rijksuniversiteit groningen

Consensus and controllability of multi-agent systems with discrete-time communication

Master Thesis Applied Mathematics

July 2015

Student: J. Popping

Student number: s2020084

First supervisor: prof. dr. M.K. Camlibel

Second supervisor: prof. dr. ir. R.C.W.P. Verstappen

Abstract

In this thesis we introduce a new framework for multi-agent systems. In this framework the communication between agents is assumed to be discrete in time rather than continuous as in the classical framework. The motivation for the new framework stems from the fact that the communication is necessarily in discrete time in practice. For this new framework we investigate the problem of consensus. We prove that consensus is reached if the corresponding undirected graph is connected and every time interval is bounded, i.e. Δ_k is bounded for every k. To provide further insights into the convergence of solutions we present simulation results illustrating how agents reach consensus within the new framework. In addition, simulation results reveal the optimal choice of Δ for the case of fixed time intervals, i.e. $\Delta_k = \Delta$. Another problem we investigate is the controllability problem. To investigate this problem for the new framework, we derive the leader-follower system associated with this new framework. We show that if the corresponding continuous-time multi-agent system is controllable then this leader-follower system is controllable for all sufficiently small Δ_k as well.

Contents

1 Introduction

Consider a situation in which a group of sensors has to measure the temperature of a large greenhouse. Although the temperature measured by each sensor will vary according to its location, the sensor group has to agree on a single value representing the temperature of the greenhouse. Therefore, each sensor has to exchange information on its local measurements with other sensors. But transmitting and receiving information requires energy, which is a sparse commodity. Hence, only sensors within a limited communication range can share information directly. Based on the possible communication about the local measurements, the group has to agree on the temperature of the greenhouse. To agree about this temperature, the group needs a protocol [13].

This group of sensors is an example of a multi-agent system (MAS). Multi-agent systems are composed of several agents that interact according to a communication law to fulfill a certain task. In the example, the sensors were the agents, the communication law was: sensors were only able to communicate with sensors within their communication range and the task they had to fulfill was: agree on a single value representing the temperature of the greenhouse. The communication law is in most multi-agent systems given by a so-called communication graph. In such a graph the vertices represent the agents and the edges the interconnection structure of the network. That means, there exists an edge between two vertices if the corresponding agents are able to communicate. Hence, the overall behavior of the network is determined by two factors: the agents' dynamics and the underlying communication graph [14].

In the situation described above, the sensor group has to agree on the temperature of the greenhouse by exchanging information on its local measurements with sensors within their communication range. When agents in a multi-agent system agree on a certain quantity of interest only through interaction with their neighbors, this is called consensus. The problem of consensus is a well-known problem in the context of networks and multi-agent systems. During the last decades there was a lot of research interest in this problem due to its wide variety of applications such as flocking of birds, groups of autonomous mobile robots, unmanned air vehicles (UAVs), autonomous underwater vehicles (AUVs), satellites and automated highway systems [21].

Consensus problems have a long history in the field of computer science, in particularly in the field of distributed computing. In this field, distributed systems are studied, these are software systems in which computers located on networked computers communicate about their actions by passing messages in order to achieve a common goal [11]. In systems and control theory distributed computation over networks started with the pioneering work of Borkar and Varaiya [1], Tsitsiklis [25] and Tsitsiklis, Bertsekas, and Athans [26]. Building on the work of Fax and Murray (see [4] and [5]), Olfati-Saber and Murray introduced the theoretical framework for posing and solving consensus problems for networked dynamic systems in [17] and [18].

In [18] Olfati-Saber and Murray have described a consensus protocol to agree on the states of n integrator agents with simple dynamics. A consensus protocol is an interaction rule that specifies the information exchange between an agent and all of its neighbors in the network. Recall that in the situation described above, the sensors used a protocol to agree on the temperature of the greenhouse. In the papers [5], [7], [9], [15] and [22] a continuous-time consensus protocol is described as well.

The continuous-time protocol described in these papers can be summarized by:

$$\dot{x}_i(t) = -\sum_{j\in N_i(t)} \alpha_{ij}(t)(x_i(t) - x_j(t)) \tag{1}$$

where $x_i(t)$ denotes the state of agent i at time t, $N_i(t)$ is the set of agents whose information is available to agent i at time t and $\alpha_{ij}(t)$ is a positive time-varying weighting factor. Hence, this protocol drives the information state of each agent towards the states of its neighbors at each time. The neighbors of each agent can be time-varying. This continuous-time linear consensus protocol (1) can be written in matrix form as follows:

$$\dot{x} = -Lx \tag{2}$$

where $x = (x_1, x_2, \ldots, x_n)^T$ and L is the Laplacian matrix corresponding to the communication graph G [21].

Correspondingly, research has been done on a discrete-time consensus protocol for n integrator agents with simple dynamics in the papers [7], [16] and [22]. The discrete-time consensus protocol described in these papers can be summarized by:

$$x_i(t_{k+1}) = \sum_{j\in N_i(t_k)\cup\{i\}} \beta_{ij}(t_k)x_j(t_k) \tag{3}$$

where $x_i(t_{k+1})$ denotes the state of agent i at time t_{k+1}, $N_i(t_k)$ is the set of agents whose information is available to agent i at time t_k, $\sum_{j\in N_i(t_k)\cup\{i\}} \beta_{ij}(t_k) = 1$ and $\beta_{ij}(t_k) > 0$ for $j \in N_i(t_k) \cup \{i\}$. In other words, this protocol updates the next state of each agent as the weighted average of its current state and the current state of its neighbors. The neighbors are possibly time-varying. The discrete-time consensus protocol given in equation (3) can be written in matrix form:

$$x(k+1) = M(k)x(k) \tag{4}$$

where $x = (x_1, x_2, \ldots, x_n)^T$ and $M(k)$ is a stochastic matrix with positive diagonal entries [21]. Note that we have written $x(t_{k+1})$ as $x(k+1)$ in this equation. We will use this notation in the sequel.

In the study of consensus problems necessary and sufficient conditions for the convergence of these protocols were derived. These conditions are different for various assumptions on the information exchange topology. For example, the assumption can be made that the neighbors of the agents are the same at each point in time. The corresponding information exchange topology is called time-invariant. Under this topology, the continuous-time consensus protocol given in (2) achieves consensus asymptotically if and only if the corresponding topology has a spanning tree. This result was based on the fact that zero is a simple eigenvalue of the Laplacian matrix if and only if its digraph has a spanning tree. This was shown in [23] by an induction argument and in [10] by a constructive approach. Under a time-invariant information exchange topology, the discrete-time consensus protocol given in (4) reaches consensus asymptotically if and only if the corresponding digraph has a spanning tree as well [22].

The next step in the research of consensus problems was to find the equilibrium state to which the consensus protocol converges. If the time-invariant information exchange topology has a spanning tree the continuous-time consensus protocol given in (2) converges to $\sum_i \nu_i x_i(0)$, where $\nu = (\nu_1, \ldots, \nu_n)$ is the non-negative left-eigenvector of the Laplacian matrix L corresponding to the eigenvalue $\lambda = 0$ and $x_i(0)$ is the initial state of agent i. Under the same topology, the discrete-time consensus protocol given in (4) reaches the value $\sum_i \mu_i x_i(0)$, where $\mu = (\mu_1, \ldots, \mu_n)$ is the non-negative left-eigenvector of the matrix $M(k)$ corresponding to the eigenvalue $\lambda = 1$. If the fixed communication graph has a spanning tree it is not clear whether each agent will converge to the final equilibrium state. If the graph is strongly connected each agent's initial state will converge to the final equilibrium state, since the left-eigenvectors ν_i's and μ_i's are all positive [6]. Furthermore, if for all $i \neq j$ we have $\nu_i = \nu_j = \frac{1}{n}$ and $\mu_i = \mu_j = \frac{1}{n}$, the final consensus equilibrium will be the average of each agent's initial condition, which is called the average consensus in [18].

Another assumption on the information exchange topology is a possibly change of the topology over time. Such a topology is called a switching information exchange topology. Many research efforts on the coordination of multiple autonomous agents under switching information exchange topologies are motivated by Viscek's model [27]. One approach to tackle these switching topologies is the algebraic graph. Another approach uses nonlinear tools. For the discrete-time consensus protocol (4), set-valued Lyapunov theory is used. For the continuous-time consensus protocol (2) a Lyapunov function candidate is proposed as $V(x) = \max\{x_1, x_2, \ldots x_n\} - \min\{x_1, x_2, \ldots x_n\}$ in [15].

In addition to the assumptions on the information exchange topology, other assumptions can be taken into consideration as well. For example, the assumption that some agents have better information than others can be made. To reflect the uncertainty of agents about their information, weighting factors should be adjusted in the model. Furthermore, time-delays in communication channels can be considered in the model. The continuous-time consensus protocol is in this case given by the equation:

$$\dot{x}_i(t) = -\sum_{j \in N_i(t)} \alpha_{ij}(t)[x_i(t - \tau_{ij}) - x_j(t - \tau_{ij})]$$

where $x_i(t)$ denotes the state of agent i at time t, $N_i(t)$ is the set of agents whose information is available to agent i at time t, $\alpha_{ij}(t)$ is a positive time-varying weighting factor and τ_{ij} denotes the time delay for information communicated from agent j to agent i. In the case that $\tau_{ij} = \tau$, results for the converges of this protocol can be found in [15].

In the research history of consensus problems a lot of research effort is put into the continuous-time consensus protocol given in (2). In this model the assumption is made that the agents can communicate continuously with each other about their states. This assumption is not very reasonable in practice. If we consider for example the group of sensors which has to agree about a single value representing the temperature of a greenhouse based on the local measurements of the sensors, one can imagine that it is impossible for these sensors to transmit and receive information continuously. In practice it is impossible for agents to communicate continuously. Hence, a more reasonable assumption would be that sensors exchange information at certain (fixed) points in time. This assumption leads to a discrete-time system such as given in (4). But for this very general system it is hard to prove that this system reaches consensus. Therefore, we will come up with an easily formulated discrete-time system in this thesis. The new framework for multi-agent systems introduced in this thesis is:

$$\dot{x}(t) = -Dx(t) + Ax(t_k) \tag{5}$$

for $t_k \leq t < t_{k+1}$, where D is the degree matrix, A is the adjacency matrix and in the sequel we will use the notation $\Delta_k = t_{k+1} - t_k$.

This new framework for multi-agent systems can be written in the same form as (4), where in this case $M(k)$ has a special structure. Because of this special structure it is easier to prove that consensus is reached if the underlying communication graph is connected. In this thesis we will show that consensus is indeed reached if the underlying communication graph is connected and every time interval Δ_k is bounded.

Another problem we investigate for this new framework of multi-agent systems (5) is the controllability problem. We want to know how to influence this system just by controlling a fraction of the agents. We call the agents that are under the forcing of external control inputs in such a system the leaders and the other agents the followers. In such a leader-follower system it is interesting to study whether a desired collective behavior can be achieved in finite time by controlling the leaders. Hence, we would like to investigate the controllability of the overall network consisting of leaders and followers.

Studying the controllability of a leader-follower multi-agent system was first done by Tanner in [24]. In his paper he derived necessary and sufficient conditions for a group of systems interconnected via nearest neighbor rules to be controllable by one of them acting as a leader. These conditions were given in terms of the eigenvectors of the Laplacian matrix corresponding to a graph G. These conditions were not graph theoretical in nature, which means that controllability could not directly be decided from the graph topology itself. But for designing leader-follower type control strategies it is more desirable to have graph theoretical conditions. A more topological result was given by Mesbahi and Rahmani in [20]. In their paper, they came up with a sufficient condition for the network to be uncontrollable in case one leader was given. Their result was related to the symmetry and automorphism group of the underlying graph. Many more papers such as [2], [3], [8], [12] and [19] were dealing with different aspects of the controllability of the leader-follower continuous-time multi-agent system:

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

where $x = (x_1, x_2, \ldots, x_n)^T$ and the matrix B encodes the vertices through which external inputs can be applied, that means it encodes the leaders of the multi-agent system. For example, necessary and sufficient conditions for the controllability of this system were derived in these papers as well as lower and/or upper bounds on the controllable subspace. Also the minimum number of leaders that renders the system controllable is studied.

In this thesis we will study the leader-follower multi-agent system corresponding to the new framework for multi-agent systems (5). This system is based on the continuous-time leader-follower system (6). The leader-follower system corresponding to the new framework for multi-agent systems is:

$$\dot{x}(t) = -Dx(t) + Ax(t_k) + Bu(t_k) \tag{7}$$

for $t_k \leq t < t_{k+1}$, where D is the degree matrix, A is the adjacency matrix, B encodes the vertices through which external inputs can be applied (the leaders) and $\Delta_k = t_{k+1} - t_k$. In this thesis we will investigate the controllability of this system. We will relate the controllability of the new framework to the controllability of the corresponding continuous-time system (6). We will show that if the corresponding continuous-time system (6) is controllable then the new framework of multi-agent systems (7) is controllable for sufficiently small Δ_k as well.

1.1 Outline of this thesis

This thesis is organized as follows. In Chapter 2 we will provide some preliminaries and basic material needed in the sequel. In particular, we will discuss some basic notions of graphs and matrices as well as describe the classical framework for multi-agent systems. In the next chapter, Chapter 3, we will introduce the new framework for multi-agent systems. In this new framework the communication between agents is assumed to be discrete in time rather than continuous as in the classical framework. For this new framework we will investigate the problem of consensus in Chapter 4. We will show that the new framework for multi-agent systems introduced in this thesis will reach consensus if the corresponding undirected graph is connected and every time-interval Δ_k is bounded. To provide further insights into the convergence of solutions we will present simulation results illustrating how agents reach consensus. In addition, simulation results reveal the optimal choice of Δ for the case of fixed time intervals, i.e. $\Delta_k = \Delta$. In Chapter 5 we will study the controllability problem corresponding to the new framework of multi-agent systems. We will derive the leader-follower multi-agent system corresponding to this new framework. We will show that if the corresponding continuous-time multi-agent system is controllable then the new framework of multi-agent systems is controllable for all sufficiently small Δ_k as well. Finally we will summarize our contributions in Chapter 6 and give some ideas for further research.

2 Preliminaries

In this section we will provide some preliminaries and basic material needed in the sequel. In particular, we will discuss some basic notions of graphs and matrices. Furthermore, we will describe the continuous-time diffusively coupled multi-agent system where the multi-agent system introduced in this thesis is based on.

2.1 Graphs

Graphs provide natural abstractions for how information is shared between agents in a network. In this section we will discuss some basic notions of graph theory.

An *undirected graph* is a pair $G = (V, E)$ with $V = \{1, 2, \dots, n\}$ the vertex set and E the edge set. The edge set E is the 2-element subset of V consisting of unordered pairs (i, j) of elements of V. The elements of V are called *vertices* and the elements of E *edges*. We only consider simple graphs, that are graphs without multiple edges or loops.

Example 1. *Consider the undirected graph G in Figure 1. The vertex set of this graph is given by the set $V = \{1, 2, \dots, 9\}$. The edge set corresponding to this graph is the set $E = \{(1, 2), (2, 3), (2, 4), (3, 4), (3, 5), (4, 5), (4, 8), (5, 6), (5, 8), (6, 7), (6, 9)\}$. This graph consists of 9 vertices and 11 edges.*

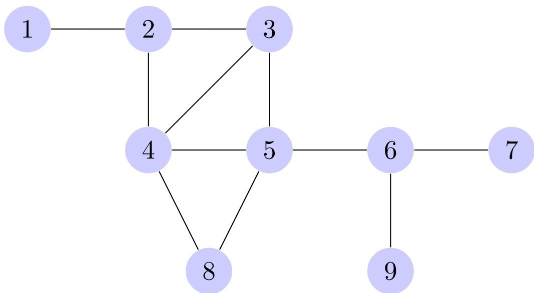

Figure 1: example of an undirected graph G.

Two vertices i and j are said to be *adjacent* if the edge between i and j is in the edge set E. The *neighborhood set* of a vertex i is the set of all vertices that are adjacent to i. This set is denoted by $N_i = \{j \in V \mid (i, j) \in E\}$. The cardinality of the neighborhood set N_i is the *degree* of vertex i. We will denote the degree of vertex i by d_i.

Example 2. *In the graph depicted in Figure 1, the vertices 1 and 2 are adjacent, but neither pair 1 and 4 nor pair 1 and 3 are adjacent. There are 4 vertices adjacent to vertex 5; that are the vertices 3, 4, 6 and 8. Hence, the neighborhood set of vertex 5 is given by the set $N_5 = \{3, 4, 6, 8\}$. The cardinality of this neighborhood set is equal to 4. Hence the degree of vertex 5 is given by $d_5 = 4$. The degrees of the other vertices can be determined in the same way. The degrees of the other vertices are: $d_1 = 1$, $d_2 = 3$, $d_3 = 3$, $d_4 = 4$, $d_6 = 3$, $d_7 = 1$, $d_8 = 2$ and $d_9 = 1$.*

A *walk* W of length k in a graph G is a sequence $\{v_{[0]}, v_{[1]}, \dots, v_{[k-1]}, v_{[k]}\}$ such that $v_{[0]}, \dots, v_{[k]} \in V$ and $e_{[i]} = \{v_{[i]}, v_{[i+1]}\} \in E$ for $i = 0, \dots, k-1$. The vertex $v_{[0]}$ is called the *initial vertex* and $v_{[k]}$ the *final vertex*. A walk W is a *trial* if all edges are distinct. That means that in a trial vertices can be repeated. A walk W is called a *path* if all the vertices in W are distinct.

Example 3. *The sequence* $\{1, 2, 3, 4, 2, 4\}$ *is an example of a walk in the graph in Figure 1. Here, vertex* 1 *is the initial and vertex* 4 *the final vertex. The sequence* $\{1, 2, 3, 4, 5, 3\}$ *is a trial, since all edges are distinct in this sequence, but vertex* 3 *is visited twice. An example of a path from vertex* 1 *to vertex* 8 *is the sequence* $\{1, 2, 3, 4, 5, 8\}$*. In this sequence all edges as well as all vertices are distinct. The length of this path is equal to* 5*. There are more paths from vertex* 1 *to* 8 *in this graph. For example* $\{1, 2, 4, 8\}$ *is a path of length* 3 *and* $\{1, 2, 3, 5, 8\}$ *a path of length* 4*.*

Two vertices i and j are called *connected* if there exists a path between them. A graph G is *connected* if every pair of vertices in the graph is connected.

Example 4. *The vertices* 1 *and* 9 *in the graph depicted in Figure 1 are connected as well as vertices* 3 *and* 7*. Since there exists a path between every pair of vertices in this graph, the graph* G *is called connected. If we consider the graph* G *given in Figure 2 the vertices* 1 *and* 2 *are connected, but neither pair* 1 *and* 3 *nor pair* 2 *and* 3 *are connected. Since there is no path between vertex* 3 *and the other vertices the graph in Figure 2 is disconnected.*

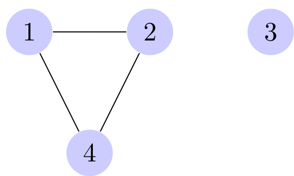

Figure 2: example of a disconnected undirected graph G.

As we have seen so far graphs can represent relations between a finite number of objects by using the terms vertices and edges. Another way of representing graphs is in terms of matrices. We will consider the Laplacian matrix L, which is defined in terms of the degree matrix D and the adjacency matrix A of a graph G.

For an undirected graph G the *degree matrix* D is the $n \times n$ diagonal matrix with the vertex-degrees on the diagonal. Hence, that means that the elements of D are:

$$d_{ij} = \begin{cases} d_i & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases}$$

The *adjacency matrix* A corresponding to an undirected graph is the symmetric $n \times n$ matrix with the following elements:

$$a_{ij} = \begin{cases} 1 & \text{if } (i, j) \in E \\ 0 & \text{otherwise.} \end{cases}$$

The *Laplacian matrix* L associated with an undirected graph is the $n \times n$ matrix:

$$L = D - A \tag{8}$$

where D is the degree matrix and A is the adjacency matrix defined above. The Laplacian matrix corresponding to an undirected graph is symmetric, i.e $L = L^T$. Denote the vector of all ones by $\mathbf{1}$. Then we know that L has an eigenvalue $\lambda = 0$ with corresponding eigenvector $\mathbf{1}$, since $L\mathbf{1} = 0$.

Example 5. *If we consider the graph G given in Figure 1 we can determine the corresponding Laplacian matrix L. First we want to determine the corresponding degree matrix D. The degree matrix is a 9×9 diagonal matrix with the vertex-degrees on the diagonal. Hence the degree matrix D corresponding to this graph G is given by:*

$$D = \mathrm{Diag}(1, 3, 3, 4, 4, 3, 1, 2, 1) \tag{9}$$

where the $n \times n$ diagonal matrix with the elements $d_{11}, d_{22}, \ldots, d_{nn}$ on the diagonal is denoted by $\mathrm{Diag}(d_{11}, d_{22}, \ldots, d_{nn})$. The next step is to determine the adjacency matrix A.

In the adjacency matrix corresponding to this graph, element (i, j) is 1 if there is an edge between i and j and 0 if there is no edge between these vertices. That means for example that element $(1, 2)$ and $(2, 1)$ in A are 1, since there is an edge between vertex 1 and 2, but element $(1, 3)$ and $(3, 1)$ are 0, since there is no edge between 1 and 3. Therefore the adjacency matrix A corresponding to this graph is equal to the symmetric (9×9) matrix:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}. \tag{10}$$

The corresponding Laplacian matrix L is equal to the difference between the degree matrix D (9) and the adjacency matrix A (10). Hence, the Laplacian matrix L corresponding to the graph G depicted in Figure 1 is given by the symmetric (9×9) matrix:

$$L = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 3 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 3 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 4 & -1 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 3 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \end{pmatrix}. \tag{11}$$

2.2 Matrices

We will introduce the following notations regarding matrices. The set of real numbers is denoted by $\mathbb{R}$ and the set of complex numbers is denoted by $\mathbb{C}$. The Euclidean space of dimension n is denoted by $\mathbb{R}^n$. The notation x^T denotes the *transpose* of a vector x and x^* denotes the *conjugate transpose* of a vector x. The notation $\|x\|$ denotes the 2-norm of the vector x, i.e. $(x^T x)^{1/2}$. A vector of all ones will be denoted by $\mathbf{1}$.

The set of $n \times m$ matrices with real elements is denoted by $\mathbb{R}^{n \times m}$. The $n \times n$ identity matrix is denoted by I_n. Sometimes we will simply write I if its dimension is clear from the context. The transpose of a matrix A is denoted by A^T and the inverse of A by A^{-1}. Let $A \in \mathbb{R}^{n \times m}$ be a matrix. We will write a_{ij} for the (i, j)th element of A.

Let A be a square matrix. As usual we say that it is *symmetric* if $A = A^T$. The matrix A is said to be *positive definite* if $x^T Ax > 0$ for all nonzero vectors x. We will often write $A > 0$ by meaning that A is positive definite. The matrix A is said to be *positive semidefinite* if $x^T Ax \geq 0$ for all vectors x. We will often write $A \geq 0$ by meaning that A is positive semidefinite. In the same way we can say that a matrix A is *negative definite* if $x^T Ax < 0$ for all nonzero vectors x. We will write $A < 0$ by meaning that A is negative definite. The matrix A is said to be *negative semidefinite* if $x^T Ax \leq 0$ for all vectors x. In case the matrix A is negative semidefinite we will write $A \leq 0$. If for two matrices A and B the following holds $x^T Ax < x^T Bx$ for all nonzero vectors x, we will denote this by $A < B$.

The column vector v satisfying $Av = \lambda v$ is called the *right-eigenvector* of A and the row vector u satisfying $uA = \lambda u$ is the *left-eigenvector* of A. The set of eigenvalues of a matrix A is called the *spectrum* of A. We will denote the spectrum of the matrix A by $\lambda(A)$. We will use the *Gershgorin disc theorem* to bound the spectrum of a square matrix in the sequel. To formulate this theorem, we first define what Gershgorin disc means.

Definition 1. *(Gershgorin disc) Let A be a complex $n \times n$ matrix. A Gershgorin disc is a closed disc centered at a_{ii} with radius $R_i = \sum_{j \neq i} |a_{ij}|$. A Gershgorin disc is denoted by $D(a_{ii}, R_i)$.*

Theorem 1. *Let A be a complex $n \times n$ matrix. Every eigenvalue of A lies in at least one the Gershgorin discs $D(a_{ii}, R_i)$.*

2.3 Multi-agent systems

In this section we will first introduce a continuous-time diffusively coupled multi-agent system associated with a graph G. We will need this multi-agent system in the sequel.

Let $G = (V, E)$ be an undirected graph with vertex set V and edge set E. Consider a diffusively coupled multi-agent system consisting of n integrator agents which are labeled by the vertex set $V = \{1, 2, \ldots, n\}$. To each agent of this diffusively coupled multi-agent system we associate the simple dynamical system given by the following equation:

$$\dot{x}_i = z_i \tag{12}$$

where $i = 1, 2, \ldots, n$, the vector x_i denotes the state of agent i and z_i the diffusively coupling term for agent i. This diffusively coupling term corresponding to this multi-agent system is based on the neighboring relations of the agents in the following way:

$$z_i = \sum_{j \in N_i} a_{ij}(x_j - x_i) \tag{13}$$

where $i = 1, 2, \ldots, n$, N_i is the neighborhood set of agent i and a_{ij} is the (i, j)th element of the adjacency matrix A corresponding to the graph G.

Example 6. *Consider a diffusively coupled multi-agent system associated with the graph G given in Figure 1. Since this graph has 9 vertices the corresponding multi-agent system consists of 9 agents. To each of these agents a linear dynamical system as given in (12) is associated. If we consider for example agent 2, the linear dynamical system associated with this agent is given by the following equation:*

$$\dot{x}_2 = z_2$$

where z_2 is the diffusively coupling term of agent 2. We can determine this diffusively coupling term z_2. Therefore we first have to know the neighborhood set of agent 2. Since the neighbors of agent 2 are agent 1, 3 and 4, this neighborhood set is given by $N_2 = \{1, 3, 4\}$. We can use equation (13) *to specify the diffusively coupling term z_2:*

$$z_2 = (x_1 - x_2) + (x_3 - x_2) + (x_4 - x_2). \tag{14}$$

Hence, the linear dynamical system associated with agent 2 is equal to the equation:

$$\dot{x}_2 = z_2 = x_1 - 3x_2 + x_3 + x_4. \tag{15}$$

In the same way we can determine the neighbors of the other agents, the diffusively coupling terms associated to these agents and the corresponding dynamical systems. If we have the dynamical systems of all agents, we can rewrite this set of equations as:

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

where $x = (x_1, x_2, \ldots, x_9)^T$ and L is the Laplacian matrix given in equation (11).

We can generalize the idea used in the example for rewriting the set of dynamical systems. In general, the multi-agent system consisting of n integrator agents with dynamical systems associated to these agents as given in equation (12) together with the diffusively coupling term given in (13) can be written in compact form as:

$$\dot{x} = -Lx \tag{16}$$

where $x = (x_1, x_2, \ldots, x_n)^T$ and L is the Laplacian matrix corresponding to the graph G.

Besides this diffusively coupled multi-agent system, we will also consider a leader-follower diffusively coupled multi-agent system associated with a graph G. We will use this multi-agent system in Chapter 5.

Let again $G = (V, E)$ be an undirected graph representing a multi-agent system consisting of n agents labeled by the set $V = \{1, 2, \ldots, n\}$. We will assign the role of leaders and followers to the agents. We denote the set of leaders by $V_L = \{v_1, v_2, \ldots, v_m\}$, where m is a positive integer with $m \leq n$. The set of followers we denote by $V_F = V \setminus V_L$. To each follower $i \in V_F$, we associate the dynamical system given by the equation:

$$\dot{x}_i = z_i. \tag{17}$$

where $i = 1, 2, \ldots, n$, the vector x_i denotes the state of agent i and z_i is the diffusively coupling term for agent i given in equation (13). To each leader i in the leader set V_L with $i = v_\ell$ we associate the dynamical system given by the following:

$$\dot{x}_i = z_i + u_\ell \tag{18}$$

where u_ℓ is the external input applied to agent $i = v_\ell$.

Example 7. *Suppose we associate a leader-follower diffusively coupled multi-agent system to our graph in Figure 1. Suppose that we assign the role of leaders to agent 2 and 6. We can determine the dynamical systems corresponding to these agents. Therefore we have to know the diffusively coupling terms of these agents. From the previous example we already know the diffusively coupling term of agent 2. This term is given in equation* (14)*. To determine the diffusively coupling term corresponding to agent 6 we first have to know its neighborhood set. The neighbors of agent 6 are agent 5, 7 and 9. Hence,* *the neighborhood set of agent* 6 *is* $N_6 = \{5, 7, 9\}$. *Therefore the diffusively coupling term associated with agent* 6 *is given by the following equation:*

$$z_6 = (x_5 - x_6) + (x_7 - x_6) + (x_9 - x_6).$$

Now we can determine the dynamics of both leaders using equation (18). *The dynamical system associated with agent* 2 *is given by the following equation:*

$$\dot{x}_2 = z_2 + u_1 = x_1 - 3x_2 + x_3 + x_4 + u_1$$

and the dynamics corresponding to the other leader which is agent 6 *is:*

$$\dot{x}_6 = z_6 + u_2 = x_5 - 3x_6 + x_7 + x_9 + u_2.$$

We can also determine the dynamical systems associated with the followers of the multi-agent system corresponding to our graph. Agent 4 *is for example one of the followers. The neighborhood set of agent* 4 *is given by* $N_4 = \{2, 3, 5, 8\}$. *Therefore the diffusively coupling term associated with agent* 4 *is equal to the following:*

$$z_4 = (x_2 - x_4) + (x_3 - x_4) + (x_5 - x_4) + (x_8 - x_4).$$

Hence, the dynamical system corresponding to agent 4 *is given by:*

$$\dot{x}_4 = z_4 = x_2 + x_3 - 4x_4 + x_5 + x_8.$$

In the same way we can determine the dynamics associated with the other 6 *followers.*

3 Communication in discrete time

In this thesis we will introduce a new framework for multi-agent systems. This new framework is based on the continuous-time diffusively coupled multi-agent system (16) in the previous chapter. The drawback of this multi-agent system is the assumption that agents communicate continuously with each other about their states. This assumption is not very reasonable in practice.

Consider for example the same situation as in the introduction where a group of sensors are to measure the temperature of a large greenhouse. In this situation the temperature measured by each sensor will vary according to its location but the group of sensors has to agree about a single value representing the temperature of the greenhouse. Each sensor exchanges information about its local measurements with other sensors. But one can imagine that it is impossible for a sensor to transmit and receive information from its neighboring sensors continuously. In fact communicating continuously is impossible in practice.

Hence, a more reasonable assumption would be that the sensors in our example will exchange information at certain (fixed) points in time. This assumption will lead to a discrete-time system. As mentioned in the introduction already several discrete-time multi-agent systems were introduced, but the drawback of these systems is the structure. Due to the structure of this systems it is for example hard to proof that the system reaches consensus. In this chapter we will come up with a discrete-time system based on the diffusively coupled multi-agent system given in (16).

In this new framework we will split the Laplacian matrix L in the degree matrix D and the adjacency matrix A. The information given in the degree matrix D is updated continuously, while the information given in the adjacency matrix A is only updated at certain points in time. The new framework of multi-agent systems is given by:

$$\dot{x}(t) = -Dx(t) + Ax(t_k) \quad \text{for} \quad t_k \leq t < t_{k+1} \tag{19}$$

where $x = (x_1, x_2, \ldots, x_n)^T$ is the state vector of the n agents, D is the degree matrix and A is the adjacency matrix. Furthermore, in the sequel we will use the notation $\Delta_k = t_{k+1} - t_k$ for the difference between point t_{k+1} and t_k in time.

For this system (19) we can determine the solution on every interval $t_k \leq t < t_{k+1}$. The solution on such an interval $t_k \leq t < t_{k+1}$ is given by the following equation:

$$x(t) = e^{-D(t-t_k)}x(t_k) + \int_{t_k}^{t} e^{-D(t-s)}Ax(t_k)\,\mathrm{d}s.$$

From this solution we can determine the solution at time t_{k+1} in terms of the solution at time t_k. If we use the notation $\Delta_k = t_{k+1} - t_k$, the solution at time t_{k+1} is:

$$x(t_{k+1}) = e^{-D\Delta_k}x(t_k) + \int_{t_k}^{t_{k+1}} e^{-D(t_{k+1}-s)}Ax(t_k)\,\mathrm{d}s.$$

If we rewrite this solution, using the fact that D is a diagonal matrix and $L = D - A$:

$$x(t_{k+1}) = \left[(e^{-D\Delta_k} - I)D^{-1}L + I\right] x(t_k).$$

In the sequel we denote this solution of the new framework of multi-agent systems by:

$$x(k+1) = M_{\Delta_k}x(k) \tag{20}$$

where the matrix M_{Δ_k} is given by:

$$M_{\Delta_k} = E_{\Delta_k} L + I \tag{21}$$

and the matrix E_{Δ_k} by:

$$E_{\Delta_k} = (e^{-D\Delta_k} - I)D^{-1}. \tag{22}$$

Note that the matrix E_{Δ_k} is a $n \times n$ diagonal matrix with the diagonal elements $\frac{1}{d_i}\big(e^{-d_i\Delta_k} - 1\big)$. Therefore we can denote this matrix also in the following way:

$$E_{\Delta_k} = \mathrm{Diag}\left(\frac{(e^{-d_1\Delta_k} - 1)}{d_1}, \dots, \frac{(e^{-d_n\Delta_k} - 1)}{d_n}\right).$$

where the $n \times n$ diagonal matrix with the elements $d_{11}, d_{22}, \dots, d_{nn}$ on the diagonal is denoted by $\mathrm{Diag}(d_{11}, d_{22}, \dots, d_{nn})$. Furthermore, note that the matrix M_{Δ_k} is a row-stochastic matrix; that is a matrix with zero row sums.

Example 8. *Consider the communication graph G given in Figure 3. The multi-agent system corresponding to this graph consists of 4 agents.*

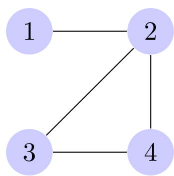

Figure 3: example of a communication graph of a multi-agent system.

The solution of the new framework of multi-agent systems is:

$$x(k+1) = M_{\Delta_k} x(k)$$

where $x = (x_1, x_2, \dots, x_4)$ and the matrix M_{Δ_k} can be specified by the following:

$$M_{\Delta_k} = \begin{pmatrix} e^{-\Delta_k} & 1 - e^{-\Delta_k} & 0 & 0 \\ \frac{1}{3}(1 - e^{-3\Delta_k}) & e^{-3\Delta_k} & \frac{1}{3}(1 - e^{-3\Delta_k}) & \frac{1}{3}(1 - e^{-3\Delta_k}) \\ 0 & \frac{1}{2}(1 - e^{-2\Delta_k}) & e^{-2\Delta_k} & \frac{1}{2}(1 - e^{-2\Delta_k}) \\ 0 & \frac{1}{2}(1 - e^{-2\Delta_k}) & \frac{1}{2}(1 - e^{-2\Delta_k}) & e^{-2\Delta_k} \end{pmatrix}.$$

Note that indeed all row sums of this matrix are equal to zero, but the sums of the columns are not equal to zero. Hence, the matrix M_{Δ_k} is not a doubly stochastic matrix, since such a matrix has zero row and column sums.

4 Consensus

In many applications involving multi-agent systems, groups of agents need to agree upon certain quantities of interest. For example in the situation mentioned in the introduction, the sensors had to agree about the temperature of the greenhouse. When all agents agree about a state value, only through interaction with their neighbors the multi-agent system achieves consensus. The problem of consensus is a well-known problem in the context of multi-agent systems. A lot of research has be done into this problem.

The consensus problem was also studied for the diffusively coupled multi-agent system $\dot{x} = -Lx$, for example in [5], [7], [9], [15] and [22]. For this multi-agent system it is known that consensus is reached if the corresponding undirected graph G is connected. In this chapter we will show a similar result for the new framework of multi-agent systems given in (19).

Recall that the solution of this new framework of multi-agent systems was given by (20).

4.1 Main result

To look at the problem of consensus for this new framework of multi-agent systems we first have to define what we mean by consensus.

Definition 2. *Let $x_i(k)$ be the state of agent i at time t_k. The multi-agent system given in* (19) *reaches consensus with respect to the graph G if for any initial state $x(0)$:*

$$\lim_{k\to\infty} x_i(k) - x_j(k) = 0 \ \textit{ for all } \ i, j \in V.$$

In other words, the state values of all the agents will go to the same value for every initial state as time goes to infinity.

For the multi-agent system given by (19) we can now state the following theorem regarding consensus of this system.

Theorem 2. *Consider the multi-agent system* (19)*. Let G be an undirected graph, and assume that G is connected. Let Δ_k be bounded, that means there exists a number K such that $\Delta_k \leq K$ for any k. Then consensus is reached for this multi-agent system* (19)*.*

Proof. To prove this theorem we will use a Lyapunov-like argument. Consider the positive semidefinite Lyapunov-like function $V(x(k)) = x(k)^T Lx(k)$. This function is non-increasing as can be shown by looking at the differences $V(x(k+1)) - V(x(k))$. Therefore we will first write these differences as given in the following equation:

$$V(x(k+1))-V(x(k)) = x(k+1)^T Lx(k+1)-x(k)^T Lx(k) = x(k)^T (M_{\Delta_k}^T LM_{\Delta_k} -L)x(k).$$

Then we write these differences using the definition of the matrix M_{Δ_k} given in (21):

$$V(x(k+1)) - V(x(k)) = x(k)^T (LE_{\Delta_k} LE_{\Delta_k} L + 2LE_{\Delta_k} L)x(k). \tag{23}$$

To show that these differences are non-positive, consider the eigenvalues of the matrix $L + 2E_{\Delta_k}^{-1}$. We know that the eigenvalues of this matrix are real, since this matrix is symmetric. We can now use *Gershgorin disc theorem* (1) to bound the spectrum of this matrix. The Gershgorin discs are given by the closed discs $D(m_{ii}, R_i)$, with:

$$m_{ii} = \frac{d_i(e^{-d_i\Delta_k}+1)}{e^{-d_i\Delta_k}-1} \text{ and } R_i = \sum_{j\neq i} |m_{ij}| = d_i$$

where $i = 1, \ldots, n$. Since we know from *Gershgorin disc theorem* (1) that all the eigenvalues of $L + 2E_{\Delta_k}^{-1}$ lie in at least one of the Gershgorin discs, we can come up with an upper bound μ_k for the eigenvalues λ_j. The upper bound for all eigenvalues λ_j is:

$$\mu_k = \frac{2\bar{d}e^{-\bar{d}\Delta_k}}{e^{-\bar{d}\Delta_k}-1} < 0.$$

where $\bar{d} = \max\{d_1, d_2, \ldots, d_n\}$. That means $\bar{d}$ denotes the maximum degree in the corresponding graph G. Note that the upperbound is negative for all k. Hence, we know that all eigenvalues of $L + 2E_{\Delta_k}^{-1}$ are negative.

At this moment we have the following relation $L + 2E_{\Delta_k}^{-1} < \mu_k I$. Using the fact that the matrix E_{Δ_k} is a diagonal matrix this relation implies $E_{\Delta_k} L E_{\Delta_k} + 2E_{\Delta_k} \leq \mu_k E_{\Delta_k}^2$. If we now use the fact that the matrix L is a symmetric matrix this relation implies:

$$LE_{\Delta_k}LE_{\Delta_k}L + 2LE_{\Delta_k}L \leq \mu_k LE_{\Delta_k}^2 L. \tag{24}$$

Here, the matrix $E_{\Delta_k}^2$ is a diagonal matrix with diagonal elements $\frac{1}{d^2}(e^{-d_i\Delta_k}-1)^2$. Hence, $E_{\Delta_k}^2 < I$ which implies $\mu_k E_{\Delta_k}^2 < \mu_k I$. Using again that L is symmetric:

$$\mu_k LE_{\Delta_k}^2 L \leq \mu_k L^2. \tag{25}$$

Now we can use (23), (24) and (25) to prove that the differences are non-positive:

$$V(x(k+1)) - V(x(k)) \leq \mu_k x(k)^T L^2 x(k) \leq 0.$$

This relation can be rewritten using the fact that $V(x(k)) = x(k)^T Lx(k)$:

$$0 \leq -\mu_k x(k)^T L^2 x(k) \leq x(k)^T Lx(k) - x(k+1)^T Lx(k+1). \tag{26}$$

Since we know that the function $V(x(k))$ is non-increasing, we know that this function has a limit, i.e. $\lim_{k\to\infty} x(k)^T Lx(k) = \hat{x}$. If k goes to infinity it follows from (26) that $\lim_{k\to\infty} -\mu_k x(k)^T L^2 x(k) = 0$. Since we assumed that Δ_k is bounded, we know that $\lim_{k\to\infty} \mu_k < 0$. Therefore, $\lim_{k\to\infty} \|Lx(k)\|^2 = 0$ and hence we can conclude $\lim_{k\to\infty} Lx(k) = 0$.

If the undirected graph G is connected then $x = \mathbf{1}$ is the only eigenvector for which $Lx = 0$. Since $\lim_{k\to\infty} Lx(k) = 0$ if $k \to \infty$, the solution $x(k+1) = M_{\Delta_k} x(k)$ will converge to a multiple of the eigenvector $x = \mathbf{1}$. Hence the multi-agent system given in (19) reaches consensus if the undirected graph G is connected and Δ_k is bounded.

□

4.2 Simulation

Consider the new framework of multi-agent systems (19) corresponding to the communication graph given in Figure 1. We will investigate how fast consensus is reached for this new framework. We will examine what the influence is of the choice of Δ_k on the speed of reaching consensus. Furthermore, we will investigate the influence of the initial positions $x(0)$ of the agents on the rate of convergence. We will study the influence of the number of different initial positions as well as the choice of the agent which initial position differs from the rest. To examine these factors, we will use simulation in

MATLAB. We will show the results in plots. In these plots, we will plot the function $V(x(k)) = x(k)^T Lx(k)$, where L is the Laplacian matrix given in (11), against k. This function $V(x(k)) = x(k)^T Lx(k)$ will show us for each k how much the positions of the agents differ at that moment. Hence, if $V(x(k)) = 0$ we know that consensus is reached.

First we investigate the influence of the choice of Δ_k on the rate of convergence. We only consider fixed values of Δ_k, i.e. $\Delta_k = \Delta$. We assume that the initial positions of the agents are given by $x(0) = (1, 2, 0, 3, 1.1, 2.5, 10, 0.5, 20)^T$. The results of this simulation are given in Figure 4.

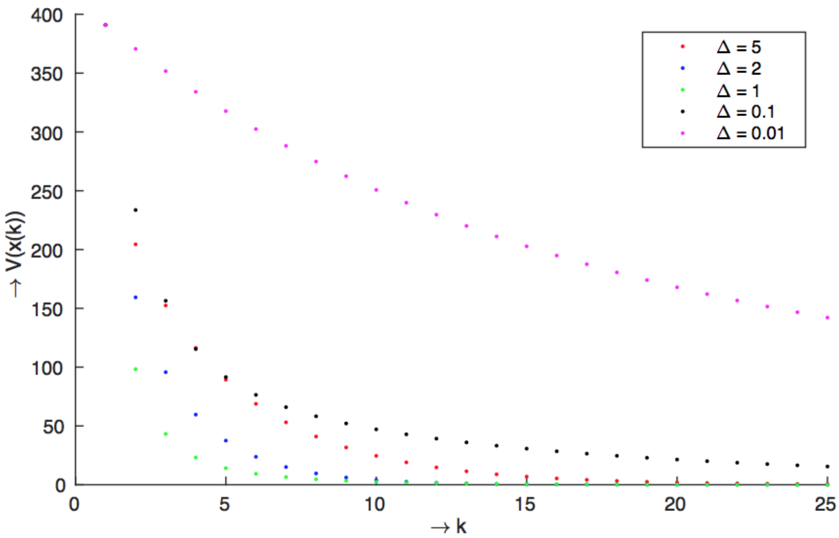

Figure 4: Different values of Δ.

Note that in Figure 4, $V(x(0))$ is the same for all values of Δ, hence there is only one dot at $k = 0$ in this figure. In Figure 4 we observe that for every value of Δ the convergence goes faster in the begin than at the end. If we consider $\Delta = 5, \Delta = 2$ and $\Delta = 1$, we see that for larger values of Δ consensus is faster reached. But if we look at $\Delta = 0.1$ and $\Delta = 0.01$ this is not the case. Both choices of Δ lead to a slower convergence. Furthermore, $\Delta = 0.01$ converges the slowest among all the values of Δ given in this figure.

Because we want to get more insight into this observation we plot in Figure 5 the function $f(\Delta)$ against Δ, where $f(\Delta) = \|V_{\Delta_1}(x(k)), V_{\Delta_2}(x(k)), \ldots, V_{\Delta_N}(x(k))\|$. We observe that there is a best choice of Δ for this example which depends on the initial position. For the initial position $x(0) = (1, 2, 0, 3, 1.1, 2.5, 10, 0.5, 20)^T$ the choice of $\Delta = 0.64$ gives the fastest convergence as we can see in Figure 5.

We study the influence of the initial positions of the agents, the vector $x(0)$, as well. We use a fixed value of $\Delta = 1$ in this case. We first investigate what the influence is of the number of initial positions that differ. The results of this simulation are given in Figure 6. We only plot the results where the number of initial positions that differ are 1, 3, 6 and 9.

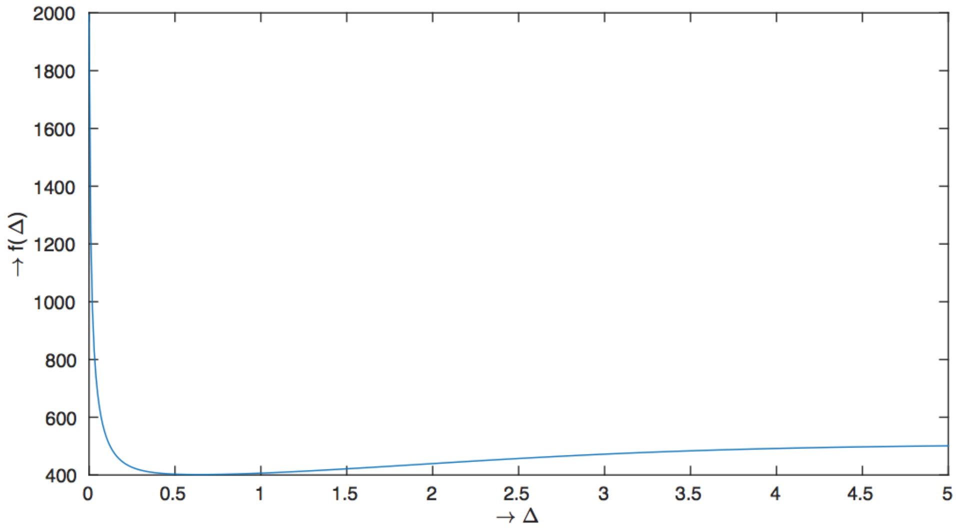

Figure 5: Optimal Δ.

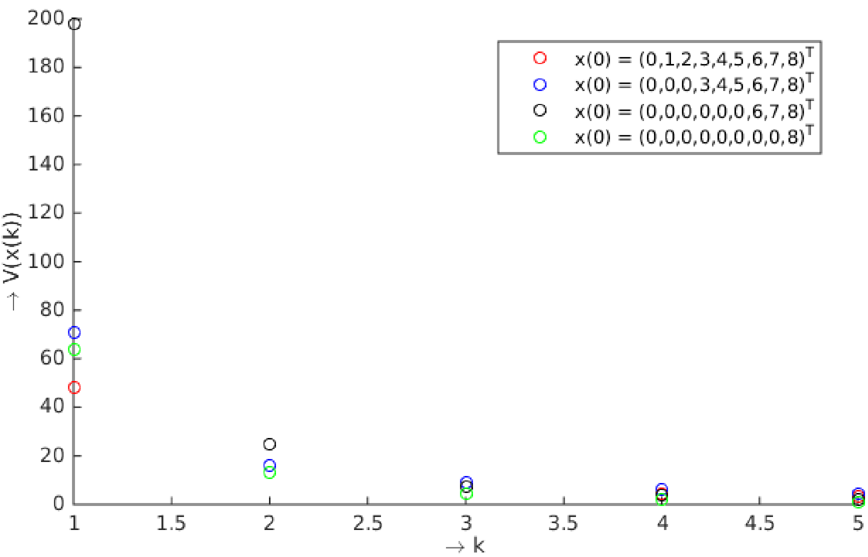

Figure 6: Different number of initial positions $x(0)$ that differ.

We observe in Figure 6 that the more the initial positions differ, the slower the convergence goes in the begin, but consensus is reached almost equally fast.

Furthermore, we examine if the choice of the agent which has a different initial position has influence on the rate of convergence. Therefore we take one of the agent's initial position equal to one and set the other equal to zero. The results of this simulation are given in Figure 7. In Figure 7 we only plot the results where agents 3, 5, 8 and 9 have a different initial position. We choose these agents, because they all have a different degree.

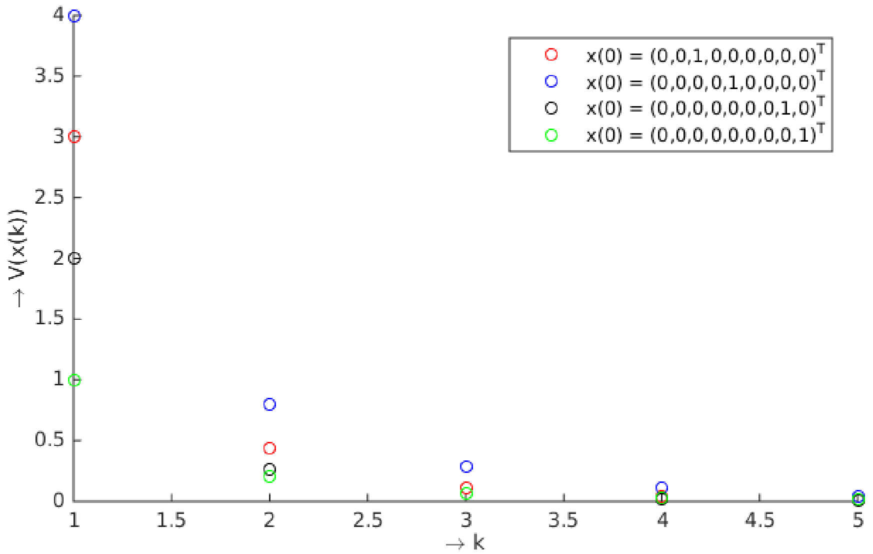

Figure 7: Different choice of the agent which initial position $x(0)$ differs from the rest.

In Figure 7 we see that if the agent with the highest degree, agent 5, differs from the rest, the convergence will go slower in the begin, but for all the initial positions $x(0)$ consensus is almost equally reached.

Finally we investigate what happens if we choose another Laplacian matrix L. That means that we have a different communication graph G. We have taken L to be:

$$L = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{pmatrix}. \tag{27}$$

As in Figure 4 we investigate the influence of the choice of Δ_k on the rate of convergence. We consider the same values of Δ and the same initial positions. The results of this simulation are given in Figure 8.

In Figure 8 we observe the same effect as in Figure 4. For large values of Δ the converges goes slower than for small values of Δ. But there is some point where this effect changes. A choice of $\Delta = 0.01$ converges the slowest among all values of Δ given in this figure. Hence, we plot the function $f(\Delta)$ against Δ in Figure 9 as well. Here we see that there is again an optimal choice of Δ which is $\Delta = 0.50$ in this case.

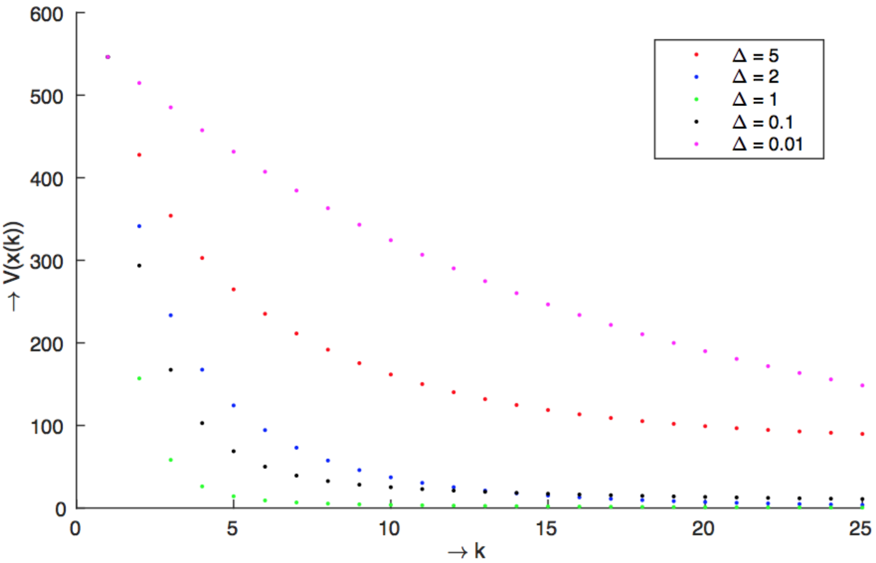

Figure 8: Different L, different values of Δ.

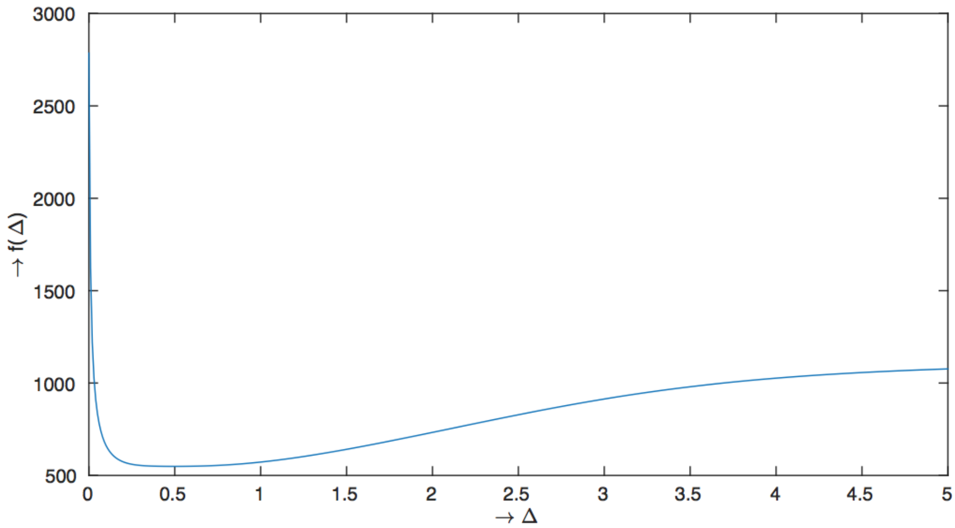

Figure 9: Different L, optimal Δ.

5 Controllability

In applications of multi-agent systems, we are interested in knowing how to influence the behavior of the overall system just by controlling a fraction of the agents in the system. For example if we consider the group of sensors measuring the temperature of the large greenhouse, it is interesting to know how we can influence the temperature of the greenhouse just by controlling some sensors in the system. Such agents that are under the forcing of external inputs are called leaders and the other agents the followers. Hence, to study whether any desired behavior can be achieved in finite time by controlling the leaders is the same as to study the controllability of the overall system where the leaders and followers are coupled together through diffusive couplings.

Such a leader-follower diffusively coupled multi-agent system was discussed in section 2.3. The diffusively coupled multi-agent system associated with a graph G given by (17) and (18), can be written in compact form as:

$$\dot{x} = -Lx + Bu \tag{28}$$

where $x = (x_1, x_2, \ldots, x_n)^T$, $u = (u_1, u_2, \ldots, u_m)^T$, L is the Laplacian matrix and $B \in \mathbb{R}^{n\times m}$ is the matrix defined by:

$$B_{i\ell} = \begin{cases} 1 & \text{if } i = v_\ell \in E \\ 0 & \text{otherwise.} \end{cases}$$

Example 9. *In the multi-agent system corresponding to the graph given in Figure 1, we choose agent* 2 *and* 6 *as leaders. That means that agent* $1, 3, 4, 5, 7, 8$ *and* 9 *are the followers. Then the leader-follower diffusively coupled multi-agent system as in* (28) *is:*

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

where $x = (x_1, x_2, \ldots, x_9)^T$*,* $u = (u_1, u_2)^T$*,* L *is the Laplacian matrix given in (11) and the matrix* B *is given by the following* 9×2 *matrix:*

$$B = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

We will introduce a new framework for multi-agent systems based on the leader-follower diffusively coupled multi-agent system given in (28) as well. We will introduce this framework because it is more convenient in practice. This new framework is:

$$\dot{x}(t) = -Dx(t) + Ax(t_k) + Bu(t_k) \quad \text{for} \quad t_k \le t < t_{k+1} \tag{29}$$

where D is the degree matrix, A is the adjacency matrix and B the matrix defined by:

$$B_{i\ell} = \begin{cases} 1 & \text{if } i = v_\ell \in E \\ 0 & \text{otherwise.} \end{cases}$$

In the same way as in the previous model we can derive the solution of the system (29):

$$x(k+1) = M_{\Delta_k} x(k) + N_{\Delta_k} u(k) \tag{30}$$

where the matrix M_{Δ_k} is given by (21) and the matrix N_{Δ_k} is defined by:

$$N_{\Delta_k} = -E_{\Delta_k} B \tag{31}$$

where the matrix E_{Δ_k} is given by (22).

5.1 Main result

To look at the problem of controllability for the system (29) we will first define the concept of controllability.

Definition 3. *The system $x((k+1)) = M_{\Delta_k} x(k) + N_{\Delta_k} u(k)$, or the pair $(M_{\Delta_k}, N_{\Delta_k})$, is said to be controllable if for any two states $x(0)$ and $x(f)$ there exists an integer $K \geq 0$ and an input sequence $u(0), u(1), \ldots, u(K)$ such that $x(K, x(0), u) = x(f)$.*

Hence, a system is controllable if an arbitrary state $x(k)$ can be reached starting from an arbitrary initial state $x(0)$ in finite time, by means of the application of a suitable admissible input sequence $u(k)$. In the following theorem we will show that if the system $\dot{x} = -Lx + Bu$ is controllable then the system (29) is controllable for all sufficiently small $\Delta_k > 0$ as well.

Theorem 3. *If the pair (L, B) is controllable then the pair $(E_{\Delta_k} L + I, E_{\Delta_k} B)$ is controllable for all sufficiently small $\Delta_k > 0$.*

Proof. Assume that the pair (L, B) is controllable. From the Hautus controllability condition we know that $\text{rank}[L - \lambda I \;\; B] = n$ for all $\lambda \in \mathbb{C}$. Suppose that the pair $(E_{\Delta_k} L + I, E_{\Delta_k} B)$ is uncontrollable for all sufficiently small $\Delta_k > 0$. That means that there exists a sequence $\{\Delta_\ell\}_{\ell=0}^{\infty}$ with $\Delta_\ell > 0$ for which $\Delta_\ell \downarrow 0$ as $\ell \to \infty$ such that $(E_{\Delta_\ell} L + I, E_{\Delta_\ell} B)$ is uncontrollable. Hence there must exist a $x^*(\ell)$ with $\|x^*(\ell)\| = 1$ such that $x^*(\ell)[E_{\Delta_\ell} L + I - \lambda_\ell I \;\; E_{\Delta_\ell} B] = 0$ for some $\lambda_\ell \in \mathbb{C}$. So there is a $x^*(\ell)$ for which $x^*(\ell)(E_{\Delta_\ell} L + I) = \lambda_\ell x^*(\ell)$ and $x^*(\ell) E_{\Delta_\ell} B = 0$.

From the above we can also state that there exists a $x^*(\ell)$ with $\|x^*(\ell)\| = 1$ such that $\lim_{\ell\to\infty} \Delta_\ell^{-1} x^*(\ell)[E_{\Delta_\ell} L + I - \lambda_\ell I \;\; E_{\Delta_\ell} B] = 0$ for some $\lambda_\ell \in \mathbb{C}$. Hence there there exists a $x^*(\ell)$ such that $\lim_{\ell\to\infty} \Delta_\ell^{-1} x^*(\ell)(E_{\Delta_\ell} L + I) = \lim_{\ell\to\infty} \Delta_\ell^{-1} \lambda_\ell x^*(\ell)$ and $\lim_{\ell\to\infty} \Delta_\ell^{-1} x^*(\ell) E_{\Delta_\ell} B = 0$.

Now we will use the fact that $\lim_{\ell\to\infty} \Delta_\ell^{-1} E_{\Delta_\ell} = -I$. With this result the previous statement can be rewritten as: there exists a $x^*(\ell)$ with $\|x^*(\ell)\| = 1$ such that $-\lim_{\ell\to\infty} x^*(\ell) L = \lim_{\ell\to\infty} \Delta_\ell^{-1}(\lambda_\ell - 1) x^*(\ell)$ and $-\lim_{\ell\to\infty} x^*(\ell) B = 0$. Since $\|x^*(\ell)\| = 1$ we know that there exists a $x^*(\ell)$ such that $\lim_{\ell\to\infty} x^*(\ell) L = \alpha x^*(\ell)$ and $\lim_{\ell\to\infty} x^*(\ell) B = 0$.

Since such a $x^*(\ell)$ exists, we have that the pair (L, B) is uncontrollable. This is in contradiction with our assumption. Therefore we know that the pair $(E_{\Delta_k} L + I, E_{\Delta_k} B)$ is controllable for all sufficiently small $\Delta_k > 0$.

□

6 Conclusions

6.1 Summary

In this thesis we introduced a new framework for multi-agent systems in Chapter 3. In this framework the communication between agents is assumed to be discrete in time rather than continuous as in the classical framework. The motivation for this new framework stems from the fact that the communication is necessarily in discrete time in practice.

For this new framework of multi-agent systems we investigated the problem of consensus in Chapter 4. The problem of consensus is a well-known problem in the context of multi-agent systems. Due to its wide variety in applications a lot of research has be done into this problem. For the continuous-time multi-agent system $\dot{x} = -Lx$ it was already shown that consensus is reached if the corresponding undirected communication graph G was connected. In this thesis we showed a similar result for the new framework of multi-agent systems.

For this new framework we showed in section 4.1 that consensus is reached if the corresponding undirected graph is connected and every time interval Δ_k is bounded. In section 4.2 we presented simulation results illustrating how agents reach consensus within the new framework. In addition, the simulation results revealed the optimal choice of Δ for the case of fixed time intervals, i.e. $\Delta_k = \Delta$.

In Chapter 5 we studied the controllability problem for the new framework of multi-agent systems. To investigate this problem for the new framework, we derived the corresponding leader-follower system. We related the controllability of the classical continuous-time multi-agent system to the controllability of the new framework of multi-agent systems. We showed that if the corresponding continuous-time multi-agent system is controllable then the new framework of multi-agent systems is controllable for all sufficiently small Δ_k as well.

6.2 Further research

In this thesis we showed that the new framework for multi-agent systems reaches consensus if the communication graph is undirected. One can try to extend this result for undirected graphs to directed ones. To proof this, the ideas used in our proof can maybe be generalized.

We only focused on the multi-agent system where the agents update the information about their states at the same points in time in this thesis. It is also interesting to study the problem of consensus when we assume that the agents do not update their information at the same moments in time.

Furthermore the problem of consensus can be studied for a system where the neighbors of each agent vary at each moment in time. The assumption that some agents have better information than others can be considered as well as time-delays in communication channels.

Also the controllability of the new framework for multi-agent system can be studied further. The necessary condition for the controllability of the system can be proven. Furthermore it is interesting to know in which way we have to choose the leaders to render this system controllable. Moreover, we want to know the minimum number of leaders that leads to the controllability of the new framework for multi-agent systems.

References

[1] V. Borkar and P.P. Varaiya. Asymptotic agreement in distributed estimation. *Automatic Control, IEEE Transactions on*, 27(3):650–655, 1982.

[2] M. Cao, S. Zhang, and M.K. Camlibel. A class of uncontrollable diffusively coupled multiagent systems with multi-chain topologies. *Automatic Control, IEEE Transactions on*, 58(2):465–469, 2013.

[3] M. Egerstedt, S. Martini, M. Cao, M.K. Camlibel, and A. Bicchi. Interacting with networks: How does structure relate to controllability in single-leader, consensus networks? *Control Systems, IEEE*, 32(4):66–73, 2012.

[4] J.A. Fax. *Optimal and Cooperative Control of Vehicle Formations*. PhD thesis, California Institute of Technology, 2001.

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

[6] R.A. Horn and C.R. Johnson. *Matrix Analysis*. Cambridge university press, 2012.

[7] A. Jadbabaie, J. Lin, and A.S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *Automatic Control, IEEE Transactions on*, 48(6):988–1001, 2003.

[8] M. Ji, A. Muhammad, and M. Egerstedt. Leader-based multi-agent coordination: Controllability and optimal control. In *American control conference*, pages 1358–1363, 2006.

[9] Z. Lin, M. Broucke, and B. Francis. Local control strategies for groups of mobile autonomous agents. *Automatic Control, IEEE Transactions on*, 49(4):622–629, 2004.

[10] Z. Lin, B. Francis, and M. Maggiore. Necessary and sufficient graphical conditions for formation control of unicycles. *Automatic Control, IEEE Transactions on*, 50(1):121–127, 2005.

[11] N.A. Lynch. *Distributed Algorithms*. Morgan Kaufmann, 1996.

[12] S. Martini, M. Egerstedt, and A. Bicchi. Controllability analysis of multi-agent systems using relaxed equitable partitions. *International Journal of Systems, Control and Communications*, 2(1):100–121, 2010.

[13] M. Mesbahi and M. Egerstedt. *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.

[14] N. Monshizadeh. *Model Reduction and Control of Complex Systems*. PhD thesis, PhD thesis, University of Groningen, 2013.

[15] L. Moreau. Stability of continuous-time distributed consensus algorithms. In *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, volume 4, pages 3998–4003. IEEE, 2004.

[16] L. Moreau. Stability of multiagent systems with time-dependent communication links. *Automatic Control, IEEE Transactions on*, 50(2):169–182, 2005.

[17] R.M. Murray and R. Olfati-Saber. Consensus protocols for networks of dynamic agents. In *Proceedings of the 2003 American Controls Conference*, 2003.

[18] R. Olfati-Saber and R.M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *Automatic Control, IEEE Transactions on*, 49(9):1520–1533, 2004.

[19] A. Rahmani, M. Ji, M. Mesbahi, and M. Egerstedt. Controllability of multi-agent systems from a graph-theoretic perspective. *SIAM Journal on Control and Optimization*, 48(1):162–186, 2009.

[20] A. Rahmani and M. Mesbahi. On the controlled agreement problem. In *American Control Conference, 2006*, pages 6–pp. IEEE, 2006.

[21] W. Ren, R.W. Beard, and E.M. Atkins. A survey of consensus problems in multi-agent coordination. In *American Control Conference, 2005. Proceedings of the 2005*, pages 1859–1864. IEEE, 2005.

[22] W. Ren, R.W. Beard, et al. Consensus seeking in multiagent systems under dynamically changing interaction topologies. *IEEE Transactions on automatic control*, 50(5):655–661, 2005.

[23] W. Ren, R.W. Beard, and T.W. McLain. Coordination variables and consensus building in multiple vehicle systems. In *Cooperative Control*, pages 171–188. Springer, 2005.

[24] H.G. Tanner. On the controllability of nearest neighbor interconnections. In *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, volume 3, pages 2467–2472. IEEE, 2004.

[25] J. N. Tsitsiklis. Problems in decentralized decision making and computation. Technical report, DTIC Document, 1984.

[26] J.N. Tsitsiklis, D.P. Bertsekas, M. Athans, et al. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE transactions on automatic control*, 31(9):803–812, 1986.

[27] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet. Novel type of phase transition in a system of self-driven particles. *Physical review letters*, 75(6):1226, 1995.