1

INTRODUCTION

Ever since the advent of the Internet and of the world wide web in the 1980's, and more recently with the upswing of the Internet of Things (IoT), one thing has become exceedingly clear: the world in which we live is becoming an increasingly interconnected place. Our phone talks to our watch, our washing machine talks to our refrigerator, and our solar panels talk to those of our neighbor. While only one of these examples is currently commonplace, people in the control community are actively working on making the other two a reality.

Interconnecting different systems creates a larger, more complex system which we call a *network*. Such networks occur when, for instance, modeling systems from physics (connected oscillators), artificial intelligence (UAV's), biological chemistry (gene regulatory networks), or energy systems (smart-grids). The behavior and properties of these networks are determined not only by the properties of the individual subsystems, but also by the specific structure with which these subsystems are interconnected. In this setup of multiple interconnected system, we speak of *complex networks* or *networked multi-agent systems*. In the theory of networked multi-agent systems, the subsystems are dynamical systems called the *agents*. These agents exchange information according to a certain communication topology. In general this topology is modeled using a graph called the *network graph*. In this graph, the nodes represent the agents of the network, while the communication links are represented by the edges. Depending on the context, the agents can have either identical dynamics in which case we talk about *homogeneous* networks, or distinct dynamics, in which case the network is called *heterogeneous*. Likewise, the network graph can be directed or undirected and weighted or unweighted. In this thesis we will consider many of the different types of networks. The manner in which the agents in the network exchange local information is called a *communication protocol*. An important object in

the theory of multi-agent systems is the *Laplacian matrix* of the network graph. Many of the properties of a multi-agent system can be expressed in terms of the eigenvalues of the Laplacian, see e.g. [39, 68].

For many different problems in the context of networked systems, the aim is to design a protocol that achieves a certain goal. One such problem is the well-known problem of *synchronization*. Other important subjects in networked systems include flocking, formation control, sensor placement, and controllability of networks, see e.g. [10, 13, 15, 18, 41, 47, 48].

1.1 SYNCHRONIZATION

One of the first problems to be researched in the theory of networked systems is the problem of synchronization of interconnected systems. In the synchronization problem the agents can be models of similar or identical physical systems, such as vehicles or oscillators, and the goal is to find conditions on the communication protocols under which the states of all the agents in the network converge to a single common trajectory. If the protocol achieves this goal, then the network is said to be synchronized. Already in 1665, the Dutch mathematician Christiaan Huygens observed that two identical pendulum clocks that have been mounted on a common wall will tend to synchronize in some sense: over time their pendulums will swing either in phase or in anti-phase [63]. Since those early beginnings in the 17th century, it has taken a few years for synchronization to really capture the attention of the control community. Starting at the end of the 20th century however, it has gathered a substantial amount of interest and research, see e.g. [34, 40, 58, 62, 70].

Closely related to the synchronization problem, and as well-known, is is the problem of *consensus*. Within the problem of consensus the agents in the network, perhaps modeling a network of sensors, again exchange local information with their neighbors only. The goal of this information exchange is for the whole network to reach agreement on certain quantities of interest depending on the states of all the agents. A protocol that achieves this aim is said to achieve consensus. Important work on the

consensus problem can be found if for instance [49, 50, 54, 55, 66], with some more recent work in e.g. [37, 38].

At first, the literature on synchronization and consensus considered mostly networks of simple systems such as scalar systems with single or double integrator agent dynamics. More recent work has diverted its attention to networks with higher order agent dynamics. In these networks the agent dynamics is a general finite dimensional linear input-output system, see [15, 37, 38]. For networks with scalar agent dynamics or networks with higher dimensional agent dynamics where the relative state is available, it is often possible to achieve synchronization or consensus with static communication protocols. However, when working with higher dimensional systems, the agents are not always able to exchange their entire relative state. In these cases dynamic *observer-based protocols*, using the measured relative output of the agents, can be used to achieve the desired goal of consensus or synchronization. Using a dynamic protocol allows us to first create an estimate the relative state of each agent, which we then feed back into the network.

An important concept in the field of control theory is *robustness*. In control theory, we often consider dynamical models representing physical real world systems. This is especially prevalent in the networked system problems of synchronization and formation control. When constructing these mathematical models we make many idealized assumptions about the dynamics of the physical system, which leads to a close but not completely accurate description of said physical system. However, it might not be unreasonable to assume that the precise dynamics of the physical system lies within a certain (to be appropriately defined) neighborhood of our idealized mathematical model, which we call the the *nominal system*. If a controller achieves a certain goal for all systems within a specified neighborhood of the nominal system then the controller is said to achieve that goal robustly. One of the most well-known robustness problems is the problem of robust stabilization. In this case, the goal is to find a controller that stabilizes the nominal system robustly.

Results from $\mathcal{H}_\infty$ and robust control theory, such as the famous small gain theorem, have been applied to the problem of *robust synchronization* for networks with uncertain agent dynamics. Here the nominal agent dy-

namics is identical for all the agents in the network, and is given by an ordinary linear input-output system. For each agent this system is then interconnected with an unknown system representing the perturbation. The exact interconnection between the nominal dynamics and the perturbation depends on the context. For networks with *additively perturbed* agent dynamics, conditions for the existence of observer-based dynamic protocols achieving robust synchronization and methods to obtain such protocols were established in [65]. In the second chapter of this thesis, we consider the problem in which the nominal agent dynamics are instead perturbed with *coprime factor perturbations*. We provide methods to construct observer-based protocols that achieve robustly synchronization of the network for all possible perturbations whose $\mathcal{H}_\infty$-norm lies within a certain achievable interval.

1.2 MODEL REDUCTION

Another topic that has gathered a great amount of interest is the problem of *model reduction*. Model reduction hinges on the following idea: if we consider a complex system which is difficult to simulate, analyze, or control; can we approximate this system by a simplified model? Another important aspect of the model reduction problem is to guarantee that certain properties of the original model are preserved in the reduced model. Furthermore it is important to establish some measure of exactly how accurate the reduced model approximates the original system. In the past few decades multiple model reduction techniques have been developed, each with a different approach to the problem.

In [44] Lyapunov-based balanced truncation was first introduced. The aim in balanced truncation is to construct a representation of the system in which states that are easy to reach are also easy to observe, and to make those that are difficult to reach difficult to observe as well. Information regarding the difficult to be reached and observed states is then truncated from the model, leading to a lower dimensional system. Hankel-norm approximation takes a different approach to model reduction. Here the goal is to find an approximating system that is optimal

with respect to a certain norm, see for instance [1, 77]. Finally, Krylov-subspace based methods, known as moment-matching methods, take a different direction altogether. In this technique Krylov-subspace projections are used to obtain a reduced order approximating system, see e.g. [16, 17, 27].

When working with large scale networks, the dynamical models can easily become extremely complex and high-dimensional. It is intuitively understandable that problems for networked systems such as analysis and controller design can benefit greatly from model reduction. However, direct application of the model reduction techniques mentioned above generally leads to a complete loss of a very important property of the system: the structure of the network itself. Often, it is impossible to again interpret the reduced system as a network of interconnected subsystems. This in turn leads to the situation that the tools that are normally available for dealing with networked systems are no longer usable after the model reduction step.

In the past few years, different model reduction techniques have been introduced that try to preserve some of the network and subsystem structure of the original networked system. For example there exist techniques that preserve the Lagrangian structure [35], the second order structure [4, 36], and the subsystem interconnection structure of the model [53, 57, 69]. For networked systems, the most important structure is of course the topology of the network. Recently model reduction techniques specifically designed for networked systems have been introduced. These techniques fall into roughly two categories: techniques that reduce the dynamic order of the individual agents, see [42], and techniques that reduce the complexity of the network topology, for instance by clustering the agents in the network graph, see [8, 24, 25]. In clustering based techniques the idea is to partition the agents of the network into disjoint sets called *clusters*, and to associate a single new node with each of the clusters in the graph of the reduced network. In this way, the number of agents in the network is reduced, leading to a reduction of the dynamical order of the entire networked system.

In this thesis we present a clustering based model reduction technique that employs a special class of graph partitions called *almost equitable par-*

titions to cluster networks with arbitrary higher dimensional agent dynamics. These techniques extend the results from [43] which considered networks with single integrator agent dynamics. In these networks, the agents are divided into two groups: a group of leaders and a group of followers. The followers can only communicate with other agents in the graph, but the leaders receive an external output. We provide a priori upper bounds on the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ approximation errors if the agents in the graph are clustered according to an almost equitable partition of the network graph. For some graphs however, it can be either difficult or impossible to find almost equitable partitions that are actually useful for clustering. This might be the case if, for instance, the graph only has trivial almost equitable partitions. If the chosen cluster is not almost equitable for the given network, one can ask oneself the following question: is it possible to modify the network graph in such a way that the chosen partition becomes an almost equitable partition of the new network graph? In the final part of the third chapter, we investigate this problem and briefly investigate how one might apply the obtained results for clustering networks according to arbitrary partitions.

Inspired by the results in the third chapter and the difficulty of finding exact expressions for the approximation error in the case of arbitrary partitions, we investigate a related model reduction problem in the fourth chapter of this thesis. In this chapter, we investigate the $\mathcal{H}_2$ approximation error when certain edges are removed from the network graph. In a graph, cycles can be considered redundant in a certain sense: for connectedness of a graph the existence of a spanning tree is necessary, and in the case of single integrator agent dynamics also a sufficient condition for the network to reach consensus.

In the theory of graph sparsification this idea is further developed and algorithms are presented to compute a sparse graph, where the number of edges is of the same order as the number of nodes, approximating a full graph with many edges, see e.g. [60, 61]. The approximating graphs are close to the original graphs in a certain sense, for instance in the sense of cut-similarity [3] or spectral-similarity [61]. If the approximating graph is spectrally similar to the original graph, then the Laplacian

eigenvalues of the approximating graph are close to that of the original graph.

While these algorithms provide an efficient way of obtaining a sparse approximating graph, they do not take any dynamics on the nodes of the graph into consideration. For resistor networks with scalar agent dynamics, the problem of model reduction by edge-removal was studied in [56] and [67]. In both papers, the nodes in the resistor networks are divided into a set of external nodes and one of internal nodes. In [56] an efficient method (reduceR) was introduced for finding an *equivalent* resistor network with the same number of external nodes, but with a reduced number of internal nodes. In [67] the number of external and internal nodes is kept the same for the reduced network. However, the approximations are not exact. The approximation error that is considered is the worst case relative error between the steady state voltages over the nodes of the original and reduced network. The upper bound on the approximation error given in [67] depends on the conductance matrices of both the original and the reduced network.

We investigate the problem of approximating networks where the agent dynamics are given by arbitrary symmetric dynamical system by a reduced network that is obtained by removing precisely those edges that close the cycles in the network graph. Inspired by the papers above, we utilize ideas from [73] and [74] on the $\mathcal{H}_2$-performance of consensus networks when adding and removing cycles in the network graph. The approximation error we consider is the $\mathcal{H}_2$-norm of the error system, comparing the output of the original and reduced networks.

1.3 OUTLINE OF THIS THESIS

This thesis is divided into two parts. In Part I, Chapter 2 we investigate the problem of robust synchronization for networks with coprime factor perturbed agent dynamics. The networks under consideration can be either directed or undirected unweighted networks. We provide a communication protocol that achieves consensus for all uncertain networks within a uncertainty interval. Part II provides several perspectives on the

problem of model reduction for networked multi-agent systems. Chapter 3 considers the problem model reduction of leader-follower networks by clustering using almost equitable partitions. Finally, in Chapter 4, we consider the problem of model reduction, or more accurately model simplification, by removing cycles from the network graph. Conclusions on the main contributions and an outlook on further research opportunities are detailed in Chapter 5.

1.4 ORIGIN OF THE CHAPTERS

Chapter 2 is based on [31], which appeared in the first special Jan C. Willems memorial issue of Systems and Control Letters. Preliminary results were first presented in [28], at the 52nd IEEE Conference on Decision and Control (CDC) in Florence, Italy, and partial results for directed networks were presented at the 21st International Symposium on Mathematical Theory of Networks and Systems in Groningen, The Netherlands [29]. Results considering the model reduction problem in Chapter 3 of networks with scalar agent dynamics were partially presented at the 54th CDC in Osaka, Japan [30], while Chapter 3 itself is based on [33], which has been submitted for publication. Finally, Chapter 4 constitutes of [32], which is currently under review.

1.5 NOTATION

Throughout this thesis, we will use fairly standard notation. The most commonly used definitions will be listed here, while chapter specific notation can be found in the chapters themselves.

Sets

The field of real numbers is denoted $\mathbb{R}$, and the field of complex numbers is denoted $\mathbb{C}$. Let $\mathbb{R}^n$ denote the linear space of vectors with n real components and $\mathbb{R}^{n\times m}$ ($\mathbb{C}^{n\times m}$) the space of real (complex) $n \times m$ matrices. The set of non-negative real numbers is denoted $\mathbb{R}^+$. We denote

the left open half of the complex plane by $\mathbb{C}^-$. The cardinality of a set $\mathcal{S}$ is denoted $|\mathcal{S}|$.

Matrices and vectors

For a set of column vectors or real numbers $\alpha_1, \alpha_2, \ldots, \alpha_n$, we define

$$\operatorname{col}(\alpha_1, \alpha_2, \ldots, \alpha_n) = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix}.$$

The trace of a square matrix A is denoted $\operatorname{tr} A$ and is the sum of the diagonal entries of A. For matrices A, B, and C of appropriate dimensions such that ABC is square, the trace of ABC satisfies

$$\operatorname{tr} ABC = \operatorname{tr} CAB = \operatorname{tr} BCA.$$

Given a matrix $A \in \mathbb{C}^{n\times m}$, let A^T and A^* denote its transpose and conjugate transpose, respectively. For a square matrix M, let $\sigma(M)$ denote its spectrum. The spectral radius of M is denoted $\rho(M)$. The largest singular value of a matrix P is denoted $\sigma_1(P)$ and satisfies $\sigma_1(P) = \sqrt{\lambda_{\max}(P^T P)}$. Let $M \in \mathbb{R}^{n\times n}$ be a symmetric matrix. We write $M > 0$ if M is positive definite, and $M < 0$ if it is negative definite. If M is positive (negative) semi-definite, we write $M \geqslant 0$ ($M \leqslant 0$).

For a rectangular matrix A, let A^+ denote its Moore–Penrose pseudoinverse. Let $B \in \mathbb{R}^{n\times m}$ with $m < n$ have full rank. A left annihilator of B is denoted $B^\perp \in \mathbb{R}^{(n-m)\times n}$ and is any full-rank matrix such that $B^\perp B = 0$.

In this thesis, the vector of ones in $\mathbb{R}^n$ is denoted $\mathbb{1}_n$, and I_n denotes the $n \times n$ identity matrix. We will sometimes omit the subscript if the appropriate dimension is clear from the context.

For given real numbers $\alpha_1, \alpha_2, \ldots, \alpha_n$, let $\operatorname{diag}(\alpha_1, \alpha_2, \ldots, \alpha_n)$ denote the $n \times n$ diagonal matrix with the α_i's on the diagonal:

$$\operatorname{diag}(\alpha_1, \alpha_2, \ldots, \alpha_n) = \begin{pmatrix} \alpha_1 & & & \\ & \alpha_2 & & \\ & & \ddots & \\ & & & \alpha_n \end{pmatrix}.$$

In the case of a collection of square matrices $A_1, A_2, \ldots, A_n$, we use $\operatorname{diag}(A_1, A_2, \ldots, A_n)$ to denote the block diagonal matrix with the A_i's as diagonal blocks.

The Kronecker product of the matrices $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{p \times q}$ is denoted $A \otimes B \in \mathbb{R}^{mp \times nq}$. The Kronecker product is bilinear and associative. We list some of the important identities for the Kronecker product used in this thesis:

$$\begin{aligned} &A \otimes (B + C) = A \otimes B + A \otimes C, \\ &(A \otimes B)(C \otimes D) = AC \otimes BD, \\ &(A \otimes B)^T = A^T \otimes B^T. \end{aligned}$$

Transfer matrix norms

The $\mathcal{H}_2$-norm of a stable strictly proper real rational matrix G is denoted $\|G\|_2$ and is defined as

$$\|G\|_2 = \sqrt{\frac{1}{2\pi} \int_{-\infty}^{\infty} \operatorname{tr} G(-i\omega)^T G(i\omega) \; d\omega}.$$

For a stable proper real rational matrix G, its $\mathcal{H}_\infty$-norm is denoted $\|G\|_\infty$ and is given by

$$\|G\|_\infty := \sup_{\omega \in \mathbb{R}} \sigma_1(G(i\omega)).$$

The space of square-integrable functions over $\mathbb{R}^+$ is denoted $\mathcal{L}_2(\mathbb{R}^+)$, see e.g. [64].

Part I

ROBUST SYNCHRONIZATION