Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 48-22 (2015) 150–155

Root Locus Design for the Synchronization of Multi-Agent Systems in General Directed Networks

Saman Khodaverdian ⋆, **Felix Lanfermann**, **Jürgen Adamy**

Institute of Automatic Control and Mechatronics; Control Methods and Robotics Lab; Technische Universität Darmstadt, Germany (e-mail: saman.khodaverdian@rmr.tu-darmstadt.de).

Abstract: This paper considers the synchronization problem of multi-agent SISO systems with general unidirectional communication structures. A distributed control strategy is presented which relies on relative output differences of neighboring agents and, thus, does not need relative state information. We propose a root locus design method to determine the synchronization gain. Since in directed networks the characteristic equation for synchronization might be complex valued, we use tools from the complex root locus technique to solve the synchronization task.

Keywords: Synchronization, multi-agent systems, directed networks, output coupling, root locus.

1. INTRODUCTION

The *synchronization* problem of *multi-agent systems* is an important task for the cooperative control of networked dynamic systems. Such networks can be used to handle various problems in different research areas. See for instance Olfati-Saber et al. (2007) and Ren et al. (2007) for a brief overview. Often, a group of interconnected agents is able to achieve a better performance than a single agent. Recently, there has been a lot of research for the coordination of such multi-agent systems. For instance, Lin et al. (2003) worked on the multi-agent rendevouz problem, Fax and Murray (2004) and Lafferriere et al. (2005) considered vehicle formation problems, and Olfati-Saber (2006) and Tanner et al. (2007) dealt with flocking algorithms.

A main objective for multi-agent systems is to find a control strategy such that the agents in the network behave uniformly or synchronous. It is usually desired to solve this task in a distributed manner, meaning that there is no central entity that controls every agent. The agents should act autonomously and coordinate themselves using a local controller and the available information from their environment.

It has been proved that the distributed synchronization task can be reduced to a stabilization problem and solved by well known control techniques (Li et al. (2010); Ma and Zhang (2010)). However, the solution of the synchronization problem depends essentially on the network structure, meaning the information exchange between the agents. Tuna (2008) proposed an LQR-based design technique which ensures synchronization under mild connectivity assumptions and is often used in literature, e.g. in Li et al. (2010); Ma and Zhang (2010); Zhang et al. (2011). Alternatively, Listmann (2012) developed an LMI-based approach to determine the controller gain. In both cases the solution depends on relative state information, meaning that the agents have knowledge about the difference between their own state vector and the state vector of their neighboring agents. From a practical point of view, it is more reasonable to assume that the agents have only relative output information and no access to relative state information. Based on this assumption, different observer-based techniques have been proposed to estimate the relative state of an agent and its neighbors, cf. Chen et al. (2009); Li et al. (2011); Wang et al. (2009); Zhang et al. (2011). However, the provided solutions lead to an increased communication effort since it is necessary to exchange internal controller states through the network. In Seo et al. (2009b), an observer-based solution was presented which works without the exchange of additional variables, but it relies on a low gain approach, and Zhao et al. (2011) proposed a similar technique using the small gain theory.

It should be noted that in all the mentioned works, the goal is to get an estimate for the relative state information. Alternatively, it might be more appropriate to use output feedback techniques. Wang et al. (2009) proposed a design scheme working on output feedback, but it is restricted to a certain class of linear systems and they only considered undirected graphs, i.e. bilateral information flow. Seo et al. (2009a) showed conditions for an output feedback controller solving the synchronization problem for directed graphs, but their solution is restricted to systems without exponentially unstable modes. Recently, Listmann (2012) proposed a root locus design technique which is less restrictive than other methods. However, the presented approach works only for a special class of graphs.

In this work, we present a root locus design for the general case of directed graphs. It should be noted that in case of directed graphs, the synchronization problem results into a stabilization problem for a set of possibly complex valued systems. While in Listmann (2012) this situation is excluded, we propose an approach which also handles complex valued systems. For this purpose, we use a complex root locus method presented in Dòria-Cerezo and Bodson (2013).

The rest of the paper is organized as follows: In Section 2, we introduce some preliminaries and formulate the problem setup. Our main result is presented in Section 3, where a complex root locus design is proposed to solve the synchronization problem. Finally, the approach is illustrated by a numerical example in Section 4, followed by a conclusion in Section 5.

⋆ This work was gratefully supported by the German Research Foundation (DFG) within the GRK 1362 "Cooperative, Adaptive and Responsive Monitoring of Mixed Mode Environments" (www.gkmm.de).

10.1016/ifacol.2015.10.322

2. PRELIMINARIES AND PROBLEM FORMULATION

2.1 Notation

Boldface letters represent vectors and matrices and scalar values are highlighted in italic letters. $\boldsymbol{I}_n$ is the identity matrix of dimension n, and $\boldsymbol{0}$ stands for the zero matrix of appropriate dimensions. A square matrix $\boldsymbol{M}$ or a polynomial $P(s)$ is called *Hurwitz* (stable) if all of its eigenvalues or zeros have strictly negative real part. The symbol $\otimes$ denotes *Kronecker* product.

2.2 Graph Theory

The communication structure of the multi-agent system is given by a time-invariant *directed graph* or *digraph* $\mathcal{G} = (\mathcal{V}_\mathcal{G}, \mathcal{E}_\mathcal{G})$, with $\mathcal{V}_\mathcal{G} = \{1, \ldots, N\}$ and $\mathcal{E}_\mathcal{G} \subseteq \mathcal{V}_\mathcal{G} \times \mathcal{V}_\mathcal{G}$. The i-th agent in the network is represented by vertex $i \in \mathcal{V}_\mathcal{G}$ and the information flow from vertex i to vertex j is described by the directed edge $(i, j) \in \mathcal{E}_\mathcal{G}$. The communication graph can also be specified by the *Laplacian* matrix (Olfati-Saber et al. (2007)), which is defined as $\boldsymbol{L}_\mathcal{G} = [l_{\mathcal{G}_{ij}}] \in \mathbb{R}^{N\times N}$,

$$l_{\mathcal{G}_{ij}} = \begin{cases} \sum_{k=1}^{N} a_{\mathcal{G}_{ki}}, & \text{if } i = j, \\ -a_{\mathcal{G}_{ji}}, & \text{if } i \neq j, \end{cases} \qquad a_{\mathcal{G}_{ij}} = \begin{cases} 1, & \text{if } (i,j) \in \mathcal{E}_\mathcal{G}, \\ 0, & \text{if } (i,j) \notin \mathcal{E}_\mathcal{G}. \end{cases}$$

Definition 1. A digraph $\mathcal{G}$ contains a *directed spanning tree* if there exists at least one vertex that can reach any other vertex in the network, using the edges contained in the set $\mathcal{E}_\mathcal{G}$.

It can be shown that a directed spanning tree exists if and only if $\boldsymbol{L}_\mathcal{G}$ has a simple eigenvalue in zero, cf. Lafferriere et al. (2005), i.e. $\lambda_1(\boldsymbol{L}_\mathcal{G}) = 0$ and $\lambda_i(\boldsymbol{L}_\mathcal{G}) \neq 0$ for $i \in \{2, \ldots, N\}$. Note that $\lambda_1(\boldsymbol{L}_\mathcal{G})$ is always zero since $\boldsymbol{L}_\mathcal{G}\boldsymbol{1} = \boldsymbol{0}$, where $\boldsymbol{1}$ is the vector with all elements one.

2.3 Problem Setup

We consider a multi-agent system of N agents with linear time-invariant SISO dynamics

$$\dot{\boldsymbol{x}}_i = \boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{b}u_i \tag{1a}$$

$$y_i = \boldsymbol{c}^\mathsf{T}\boldsymbol{x}_i \tag{1b}$$

$$\zeta_i = \sum_{j=1}^{N} a_{\mathcal{G}_{ji}}(y_i - y_j). \tag{1c}$$

The state, input and output variables of agent i are represented by $\boldsymbol{x}_i \in \mathbb{R}^n$, $u_i \in \mathbb{R}$ and $y_i \in \mathbb{R}$, respectively. Furthermore, $\zeta_i \in \mathbb{R}$ denotes relative output information which is available to agent i. For instance, assuming a network of autonomous vehicles, ζ_i can be interpreted as a sum of position distances to neighboring agents which can be measured by agent i. Note that the network consists of N agents with identical dynamics $(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}^\mathsf{T})$. The goal is to design distributed output feedback controllers

$$u_i = -k\,\zeta_i = -k\sum_{j=1}^{N} a_{\mathcal{G}_{ji}}(y_i - y_j) \tag{2}$$

which use the available information ζ_i to synchronize all agents such that

$$\lim_{t\to\infty} \|\boldsymbol{x}_i(t) - \boldsymbol{x}_j(t)\| = 0, \quad \forall i, j \in \{1, \ldots, N\}. \tag{3}$$

Remark 1. In case that $\boldsymbol{A}$ is a Hurwitz matrix, the problem is trivially solved since then $\lim_{t\to\infty} \boldsymbol{x}_i(t) = \boldsymbol{0}$ for all $i \in \{1, \ldots, N\}$, even if there is no control effort.

In order to solve problem (3) with control law (2), we make the following assumptions.

Assumption 1. $(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}^\mathsf{T})$ is stabilizable and detectable.

Assumption 2. The network graph $\mathcal{G}$ contains a directed spanning tree, i.e. $\boldsymbol{L}_\mathcal{G}$ has a simple eigenvalue in zero.

2.4 Synchronization as Stabilization Problem

It is well known that the described synchronization problem can be reformulated to a stabilization problem and solved by standard feedback techniques. In the following, we summarize some results which are needed in the next sections. For detailed derivations or proofs, see e.g. Ma and Zhang (2010).

Combining (1) and (2) and considering the closed-loop system of the overall multi-agent system, we get

$$\dot{\boldsymbol{x}} = \left(\boldsymbol{I}_n \otimes \boldsymbol{A} - \boldsymbol{L}_\mathcal{G} \otimes \boldsymbol{b}k\boldsymbol{c}^\mathsf{T}\right)\boldsymbol{x},$$

where $\boldsymbol{x} = [\boldsymbol{x}_1^\mathsf{T} \cdots \boldsymbol{x}_N^\mathsf{T}]^\mathsf{T}$. Now, there exists a regular state transformation $z = \left(\boldsymbol{T}^{-1} \otimes \boldsymbol{I}_n\right)\boldsymbol{x}$ (cf. Tuna (2008)) such that

$$\dot{z} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{0} & \overline{\boldsymbol{A}} \end{bmatrix} z,$$

with

$$\overline{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{A} - \lambda_2(\boldsymbol{L}_\mathcal{G})\boldsymbol{b}k\boldsymbol{c}^\mathsf{T} & \cdots & \star \\ \vdots & \ddots & \vdots \\ \boldsymbol{0} & \cdots & \boldsymbol{A} - \lambda_N(\boldsymbol{L}_\mathcal{G})\boldsymbol{b}k\boldsymbol{c}^\mathsf{T} \end{bmatrix}$$

and $\star$ denotes elements that are not of further interest. It can be shown that synchronization will be achieved if and only if $\overline{\boldsymbol{A}}$ is a Hurwitz matrix. Since $\overline{\boldsymbol{A}}$ is block triangular, the matrices $\boldsymbol{A} - \lambda_i(\boldsymbol{L}_\mathcal{G})\boldsymbol{b}k\boldsymbol{c}^\mathsf{T}$ must be Hurwitz for all $i \in \{2, \ldots, N\}$. Therefore, we conclude that synchronization will be achieved if the controller gain k is determined such that the virtual system

$$\dot{\boldsymbol{v}} = \boldsymbol{A}\boldsymbol{v} + \boldsymbol{b}u_v \tag{4a}$$

$$y_v = \boldsymbol{c}^\mathsf{T}\boldsymbol{v} \tag{4b}$$

$$u_v = -k\lambda_i(\boldsymbol{L}_\mathcal{G})y_v \tag{4c}$$

is stable for all $i \in \{2, \ldots, N\}$. Note that $\lambda_2(\boldsymbol{L}_\mathcal{G}), \ldots, \lambda_N(\boldsymbol{L}_\mathcal{G})$ are non-zero since we have assumed that the network graph contains a directed spanning tree (Assumption 2). The difficulty is to find a controller gain which simultaneously stabilizes system (4) for the given $\lambda_i(\boldsymbol{L}_\mathcal{G})$. Tuna (2008) proposed an LQR-based controller design which ensures stability for all $\lambda_i(\boldsymbol{L}_\mathcal{G})$ if the smallest real part of the non-zero eigenvalues is known. However, the solution relies on relative state information and, in general, is only applicable if $\boldsymbol{c}^\mathsf{T} = \boldsymbol{I}_n$.

Alternatively to the state space description, the virtual system (4) can be represented by its transfer function. While the open-loop system (4a), (4b) reads

$$G_{ol}(s) = \frac{N_{ol}(s)}{D_{ol}(s)} = \boldsymbol{c}^\mathsf{T}\left(s\boldsymbol{I}_n - \boldsymbol{A}\right)^{-1}\boldsymbol{b},$$

the closed-loop system (4a), (4b) and (4c), is given by

$$G_{cl}(s) = \frac{N_{ol}(s)}{D_{ol}(s) + k\lambda_i(\boldsymbol{L}_\mathcal{G})N_{ol}(s)}. \tag{5}$$

Thus, by closing the loop, the virtual system depends on the eigenvalues of the Laplacian matrix $\boldsymbol{L}_\mathcal{G}$ which represents the communication network of the multi-agent system.

From (5), it is clear that stabilization of the virtual system or synchronization of the multi-agent system can be analyzed by the closed-loop characteristic polynomial

$$\begin{aligned} D_{cl}(s) &= D_{ol}(s) + k_i^\lambda N_{ol}(s), \\ k_i^\lambda &= k\lambda_i(\boldsymbol{L}_\mathcal{G}), \quad i = 2, \ldots, N. \end{aligned} \tag{6}$$

The closed-loop polynomial (6) has the standard output feedback structure which is well known in control theory, but note that $\lambda_i(\boldsymbol{L}_\mathcal{G})$ might be complex valued.

In case that $\lambda_i(\boldsymbol{L}_\mathcal{G})$ is real valued, (6) can be analyzed by conventional output feedback techniques. For instance, Listmann (2012) proposed a root locus design for network graphs where the eigenvalues of the Laplacian matrix have zero imaginary part (which includes the special case of undirected graphs). In this case, it is possible to calculate the set of stabilizing controller gains k_i^λ and then deduce the set of valid synchronization gains k under consideration of the relation $k = \frac{k_i^\lambda}{\lambda_i(\boldsymbol{L}_\mathcal{G})}$. Note that the analysis must be done for all eigenvalues $\lambda_i(\boldsymbol{L}_\mathcal{G})$ (except the zero eigenvalue). However, the classical root locus design is only applicable if the system parameters are real valued, wherefore Listmann (2012) excluded communication graphs with complex valued Laplacian eigenvalues.

3. SYNCHRONIZATION CONTROLLER DESIGN BY COMPLEX ROOT LOCUS METHODS

In the following, we present a root locus design for the synchronization of networks with general directed graphs and, thus, possibly complex valued eigenvalues. First, we summarize some facts for the root locus design for polynomials with complex coefficients. Then, we present a solution for the case that the Laplacian graph or its eigenvalues $\lambda_i(\boldsymbol{L}_\mathcal{G})$ are exact known, and we also provide a design method for the case that the communication network is not known a priori.

3.1 Root Locus for Polynomials with Complex Coefficients

Recently, Dòria-Cerezo and Bodson (2013) presented root locus rules for polynomials of the form

$$D_{cl}(s) = D_{ol}(s) + kk_c N_{ol}(s), \tag{7}$$

where $D_{ol}(s)$ and $N_{ol}(s)$ contain complex coefficients and k_c is also a complex parameter. Comparing (7) with (6), it is obvious that the complex root locus method is well suited for the synchronization problem, since k_c is replaced by $\lambda_i(\boldsymbol{L}_\mathcal{G})$.

As in the classical case, the root locus with complex coefficients is described by $D_{cl}(s) = 0$ as k varies from zero to infinity. Note that $D_{ol}(s)$ and $N_{ol}(s)$ are polynomials with degrees n and m, respectively, and the roots of $D_{ol}(s)$ and $N_{ol}(s)$ are the open-loop poles and zeros of the system $(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}^\top)$. From the characteristic equation

$$D_{ol}(s) + kk_c N_{ol}(s) = 0, \tag{8}$$

the following magnitude and phase condition can be deduced:

$$k\,|k_c| \left|\frac{N_{ol}(s)}{D_{ol}(s)}\right| = 1 \tag{9a}$$

$$\angle k_c + \sum_{l=1}^{m} \angle(s - z_i) - \sum_{l=1}^{n} \angle(s - p_i) = q\pi, \tag{9b}$$

where $z_1, \ldots, z_m$ and $p_1, \ldots, p_n$ are the open-loop zeros and poles, respectively, and $q = \pm 1, \pm 3, ...$

Taking (8) and (9) into account, some simple construction rules for the complex root locus can be derived which are similar to the classical case. Obviously, the number of root locus branches is equal to n since $D_{cl}(s)$ is a polynomial of degree n. Furthermore, as in the classical case, it is apparent that the root locus starts at the open-loop poles for $k = 0$ and it can be shown that for $k \to \infty$, m branches end in open-loop zeros and $n - m$ branches tend to infinity along specified asymptotes. For polynomials with complex coefficients, there are also rules for calculating break-away and break-in points, angle of departure from poles or arrival at zeros, and crossing points with the imaginary axis. We omit the details due to space and brevity reasons and refer to Dòria-Cerezo and Bodson (2013).

It should be noted that in contrast to the classical case, the branches of the complex root locus do not have to be symmetric with respect to the real axis. However, the following relation proves beneficial for our further analysis.

Lemma 1. Given the root locus for a complex number k_c, the root locus for the complex conjugate $\overline{k}_c$ is obtained by mirroring the root locus for k_c on the real axis.

Proof. From the magnitude and phase condition (9) and the relations $|\overline{k}_c| = |k_c|$ and $\angle\overline{k}_c = \angle k_c + \pi$, it is clear that the root locus for $\overline{k}_c$ is rotated by 180° compared to the root locus for k_c and, thus, mirrored on the real axis.

3.2 Known Communication Topology

With the help of the root locus design rules for complex polynomials, we can simply check and, if available, calculate the valid gains k which stabilize system (4) for a given $\lambda_i(\boldsymbol{L}_\mathcal{G})$. Note that as in the real coefficient case, system (4) or polynomial (6) is said to be Hurwitz stable if all eigenvalues or poles are in the open left-half of the complex plane. Thus, if the communication graph or the eigenvalues of the Laplacian matrix are known, we can construct the complex root locus for each eigenvalue of the Laplacian matrix and check the solution set for each eigenvalue. Since the same controller gain k must be applied to all systems, we need to find the gains which simultaneously stabilize (4) for all $\lambda_i(\boldsymbol{L}_\mathcal{G})$, $i = 2, \ldots, N$. Therefore, we can construct the root locus for each $\lambda_i(\boldsymbol{L}_\mathcal{G})$ separately and calculate the set of valid gains $\mathcal{K}_{\lambda_i(\boldsymbol{L}_\mathcal{G})} \subset \mathbb{R}$ which lead to a stable polynomial (6). Then, synchronization will be achieved for any controller gain $k \in \mathcal{K}_{\boldsymbol{L}_\mathcal{G}}$, where $\mathcal{K}_{\boldsymbol{L}_\mathcal{G}}$ denotes the intersection $\mathcal{K}_{\boldsymbol{L}_\mathcal{G}} = \mathcal{K}_{\lambda_2(\boldsymbol{L}_\mathcal{G})} \cap \ldots \cap \mathcal{K}_{\lambda_N(\boldsymbol{L}_\mathcal{G})}$. This means that any $k \in \mathcal{K}_{\boldsymbol{L}_\mathcal{G}}$ stabilizes the virtual system (4) for every given $\lambda_i(\boldsymbol{L}_\mathcal{G})$, $i = 2, \ldots, N$.

If there is no valid solution, i.e. $\mathcal{K}_{\boldsymbol{L}_\mathcal{G}} = \emptyset$, it is not possible to synchronize the agents with a static output feedback controller of the form (2). However, it might be possible to achieve synchronization with a dynamic output feedback controller

$$u_i(s) = -K(s)\zeta_i(s) = -k\frac{N_k(s)}{D_k(s)}\zeta_i(s). \tag{10}$$

It is easy to see that the closed-loop transfer function of the virtual system (4) is then given by

$$G_{cl}(s) = \frac{N_{ol}(s)N_k(s)}{D_{ol}(s)D_k(s) + k\lambda_i(\boldsymbol{L}_\mathcal{G})N_{ol}(s)N_k(s)}.$$

Thus, under consideration of the complex root locus rules, we can design the polynomials $N_k(s)$ and $D_k(s)$ of the dynamic controller (10) such that there exists a valid controller gain k, i.e. a non-empty set $\mathcal{K}_{\boldsymbol{L}_\mathcal{G}}$, solving the synchronization problem.

3.3 Unknown Communication Topology

If the network connection or the communication graph is known, it is not difficult to design a static or dynamic output feedback synchronization controller, as described in Section 3.2. But if the network connection is not known a priori or if the same controller should hold for variable network connections, the Laplacian eigenvalues are not specified. Then, the solution is more demanding and the approach must be modified.

Note that the region of the Laplacian eigenvalues is restricted due to the following properties:

- From the *Gersgorin Disc Theorem* (Horn and Johnson (1990)), we know that all eigenvalues of the Laplacian matrix are contained in a circle with radius $N-1$ centered at $N-1$ in the complex plane (N is the number of agents).
- Dmitriev and Dynkin (1945) showed that the angles of the Laplacian eigenvalues satisfy
$$-\left(\frac{\pi}{2}-\frac{\pi}{N}\right) \le \angle\lambda_i(\boldsymbol{L}_{\mathcal{G}}) \le \left(\frac{\pi}{2}-\frac{\pi}{N}\right). \quad (11)$$
- Furthermore, Agaev and Chebotarev (2005) proved that the eigenvalues lie in a second circle also with radius $N-1$ centered at 1, and their imaginary parts are bounded by $|\Im\mathfrak{m}\{\lambda_i(\boldsymbol{L}_{\mathcal{G}})\}| \le \frac{1}{2}\cot(\frac{\pi}{2N})$. They also showed that shifting the imaginary axis to the point N, the angles of the eigenvalues satisfy
$$\left(\frac{\pi}{2}+\frac{\pi}{N}\right) \le \angle\lambda_i(\boldsymbol{L}_{\mathcal{G}}) \le -\left(\frac{\pi}{2}+\frac{\pi}{N}\right).$$

Summarizing the above results, we get an eigenvalue region depicted in Fig. 1, where the Laplacian eigenvalues for a network of N agents lie in the shaded region. In the following, we propose a method to analyze the complex root locus for the closed-loop polynomial (6) as a function of k and $\lambda_i(\boldsymbol{L}_{\mathcal{G}})$.

For this purpose, we first reformulate the possibly complex eigenvalues as
$$\lambda_i(\boldsymbol{L}_{\mathcal{G}}) = \sigma_i + j\omega_i = \sigma_i(1 + jm_i), \quad (12)$$
where $m_i = \frac{\omega_i}{\sigma_i}$ and j is the imaginary unit $j = \sqrt{-1}$. Taking (12) into account and substituting $k_i = k\sigma_i$ and $k_c(m_i) = (1 + jm_i)$, (6) can be written as
$$D_{ol}(s) + k_i k_c(m_i) N_{ol}(s) = 0. \quad (13)$$
Note that k_i is real and $k_c(m_i)$ is a complex number. From (13), we can derive that all Laplacian eigenvalues that lie on the same straight line through the origin lead to the same root locus curve. This follows from the fact that the slope m_i is equal for all points on a straight line, only k_i will be scaled by different parameters σ_i. Hence, for a fixed m_i we can construct the root locus and consider the set of stabilizing gains k_i and, then, deduce the valid solutions k.

Let us consider an eigenvalue $\lambda_i(\boldsymbol{L}_{\mathcal{G}}) = \sigma' + j\omega'$ with slope $m_i = m'$ and define $k_{lb}(m')$ and $k_{ub}(m')$ such that if $k' < k_{lb}(m')$ or $k' > k_{ub}(m')$ the closed-loop system is unstable. Then, the controller gain must satisfy
$$k_{lb}(m') < k' < k_{ub}(m') \quad (14)$$
to get a stable closed-loop system. From (14) and with the relation $k = \frac{k'}{\sigma'}$, it is possible to determine the set of synchronization gains for this slope as
$$\frac{k_{lb}(m')}{\sigma'} < k < \frac{k_{ub}(m')}{\sigma'}. \quad (15)$$
The problem is that σ' is unknown since the network graph and the Laplacian eigenvalues are unknown. However, if we have a lower and upper bound for σ', it is possible to estimate the set of synchronization gains k. Suppose that $\sigma_{lb}(m') \le \sigma' \le \sigma_{ub}(m')$, then it is clear that all gains satisfying
$$\frac{k_{lb}(m')}{\sigma_{lb}(m')} < k < \frac{k_{ub}(m')}{\sigma_{ub}(m')}$$
are also contained in (15).

As long as Assumption 2 holds, there is a lower bound σ_{lb} with $0 < \sigma_{lb} \le \min_{i\ge2} \Re\mathfrak{e}\{\lambda_i(\boldsymbol{L}_{\mathcal{G}})\}$. However, to the best of the

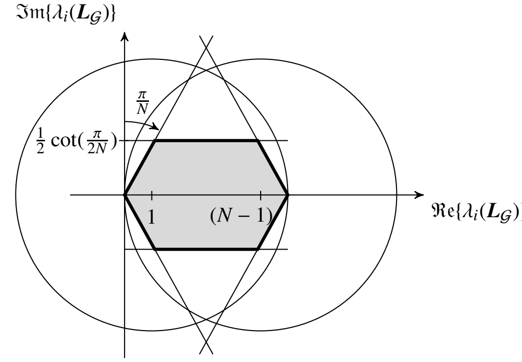

Fig. 1. The shaded area represents the region of the Laplacian eigenvalues (according to Agaev and Chebotarev (2005)).

authors' knowledge, there does not exist an estimation for the minimum of the real part of the non-zero Laplacian eigenvalues. In the synchronization literature it is usually assumed that $\min_{i\ge2} \Re\mathfrak{e}\{\lambda_i(\boldsymbol{L}_{\mathcal{G}})\}$ is known or σ_{lb} is chosen sufficiently small. We discuss this issue in Section 4 again, but in the following it is assumed that $\sigma_{lb}(m_i) = \sigma_{lb}$ for all slopes m_i.

An estimation for the upper bound $\sigma_{ub}(m_i)$ can easily be determined by considering the eigenvalue region shown in Fig. 1. Given a certain slope m', an upper bound for σ' is obviously determined by the real part of the intersection point described by the boundary of the eigenvalue region with a straight line through the origin and slope m'. Fig. 2 illustrates the idea, where some slopes are plotted for demonstration purposes.

Altogether, an estimation for the valid synchronization gains can be calculated as a function of the slope m_i and is given by
$$\frac{k_{lb}(m_i)}{\sigma_{lb}} < k < \frac{k_{ub}(m_i)}{\sigma_{ub}(m_i)}. \quad (16)$$
Therefore, it is possible to construct the complex root locus for the polynomial (13) as a function of k_i and m_i. Herein, m_i varies from zero to $m_{\max} = \tan(\frac{\pi}{2} - \frac{\pi}{N})$ and k_i varies from zero to infinity. Note that since the Laplacian eigenvalues appear in complex conjugate pairs, it suffices to consider only positive (or negative) slopes m_i, as depicted in Fig. 2. From Lemma 1, we know that the root locus for the complex conjugate counterpart results from mirroring on the real axis.

Remark 2. The estimated range (16) might be empty, while (given the Laplacian eigenvalues are known) the exact range (15) could have solutions. However, every gain satisfying (16) is also a solution for (15) and, thus, guarantees synchronization.

Remark 3. The eigenvalue region shown in Fig. 1 or Fig. 2 is a hexagon and appears for $3 < N < 19$. For $N = 2$ the region shrinks to the segment $[0, 2]$ (cf. (11)), for $N = 3$ we get a rhombus, and for $N \ge 19$ the relevant eigenvalue region contains arc segments (Agaev and Chebotarev (2005)). However, the described procedure is basically the same.

Remark 4. Instead of a stability range like (14), it is also possible to get conditions like
$$k_i < k_{lb}(m_i) \quad \vee \quad k_i > k_{ub}(m_i).$$
The estimation for the valid synchronization gains is then
$$k < \frac{k_{lb}(m_i)}{\sigma_{ub}(m_i)} \quad \vee \quad k > \frac{k_{ub}(m_i)}{\sigma_{lb}}.$$
There might also be other combinations of stability ranges. In general, it must be considered that the correct bound σ_{lb} or $\sigma_{ub}(m_i)$ is chosen as denominator to get a valid estimation.

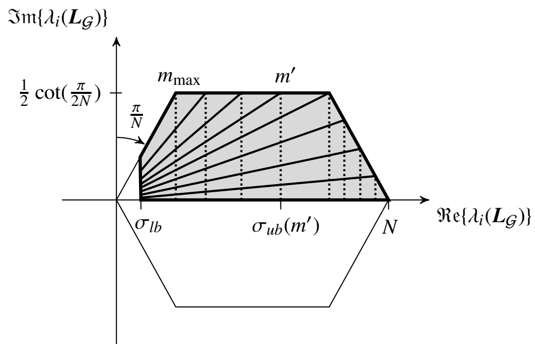

Fig. 2. Eigenvalue region (for $3 < N < 19$) and relevant lower and upper bounds for σ_i.

3.4 Concluding Notes

It is worth mentioning that the synchronization gain designed in Section 3.3 is robust against communication malfunctions or drop outs of agents. This results from the fact that the controller gain (16) is designed to synchronize a network of N agents for all possible communication structures (provided it contains a directed spanning tree). Additionally, if one or more agents drop out, we get a new graph with $N^\star < N$. However, from Fig. 1 it is apparent that the Laplacian eigenvalue region for $N^\star$ is fully contained in the eigenvalue region for N.

Furthermore, it should be considered that there exist also other techniques to determine the synchronization gain. For instance, stability of the characteristic polynomial (6) can be analyzed by the Hurwitz test for complex polynomials (Frank (1946)). However, the root locus design is a clear method which can be illustrated graphically, while the Hurwitz test leads to a set of nonlinear inequalities. This is difficult to solve and intransparent in general, especially in case of higher order polynomials.

So far, we have considered the asymptotic synchronization problem. Moreover, an interesting point for future work would be to optimize performance issues like synchronization speed and overshoot. Note that stability of the virtual system (4) is a measure for the transient synchronization behavior. Thus, taking the root locus rules into account, it is possible to design (dynamic) output feedback controllers such that the characteristic polynomial (6) satisfies a desired pole region. Due to its graphical interpretability, the root locus design method is well suited for this task.

4. EXAMPLE

As an example, we consider a network of $N = 6$ agents with

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -6 & -1 & -6 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \boldsymbol{c}^\mathsf{T} = \begin{bmatrix} 3 & 1 & 0 \end{bmatrix},$$

or

$$G_{ol}(s) = \frac{N_{ol}(s)}{D_{ol}(s)} = \frac{s+3}{(s^2+1)(s+6)}. \tag{17}$$

The open-loop poles and zeros are $p_{1,2} = \pm j$, $p_3 = -6$, and $z_1 = -3$, respectively. We assume that the network structure is given by a ring topology, leading to the Laplacian eigenvalues $\lambda_1(\boldsymbol{L}_\mathcal{G}) = 0$, $\lambda_{2,3}(\boldsymbol{L}_\mathcal{G}) = 0.5 \pm j0.866$, $\lambda_{4,5}(\boldsymbol{L}_\mathcal{G}) = 1.5 \pm j0.866$, and $\lambda_6(\boldsymbol{L}_\mathcal{G}) = 2$.

Known Communication Topology

First, suppose that the communication structure or the Laplacian eigenvalues are known. Then, given the static controller (2), the characteristic equation for synchronization reads

$$(s^2+1)(s+6) + k\lambda_i(\boldsymbol{L}_\mathcal{G})(s+3) = 0. \tag{18}$$

Calculating the root locus for $\lambda_2(\boldsymbol{L}_\mathcal{G})$, $\lambda_4(\boldsymbol{L}_\mathcal{G})$ and $\lambda_6(\boldsymbol{L}_\mathcal{G})$ separately (note that it suffices to consider only one of the complex conjugate pairs), we get the curves shown in Fig. 3. Obviously, there is no controller gain leading to a Hurwitz polynomial (18). Even though the closed-loop polynomial for $\lambda_6(\boldsymbol{L}_\mathcal{G})$ is Hurwitz for all $k > 0$, at least one branch of the root loci for $\lambda_2(\boldsymbol{L}_\mathcal{G})$ and $\lambda_4(\boldsymbol{L}_\mathcal{G})$ is always in the open right-half of the complex plane. Hence, the set of synchronization gains k is empty ($\mathcal{K}_{\boldsymbol{L}_\mathcal{G}} = \emptyset$).

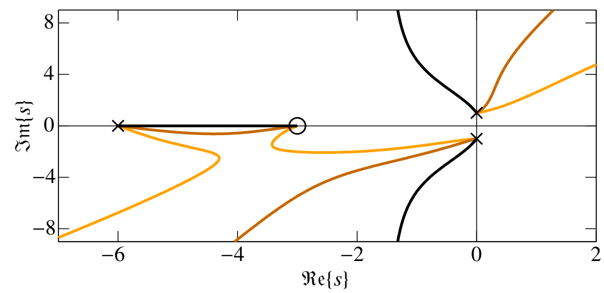

Fig. 3. Complex root locus plots of (18) for $\lambda_2(\boldsymbol{L}_\mathcal{G})$ (orange), $\lambda_4(\boldsymbol{L}_\mathcal{G})$ (brown), and $\lambda_6(\boldsymbol{L}_\mathcal{G})$ (black). The open-loop poles and zeros are marked with an "×" and "○", respectively.

Nevertheless, to get a valid synchronization gain, we use a dynamic output feedback controller (10). Under consideration of the complex root locus rules, we choose $D_k(s) = s + 20$ and $N_k(s) = s + 2$, leading to the characteristic equation

$$(s^2+1)(s+6)(s+20) + k\lambda_i(\boldsymbol{L}_\mathcal{G})(s+3)(s+2) = 0. \tag{19}$$

Again, calculating the root locus for the given $\lambda_i(\boldsymbol{L}_\mathcal{G})$, we get the following solution sets:

$$\mathcal{K}_{\lambda_{2,3}(\boldsymbol{L}_\mathcal{G})} = [51.044, 327.125]$$
$$\mathcal{K}_{\lambda_{4,5}(\boldsymbol{L}_\mathcal{G})} = (0, 897.404]$$
$$\mathcal{K}_{\lambda_6(\boldsymbol{L}_\mathcal{G})} = (0, \infty).$$

The intersection is determined by $\mathcal{K}_{\boldsymbol{L}_\mathcal{G}} = \mathcal{K}_{\lambda_{2,3}(\boldsymbol{L}_\mathcal{G})}$. Hence, synchronization will be achieved for all gains $k \in \mathcal{K}_{\lambda_{2,3}(\boldsymbol{L}_\mathcal{G})}$. The corresponding root loci are depicted in Fig. 4.

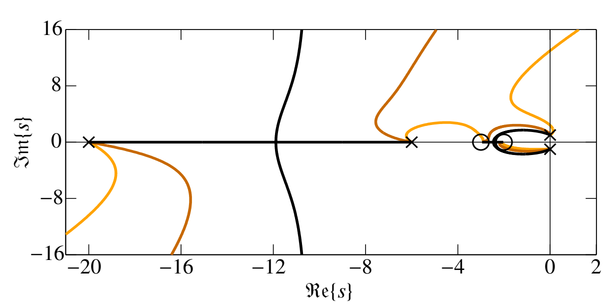

Fig. 4. Complex root locus plots of (19) for $\lambda_2(\boldsymbol{L}_\mathcal{G})$ (orange), $\lambda_4(\boldsymbol{L}_\mathcal{G})$ (brown), and $\lambda_6(\boldsymbol{L}_\mathcal{G})$ (black).

Unknown Communication Topology

Next, suppose that the network connection is unspecified. As described in Section 3.3, in this case we consider the root locus of the characteristic equation

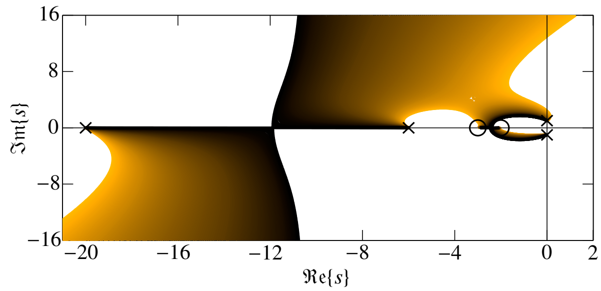

Fig. 5. Complex root locus plot for unknown communication topology, with m_i varying from $m_i = 0$ (black) to $m_i = \tan(\frac{\pi}{3})$ (orange).

$$(s^2 + 1)(s + 6)(s + 20) + k_i(1 + jm_i)(s + 3)(s + 2) = 0 \quad (20)$$

as a function of k_i and m_i. Varying m_i from 0 to $m_{\max} = \tan(\frac{\pi}{3})$ and k_i from 0 to infinity, we get a family of root locus curves as shown in Fig. 5. Under consideration of the Laplacian eigenvalue region (cf. Fig. 2), the following estimation for the set of synchronization gains is calculated:

$$\frac{25.9}{\sigma_{lb}} \leq k \leq 151.761,$$

where σ_{lb} is a lower bound for the real part of the non-zero Laplacian eigenvalues. For the given ring topology, a valid choice is $\sigma_{lb} = 0.5$ leading to

$$51.8 \leq k \leq 151.761$$

which is coherent with the calculated exact set $\mathcal{K}_{L_{\mathcal{G}}}$. Furthermore, we can see that the estimated synchronization set is empty for $\sigma_{lb} < 0.171$. Thus, the estimation delivers a solution for all communication structures where the real part of the non-zero Laplacian eigenvalues is greater than 0.171. Since the lower bound σ_{lb} is not known in general, it is beneficial to choose k large enough. For instance, the choice $k = 150$ synchronizes the multi-agent system for all network connections where the real part of the non-zero Laplacian eigenvalues is greater than or equal to 0.173.

5. CONCLUSION

We have proposed a root locus design strategy for the synchronization of multi-agent SISO systems in directed networks. In contrast to existing results, our approach is not restricted to a special class of network connections or agent models (exept that they are SISO). Furthermore, since the developed controller is an output feedback controller, it is not necessary to have access to relative state information or to exchange additional information through the network. We have presented solutions for the case that the communication structure is known, and also if it is not known a priori.

REFERENCES

Agaev, R. and Chebotarev, P. (2005). On the spectra of nonsymmetric laplacian matrices. *Linear Algebra and its Applications*, 399, 157–168.

Chen, T., Zhu, J., and Shi, Y. (2009). Consensus of linear time-invariant multi-agent systems based on dynamic output feedback. In *Proc. of the Chinese Control and Decision Conference*, 4201–4205.

Dmitriev, N. and Dynkin, E. (1945). On the characteristic numbers of a stochastic matrix. *Doklady Acad. Sci. U.S.S.R.*, 49(3), 159–162.

Dòria-Cerezo, A. and Bodson, M. (2013). Root locus rules for polynomials with complex coefficients. In *Proc. of the 21st Mediterranean Conf. on Control & Automation*, 663–670.

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

Frank, E. (1946). On the zeros of polynomials with complex coefficients. *Bulletin of the American Mathematical Society*, 5(2), 144–157.

Horn, R.A. and Johnson, C.R. (1990). *Matrix Analysis*. Cambridge University Press.

Lafferriere, G., Williams, A., Caughman, J., and Veerman, J.J.P. (2005). Decentralized control of vehicle formations. *Systems & Control Letters*, 54(9), 899–910.

Li, Z., Duan, Z., Chen, G., and Huang, L. (2010). Consensus of multiagent systems and synchronization of complex networks: A unified viewpoint. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 57(1), 213–224.

Li, Z., Liu, X., Lin, P., and Ren, W. (2011). Consensus of linear multi-agent systems with reduced-order observer-based protocols. *Systems & Control Letters*, 60(7), 510–516.

Lin, J., Morse, A.S., and Anderson, B.D.O. (2003). The multi-agent rendevouz problem. In *Proc. of the 42nd IEEE Conference on Decision and Control*, 1508–1513.

Listmann, K.D. (2012). *Synchronization of Networked Linear Systems - an LMI Approach*. Ph.D. thesis, Technische Universität Darmstadt, Darmstadt, Germany.

Ma, C.Q. and Zhang, J.F. (2010). Necessary and sufficient conditions for consensusability of linear multi-agent systems. *IEEE Transactions on Automatic Control*, 55(5), 1263–1268.

Olfati-Saber, R. (2006). Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3), 401–420.

Olfati-Saber, R., Fax, J.A., and Murray, R.M. (2007). Consensus and cooperation in networked multi-agent systems. *Proc. of the IEEE*, 95(1), 215–233.

Ren, W., Beard, R.W., and Atkins, E.M. (2007). Information consensus in multivehicle cooperative control. *IEEE Control Systems Magazine*, 27(2), 71–82.

Seo, J.H., Shim, H., and Back, J. (2009a). Consensus and synchronization of linear high-order systems via output coupling. In *Proc. of the European Control Conf.*, 767–772.

Seo, J.H., Shim, H., and Back, J. (2009b). Consensus of high-order linear systems using dynamic output feedback compensator: Low gain approach. *Automatica*, 45(11), 2659–2664.

Tanner, H.G., Jadbabaie, A., and Pappas, G.J. (2007). Flocking in fixed and switching networks. *IEEE Transactions on Automatic Control*, 52(5), 863–868.

Tuna, S.E. (2008). LQR-based coupling gain for synchronization of linear systems. [Online] `http://arxiv.org/abs/0801.3390`.

Wang, J., Liu, Z., and Hu, X. (2009). Consensus of high order linear multi-agent systems using output error feedback. In *Proc. of the 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, 3685–3690.

Zhang, H., Lewis, F.L., and Das, A. (2011). Optimal design for synchronization of cooperative systems: State feedback, observer and output feedback. *IEEE Transactions on Automatic Control*, 55(8), 1948–1952.

Zhao, Y., Wen, G., Duan, Z., and Xu, X. (2011). A new observer-type consensus protocol for linear multi-agent dynamical systems. In *Proc. of the 30th Chinese Control Conference*, 5975–5980.