. Recall that there is an arc from a node p to another node q in $\hat{G}$ if and only if there is an arc from a node in cell C_p to a node in cell C_q in the original graph G. Then clearly, since G has a spanning tree, $\hat{G}$ has a spanning tree as well. Again by Proposition 1 from [6] it follows that $\operatorname{rank} \hat{L} = r-1$, and therefore the reduced multi-agent system given by (15) reaches consensus. $\square$

4 Spanning trees

Before we move on to the main results of this thesis, we will study the spanning trees of graphs. In this chapter we will obtain some results that are needed before we investigate the input-output approximation of multi-agent systems using the model reduction that was introduced in the previous chapter.

Recall that for a weighted directed graph we defined the spanning tree vector $\mu(G)$ as the vector whose i^{th} component is equal to the sum of the weights of all spanning trees of G rooted at i. We then defined the spanning tree matrix of a graph G as the diagonal matrix $D = \mathrm{diag}(\mu_1, \mu_2, \ldots, \mu_n)$, where n is the number of nodes in G. In this chapter we will prove that the spanning tree vector is a left eigenvector of the Laplacian matrix of the graph associated with the zero eigenvalue, i.e., $\mu^{\mathrm{T}} L = 0$. For this, we will first investigate some properties of the spanning tree vector. We will investigate how a change in the graph, such as deletion of an arc, affects the elements of this vector. First, we will introduce the procedures of cutting and contracting an arc from G. We will provide an expression for $\mu(G)$ in terms of the spanning tree vectors of the resulting graphs. After that, we will provide an expression for the spanning tree vector in terms of the Laplacian matrix.

Let $G = (V, E, A)$ be a strongly connected weighted directed graph with n nodes and suppose $(i, j) \in E$. We denote the graph that is obtained from G by *contracting* the arc (i, j) by $G/(i, j)$. By contracting the arc (i, j), we mean that we delete the arc and merge the nodes i and j to a new node, which we give the label i. If (j, i) is also an arc in G, then this arc is deleted as well. Every other outgoing arc from either i or j in G becomes an outgoing arc from i in $G/(i, j)$ and every other incoming arc to either i or j in G becomes an incoming arc to i in $G/(i, j)$. Figure 3 shows an example of this process. Note that $G/(i, j)$ is a directed multigraph with $n - 1$ nodes, that is, there may exist multiple arcs in the same direction between two nodes.

(a) Graph G (b) Graph $G/(1, 2)$

Figure 3: Contracting arc $(1, 2)$.

For a multigraph, the (p, q)-th element of the adjacency matrix is given by the sum of the weights of all arcs going from node q to node p. Thus, the Laplacian matrices of the graphs depicted in Figure 3 are given by

$$L(G) = \begin{pmatrix} 2 & -1 & 0 & -1 \\ -1 & 3 & 0 & -2 \\ 0 & -4 & 4 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix} \qquad L(G/(1,2)) = \begin{pmatrix} 3 & 0 & -3 \\ -4 & 4 & 0 \\ 0 & -1 & 1 \end{pmatrix}$$

Let $G - (i, j)$ denote the graph that is obtained from G by cutting (deleting) the arc (i, j). The following lemma provides an expression for the spanning tree vector of G. The idea of the lemma is that the number $\mu_i(G - (i, j))$ is equal to the sum of the weights of the spanning trees rooted at i that do not contain arc (i, j), and that the number $a_{ji}\mu_i(G/(i, j))$ is equal to the sum of the weights of the remaining spanning trees rooted at i. This result was already mentioned for undirected graphs in [3, equation (4) on p. 57]. Here, we extend this result to directed graphs. The proof can be found in the Appendix.

Lemma 1. . *Let G be a weighted directed graph and let $\mu(G)$ denote the spanning tree vector of G. Suppose $(i, j) \in E$ for some $i, j \in V$. Then*

$$\mu_i(G) = \mu_i(G - (i, j)) + a_{ji}\mu_i(G/(i, j)). \tag{16}$$

Proof. See Appendix. $\square$

Let G be a weighted directed graph with n nodes and let $L(G)$ denote the $n \times n$ Laplacian matrix of G. Let $L_i(G)$ denote the $(n-1) \times (n-1)$ matrix that is obtained from $L(G)$ by removing the row and column that correspond to node i. With this notation and using Lemma 1, we can provide an expression for the spanning tree vector of G in terms of the Laplacian matrix. The same expression was mentioned in [5, Section 3].

Lemma 2. *Let G be a weighted directed graph with at least two nodes, let $L(G)$ denote the corresponding Laplacian matrix and let $\mu_i(G)$ denote the spanning tree vector of G. Then for every node i of G we have*

$$\det L_i(G) = \mu_i(G). \tag{17}$$

Proof. Here we will only give an outline of the proof. For a full proof, see the Appendix. The proof is by induction on the number of arcs of G. If G has no arcs at all, $\det L_i = 0$ for all i. In that case we also have $\mu_i(G) = 0$, so the statement follows for graphs with with no arcs. Now, suppose the statement holds for graphs for which the number of arcs is less than or equal to k. Let $G = (V, E, A)$ be a graph with $k+1$ arcs and let $i \in V$. Suppose $(i, j) \in E$ for some $j \in V$. Then by Lemma 1 we have

$$\mu_i(G) = \mu_i(G - (i, j)) + a_{ji}\mu_i(G/(i, j)). \tag{18}$$

It can be shown that a similar relation holds for the determinant of $L_i(G)$:

$$\det L_i(G) = \det L_i(G - (i, j)) + a_{ji} \det L_i(G/(i, j)). \tag{19}$$

Observe that both $G - (i, j)$ and $G/(i, j)$ contain at most k arcs, so by induction hypothesis, the right-hand sides of (18) and (19) are equal. We conclude that $\mu_i(G) = \det L_i$, which completes the proof. □

With the results stated in the previous lemmas, we are now able to prove the surprising result stated in the following theorem.

Theorem 2. *Let G be a weighted directed graph, let L denote the Laplacian matrix of G and let $\mu = \mu(G)$ denote the spanning tree vector of G. Then*

$$\mu^T L = 0.$$

Proof. Denote the (i, j)-th cofactor of L by $C_{ij} = (-1)^{i+j} M_{ij}$, where M_{ij} is the determinant of the (i, j)-th minor of L. Observe that $C_{ii} = \det L_i$ for all i. For the adjoint matrix $\operatorname{adj}(L)$, defined as the $n \times n$ matrix with (i, j)-th element given by C_{ji}, it is well-known that

$$\operatorname{adj}(L)L = L \operatorname{adj}(L) = \det(L)I = 0. \tag{20}$$

Then it follows that the columns of $\operatorname{adj}(L)$ are in $\ker(L) = \operatorname{span}\{\mathbb{1}\}$, which implies that the (i, j)-th element of $\operatorname{adj}(L)$ does not depend on i, i.e. C_{ji} does not depend on i. It follows that $\det L_i = C_{ii} = C_{i1}$ for all i, so by Lemma 2 the spanning tree vector of G can be written as

$$\mu = \begin{pmatrix} C_{11} & C_{21} & \cdots & C_{n1} \end{pmatrix}^{\mathrm{T}}. \tag{21}$$

Hence, μ^{T} is equal to the first row of $\operatorname{adj}(L)$, so by (20) we have $\mu^{\mathrm{T}} L = 0$. □

5 Approximation of multi-agent systems

If we are given a multi-agent system, we can perform the model reduction technique that was introduced in Chapter 3, using a graph partition of our own choice, to obtain a reduced multi-agent system that approximates the original system. We saw that the property of reaching consensus is preserved if we perform this model reduction. In this chapter we will investigate the error that is involved in approximating the multi-agent system for a certain class of graph partitions. We will assign an output to both the original and the reduced multi-agent system and compare the corresponding transfer matrices of the systems. We will compute the H_2-norm of the difference of the transfer matrices of the original and the reduced system. Throughout this section we assume that the multi-agent system is defined on a strongly connected graph.

5.1 Input-output approximation

In this section we will assign outputs to the multi-agent systems given by (6) and (7). Since the disagreement among the agents plays a crucial role in distributed control, it is reasonable to take the differences of the states of the agents as outputs for the systems. Note that these differences are reflected by the Laplacian matrix L of the underlying graph G of the multi-agent system. In fact, the i^{th} element of Lx is equal to the right-hand side of (4), which represents the disagreement of agent i with its neighbors in accordance with the weights of the corresponding information flows. Also note that the influence of an agent on the entire system is reflected by the spanning trees in the underlying graph that are rooted at the corresponding node. If for some agent i the sum of the weights of the spanning trees of the graph rooted at node i, μ_i, is relatively large, the agent has much influence on the dynamics of the entire system. Hence, the vector DLx, where D denotes the spanning tree matrix, represents the disagreement of each agent with its neighbors, multiplied by its influence on the entire system.

As a matter of fact, DL denotes the Laplacian matrix of a strongly connected balanced graph. To see this, observe that as the underlying graph of the multi-agent system is strongly connected, each node of the graph is a root. Hence, each term of the spanning tree matrix D is positive. Therefore left-multiplying the Laplacian matrix by the diagonal matrix D simply means that we multiply each row of L by a positive number, so the properties of Laplacian matrices still hold for DL. It follows that DL is the Laplacian matrix of some graph G^*. Furthermore, note that G^* is obtained from G by multiplying the weights of the arcs by some positive numbers, but the network topology remains the same. Therefore, G^* is strongly connected as well. Now, observe that $\mathbb{1}^{\mathrm{T}}D = \mu^{\mathrm{T}}$, where μ is the spanning tree vector of G. Then it follows from Theorem 2 that $\mathbb{1}^{\mathrm{T}}DL = 0$, so G^* is a balanced graph.

Since DL is the Laplacian matrix of a balanced graph, it follows from [11, Lemma 1.2] that $x^{\mathrm{T}}DLx \geq 0$ for all $x \in R^n$. Hence, the symmetric matrix $DL + L^{\mathrm{T}}D$ is positive semi-definite, and therefore it has a square root. Then the output we assign to system (6) is given by

$$y = (DL + L^{\mathrm{T}}D)^{\frac{1}{2}}x.$$

Observe that the disagreement of each agent with its neighbors, multiplied by its influence on the system is reflected in the output. Furthermore, as G^* is strongly connected , again by [11, Lemma 1.2] we have that $||y||^2 = x^{\mathrm{T}}(DL + L^{\mathrm{T}}D)x = 0$ if and only if the agents have reached an agreement regarding their states.

The original model is now represented by

$$\dot{x} = -Lx + Mu \tag{22a}$$

$$y = (DL + L^{\mathrm{T}}D)^{\frac{1}{2}}x. \tag{22b}$$

and the reduced-order model by

$$\dot{\hat{x}} = -\hat{L}\hat{x} + \hat{M}u \tag{23a}$$

$$y = (DL + L^{\mathrm{T}}D)^{\frac{1}{2}}P\hat{x}. \tag{23b}$$

If we decide upon the partition we use, we only consider the graph topology, as the dynamics of the individual nodes are identical. For any graph G there are two trivial partitions. The first one is the partition with one cell containing all nodes and the second is the partition in which each node appears as a singleton. In the first case we obtain a reduced order model with state space dimension $r = 1$, whereas in the second case the approximation error is zero. Clearly, deciding which partition we take is a compromise between the order of the reduced model and the accuracy of the approximation.

In order to obtain an accurate approximation, we want to cluster agents that are connected to the rest of the network in a similar way. To formalize this idea, we introduce the notion of almost equitability of a partition. We will distinguish the class of almost equitable partitions from other partitions.

In order to define almost equitability of a partition, we first introduce the definition of the in- and outdegree of a node with respect to a cell of π. Let $\pi = \{C_1, C_2, \ldots, C_r\}$ be a partition of G. For $p, q \in$

$\{1, 2, \dots, r\}$ and $i \in C_p$ we define the *outdegree of i with respect to cell C_q* as the number

$$w_{pq}^{\text{out}}(i) := \sum_{j \in C_q} a_{ji},$$

and the *indegree of i with respect to cell C_q* as

$$w_{pq}^{\text{in}}(i) := \sum_{j \in C_q} a_{ij}.$$

For a weighted directed graph $G = (V, E, A)$, we call a partition $\pi = \{C_1, C_2, \dots, C_r\}$ an *almost equitable partition* (AEP) of G if for each $p, q \in \{1, 2, \dots, r\}$, $w_{pq}^{\text{in}}(i)$ and $w_{pq}^{\text{out}}(i)$ do not depend on $i \in C_p$, i.e.

$$w_{pq}^{\text{in}}(i_1) = w_{pq}^{\text{in}}(i_2)$$

and

$$w_{pq}^{\text{out}}(i_1) = w_{pq}^{\text{out}}(i_2)$$

for all $i_1, i_2 \in C_p$. Then, for all p, q with $p \neq q$ we have for all $i \in C_p$,

$$w_{pq}^{\text{in}}(i) = \hat{a}_{pq}, \tag{24}$$

with $\hat{a}_{pq}$ the (p, q)-th entry of the reduced Laplacian, as given by (10). Also, there exists $y_{pq} \in \mathbb{R}$ such that $w_{pq}^{\text{out}}(i) = y_{pq}$ for all $i \in C_p$.

Note that for undirected graphs, we have $w_{pq}^{\text{out}}(i) = w_{pq}^{\text{in}}(i)$, so in this case a partition is called almost equitable if $w_{pq}^{\text{in}}(i)$ does not depend on $i \in C_p$. For undirected graphs an AEP has the important property that $\operatorname{im} P$ is L-invariant (see e.g. [4]). It can be shown that the L-invariance property of $\operatorname{im} P$ remains valid in the case of weighted directed graphs (see e.g. [1]). However, for undirected graphs the Laplacian matrix L satisfies some useful properties that no longer apply in the case of directed graphs. For example, for undirected graphs L is symmetric and therefore $\operatorname{im} P$ is both L- and L^{T}-invariant. For directed graphs L is not necessarily symmetric. As we will see later on, in order for the L^{T}-invariance property to remain valid in the case of directed graphs, the partition must satisfy an additional condition. For this, we introduce the definition of *balanced* partition.

A partition $\pi = \{C_1, C_2, \dots, C_r\}$ is called *balanced* if for each $p \in \{1, 2, \dots, r\}$ we have

$$w_{pp}^{\text{out}}(i) = w_{pp}^{\text{in}}(i) \tag{25}$$

for all $i \in C_p$. In other words, π is balanced if and only if for each p the subgraph with node set C_p and arc set $E_p = \{(i, j) \in E \mid i, j \in C_p\}$ is a balanced graph.

For the graph depicted in Figure 1, the partition π given by (13) is an example of a balanced almost equitable partition.

Note that the definition of almost equitable partition for weighted directed graphs includes almost equitable partitions for weighted undirected graphs as a special case, since for undirected graphs for each $p, q \in \{1, 2, \dots, r\}$ we have $w_{pq}^{\text{out}}(i) = w_{pq}^{\text{in}}(i)$ for all $i \in C_p$. Also note that a partition of an undirected graph is automatically balanced. Moreover, for a balanced almost equitable partition of a weighted directed graph $\operatorname{im} P$ is both L- and L^{T}-invariant, as stated in the following lemma.

Lemma 3. *Let $\pi = \{C_1, C_2, \dots, C_r\}$ be a partition of a weighted directed graph G and let L denote the Laplacian matrix of G. Then π is a balanced almost equitable partition if and only if there exist matrices X and Y such that*

$$\begin{aligned} LP &= PX, \\ L^T P &= PY. \end{aligned} \tag{26}$$

In that case we have $X = \hat{L}$ and $Y = (P^T P)^{-1} \hat{L}^T P^T P$.

Proof. Let $p \in \{1, 2, \ldots, r\}$. Observe that for all $i \in C_p$, by definition of the indegree d_i in (1) we have

$$d_i = \sum_q w_{pq}^{\text{in}}(i). \tag{27}$$

Let $i \in C_p$. Then we can write the (i, p)-th element of the matrix product LP as

$$(LP)_{ip} = d_i - \sum_{j \in C_p} a_{ij} = \sum_{q \neq p} w_{pq}^{\text{in}}(i). \tag{28}$$

Furthermore, for any $q \in \{1, 2, \ldots, r\}$ with $q \neq p$ the (i, q)-th element of LP is given by

$$(LP)_{iq} = \sum_{j \in C_q} -a_{ij} = -w_{pq}^{\text{in}}(i). \tag{29}$$

Also, the (i, p)-th element of $L^{\mathrm{T}}P$ is given by

$$(L^{\mathrm{T}}P)_{ip} = d_i - \sum_{j \in C_p} a_{ji} = d_i - w_{pp}^{\text{out}}(i). \tag{30}$$

and for any $q \in \{1, 2, \ldots, r\}$ with $q \neq p$, its (i, q)-th element is given by

$$(L^{\mathrm{T}}P)_{iq} = \sum_{j \in C_q} -a_{ji} = -w_{pq}^{\text{out}}(i). \tag{31}$$

Now, suppose π is a balanced almost equitable partition of G. Using (28) and (29), we can write the i^{th} row of LP as

$$\begin{aligned}(LP)_i &= \begin{pmatrix} -w_{p1}^{\text{in}}(i) & \cdots & -w_{p,p-1}^{\text{in}}(i) & \sum\limits_{q \neq p} w_{pq}^{\text{in}}(i) & -w_{p,p+1}^{\text{in}}(i) & \cdots & -w_{pr}^{\text{in}}(i) \end{pmatrix} \\ &= \begin{pmatrix} -\hat{a}_{p1} & \cdots & -\hat{a}_{p,p-1} & \sum\limits_{q \neq p} \hat{a}_{pq} & -\hat{a}_{p,p+1} & \cdots & -\hat{a}_{pr} \end{pmatrix},\end{aligned} \tag{32}$$

as follows from (24). One easily checks that the right-hand side of (32) is equal to the i^{th} row of $P\hat{L}$ and therefore, $LP = P\hat{L}$. Since π is an AEP, $w_{pq}^{\text{out}}(i)$ and $w_{pq}^{\text{in}}(i)$ do not depend on $i \in C_p$ if $p \neq q$, and therefore we can define the matrix $Y = (y_{pq})$ with $p, q \in \{1, 2, \ldots, r\}$ as

$$y_{pq} = \begin{cases} -w_{pq}^{\text{out}}(i) & \text{if } p \neq q, \\ \sum\limits_{j \neq p} w_{pj}^{\text{in}}(i) & \text{if } p = q, \end{cases}$$

where i is any node in C_p. For $i \in C_p$, using (30) and (31) we can write the i^{th} row of $L^{\mathrm{T}}P$ as

$$\begin{aligned}(L^{\mathrm{T}}P)_i &= \begin{pmatrix} -w_{p1}^{\text{out}}(i) & \cdots & -w_{p,p-1}^{\text{out}}(i) & d_i - w_{pp}^{\text{out}}(i) & -w_{p,p+1}^{\text{out}}(i) & \cdots & -w_{pr}^{\text{out}}(i) \end{pmatrix} \\ &= \begin{pmatrix} -w_{p1}^{\text{out}}(i) & \cdots & -w_{p,p-1}^{\text{out}}(i) & \sum\limits_{q \neq p} w_{pq}^{\text{in}}(i) & -w_{p,p+1}^{\text{out}}(i) & \cdots & -w_{pr}^{\text{out}}(i) \end{pmatrix} \\ &= \begin{pmatrix} y_{p1} & \cdots & y_{p,p-1} & y_{pp} & y_{p,p+1} & \cdots & y_{pr} \end{pmatrix},\end{aligned}$$

where the second equality follows from (27) and (25). Clearly, this is equal to the i^{th} row of PY and hence, $L^{\mathrm{T}}P = PY$. Then it directly follows that $P^{\mathrm{T}}PY(P^{\mathrm{T}}P)^{-1} = \hat{L}^{\mathrm{T}}$ and therefore $Y = (P^{\mathrm{T}}P)^{-1}\hat{L}^{\mathrm{T}}P^{\mathrm{T}}P$.

Conversely, suppose π is a partition of G that satisfies (26) for some matrices X and Y. Let $p, q \in \{1, 2, \ldots, r\}$ with $p \neq q$. By definition, the i^{th} row of P is constant for $i \in C_p$, so $(PX)_{iq}$ does not depend on $i \in C_p$. From (29) it follows that $(PX)_{iq} = (LP)_{iq} = -w_{pq}^{\text{in}}(i)$, and therefore $w_{pq}^{\text{in}}(i)$ does not depend on

$i \in C_p$. Similarly, $(PY)_{iq} = (L^{\mathrm{T}}P)_{iq}$ does not depend on $i \in C_p$. From (31) it then follows that $w_{pq}^{\text{out}}(i)$ does not depend on $i \in C_p$ and hence π is almost equitable.

Since $(PY)_{ip} = (L^{\mathrm{T}}P)_{ip}$ does not depend on $i \in C_p$, it follows from (30) that $d_i - w_{pp}^{\text{out}}(i)$ does not depend on $i \in C_p$. From (27) it follows that

$$d_i - w_{pp}^{\text{out}}(i) = \sum_{q \neq p} w_{pq}^{\text{in}}(i) + w_{pp}^{\text{in}}(i) - w_{pp}^{\text{out}}(i).$$

Since π is an AEP, $\sum_{q \neq p} w_{pq}^{\text{in}}(i)$ does not depend on $i \in C_p$ and therefore $w_{pp}^{\text{in}}(i) - w_{pp}^{\text{out}}(i)$ does not depend on $i \in C_p$. Hence, there exists an integer s_p such that $s_p = w_{pp}^{\text{in}}(i) - w_{pp}^{\text{out}}(i)$ for all $i \in C_p$. Clearly,

$$\sum_{i \in C_p} w_{pp}^{\text{in}}(i) = \sum_{i \in C_p} w_{pp}^{\text{out}}(i),$$

and it follows that

$$0 = \sum_{i \in C_p} \left(w_{pp}^{\text{in}}(i) - w_{pp}^{\text{out}}(i) \right) = \sum_{i \in C_p} s_p = |C_p| s_p.$$

Therefore $s_p = 0$, so $w_{pp}^{\text{in}}(i) = w_{pp}^{\text{out}}(i)$ and we conclude that π is balanced, which completes the proof. □

The result of Theorem 2 can be used to proof the following lemma. It states that for any balanced almost equitable partition with characteristic matrix $P = P(\pi)$, $\operatorname{im} P$ is D-invariant.

Lemma 4. *Let G be a strongly connected weighted directed graph, let μ denote the spanning tree vector of G and let D denote the spanning tree matrix of G. If π is a balanced AEP, then $\mu \in \operatorname{im} P$. Moreover, there exists a matrix Z such that*

$$DP = PZ. \tag{33}$$

Proof. Let π be a balanced almost equitable partition of G. By Lemma 3 we know that the matrix $Y = (P^{\mathrm{T}}P)^{-1}\hat{L}^{\mathrm{T}}P^{\mathrm{T}}P$ satisfies $P^{\mathrm{T}}L = YP^{\mathrm{T}}$. Since G is strongly connected, the graph of the reduced-order model $\hat{G}$ is strongly connected as well. Hence, from [9] it follows that $\operatorname{rank} \hat{L} = r - 1$, so clearly $\operatorname{rank} Y = r - 1$. Then there exists a nonzero vector $v \in \mathbb{R}^r$ such that $v^{\mathrm{T}}Y = 0$, and therefore we have

$$(Pv)^{\mathrm{T}}L = v^{\mathrm{T}}P^{\mathrm{T}}L = v^{\mathrm{T}}YP^{\mathrm{T}} = 0.$$

Since Pv is nonzero and $\operatorname{rank} L = n - 1$, it follows from Theorem 2 that there exists a nonzero constant $\alpha \in \mathbb{R}$ such that

$$\mu = \alpha P v, \tag{34}$$

and hence $\mu \in \operatorname{im} P$. Using (3) we write the matrix product DP as

$$DP = \begin{pmatrix} Dp(C_1) & Dp(C_2) & \cdots & Dp(C_r) \end{pmatrix}.$$

From (2) one observes that for $j = 1 \ldots, r$ the column vector $Dp(C_j)$ is given by

$$(Dp(C_j))_i = \begin{cases} \mu_i & \text{if } i \in C_j, \\ 0 & \text{otherwise.} \end{cases} \tag{35}$$

From (34) it follows that $\mu_i = \alpha v_j$, where j is such that $i \in C_j$. Then it follows from (35) that $Dp(C_j) = \alpha v_j p(C_j)$ and hence, we can write

$$DP = \begin{pmatrix} \alpha v_1 p(C_1) & \alpha v_2 p(C_2) & \cdots & \alpha v_r p(C_r) \end{pmatrix}. \tag{36}$$

Now we define $Z = \alpha \operatorname{diag}(v_1, v_2, \ldots, v_r)$. Observe that the effect of post-multiplying P by the diagonal matrix Z is that the j^{th} column of P is multiplied by αv_j. Hence, the j^{th} column of the matrix product PZ is equal to $\alpha v_j p(C_j)$. Comparing this to (36), one sees that

$$DP = PZ,$$

which completes the proof. □

5.2 Approximation error

Now we are ready to provide an approximation error for the model reduction procedure that was introduced in Section 3. Consider the multi-agent system (22) and suppose that the system is defined on a strongly connected weighted directed graph. Let (23) be the approximation of the original multi-agent system (22) obtained by applying the proposed clustering method, using a balanced almost equitable partition. Then the following theorem provides an explicit expression for the relative approximation error.

Theorem 3. *Let G be a strongly connected weighted directed graph with n nodes. Let $\pi = \{C_1, C_2, \ldots, C_r\}$ be a balanced almost equitable partition of G and suppose (23) is obtained from (22) by using the partition π. Let S and $\hat{S}$ denote the transfer matrices from u to y in (22) and (23) respectively. Let $\mu \in \mathbb{R}^n$ denote the spanning tree vector of G. Then we have*

$$\frac{||S-\hat{S}||_2^2}{||S||_2^2} = \frac{\sum_{i\in V_L} \mu_i\big(1-\frac{1}{|C_i|}\big)}{\sum_{i\in V_L} \mu_i\big(1-\frac{\mu_i}{\sum_{j=1}^n \mu_j}\big)}, \tag{37}$$

where V_L denotes the set of leaders of the multi-agent system.

Before we move on to a proof of this theorem, we want to make some comments on the right-hand side of (37). First of all, note that the approximation error will be identically zero if all the leaders of the multi-agent systems appear as singletons in π, as in that case $C_i = 1$ for all $i \in V_L$. For the example in Figure 4, this is the case if the leaders are agent 7 and 8. Furthermore, we can decrease the error by adding arcs to the graph. To see this, suppose we add an arc to G such that the new spanning trees that arise are all rooted at followers of G. Then clearly, μ_i increases for all followers $i \in V_F$, but μ_i stays the same for all leaders $i \in V_L$. Hence, the numerator of the right-hand side of (37) does not change. However, as $\sum_{j=1}^n \mu_j$ increases, the denominator increases and consequently, the error decreases. Analogously, we can increase the weight of an arc that only appears in spanning trees rooted at followers to obtain a smaller error. Clearly, we have to make sure that the partition remains balanced almost equitable, since otherwise the conditions of the theorem are no longer satisfied. Now we move on to the proof of the theorem.

Proof. The columns of P are orthogonal, so we can construct an $n \times n$ matrix T as $T = \begin{pmatrix} P & Q \end{pmatrix}$ where Q is an $n \times (n-k)$ matrix such that the columns of T are an orthogonal set of vectors in $\mathbb{R}^n$. Then we have $P^{\mathrm{T}}Q = 0$. If we apply the state space transformation $x = T\tilde{x}$ to the system given by (22), we obtain the following system:

$$\begin{aligned} \begin{pmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{pmatrix} &= -\begin{pmatrix} (P^{\mathrm{T}}P)^{-1}P^{\mathrm{T}}LP & (P^{\mathrm{T}}P)^{-1}P^{\mathrm{T}}LQ \\ (Q^{\mathrm{T}}Q)^{-1}Q^{\mathrm{T}}LP & (Q^{\mathrm{T}}Q)^{-1}Q^{\mathrm{T}}LQ \end{pmatrix} \begin{pmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{pmatrix} + \begin{pmatrix} (P^{\mathrm{T}}P)^{-1}P^{\mathrm{T}}M \\ (Q^{\mathrm{T}}Q)^{-1}Q^{\mathrm{T}}M \end{pmatrix} u \\ y &= \begin{pmatrix} (DL+L^{\mathrm{T}}D)^{\frac{1}{2}}P & (DL+L^{\mathrm{T}}D)^{\frac{1}{2}}Q \end{pmatrix} \begin{pmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{pmatrix} \end{aligned} \tag{38}$$

Then the transfer matrices from u to y in (22) and (38) are the same. Since π is an AEP of G, by Lemma 3 we know that there exists matrices X and Y such that $LP = PX$ and $L^{\mathrm{T}}P = PY$ and by Lemma 4 there exists a matrix Z such that $DP = PZ$. Thus we have $DLP = PZX$ and $L^{\mathrm{T}}DP = PYZ$ and it follows that

$$\begin{aligned} Q^{\mathrm{T}}DLP &= 0, \\ Q^{\mathrm{T}}L^{\mathrm{T}}DP &= 0. \end{aligned} \tag{39}$$

If we truncate the state components $\tilde{x}_2$ in (38), we obtain the reduced-order model. Therefore, using the equalities in (39), the transfer matrices S and $\hat{S}$ are related by

$$S(s) = \hat{S}(s) + \Delta(s),$$

where

$$\Delta(s) = (DL+L^{\mathrm{T}}D)^{\frac{1}{2}}Q(sI + (Q^{\mathrm{T}}Q)^{-1}Q^{\mathrm{T}}LQ)^{-1}(Q^{\mathrm{T}}Q)^{-1}Q^{\mathrm{T}}M.$$

From (39) it follows that $\hat{S}^{\mathrm{T}}(-s)\Delta(s) = 0$. Therefore, we have

$$||S||_2^2 = ||\hat{S}||_2^2 + ||\Delta||_2^2. \tag{40}$$

In order to compute $||S||_2^2$ and $||\hat{S}||_2^2$ we define the matrices $X_1 \in \mathbb{R}^{n\times n}$ and $Y_1 \in \mathbb{R}^{r\times r}$ as:

$$\begin{aligned} X_1 &= \int_0^\infty e^{-L^{\mathrm{T}}t}(DL + L^{\mathrm{T}}D)e^{-Lt}dt, \\ Y_1 &= \int_0^\infty e^{-\hat{L}^{\mathrm{T}}t}P^{\mathrm{T}}(DL + L^{\mathrm{T}}D)Pe^{-\hat{L}t}dt, \end{aligned}$$

where $\hat{L}$ is given by (8). We compute X_1 as

$$\begin{aligned} X_1 &= \int_0^\infty e^{-L^{\mathrm{T}}t}(DL + L^{\mathrm{T}}D)e^{-Lt}dt = -\int_0^\infty \frac{d}{dt}\left(e^{-L^{\mathrm{T}}t}De^{-Lt}\right)dt = -e^{-L^{\mathrm{T}}t}De^{-Lt}\Big|_0^\infty \\ &= D - \lim_{t\to\infty} e^{-L^{\mathrm{T}}t}De^{-Lt}. \end{aligned} \tag{41}$$

In order to compute the limit, we rewrite the matrix L. Since all nontrivial eigenvalues of L have positive real part, and the zero eigenvalue has multiplicity one with corresponding eigenvector $\mathbb{1}$, there exist matrices $V \in \mathbb{R}^{n\times n}$ and $L_2 \in \mathbb{R}^{(n-1)\times(n-1)}$ such that

$$L = V\begin{pmatrix} 0 & 0 \\ 0 & L_2 \end{pmatrix}V^{-1}, \tag{42}$$

where all eigenvalues of L_2 have positive real part and the first column of V is equal to $\mathbb{1}$. Now let $V_2 \in \mathbb{R}^{(n-1)\times n}$ be such that

$$V = \begin{pmatrix} \mathbb{1} & V_2 \end{pmatrix}. \tag{43}$$

Observe that $\mathbb{1}^{\mathrm{T}}D = \mu^{\mathrm{T}}$, so it follows that

$$V^{\mathrm{T}}DV = \begin{pmatrix} \mathbb{1}^{\mathrm{T}} \\ V_2^{\mathrm{T}} \end{pmatrix} D \begin{pmatrix} \mathbb{1} & V_2 \end{pmatrix} = \begin{pmatrix} \mu^{\mathrm{T}} \\ V_2^{\mathrm{T}}D \end{pmatrix}\begin{pmatrix} \mathbb{1} & V_2 \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^n \mu_i & \mu^{\mathrm{T}}V_2 \\ V_2^{\mathrm{T}}\mu & V_2^{\mathrm{T}}DV_2 \end{pmatrix}. \tag{44}$$

From (42) it follows that the first row of $V^{-1}L$ is equal to zero. Since the first row of $V^{-1}L$ is the first row of V^{-1} times L, it follows that the first row of V^{-1} is equal to $\beta\mu^{\mathrm{T}}$ for some $\beta \in \mathbb{R}$. Then the first row of $V^{-1}V$ reads

$$(V^{-1}V)_1 = \beta\mu^{\mathrm{T}}V = \begin{pmatrix} \beta\sum_{i=1}^n \mu_i & \beta\mu^{\mathrm{T}}V_2 \end{pmatrix},$$

as follows from (43). Since $V^{-1}V = I_n$, we have that $\beta = 1/\sum_{i=1}^n \mu_i$ and $\mu^{\mathrm{T}}V_2 = 0$. Plugging this into (44), we obtain

$$V^{\mathrm{T}}DV = \begin{pmatrix} \sum_{i=1}^n \mu_i & 0 \\ 0 & V_2^{\mathrm{T}}DV_2 \end{pmatrix}. \tag{45}$$

Furthermore, we now have

$$V^{-1} = \begin{pmatrix} \frac{1}{\sum_{i=1}^n \mu_i}\mu^{\mathrm{T}} \\ * \end{pmatrix} \tag{46}$$

Using (42) and (45), we can write

$$
\begin{aligned}
\lim_{t\to\infty} e^{-L^{\mathrm{T}}t} D e^{-Lt} &= \lim_{t\to\infty} V^{-\mathrm{T}} e^{\begin{pmatrix} 0 & 0 \\ 0 & -L_2^{\mathrm{T}}t \end{pmatrix}} V^{\mathrm{T}} D V e^{\begin{pmatrix} 0 & 0 \\ 0 & -L_2 t \end{pmatrix}} V^{-1} \\
&= V^{-\mathrm{T}} \lim_{t\to\infty} \begin{pmatrix} 1 & 0 \\ 0 & e^{-L_2^{\mathrm{T}}t} \end{pmatrix} \begin{pmatrix} \sum_{i=1}^n \mu_i & 0 \\ 0 & V_2^{\mathrm{T}} D V_2 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & e^{-L_2 t} \end{pmatrix} V^{-1} \\
&= V^{-\mathrm{T}} \lim_{t\to\infty} \begin{pmatrix} \sum_{i=1}^n \mu_i & 0 \\ 0 & e^{-L_2^{\mathrm{T}}t} V_2^{\mathrm{T}} D V_2 e^{-L_2 t} \end{pmatrix} V^{-1} \\
&= V^{-\mathrm{T}} \begin{pmatrix} \sum_{i=1}^n \mu_i & 0 \\ 0 & 0 \end{pmatrix} V^{-1} \\
&= \begin{pmatrix} \frac{1}{\sum_{i=1}^n \mu_i} \mu & * \end{pmatrix} \begin{pmatrix} \sum_{i=1}^n \mu_i & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sum_{i=1}^n \mu_i} \mu^{\mathrm{T}} \\ * \end{pmatrix} \\
&= \frac{1}{\sum_{i=1}^n \mu_i} \mu \mu^{\mathrm{T}},
\end{aligned}
$$

where the fifth equality follows from (46). Plugging this expression into (41), we obtain

$$
X_1 = D - \frac{1}{\sum_{i=1}^n \mu_i} \mu \mu^{\mathrm{T}}. \tag{47}
$$

Furthermore, we have

$$
\begin{aligned}
T^{\mathrm{T}} X_1 T &= \int_0^\infty T^{\mathrm{T}} e^{-L^{\mathrm{T}}t} (DL + L^{\mathrm{T}} D) e^{-Lt} T dt \\
&= \int_0^\infty e^{-(T^{-1} L^{\mathrm{T}} T)^{\mathrm{T}} t} T^{\mathrm{T}} (DL + L^{\mathrm{T}} D) T e^{-T^{-1} L T t} dt \\
&= \int_0^\infty e^{\begin{pmatrix} \hat{L}^{\mathrm{T}} & 0 \\ 0 & * \end{pmatrix}} \begin{pmatrix} P^{\mathrm{T}} (DL + L^{\mathrm{T}} D) P & 0 \\ 0 & * \end{pmatrix} e^{\begin{pmatrix} \hat{L} & 0 \\ 0 & * \end{pmatrix}} dt,
\end{aligned}
$$

where the last equality follows from (39). Now observe that

$$
T^{\mathrm{T}} X_1 T = \begin{pmatrix} Y_1 & 0 \\ 0 & * \end{pmatrix},
$$

from which it follows that $Y_1 = P^{\mathrm{T}} X_1 P$. From (47) we obtain

$$
Y_1 = P^{\mathrm{T}} D P - \frac{1}{\sum_{i=1}^n \mu_i} P^{\mathrm{T}} \mu \mu^{\mathrm{T}} P. \tag{48}
$$

Now we can compute $||S||_2^2$ and $||\hat{S}||_2^2$. Clearly, we have $||S||_2^2 = \operatorname{trace} M^{\mathrm{T}} X_1 M$, so by (47) we have

$$
||S||_2^2 = \operatorname{trace} M^{\mathrm{T}} \big(D - \frac{1}{\sum_{i=1}^n \mu_i} \mu \mu^{\mathrm{T}}\big) M = \operatorname{trace} M M^{\mathrm{T}} \big(D - \frac{1}{\sum_{i=1}^n \mu_i} \mu \mu^{\mathrm{T}}\big).
$$

By construction of M, the matrix MM^{T} is a diagonal matrix, where the i^{th} diagonal element is equal to 1 if $i \in V_L$ and zero otherwise. Then it is clear that

$$
||S||_2^2 = \sum_{i \in V_L} \mu_i \big(1 - \frac{\mu_i}{\sum_{j=1}^n \mu_j}\big) \tag{49}
$$

We can compute $||\hat{S}||_2^2$ as

$$\begin{aligned}||\hat{S}||_2^2 &= \operatorname{trace} M^{\mathrm{T}} P(P^{\mathrm{T}}P)^{-1} Y_1 (P^{\mathrm{T}}P)^{-1} P^{\mathrm{T}} M \\ &= \operatorname{trace} M^{\mathrm{T}} P(P^{\mathrm{T}}P)^{-1} \big(P^{\mathrm{T}} D P - \frac{1}{\sum_{i=1}^n \mu_i} P^{\mathrm{T}} \mu\mu^{\mathrm{T}} P\big)(P^{\mathrm{T}}P)^{-1} P^{\mathrm{T}} M \\ &= \operatorname{trace} M M^{\mathrm{T}} \big(P(P^{\mathrm{T}}P)^{-1} P^{\mathrm{T}} D P (P^{\mathrm{T}}P)^{-1} P^{\mathrm{T}} - \frac{1}{\sum_{i=1}^n \mu_i} P(P^{\mathrm{T}}P)^{-1} P^{\mathrm{T}} \mu\mu^{\mathrm{T}} P(P^{\mathrm{T}}P)^{-1} P^{\mathrm{T}}\big).\end{aligned}$$

Since $DP = PZ$, we have

$$P(P^{\mathrm{T}}P)^{-1} P^{\mathrm{T}} D P = \alpha P(P^{\mathrm{T}}P)^{-1} P^{\mathrm{T}} P \hat{D} = \alpha P \hat{D} = DP.$$

Furthermore, from Lemma 4 it follows that there exists a vector $v \in \mathbb{R}^r$ such that $\mu = Pv$, and therefore $P(P^{\mathrm{T}}P)^{-1}P^{\mathrm{T}}\mu = P(P^{\mathrm{T}}P)^{-1}P^{\mathrm{T}}Pv = Pv = \mu$. Hence, we can write

$$||\hat{S}||_2^2 = \operatorname{trace} M M^{\mathrm{T}} \big(DP(P^{\mathrm{T}}P)^{-1} P^{\mathrm{T}} - \frac{1}{\sum_{i=1}^n \mu_i} \mu\mu^{\mathrm{T}}\big).$$

Hence,

$$||\hat{S}||_2^2 = \sum_{i \in V_L} \mu_i \big(\frac{1}{|C_i|} - \frac{\mu_i}{\sum_{j=1}^n \mu_j}\big) \tag{50}$$

and from (40) it then follows that

$$||\Delta||_2^2 = \sum_{i \in V_L} \mu_i \big(1 - \frac{1}{|C_i|}\big). \tag{51}$$

The result follows from (49) and (51). □

Remark 1. For a strongly connected weighted *undirected* graph we have $\mathbb{1}^{\mathrm{T}} L = 0$. Since $\operatorname{rank} L = n - 1$, it follows from Theorem 2 that $\mu = \alpha \mathbb{1}$ for some $\alpha \in \mathbb{R}$, so $\mu_i = \alpha$ for all $i = 1, 2, \ldots, n$. Hence as a special case, for undirected graphs the right-hand side of (37) is given by

$$\frac{\sum_{i \in V_L} \mu_i \big(1 - \frac{1}{|C_i|}\big)}{\sum_{i \in V_L} \mu_i \big(1 - \frac{\mu_i}{\sum_{j=1}^n \mu_j}\big)} = \frac{\sum_{i \in V_L} \big(1 - \frac{1}{|C_i|}\big)}{\sum_{i \in V_L} \big(1 - \frac{1}{n}\big)},$$

which is exactly the expression that was found by Monshizadeh *et al.* in [4].

Example 2. Consider again the multi-agent system given by (11) with underlying graph G, depicted in Figure 1, and the partition π given by (13). The output of the system is given in (22b). The reduced-order model (14) is obtained from the model reduction using the graph partition π, and we assign the output as in (23b) to this model. Recall that the set of leaders is given by $\{1, 3\}$. Also recall that the partition π is a balanced almost equitable partition. Furthermore, we have $1 \in C_1 = \{1, 2, 5\}$ and $3 \in C_2 = \{3\}$. Since the graph is strongly connected and the graph partition π is a balanced almost equitable partition of the graph, we can use Theorem 3 to compute the approximation error that is involved in the approximation of (11). In order to do so, we first need to compute the spanning tree vector of the graph G.

There is one spanning tree rooted at node 1 with corresponding weight $1 \times 5 \times 2 \times 2 = 20$. Therefore, $\mu_1 = 20$. Similarly, we have $\mu_2 = \mu_5 = 20$. The graph has three spanning trees that are rooted at node 3, each with weight 6. Hence, $\mu_3 = 18$. Finally, there is exactly one spanning tree rooted at node 4, and the corresponding weight is 40, so $\mu_4 = 40$. Thus, the spanning tree of the graph is given by $\mu = \begin{pmatrix} 20 & 20 & 18 & 40 & 20 \end{pmatrix}^{\mathrm{T}}$. Then, by Theorem 3, we can compute the relative approximation error $\Xi(\pi)$ using the right-hand side of (37):

$$\Xi(\pi) = \frac{\mu_1\big(1 - \frac{1}{|C_1|}\big) + \mu_3\big(1 - \frac{1}{|C_2|}\big)}{\mu_1\big(1 - \frac{\mu_1}{\sum_{j=1}^5 \mu_j}\big) + \mu_3\big(1 - \frac{\mu_3}{\sum_{j=1}^5 \mu_j}\big)} = \frac{20\big(1 - \frac{1}{3}\big) + 18\big(1 - \frac{1}{1}\big)}{20\big(1 - \frac{20}{118}\big) + 18\big(1 - \frac{18}{118}\big)} \approx 0.42$$

6 Example

In this chapter we provide a large-scale example of the model reduction procedure.

In this example we consider the multi-agent system with underlying graph G depicted in Figure 4, where the agents 1 and 7 are the leaders of the system. The multi-agent system is given by

$$\dot{x} = -Lx + Mu,$$

where

$$L = \begin{pmatrix} 3 & -1 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & -1 & 0 & -2 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 3 & 0 & 0 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & -2 & 0 & 0 & 0 & 0 \\ -4 & -3 & -1 & 0 & 11 & 0 & -3 & 0 & 0 & 0 \\ 0 & -1 & -3 & -4 & 0 & 11 & -3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2 & -2 & 0 & 6 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 0 & 2 \end{pmatrix}, \qquad M = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

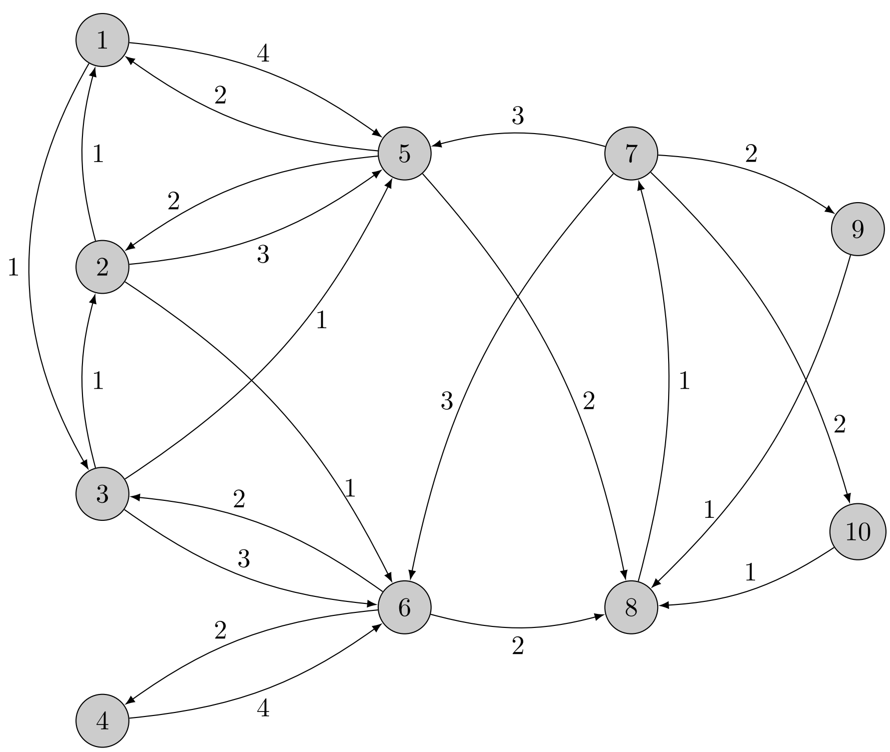

Figure 4: Directed graph G.

We partition the nodes of the graph as

$$\pi = \{C_1, C_2, C_3, C_4, C_5\} = \{\{1,2,3,4\},\{5,6\},\{7\},\{8\},\{9,10\}\}. \tag{52}$$

Observe that this is a balanced almost equitable partition of G. The corresponding characteristic matrix $P = P(\pi)$ is given by

$$P = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}^{\mathrm{T}}.$$

Now we perform the model reduction using the given partition π and obtain the following reduced-order model:

$$\dot{\hat{x}} = -\hat{L}\hat{x} + \hat{M}u,$$

where the matrices $\hat{L}$ and $\hat{M}$ are given by

$$\hat{L} = (P^{\mathrm{T}}P)^{-1}P^{\mathrm{T}}LP = \begin{pmatrix} 2 & -2 & 0 & 0 & 0 \\ -8 & 11 & -3 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & -4 & 0 & 6 & -2 \\ 0 & 0 & -2 & 0 & 2 \end{pmatrix}, \qquad \hat{M} = (P^{\mathrm{T}}P)^{-1}P^{\mathrm{T}}M = \begin{pmatrix} 0.25 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}. \tag{53}$$

The underlying graph $\hat{G}$ of the reduced-order multi-agent system is depicted in Figure 5.

Let L_i denote the matrix that is obtained from the Laplacian L of G by removing the row and column corresponding to node i. By Lemma 2, we have $\mu_i = \det L_i$ for all $i = 1, 2, \ldots, 10$. Therefore, using matlab to compute the determinant of L_i for $i = 1, 2, \ldots, 10$, we obtain the elements of the spanning tree vector:

$$\mu = \begin{pmatrix} 5312 & 5312 & 5312 & 5312 & 2656 & 2656 & 23904 & 3984 & 1992 & 1992 \end{pmatrix}^{\mathrm{T}}.$$

The leader set is given by $\{1, 7\}$, and clearly, $1 \in C_1$ and $7 \in C_3$, where $|C_1| = 4$ and $|C_3| = 1$.

Since the graph G is strongly connected and the partition π is a balanced almost equitable partition of the graph, we can again apply Theorem 3 to compute the relative approximation error. We compute the right-hand side of (37) and obtain the relative approximation error $\Xi(\pi) \approx 0.21$. $\square$

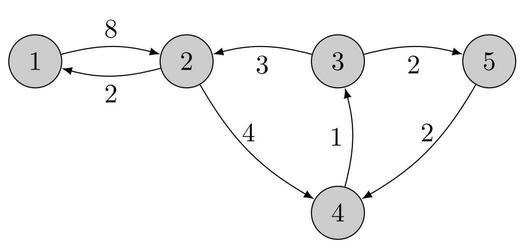

Figure 5: Reduced directed graph $\hat{G}$.

7 Conclusions

In this thesis we have extended the results from [4] to multi-agent systems defined on weighted directed graphs. We have seen that the reduction procedure as introduced in [4] can be applied to such systems and that the network structure of the system is preserved. The reduced-order models again represent a multi-agent system defined on a smaller graph. Also, we have shown that consensus is preserved if we apply the model reduction. We have introduced the notion of balanced almost equitable partitions and subsequently we have provided an explicit expression of the relative error of the approximation that is obtained from performing the model reduction using a balanced almost equitable partition. The proof of this expression was based on ideas of Monshizadeh *et al.* in [4]. An expression for the error can be derived easily from the graph partition and the weights of the spanning trees of the graph. We have briefly discussed the expression and how one could reduce the error by changing the interconnection between the agents. We have given two examples of the model reduction procedure.

Future research could focus on the computation of a bound for the approximation error for partitions that are not balanced almost equitable. Research could also focus on the right-hand side of the expression for the approximation error given in (37) and how the error of the approximation depends on the topology and weights of the graph. This is interesting since the error expression can be influenced by changing the topology and the weights of the graph.

8 Appendix

A. Proof of Lemma 1.

Denote the set of spanning trees of G rooted at i by $\mathcal{T}_i(G)$. It is clear that the spanning trees of $G-(i,j)$ rooted at i are exactly the spanning trees in $\mathcal{T}_i(G)$ that do not contain arc (i,j). So it remains to show that $a_{ji}\mu_i(G/(i,j))$ is equal to the sum of the weights of the remaining spanning trees in $\mathcal{T}_i(G)$. Now, if a spanning tree $T \in \mathcal{T}_i(G)$ contains (i,j), then contraction of (i,j) yields a spanning subgraph of $G/(i,j)$ rooted at i with $n-2$ arcs, so $T/(i,j)$ is a spanning tree of $G/(i,j)$ rooted at i. We will show that the mapping $T \mapsto T/(i,j)$ defines a bijection from spanning trees in $\mathcal{T}_i(G)$ containing (i,j) to spanning trees of $G/(i,j)$ rooted at i.

To see that contracting (i,j) is injective, observe that all the other arcs in the spanning tree maintain their identity, so two different spanning trees in $\mathcal{T}_i(G)$ containing (i,j) are never mapped to the same spanning tree of $G/(i,j)$. Also, observe that any spanning tree of $G/(i,j)$ is formed by contracting (i,j) in a spanning tree of G and therefore contraction indeed defines a bijection.

Furthermore, when we contract (i,j), the other arcs of T do not change, so clearly, $w(T) = a_{ji}w(T/(i,j))$. Now, since $T \mapsto T/(i,j)$ is a bijection from spanning trees in $\mathcal{T}_i(G)$ containing (i,j) to spanning trees of $G/(i,j)$ rooted at i, it follows that the sum of the weights of all spanning trees in $\mathcal{T}_i(G)$ that contain (i,j) is equal to $a_{ji}\mu_i(G/(i,j))$. This completes the proof.

B. Proof of Lemma 2.

We will proof the lemma by induction on the number of arcs of G. Clearly, if G has zero arcs, all elements of the Laplacian matrix are zero, so $\det L_i = 0$ for all i. Furthermore, the graph does not have any spanning trees, and hence $\mu_i(G) = 0$ as well for all nodes i, so the statement holds for graphs with zero arcs. Now, suppose the statement holds for graphs for which the number of arcs is less than or equal to k.. Let $G = (V, E, A)$ be a graph with $k+1$ arcs and let $i \in V$. Suppose $(i,j) \in E$ for some $j \in V$ (if such j does not exist, we have $\det L_i = \mu_i(G) = 0$). Then by Lemma 1 we have

$$\mu_i(G) = \mu_i(G-(i,j)) + a_{ji}\mu_i(G/(i,j)). \tag{54}$$

Let $L_{ij}(G)$ denote the $(n-2)\times(n-2)$ matrix that is obtained from $L(G)$ by removing the rows and columns that correspond to nodes i and j. Observe that the Laplacian matrix of $G/(i,j)$ is obtained from $L(G)$ by removing the j^{th} row and column and altering only the i^{th} row and column. The rest of the elements remain identical. Then it follows that

$$\det L_i(G/(i,j)) = \det L_{ij}(G). \tag{55}$$

Observe that $L(G-(i,j))$ is obtained from $L(G)$ by replacing the (j,i)-th element $-a_{ji}$ with zero and the (j,j)-th element d_j with $d_j - a_{ji}$. Hence, the only difference between the matrices $L_i(G)$ and $L_i(G-(i,j))$ is the (j,j)-th element. Therefore, $\det L_i(G) = \det L_i(G-(i,j)) + a_{ji} \det L_{ij}(G)$, so from (55) it follows that

$$\det L_i(G) = \det L_i(G-(i,j)) + a_{ji} \det L_i(G/(i,j)). \tag{56}$$

Observe that both $G-(i,j)$ and $G/(i,j)$ contain at most k arcs, so by induction hypothesis, the right-hand sides of (54) and (56) are equal. We conclude that $\mu_i(G) = \det L_i$, which completes the proof.

9 Acknowledgements

I would like to thank my supervisor Harry Trentelman for providing a very good introduction into multi-agent systems and for his guidance during the project. Your very clear explanations helped me to understand the material and your enthusiasm for the subject and your numerous suggestions really motivated me during the research.

References

[1] C.O. Aguilar and B. Gharesifard. On almost equitable partitions and network controllability. *American Control Conference*, 2016.

[2] A.C. Antoulas. Approximation of large-scale dynamical systems. *SIAM, Philadelphia, Pa, USA*, 2005.

[3] B. Bollobas. *Modern Graph Theory.* Springer-Verlag New York, 1998.

[4] N. Monshizadeh M.K. Camlibel and H.L. Trentelman. Projection based model reduction of multi-agent systems using graph partitions. *IEEE Transactions on Control of Network Systems, 1(2), 145-154*, 2014.

[5] A. Van der Schaft. Complex and detailed balancing of chemical reaction networks revisited. *Journal of Mathematical Chemistry*, 2015.

[6] Jiu-Gang Dong and Li Qiu. Complex laplacians and applications in multi-agent systems. *arXiv:1406.1862*, 2015.

[7] J.A. Fax and R.M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control, 49(9):1465-1476*, 2004.

[8] T. Karatas J. Cortes, S. Martinez and F. Bullo. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation, 20(2):243-255*, 2004.

[9] J.A. Fax R. Olfati-Saber and R.M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE, vol. 95, pp. 215-233*, 2007.

[10] H. Sandberg and R.M. Murray. Model reduction of interconnected linear systems. *Optimal Control Applications and Methods*, 2007.

[11] Yongcan Cao Wei Ren. *Distributed Coordination of Multi-agent Networks.* Springer-Verlag, 2011.