POLITECNICO DI TORINO

Master‘s degree in Mechatronic Engineering

Master‘s degree thesis

Control of heterogeneous multi-agent systems and aerospace applications

Supervisors:
Prof. Elisa Capello
Prof. Yasumasa Fujisaki

Candidate:
Rodolfo Pietrasanta

A.Y. 2020/2021

Contents

1 Introduction

1.1 Multi-agent systems

In recent times, multi-agent system are becoming increasingly important in many different engineering fields. In literature we can find different definitions of multi-agent system, they can be described [22] as a set of subsystems that are, at least partially, autonomous: each agent has its own behaviour or dynamics depending on the context. The main characteristics of Multi-Agent Systems (MAS) are [23]:

- **Sociability**: ability to share and request information from other agents.
- **Autonomy**: Each agent can take action independently.
- **Proactivity**: Each agent uses its history, sensed parameters and information of the other agents.

MAS can be classified depending on their feature[23]:

- **Leadership**: MAS can be *leaderless*, a configuration in which each agents acts to reach its own objective, or *leader-follower* in which there are one or more agents that are leaders and the other chase their behaviour. In this latter case, leader(s) can be predefined or collaboratively chose by the agents. In this work, particular attention is posed on leader-follower configurations with one leader .
- **Decision function**: This feature can be linear or non-linear. In the first case the decision function of each agent is proportional to the parameters sensed from the environment. Clearly in the non-linear case the relation is not a simple proportionality.
- **Hetrogeneity**: In *homogeneous* MAS, all the agents have the same identical characteristics, while in *heterogenous* MAS, agents are different. In this work the focus is on heterogenous agents
- **Delay**: The time delay that occurs when sharing information among the agents can be taken in consideration or not, in this thesis, time delay is neglected.
- **Topology**: topology refers to reciprocal locations and relations among agents. It can be static or dynamic, up to section 4 static topologies are considered, in section 5 some cases with switching topologies are considered.

The overall system is studied in order to find more efficient ways of scheduling, path finding, obstacle avoidance etc. In a physical context, MAS are used to control swarms of physical objects such as robots or satellites, in 1996 the first article that approached the problem of coordination among a set of robots was published [24]. In [25] we can find the definitions of the two main problems:

- Coordination among robots
- Planning trajectory

In this thesis, the main field of interest is the Aerospace, in particular considering sworms of satellites, and the main objective is the coordination, in particular the reach of consensus. Consensus is a condition in wich each agent in the MAS has a common charachteristic with the others, for example position and/or velocity. In this context an attempt to find better coordination laws is made, in order to reduce the global energy consumpion of the system and obtain better performances to reach consensus.

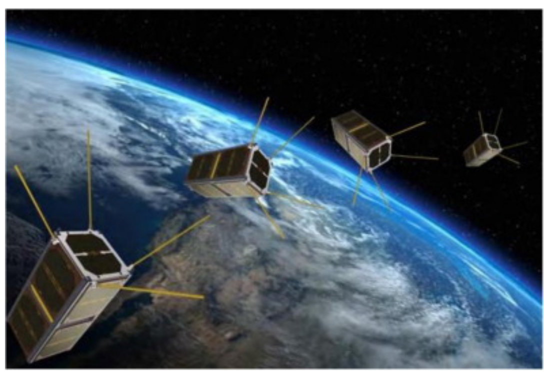


Figure 1: Concept of a group of satellites orbiting around the earth.

Satellites orbiting around the earth are used for many different purposes such as pollution control, disasters monitoring (earthquakes, forest fires...), communications, space observation and so on [14]. For all these purposes, they ususally need an high pointing accuracy (the variables of interest are considered to be angular displacement and velocity) as we can find in [1],[2], and to be optimized in terms of costs. In [26], [27] and [28] we can find some conrol laws for fisrt order dynamics MAS, considering time delays, and the preliminary concepts needed to face such problems. In particular, control laws involving Laplacians and some lemmas about parameters bound and graph properties are shown. Satellites are considered to have a rigid body described by Newton's law, and a flexible appendage, that could be, for example, a solar panel, described by a second order differential equation coupled with the rigid body equation that can be found from different sources in literature ([15],[16],[17] and [18]). In this way the model is more accurate, higher pointing accuracy is obtained and unwanted flexible parts excitments are avoided. For all these purposes, a background of graph theory and algebraic graph theory was studied starting from some reference books as [10],[11],[12] and [13], then some new methods to tune the transient of such systems using directed topologies are presented starting from the ideas found in [6], [7] and [8]. In particular, a subcase for a one-leader-follower MAS consensus under directed topologies is demonstrated using a parameter inequality already present in [3], then it is extended to the flexible dynamics case and a method to reduce the global control input without lowering performances is shown. In [4] we can find a resume of the thoeretical results for first order MAS in continous and discrete time, also with switching topologies.

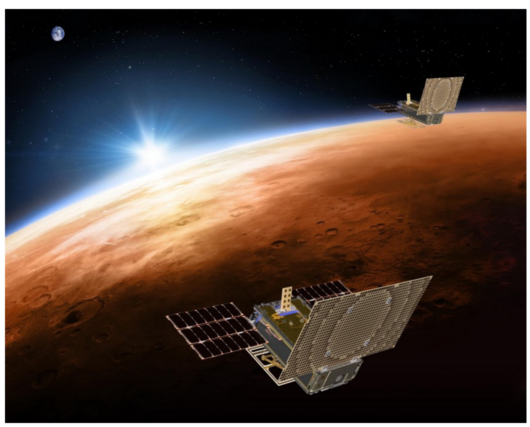


Figure 2: Concept of a CubeSat with flexible appendages.

Moreover, most relevant external disturbances found in Low Earth Orbit are modeled following [19],[20] and [21]. Since often satellites are not equipped with gyroscopes, and, otherwise, to take in account possible failures, the absence of a velocity measure is also considered by translating the MAS in discrete time and using differentiators to estimate the velocity.

1.2 Thesis outline

This work is divided in four main sections:

- Section 2: This section is dedicated to the mathematical introduction, with particular focus on graph theory and algebraic graph theory. The basic notions needed to understand succeding sections are introduced such as basic definitions and properties about graphs and related matrices. Moreover some of the "less common" matrix properties are shown with some simple proofs.

- Section 3: this section provides an overview of the problem starting from the work already present in literature. Consensus of second order heterogeneous multi-agent systems with flexible appendages and undirected communication topologies is considered. At first velocity is supposed to be measurable, so each agent is assumed to be equipped with instrumentation able to perform the measure. Then a control law able to stabilize the system also without velocity measures is presented. Some numerical simulations with different communication topologies among the agents are shown.

- Section 4: this section are present the main results of the work. Directed topologies are taken into account in order to propose methods to better tune the transient of the system and reduce the number of communication. In this section leader-follower control is considered, so all the agents have to reach the same state (consensus) of an agent called Leader. Consensus reachability over directed topologies is prooved for first and second order dynamics, also with flexible dynamics. The system is translated in discrete time to involve differentiators to estimate velocity in case it is not measurable. At last, external disturbances models are presented and numerical simulations are performed.

- Section 5: this section deals with the possible failure of one or more of the agents, the communication topology is checked to be still capable reaching consensus and it is recomputed if it‘s not. Some simulations with switching topologies are presented.

Section 6 and 7 are dedicated to conclusions and references.

2 Mathematical tools

This section introduces the main mathematical tools used in the succeeding sections.

2.1 Graph theory

A graph G consists of a finite set of Vertices $V(G)$ and a set of Edges $E(G)$. We take $V(G)$ to be $\{1,2,...,n\}$ and $E(G)$ to be $\{e_1, e_2, ..., e_m\}$. An edge $e_k = \{i, j\}$ connects the $i-th$ and $j-th$ vertices. The following figure shows a simple graph:

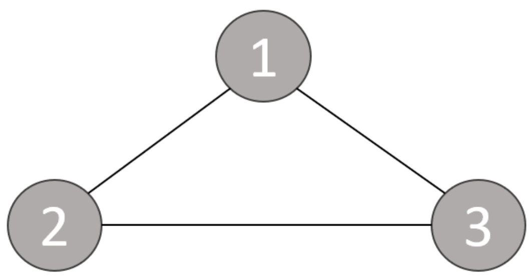
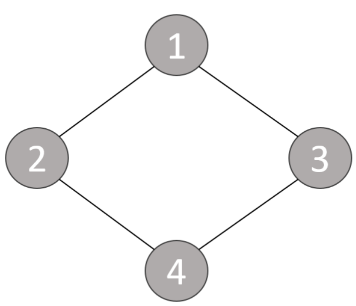


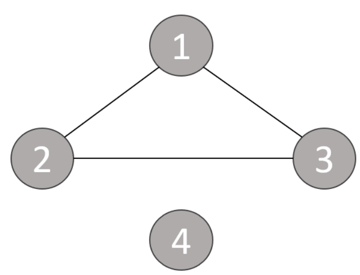
Figure 3: Simple Graph

Now some of the graphs main properties are listed.

Connected graph: a graph is said to be connected if there exist a path from any vertex to any other. For example:



(a) Connected graph



(b) Unconnected graph

The graph in figure (b) is not connected because does not exist a path from vertex 4 to the others.

Directed graph: A directed graph is a graph in wich the pairs of vertexes connected by an edge is an ordered pair, so the edge connecting {i,j} does not connect {j,i} in general. In other words, the edges have a direction:

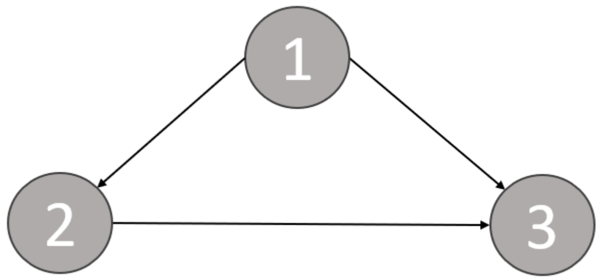


Figure 5: Directed Graph

In this example the edge connecting 2 to 1 does not connect 1 to 2. A directed graph is said to be **weakly connected** if the underlying undirected graph is connected. This is obtained replacing each directed edge with an undirected one. The graph in Figure 3 is an example of weakly connected graph.
Self loop graph: this graphs has at least one edge connecting a vertex to itself:

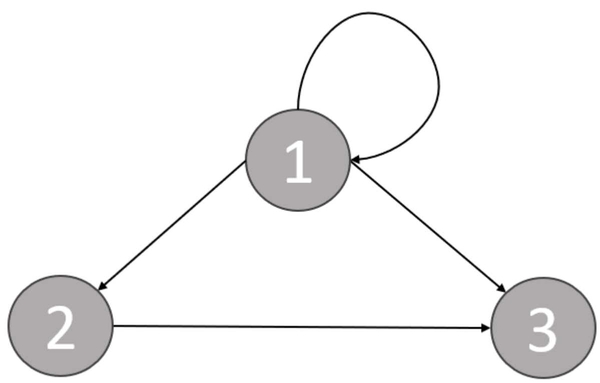


Figure 6: Self loop Graph

A graph with no self loops is called **loopless**.

A **source** is a vertex from which all the edges are pointing out. For example in the next figure vertex 1 is a source.

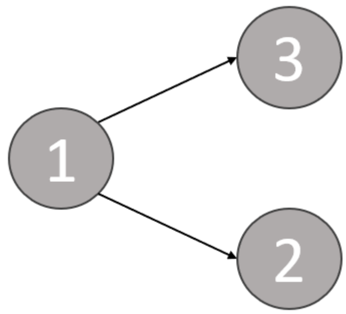


Figure 7: Graph with a source

If on the contrary a vertex has all the edges pointing in, it is called a *sink*.

Weighted graph: a weighted graph is a directed graph in whch a positive real number a_e is associated to each edge: $G = (V(G), E(G), \{a_e\}_{e \in E(G)})$ Note that if the edge connecting vertex i to vertex j is not present, it can be interpreted as a 0 weigth associated to that edge. An *unweighted graph* has $a_i = 1 \quad \forall i$. The following picture is an example of weighted graph:

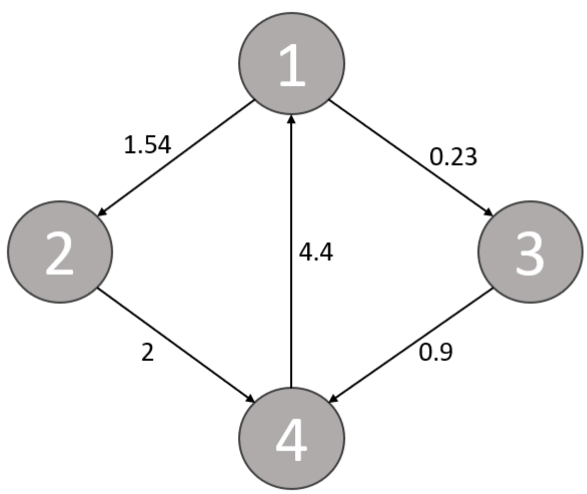


Figure 8: Weighted Graph

2.2 Algebraic graph theory

Let's see now how to manipulate graphs and their properties using matrices. The first important definition is the *Adjacency Matrix A*. In general it is an $n \times n$ matrix (n number of vertex) defined as follows:

$$A = \begin{cases} a_{ij} & if \quad \{i, j\} \in E(G) \\ 0 & otherwise \end{cases}$$

So it is a matrix where all the entries are equal to the weight of the edge connecting vertex i to vertex j. Clearly if an edge does not exist its correspondent weight is 0. For example:

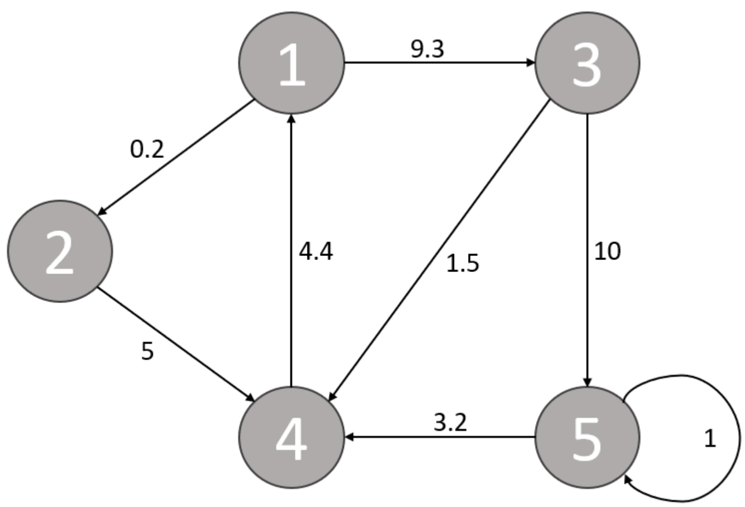


Figure 9: Weighted Directed Graph

The graph in Figure 7 is associated to the following adjacency matrix:

$$A = \begin{bmatrix} 0 & 0.2 & 9.3 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 1.5 & 10 \\ 4.4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3.2 & 1 \end{bmatrix}$$

Notice that the transpose of A is the matrix that has as entries the weights of the edges connecting vertex j to vertex i, so an interpretation is that the rows of A represent the information that the correspondent vertex is giving to the others, and the colums (so the rows of A^T) the information tha the correspondent vertex is receiving from the others. Some properties can be extrapolated from this definition:

- **Loopless graph**: the adjacency matrix of a loopless graph has all the diagonal entries equal to 0 since there is no edge from a vertex to itself.

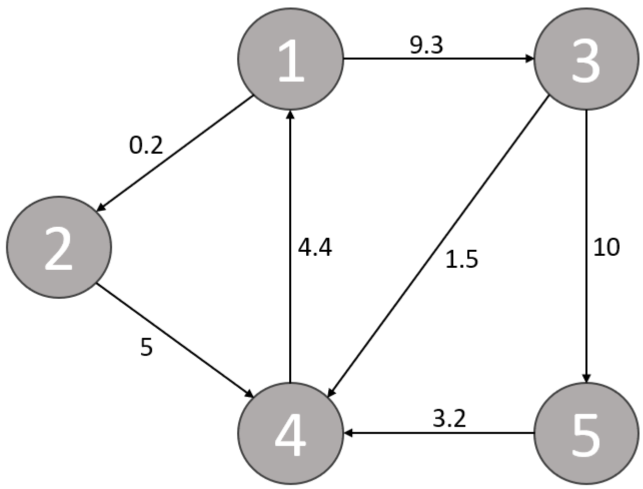


Figure 10: Weighted Directed Graph

$$A = \begin{bmatrix} 0 & 0.2 & 9.3 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 1.5 & 10 \\ 4.4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3.2 & 0 \end{bmatrix}$$

- **Unweighted graph**: all entries of an unweighted graph are either 1 or 0.

- **Undirected graph**: the adjacency matrix of an undirected graph is symmetric since the edge connecting i to j also connects j to i with the same weight.

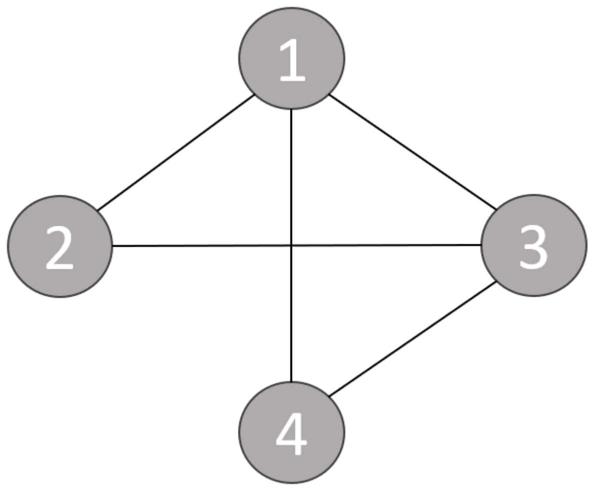


Figure 11: Undirected Graph

$$A = A^T = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

- **Unconnected graph**: an unconnected (undirected) graph has the $k-th$ row and column all set to 0, where the $k-th$ vertexes are the not connected once. This is because they are not giving or receiving information.

- **Graph with a source**(or a sink): a graph with a source has the correspondent column set to 0 since that vertex is only giving information to the others. The dual of the source is the sink, in that case the correspondent row is 0 since it only receives information from the others.

For example the graph in Figure 5 has a source and two sinks, so the adjacency matrix is:

$$A = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

- **Complete graph**: a graph is said to be complete if all the entries of the adjacency matrix are not 0. This means that exist an edge from all vertex to any other, themselves included.

Another important definition is the ***Degree matrix***. In general it can be defined in two ways:

$$D_{out} = diag(A\mathbf{1}_n) = \textit{row-sum of } A$$

$$D_{in} = diag(A^T\mathbf{1}_n) = \textit{column-sum of } A$$

where 1_n is the column vector of all 1. The first is the *out-degree* matrix, that isa a diagonal matrix in which any entry is the sum of the weights of the edges pointing out the correspondent vertex. The second is the same but each entry is the sum of the weights of the edges pointing in the correspondent vertex.Clearly if the graph is not weighted the sum of the weights corresponds to the number of edges. The graph in Figure 8 has the following degree matrices:

$$D_{out} = \begin{bmatrix} 9.5 & 0 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 11.5 & 0 & 0 \\ 0 & 0 & 0 & 4.4 & 0 \\ 0 & 0 & 0 & 0 & 3.2 \end{bmatrix}$$

$$D_{in} = \begin{bmatrix} 4.4 & 0 & 0 & 0 & 0 \\ 0 & 0.2 & 0 & 0 & 0 \\ 0 & 0 & 9.3 & 0 & 0 \\ 0 & 0 & 0 & 9.7 & 0 \\ 0 & 0 & 0 & 0 & 10 \end{bmatrix}$$

Notice that if the graph is undirected A is symmetric, so $D_{out} = D_{in}$ since $A = A^T$.

Now we can define the ***Laplacian Matrix***. In general the Laplacian of a graph is a matrix defined as:

$$L = D_{out} - A$$

So the off-diagonal elements are the opposite of the weights and if the graph is loopless the diagonal ones are the absolute value of the sum of the off diagonal on the same row:

$$L = [l_{ij}] = \begin{cases} -a_{ij} & \textit{for } i \neq j \\ \sum_{h=1, h\neq i}^{n} a_{ih} & \textit{for } i = j \end{cases}$$

For example:

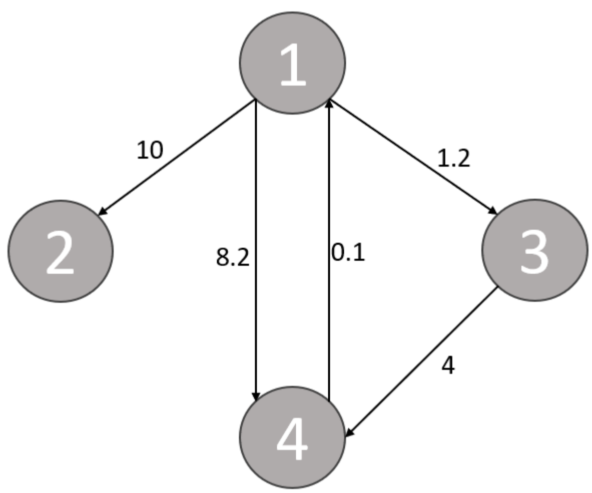


Figure 12: Directed loopless graph

$$L = \begin{bmatrix} 19.4 & -10 & -1.2 & -8.2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & -4 \\ -0.1 & 0 & 0 & 0.1 \end{bmatrix}$$

It follows from the properties of the Laplacian that:

- $L\mathbf{1}_n = 0$: the row-sum is always 0.
- If the graph is undirected L is symmetric, so $L = L^T$ and $\mathbf{1}^T L = 0$. This latter identity can be obtained computing the transpose of $L\mathbf{1}_n = 0$
- $[Lx]_i = d_{out}(i)(x_i - \textit{weighted-evrage}(x_j)$ for all j neighbour of i if the graph is loopless.
- The diagonal elements are always non-negative, while the off diagonal are always non-positive.
- All the eigenvalues $\lambda_i(L)$ have non-negative real part.

This latter property can be easily prooved using the first Gershgorin theorem: The theorem states that, given a square $n \times n$ matrix $A = [a_{ij}]$ and calling

$$Ki^r = \{z \in \mathbb{C} : |z - a_{ii}| <= R_i^r\}$$
$$Ki^c = \{z \in \mathbb{C} : |z - a_{ii}| <= R_i^c\}$$

where a_{ii} is a diagonal element of A and

$$R_i^r = \sum_{j,j \neq i}^{n} |a_{ij}| \quad \textit{row-sum}$$

$$R_i^c = \sum_{j,j \neq i}^{n} |a_{ji}| \quad \textit{column-sum}$$

all the eigenvalues of A are inside the intersection between the union of all the K_i^r and the union of all the K_i^c.

So if

$$K^r = \bigcup_i K_i^r \quad \textit{and} \quad K^c = \bigcup_i K_i^c$$
$$\lambda_i(A) \in K^r \cap K^c \quad \forall i$$

In our case $A = L$ so by definition

$$R_i^r = \sum_{j,j\neq i}^{n} |a_{ij}| = a_{ii} \quad \forall i$$

So this means that all the K_i^r are circles in the Gauss plane centered in $(R_i^r, 0)$ with radius R_i^r:

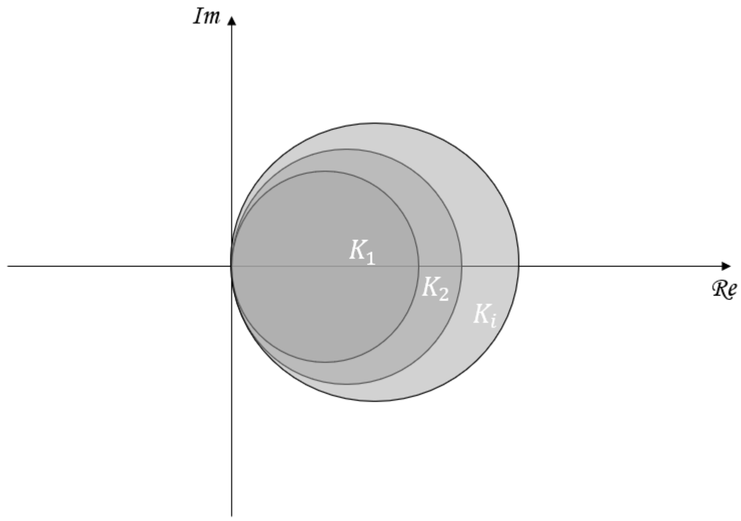


Figure 13: Row-Ghershgorin circles

So K^r, which is the union of alla the circles, corresponds to the biggest one and can be written as

$$K^r = \{z \in \mathbb{C} : |z - \frac{\|L\|_\infty}{2}| \leq \frac{\|L\|_\infty}{2}\}$$

Since the intersection of K^r with any other set is either empty or inside K^r itself, we can conclude that the real part of the eigenvalues of a Laplacian matrix are all non negative.

We can define as a **generalized Laplacian** any L such that:

$$L = \begin{cases} l_{i,j} < 0 & if \;\; i \neq j \;\; and \;\; i \;\; adjacent \;\; to \;\; j \\ l_{i,j} = 0 & if \;\; i \neq j \;\; and \;\; i \;\; is \;\; not \;\; adjacent \;\; to \;\; j \\ \textbf{any} \quad \textbf{number} & otherwise \end{cases}$$

This can be useful in some cases, for example if we consider

$$L = D_{in} - A^T$$

L is a generalized Laplacian (notice that it corresponds to the Laplacian obtained inverting the direction of the edges in the graph). In this way the quantity Lx can be seen as the difference between the information of each node and the information that it receives from the others, for example:

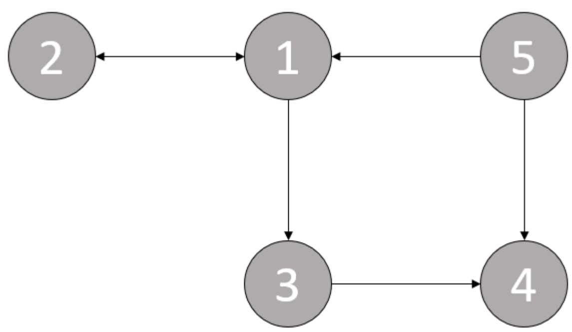


Figure 14: Graph example

$$L = \begin{bmatrix} 2 & -1 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$Lx = \begin{bmatrix} 2x_1 - x_2 - x_5 \\ x_2 - x_1 \\ x_3 - x_1 \\ 2x_4 - x_3 - x_5 \\ 0 \end{bmatrix}$$

2.3 Some properties of matrices

In this section some matrix properties that will be used in the next sections are given.
Block matrix determinant: considering a block matrix as

$$H = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

we can factorize it as:

$$H = \begin{bmatrix} I & 0 \\ CA^{-1} & I \end{bmatrix} \begin{bmatrix} A & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} I & A^{-1}B \\ 0 & I \end{bmatrix} = H_1 H_2 H_3$$

where S is the Schur's complement of H, $S = D - CA^{-1}B$. It holds that $det(H) = det(H_1)det(H_2)det(H_3)$ and since $det(H_1) = det(H_2) = 1$, $det(H) = det(H_2)$. So now we can write that:

$$det(H) = det(\begin{bmatrix} A & 0 \\ 0 & S \end{bmatrix}) = det(A)det(S) = det(A)det(D - CA^{-1}B)$$

Matrix pseudo-inverse: given any $A \in \mathbb{R}^{n,m}$,it is defined $A^\dagger$ such that:

- $AA^\dagger A = A$
- $A^\dagger AA^\dagger = A^\dagger$
- $AA^\dagger$ and $A^\dagger A$ symmetric

$A^\dagger$ is called the *Moore-Penrose* inverse of A, or *pseudo*-inverse of A. Some properties of the pseudo inverse are:

- If A is invertible $A^\dagger = A^{-1}$
- $(A^\dagger)^\dagger = A$
- $(\alpha A)^\dagger = \frac{1}{\alpha} A^\dagger$
- If λ_i is an eigenvalue of A and $\lambda_i \neq 0$, then $\frac{1}{\lambda_i}$ is an eigenvalue of $A^\dagger$
- If λ_i is an eigenvalue of A and $\lambda_i = 0$, then also the correspondent eigenvalue of $A^\dagger$ is 0.

2.4 Kronecker Product

Given any two matrices $A = [a_{i,j}] \in \mathbb{R}^{p,q}$ and $B = [b_{i,j}] \in \mathbb{R}^{n,m}$, the *Kronecker product* is defined as:

$$A \otimes B = \begin{bmatrix} a_{11}B & ... & a_{1q}B \\ ... & ... & ... \\ a_{p1}B & ... & a_{pq}B \end{bmatrix}$$

So it is always doable and the result is a matrix $\in \mathbb{R}^{pn,qm}$.
For example:

$$\begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix} \otimes \begin{bmatrix} 0 & 7 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 0 & 7 & 0 & 14 \\ 1 & -1 & 2 & -2 \\ 0 & 14 & 0 & 35 \\ 2 & -2 & 5 & -5 \end{bmatrix}$$

Some properties:

- $A \otimes (B + C) = A \otimes B + A \otimes C$
- $A \otimes B \neq B \otimes A$ in general
- $(\alpha A) \otimes B = A \otimes (\alpha B) = \alpha(A \otimes B)$

An useful concept that will be used later is the Kroneker product between a Laplacian matrix and a diagonal matrix. Let‘s consider an example:

$$L = L_1 \otimes R$$

where $L_1 = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$ and $R = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$.
So we get

$$L = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 2 & 0 & -2 & 0 & 0 \\ -1 & 0 & 2 & 0 & -1 & 0 \\ 0 & -2 & 0 & 4 & 0 & -2 \\ 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & -2 & 0 & 2 \end{bmatrix}$$

L_1 is related to the following graph:

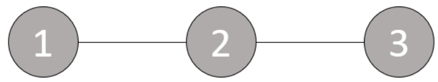


Figure 15: L_1 graph

While L is related to the following one:

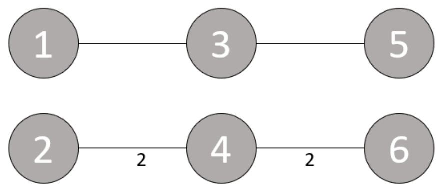


Figure 16: L graph

So basically the result is a set of disjointed graphs with all the weights scaled by the correspondent diagonal entry of R. Notice that, even if the Kroneker product is not commutative in general, the effect on the graph topology is the same inverting the product order. We just get a different nodes enumeration.

$$L_c = R \otimes L_1 = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & -2 & 0 \\ 0 & 0 & 0 & -2 & 4 & -2 \\ 0 & 0 & 0 & 0 & -2 & 2 \end{bmatrix}$$

That corresponds to the following graph:

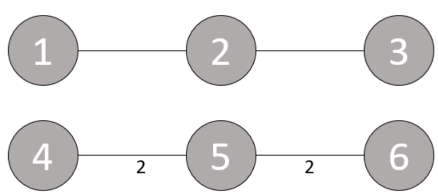


Figure 17: L_c graph

2.5 Differential equations canonical form

Any linear ordinary differential equation of order n can be reduced to a system of n linear ordinary differential equations of order 1. Given:

$$a_n y(t)^{(n)} + a_{n-1} y(t)^{(n-1)} + ... + a_2 \ddot{y}(t) + a_1 \dot{y}(y) + a_0 y(t) = g(t)$$

we ca call:

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ ... \\ x_n(t) \end{bmatrix} = \begin{bmatrix} y(t) \\ \dot{y}(t) \\ ... \\ y^{(n-1)}(t) \end{bmatrix} ; \quad \dot{x}(t) \begin{bmatrix} \dot{y}(t) \\ \ddot{y}(t) \\ ... \\ y^{(n)}(t) \end{bmatrix}$$

So the equation can be written as:

$$\begin{cases} a_n \dot{x}_n + a_{n-1} x_n + ... + a_1 x_2 + a_0 x_1 = g(t) \\ \dot{x}_1 = x_2 \\ ... \\ \dot{x}_{n-1} = x_n \end{cases}$$

The latter systme can be written in a compact matrix form:

$$A\dot{x}(t) + Bx(t) = C(t)$$

where:

$$A = \begin{bmatrix} 1 & 0 & ... & 0 \\ 0 & 1 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & a_n \end{bmatrix} ; \quad B = \begin{bmatrix} 0 & -1 & 0 & 0 & ... & 0 \\ 0 & 0 & -1 & 0 & ... & 0 \\ 0 & 0 & 0 & -1 & ... & 0 \\ ... & ... & ... & ... & ... & ... \\ a_0 & a_1 & a_2 & a_3 & ... & a_{n-1} \end{bmatrix}$$

$$C(t) = \begin{bmatrix} 0 \\ ... \\ g(t) \end{bmatrix}$$

If a_i are scalar then A is always invertible and we can reduce to a simpler form $\dot{x}(t) + Ax(t) = g(t)$, if a_i are matrices the method can still be used, we will have I (identity matrix) instead of 1 and A, B will be block matrices.

3 Undirected graphs case

In this section the problem of controlling a MAS under undirected connected topologies is faced analyzing the main methods found in literature. In particular, the problem is applied in the aerospace field in which each agent represents a satellite orbiting in Low-Earth Orbit and the graph topology the capability of agents to share information on angular position and/or velocity. The set of equations describing the single agent dynamic is the following [1],[2]:

$$\begin{cases} J_i\ddot{\theta}_i(t) + \delta_i^T\ddot{\eta}_i(t) = u_i(t) \\ \delta_i\ddot{\theta}_i(t) + \ddot{\eta}_i(t) + C_i\dot{\eta}_i(t) + K_i\eta_i(t) = 0 \end{cases}$$

These equations describe the rotational motion of a rigid body with flexible appendages. Considering the rotations in a 3-dimensional space, $\theta_i(t) \in \mathbb{R}^3$ is the vector of rotations around the three axes and J_i is the inertia matrix. Vector $\eta_i(t) \in \mathbb{R}^{m_j}$ is the modal coordinates vector representing the flexible modes of the appendages, m_j is the number of modes. $C_i \in \mathbb{R}^{m_j,m_j}$ is the damping coefficients matrix and $K_i \in \mathbb{R}^{m_j,m_j}$ the stiffness matrix. $\delta_i \in \mathbb{R}^{3,m_j}$ is the coupling matrix between the space of $\theta_i(t)$ and the space of $\eta_i(i)$. At last, the control input is $u_i(t) \in \mathbb{R}^3$.

Considering a set of N agents, $i = 1, 2, ..., N$, so we have a set of $2N$ equations. It can be grouped using block matrices:

$$\theta(t) = \begin{bmatrix} \theta(t)_1^T & \theta(t)_2^T & ... & \theta(t)_N^T \end{bmatrix}^T$$

$$\eta(t) = \begin{bmatrix} \eta(t)_1^T & \eta(t)_2^T & ... & \eta(t)_N^T \end{bmatrix}^T$$

$$u(t) = \begin{bmatrix} u(t)_1^T & u(t)_2^T & ... & u(t)_N^T \end{bmatrix}^T$$

and

$$\begin{cases} J = blockdiag(J_1, J_2, ..., J_N) \\ C = blockdiag(C_1, C_2, ..., C_N) \\ K = blockdiag(K_1, K_2, ..., K_N) \\ \Delta = blockdiag(\delta_1, \delta_2, ..., \delta_N) \end{cases}$$

where $\theta(t) \in \mathbb{R}^{3N}$, $\eta(t) \in \mathbb{R}^{m_{tot}N}$, with $m_{tot} = \sum_j m_j$, $u(t) \in \mathbb{R}^{3N}$, $J \in \mathbb{R}^{3N,3N}$, $C \in \mathbb{R}^{m_{tot},m_{tot}}$, $K \in \mathbb{R}^{m_{tot},m_{tot}}$, $\Delta \in \mathbb{R}^{m_{tot},3N}$.

The set of equations can be written as:

$$\begin{cases} J\ddot{\theta}(t) + \Delta^T\ddot{\eta}(t) = u(t) \\ \Delta\ddot{\theta}(t) + \ddot{\eta}(t) + C\dot{\eta}(t) + K\eta(t) = 0 \end{cases}$$

In addition, coefficients matrices have some properties:

$$J = J^T > 0; \;\; C = C^T > 0; \;\; K = K^T > 0;$$

$$\begin{bmatrix} J & \Delta^T \\ \Delta & I \end{bmatrix} > 0$$

This latter property is found in literature [1],[2], and the necessity of it is also prooved in section 4.4. ">" stands for positive definite matrix. Matrices C_i and K_i can be defined from the natural frequency ω_j and the damping coefficient ζ_j of each fexible mode in the following way:

$$C_i = diag(2\zeta_j\omega_j); \;\; K_i = diag(\omega_j^2) \quad \forall\, j$$

So also C and K are diagonal matrices. The following table reports ans example used as a reference for next sections:

Mode	Natural frequency ω	Damping ζ
Mode 1	0.7681	0.05607
Mode 2	1.1038	0.00862
Mode 3	1.8733	0.01283
Mode 4	2.5496	0.02516

Figure 18: Table 1

The modes are considered to be 4 for all the agents and a random variance added from the nominal values in Table 1. So $C_i \in \mathbb{R}^{4,4}$ and $K_i \in \mathbb{R}^{4,4}$

3.1 Measurable velocities

In this first case, angular velocity and displacement are considered to be measurable by the agents, so they are supposed to be equipped with gyroscopes. For now, the objective is considered to be the reach of consensus without the presence of a leader (reference), that will be introduced later. Formally, consensus is:

$$\lim_{t\to\infty} ||\theta_i(t) - \theta_j(t)|| = 0 \quad \forall\, i, j$$
$$\lim_{t\to\infty} ||\dot{\theta}_i(t) - \dot{\theta}_j(t)|| = 0 \quad \forall\, i, j$$

meaning that all the agents reach the same state, or the same position and velocity.

The proposed control law is the following:

$$u_i(t) = -g\sum_{j}^{N} p_{ij} R(\dot{\theta}_i(t) - \dot{\theta}_j(t)) - \sum_{j}^{N} q_{ij} R(\theta_i(t) - \theta_j(t))$$

R is, at least, a positive definite matrix, that in general should be considered also diagonal for sake of simplicity. p_{ij}, $q_i j$ and g are the control parameters. It is well known from literature [9], that, if $p_{ij} = p_{ji}$, $q_{ij} = q_{ji}$ and $g > 0$, consensus is reached, and the final value of the angular positions and velocities is the average among the initial conditions (time averaging consensus). This control law can be better understood in matrix form: calling

$$L_v = [l_{ij}]_v = \begin{cases} \sum_j p_{ij} & i = j \\ -p_{ij} & i \neq j \end{cases} \quad ; \quad L_d = [l_{ij}]_d = \begin{cases} \sum_j q_{ij} & i = j \\ -q_{ij} & i \neq j \end{cases} \quad ;$$

the control law is:

$$u(t) = -(gL_v \otimes R)\dot{\theta}(t) - (L_d \otimes R)\theta(t)$$

Hence, it can be seen as a PD-like (Proportional Derivative) controller. Matrices L_v and L_d are the Laplacian matrices corresponding to the graph of the communication topology of angular velocities and displacements. Since $p_{ij} = p_{ji}$ and $qij = q_{ji}$, the graph is undirected and $L_v = L_v^T \geq 0$, $L_d = L_d^T \geq 0$.

It is easier to understand now the role of matrix R since, as explained in section 2.4, if it is diagonal, it splits the graph into 3 independent subgraphs, one for each of the three coordinates.

Some simulations are now performed with different topologies:

Simulation 1

A total of $N = 5$ agents is considered. The communication topology is considered to be the same for angular velocities and displacements, so $L_v = L_d = L$. $L = D - A$ corresponds to the following graph:

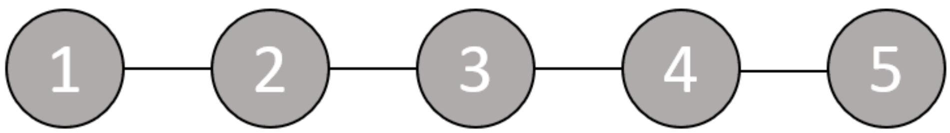


Figure 19: Simulation 1 graph

The simulation parameters are:

$$L = 100 \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}; \;\; g = 4$$

The initial displacements are set at random between -1 and 1, while initial velocities are set to 0 as the initial conditions of the modal vectors. C and K are defined randomly starting from Table 1. J and Δ are also defined randomly starting from:

$$J_0 = \begin{bmatrix} 100 & 3 & 4 \\ 3 & 280 & 10 \\ 4 & 10 & 190 \end{bmatrix}; \;\; \delta_0 = \begin{bmatrix} 6.4564 & 1.2781 & 2.1563 \\ -1.2562 & 0.9176 & -1.6726 \\ 1.1169 & 2.4890 & -0.8367 \\ 1.2364 & -2.6581 & -1.1250 \end{bmatrix}$$

while the matrix R are set as the identity matrix.

The following Figures (20-21) show the transient of the roations around the $y - axis$ and the space of phases of $x - y$ positions and velocities:

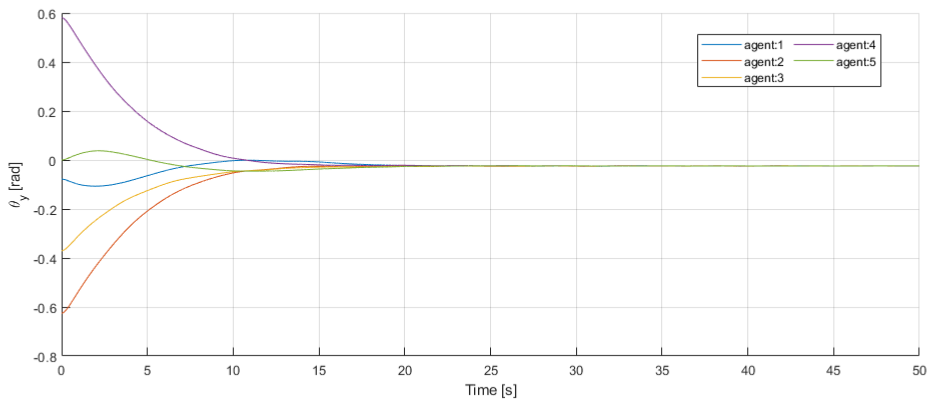


Figure 20: y-axis rotations of agents 1 to 5, Simulation 1

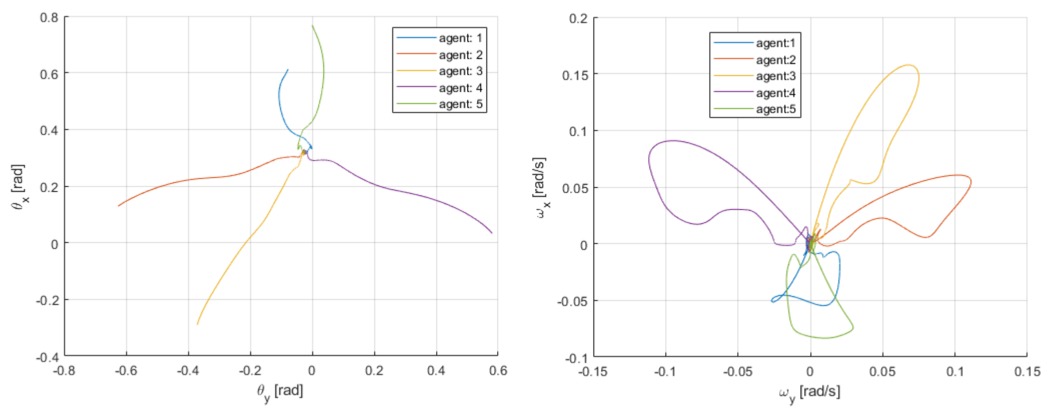


Figure 21: x-y plane of rotations (left) and velocities (right)

As expected, angular positions reach consensus to the average among the initial positions (Figure 20). In the phase space plots (Figure 21), we can also see how velocities start and finish at 0, so the paths are closed. Flexible modes all converge to 0 since $K > 0$ and $C > 0$.

Simulation 2

In this simulation $N = 7$ agents are considered:

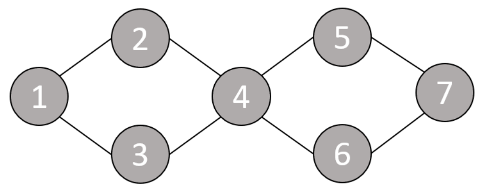


Figure 22: Simulation 2 topology, 7 agents

The simulation setting are the same as in simulation 1, but in this case velocities are also set at random value, different from 0. The following Figures (23-24) show the velocity and displacements transients and the x-y plane portraits:

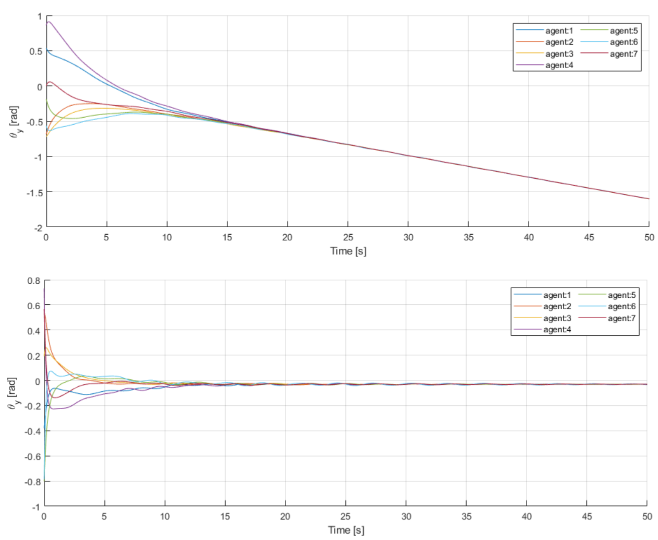


Figure 23: y-axis rotations and velocities of agents 1 to 7, Simulation 2

Also in this case both velocities and displacements reach consensus. With respect to the previous situation, in this case positions trend is to diverge as ramps with a slope equal to the average initial velocity.

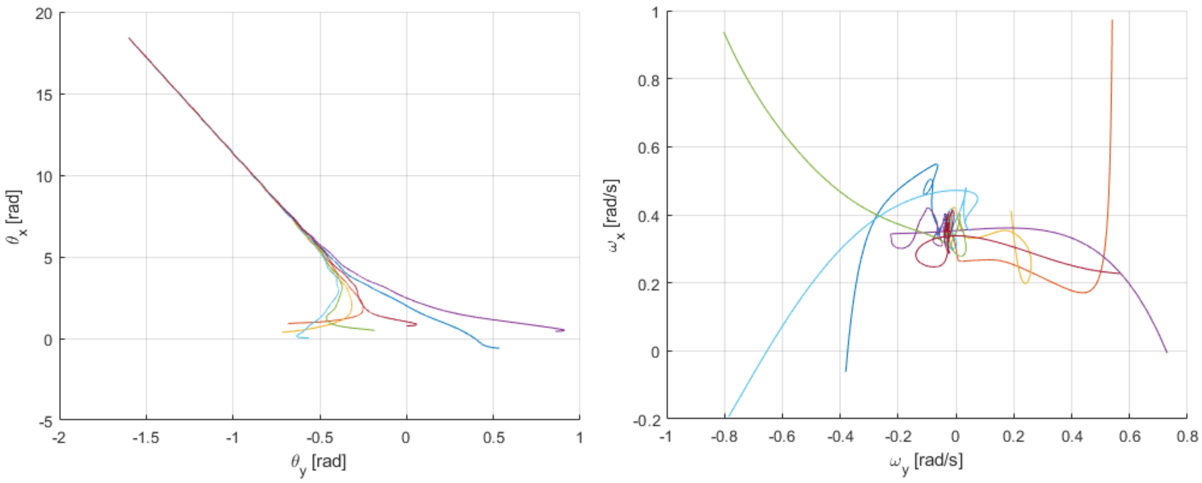


Figure 24: x-y plane portraits, Simulation 2

$$L = 100 \begin{bmatrix} 2 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 2 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 2 & -1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 4 & -1 & -1 & 0 \\ 0 & 0 & 0 & -1 & 2 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 & -1 & -1 & 2 \end{bmatrix}$$

3.2 Non-measurable velocities

Often satellites are not equipped with gyroscopes, so they are not able to get measure of angular velocity. For this reason, the control law in section 3.1 can be modified [1] as follows:

$$\begin{cases} u_i(t) = -z_i(t) - g\sum_j^N p_{ij} R(\theta_i(t) - \theta_j(t)) \\ \dot{z}_i(t) = -g z_i(t) - g\sum_j^N (g p_{ij} - q_{ij}) R(\theta_i(t) - \theta_j(t)) \end{cases}$$

Also in this case we can use block matrices to write the equations $\forall\, i$:

$$\begin{cases} u(t) = -z(t) - (gL_v \otimes R)\theta(t) \\ \dot{z}(t) = -g z(t) - g((gL_v - L_d) \otimes R)\theta(t) \end{cases}$$

In this case, $z(t) = \begin{bmatrix} z_1(t)^T & z_2(t)^T & \dots & z_N(t)^T \end{bmatrix}^T$ is the controller state that changes dynamically following the second equation in the set. In this case the parameter g has a lower bound, and in literature [1] we can find that always exists a g sufficiently large such that the system reaches consensus.

Simulation 2 of section 3.1 is reproposed with this modified control law, initial velocities are set to 0, $g = 100$, $L_v = 4L_d = 4L$

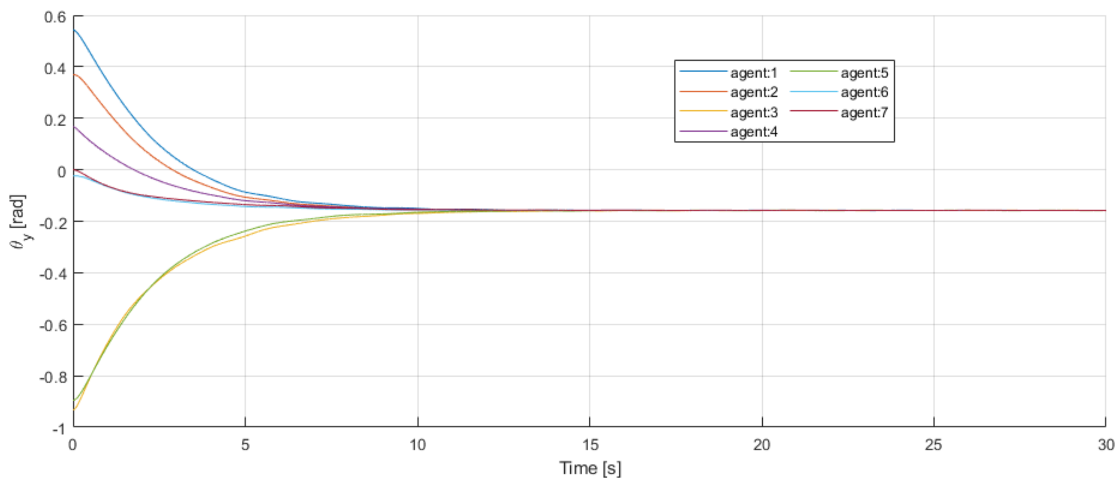


Figure 25: y-axis rotations transient, Simulation 2

Also in this case consensus is reached, parameters can be tuned to trade between settling time and control input, or to avoid chattering. For sake of completeness, the transient of the modal vectors is shown in the next Figure (26):

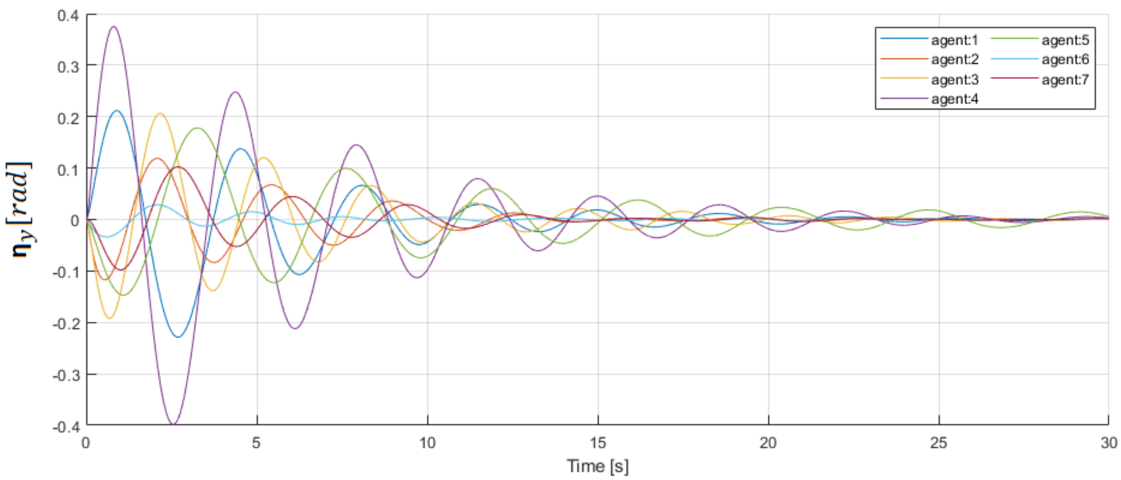


Figure 26: modal coordinates transient

Now a last case is analyzed introducing the concept of leader. The leader in undirected topologies is an agent, that can be a real physical object like others or equivalently just a signal, that acts as a reference for the others. So in this case consensus is the reaching of the same state of the leader:

$$\lim_{t\to\infty} ||\theta_i(t) - \theta_{leader}(t)|| = 0 \quad \forall\, i$$

$$\lim_{t\to\infty} ||\dot{\theta}_i(t) - \dot{\theta}_{leader}(t)|| = 0 \quad \forall\, i$$

As found in [29], control laws are able to guarantee consensus also in the presence of a leader. To make the leader a constant reference we set its dynamic to be:

$$J_{leader}\ddot{\theta}_{leader}(t) = 0; \;\; \eta_{leader}(t) = 0$$

A simulation is now presented with the following topology. The leader is

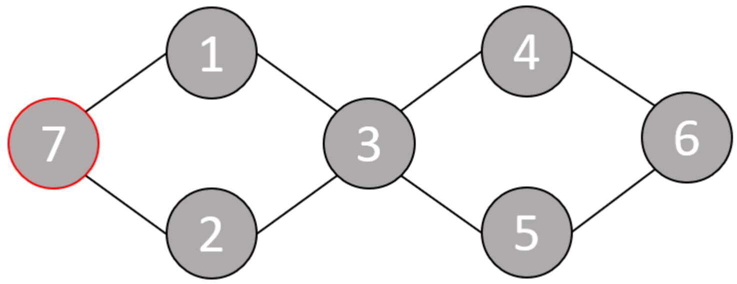


Figure 27: Directed topology with the presence of a leader

set as a convention in the last position in the enumeration (Agent 7, circled in red). Since the leader is not moving, its state is constant and equal to the initial conditions, the other agents should converge to it. In this example leader's initial conditions are:

$$\theta_{leader}(0) = \begin{bmatrix} 0.7684 & 0 & 0.224 \end{bmatrix}^T; \; \dot{\theta}_{leader}(0) = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$$

The following figure shows the transient of the rotations around the x-axis:

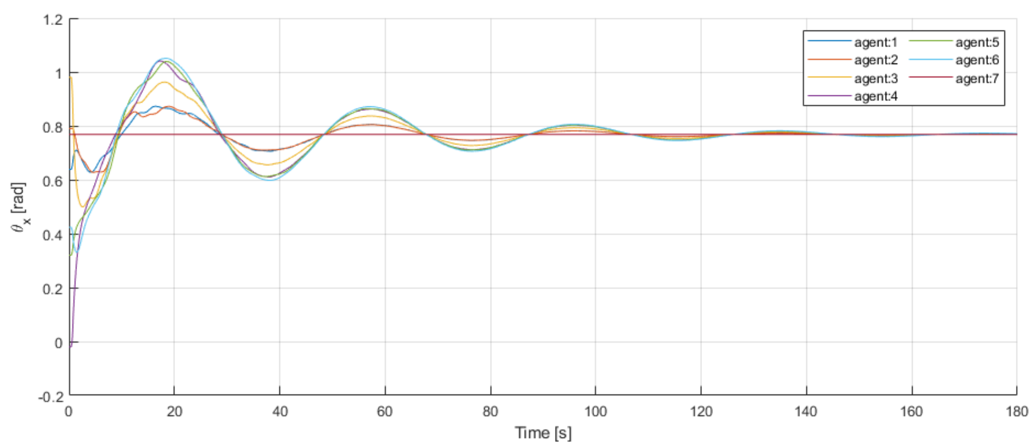


Figure 28: Directed topology with the presence of a leader

4 Directed graph cases

In this section directed topology cases are be introduced, the main results of this work and related proofs are shown.

4.1 Eigenprojections in dynamic systems

To introduce the main results eigenprojections and their applications in dynamic systems are shown. In general the eigenprojection of a matrix A is an idempotent matrix A^+ that satisfies:

$$Range(A^+) = Null(A^\nu)$$

$$Range(A^\nu) = Null(A^+)$$

where ν is the smallest natural number such that $rank(A^\nu) = rank(A^{\nu+1})$. Since A^+ is idempotent, it is completely determined by its range and null space. Let's take an example:

$$A = \begin{bmatrix} 2 & -1 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \quad A^+ = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

In this case $rank(A^0) \neq rank(A^1)$,$rank(A^1) = rank(A^2)$ so $\nu = 1$. A basis of the range of A can be $\{\begin{bmatrix} 2 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}\}$ and a basis of the null space is $\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$. So the dimension of $null(A)$ is 1, that will correspond to the dimension of $range(A^+)$, while the dimension of $range(A)$ is 2, that will correspond to the dimension of $null(A^+)$. This means that $rank(A^+) = 1$. It is a property of idempotent matrices to have the trace equal to the rank [6] so $trace(A^+) = rank(A^+) = 1$. Combining this with $range(A^+) = span(\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix})$ we get:

$$A^+ = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

Notice that the non-zero column must be the third otherwise $null(A^+) \neq range(A)$. From this example we can see a useful property of eigenprojections: if a square matrix A such that $rank(A) = rank(A^2)$ has a row all set to 0, its eigenprojection is a matrix with the correspondent column set to 1 and all other entries set to 0.

It is a property of eigenprojections that:

$$A^+ = \lim_{|\tau|\to\infty} (I + \tau A^\nu)^{-1}$$

that can be used to link eigenprojections with dynamic systems. To this end, let‘s consider the following differential equation:

$$\dot{x} + Ax = 0$$

and its Laplace transform:

$$sx(s) - x_0 + Ax(s) = 0 \quad \to \quad x = (sI + A)^{-1}x_0$$

We know that the solution in time domain is $x(t) = e^{-At}x_0$, so using final value theorem if $\lim_{t\to\infty} x(t)$ exists and is finite we can write:

$$\lim_{s\to 0} s(sI + A)^{-1}x_0 = \lim_{t\to\infty} e^{-At}x_0$$

Since in general $(CB)^{-1} = B^{-1}C^{-1}$, if $C = sI + A$ and $B = \frac{1}{s}I$:

$$\lim_{s\to 0} s(sI + A)^{-1} = \lim_{s\to 0}[(sI + A)\frac{1}{s}I]^{-1} = \lim_{s\to 0}(I + \frac{1}{s}A)^{-1}$$

Now calling $\tau = \frac{1}{s}$ and if $\nu = 1$:

$$\lim_{t\to\infty} e^{-At}x_0 = \lim_{|\tau|\to\infty} (I + \tau A)^{-1}x_0 = A^+x_0$$

So we can conclude that if the solution of a first order differential equation like $\dot{x} + Ax = 0$ has a finete limit to infinity, it is equal to A^+x_0 if $rank(A) = rank(A^2)$. We can now apply this to some simple cases involving Laplacians. Let‘s consider

$$\dot{x} + Lx = 0$$

where $L = D_{in} - A^T$ is the generalized Laplacian associated to a general directed graph one source(leader) and a directed spanning tree starting from it, for example:

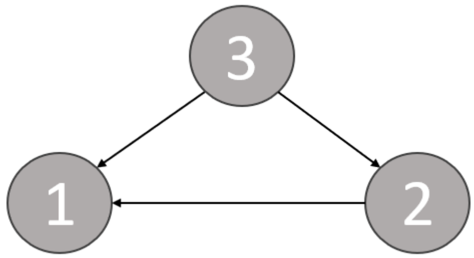


Figure 29: Graph example

$$L = \begin{bmatrix} 2 & -1 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \quad L^+ = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

Since the graph has a source, the correspondent third row is 0. This means that, since $rank(L) = rank(L^2)$, the eigenprojection of L has the third column set to 1 and all other entries set to 0, as already shown. Moreover the 0 eigenvalue of L is simple, so since other eigenvalues are all positive, the solution converges. This means that:

$$\lim_{t\to\infty} x(t) = L^+ x_0 = \begin{bmatrix} x_0(3) \\ x_0(3) \\ x_0(3) \end{bmatrix}$$

meaning that all the agents will converge to the source‘s (leader) initial condition (consensus).

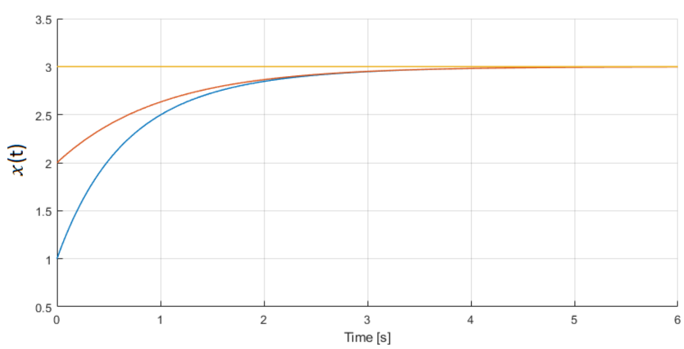


Figure 30: First order transient example

This latter example can be generalized for all directed graphs with a source, since $rank(L) = rank(L^2)$ holds in general. To show it let's take the Jordan form of such Laplacian:

$$L = TJT^{-1}; \quad J = \begin{bmatrix} J_0 & ... & 0 & 0 \\ 0 & J_1 & ... & 0 \\ ... & ... & ... & 0 \\ 0 & 0 & ... & J_n \end{bmatrix}$$

where J_i is the Jordan block associated to the i_{th} eigenvalue of L. Now taking the square of L: $L^2 = LL = TJT^{-1}TJT^{-1} = TJ^2T^{-1}$. Since T is full-rank, the rank of L^2 is the same of J^2.

$$J^2 = \begin{bmatrix} J_0^2 & ... & 0 & 0 \\ 0 & J_1^2 & ... & 0 \\ ... & ... & ... & 0 \\ 0 & 0 & ... & J_n^2 \end{bmatrix}$$

Consider in a slightly more general case d as the number of sources in the graph, $rank(L) = n - d$, moreover d is also the number of rows set to 0 in the Laplacian (if $d = 0$ then $rank(L) = n - 1$ since we have the $\mathbf{1}$ vector that is a basis of $null(L)$, but the reasoning is the same). This means that the algebraic multiplicity of the 0 eigenvalue is always equal to geometric multiplicity, this can be shown writing g (geometric multiplicity) as $g = dim\{null(L - 0I)\} = dim\{null(L)\}$ that is equal to the number of *zero* rows of L. This means that the 0 eigenvalue is semisimple and that the correspondent Jordan blocks are of dimension 1. All other Jordan blocks are full-rank so this implies that $rank(J^2) = rank(J)$. For example:

$$L = \begin{bmatrix} 2 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad J = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$$

$rank(L^2) = rank(L) = rank(J) = rank(J^2)$.

Let's now analyze the second order case, starting from a general case like:

$$\ddot{x_c} + gL\dot{x_c} + Lx_c = 0$$

where $g \geq 0$ is a real number. We can transform the system in a first order one as in (2.5):

$$\dot{x} + Ax = 0$$

where $A = \begin{bmatrix} 0 & -I \\ L & gL \end{bmatrix}$ and $x = \begin{bmatrix} x_c \\ \dot{x}_c \end{bmatrix}$. In this case $rank(A) \neq rank(A^2)$ so we can't use the same approach as for the first order dynamics. Let's at first study the sign of the eigenvalues of the system: using block matrices properties (2.3) we can write that:

$$det(\begin{bmatrix} H_1 & H_2 \\ H_3 & H_4 \end{bmatrix}) = det(H_1)det(H_4 - H_3H_1^{-1}H_2)$$

and apply it to $A - sI$:

$$det(\begin{bmatrix} -sI & -I \\ L & gL - sI \end{bmatrix}) = det(-sI)det\{gL - sI - L(-\frac{1}{s}I)(-I)\} =$$

$$= det(-sI)det(gL - sI - \frac{1}{s}L) = det(-sI)det((g - \frac{1}{s})L - sI) =$$

$$= det(-sI)det(L - \frac{s}{g - \frac{1}{s}}I)$$

So if $\lambda_i = \frac{s_i}{g - \frac{1}{s_i}}$ is an eigenvalue of L, s_i is an eigenvalue of A. We can now solve the latter equation to find the eigenvalues of A as function of those of L:

$$s_i^2 - g\lambda_i s + \lambda_i = 0$$

$$s_i^{1,2} = \frac{g\lambda_i \pm \sqrt{g^2\lambda_i^2 - 4\lambda_i}}{2}$$

If all the eigenvalues of L are real, we know that $\lambda_i \geq 0$, so if $g^2\lambda_i^2 < 4\lambda_i$ the square root is complex and $Re\{s_i\} = g\lambda_i \geq 0$ since $g \geq 0$. If $g^2\lambda_i^2 \geq 4\lambda_i$ the square root is a real number, but $\sqrt{g^2\lambda_i^2 - 4\lambda_i} \leq g\lambda_i$ so also in this case $s_i \geq 0$. If the eigenvalues of L are in general complex, a more strict bound for g that guarantees the eigenvalues s_i to have non negative real part can be found in literature [3], in particular

$$g^2 > \frac{Im^2(\lambda_i)}{Re(\lambda_i)||\lambda_i||^2} \quad \forall\, i|\lambda_i \neq 0$$

If the two Laplacians matrices are not the same, the explicit computation of eigenvalues is not trivial and can be done using numerical tools. This shows that all the eigenvalues of the system are non-negative, but still we have to proove consensus.

To this end, let‘s consider the solution in the time domain $x(t) = e^{-At}x_0$. The exponential can be replaced by its Taylor series:

$$e^{-At} = I + (-A)t + (-A)^2\frac{t^2}{2} + (-A)^3\frac{t^3}{3!} + ...$$

Now substituting the canonical Jordan form of $A = TJT^{-1}$ we get:

$$e^{-At} = I + (-TJT^{-1})t + (-TJT^{-1})^2\frac{t^2}{2} + (-TJT^{-1})^3\frac{t^3}{3!} + ...$$

Since all the powers can be written as $(TJT^{-1})^n = TJ^nT^{-1}$, the whole exponential can be rewritten as:

$$e^{-At} = Te^{-Jt}T^{-1}$$

From now on, let‘s consider A as $(-A)$ for sake of simplicity, so all the eigenvalues of A are non-positive. The Jordan matrix J can be written as $J = diag(J_0, J_1, ..., J_n)$ where J_i is the Jordan block associated to the $i-th$ eigenvalue. If we consider J_0 as the Jordan block of the 0 eigenvalue, since it has algebraic multiplicity 2 and geometric multiplicity 1, it is equal to:

$$J_0 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$

Moreover $e^{diag(j_i)} = diag(e^{J_i})$ so we can write:

$$e^{Jt} = \begin{bmatrix} e^{J_0} & 0 \\ 0 & e^{Ft} \end{bmatrix}$$

where F is the block matrix containing all the non-zero eigenvalues of A. Notice that $J_0^n = 0$ for any natural $n > 1$ so:

$$e^{Jt} = \begin{bmatrix} I + J_0t & 0 \\ 0 & e^{Ft} \end{bmatrix}$$

Let‘s now consider the final value taking the limit:

$$\lim_{t\to\infty} e^{Jt} \approx \begin{bmatrix} I + J_0t & 0 \\ 0 & 0 \end{bmatrix}$$

since e^{Ft} goes to 0 having all strictly negative eigenvalues.

Now we can say that

$$\lim_{t\to\infty} e^{At} \approx T \begin{bmatrix} I + J_0 t & 0 \\ 0 & 0 \end{bmatrix} T^{-1} = T \begin{bmatrix} \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix} & 0 \\ 0 & 0 \end{bmatrix} T^{-1}$$

T is a matrix whose columns are the right generalized eigenvectors of A, while T^{-1} is a matrix whose rows are the left generalized eigenvectors of A. Since in J we put the two zero eigenvalues in the first to places, the correspondent left and right eigenvectors will be in the first two columns and rows of T^{-1} and T that are respectively $l_1 = \begin{bmatrix} 1 & 0 & 0 & .. & 0 \end{bmatrix}^T$ and $l_2 = \begin{bmatrix} 0 & 1 & 0 & .. & 0 \end{bmatrix}^T$, $r_1 = \begin{bmatrix} 1 & 0 & 1 & 0.. \end{bmatrix}^T$ and $\begin{bmatrix} 0 & 1 & 0 & 1.. \end{bmatrix}^T$. This can be intuitively seen considering that if we take a vector v that has all zero entries exept one set to 1, $v^T A = 0$ if the 1 is in the same place of the 0 row of L. Moreover if we take a vector v with half of the entries set to 1, and the other half set to 0 ($\begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}^T$ for instance), $Av = 0$. Then Jordan form "rearrange" variables in order to have the alternation of x_i and $\dot{x}_i$, so also the entries of the eigenvectors will follow that pattern. Now, calling x_l and $\dot{x}_l$ the initial conditions of the leader, we can compute:

$$\lim_{t\to\infty} e^{At} x_0 \approx T \begin{bmatrix} \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix} & 0 \\ 0 & 0 \end{bmatrix} T^{-1} x_0 = \begin{bmatrix} x_l + \dot{x}_l t \\ \dot{x}_l \\ x_l + \dot{x}_l t \\ \dot{x}_l \\ ... \end{bmatrix}$$

where $x_0 = \begin{bmatrix} x_l & \dot{x}_l & x_{c0}(1) & \dot{x}_{c0}(1) & x_{c0}(2) & \dot{x}_{c0}(2) & ... \end{bmatrix}^T$ is the initial conditions vector with the initial position and velocity alternated for each agent. This establishes consensus since the positions and velocities of all the agents (leader included) are the same. Moreover notice that position diverges as a ramp, and that if $\dot{x}_l = 0$ all the agents converge to the same constant position. At last it can be useful write this latter result as

$$\lim_{t\to\infty} e^{At} x_0 \approx \begin{bmatrix} L^+ & L^+ t \\ 0 & L^+ \end{bmatrix} x_0'$$

where $x_0' = \begin{bmatrix} x_c(0) \\ \dot{x}_c(0) \end{bmatrix}$.

4.2 Rigid dynamics, continuous time

Goin back in the aerospace context, let‘s start with simple cases, so flexible dynamics and external disturbances will be neglected. Considering N agents, we can write the single agent dynamic as $J_i\ddot{\theta}_i = u_i$ where J_i is the inertia matrix of the i-th agent, $\theta_i = [\theta_{ix}\ \theta_{iy}\ \theta_{iz}]^T$ and $u_i = [u_{ix}\ u_{iy}\ u_{iz}]^T$.

We can apply to the dynamic system the following control law:

$$\begin{cases} J\ddot{\theta} = u \\ u = -gL\dot{\theta} - L\theta \end{cases}$$

where $J = blkdiag\{J_1, J_2, ..., J_N\}$, $\theta = [\theta_1^T, \theta_2^T, ..., \theta_N^T]^T$ and $L = L_1 \otimes R$ with R a diagonal positive matrix.

L_1 represents the communication topology between the agents and $L = L_1 \otimes R$ splits in 3 identical graphs that do not share information each other, one for each coordinate, as explained in section 2.4. In general the comunication topology among the agents is considered to have one *Leader* wich is a source and a directed spanning tree. The Laplacian is a generalized one computed as $L = D_{in} - A^T$.

As before (referring to section 3, not done yet), consensus is reached if:

$$\lim_{t\to\infty} ||\theta_i(t) - \theta_j(t)|| = 0 \quad \forall \quad i, j$$

$$\lim_{t\to\infty} ||\dot{\theta}_i(t) - \dot{\theta}_j(t)|| = 0 \quad \forall \quad i, j$$

Since J is positive definite, also J^{-1} is and $\ddot{\theta} + gJ^{-1}L\dot{\theta} - J^{-1}L\theta = 0$ respects all the hypothesis used in section (4.1) since multiplying by a positive definite matrix doesn‘t change the sign of the eigenvalues, so consensus is reached. The g parameter and the weights in the communication graph can be changed to tune the system behaviour as shown in the following numerical examples.

Example 1: in this first simulation 4 agents and a leader are considered. System parameters are the following:

$$L_1 = \epsilon(D_{in} - A^T) = 200 \begin{bmatrix} 1 & 0 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}; \quad J_0 = \begin{bmatrix} 350 & 3 & 4 \\ 3 & 280 & 10 \\ 4 & 10 & 190 \end{bmatrix};$$

$J_i = rand(0.5 \div 2)J_0$: the inertia matrices of the agents are randomly computed starting from J_0

The leader is considered to be, as a convention, the last in the agents enumeration.

$$\begin{bmatrix} \theta(0) \\ \dot{\theta}(0) \end{bmatrix} = \begin{bmatrix} rand(12,1) \\ \theta_f \\ 0 \\ 0 \\ \dots \end{bmatrix}$$

So the initial positions of the agents are random except for the leader that has $\theta_5(0) = \theta_f = [0.7684, 0, 0.224]^T$ that will be the reference for other agents. The g parameter is set to 2. The following Figures(31-32) shows the graph topology and the transient of the θ_y and $\dot{\theta}_y$ for all the agents:

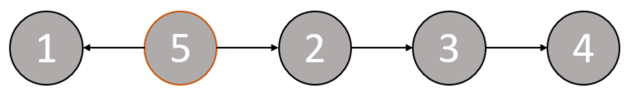


Figure 31: Graph topology for rigid dynamics, agent 5 (red) is the leader

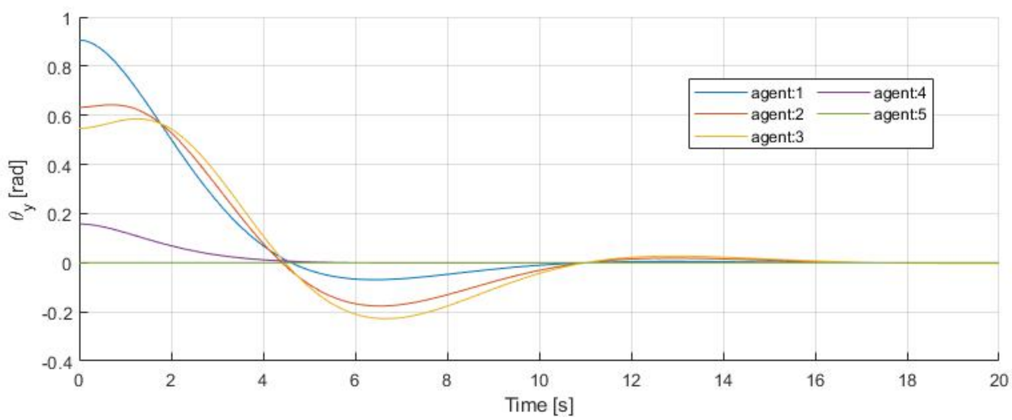


Figure 32: y-axis rotation for rigid dynamics

Notice that L has all real eigenvalues, so in this case $g > 0$ is sufficient to guarantee consensus. In Figure 34 a comparison with different values of g is shown.

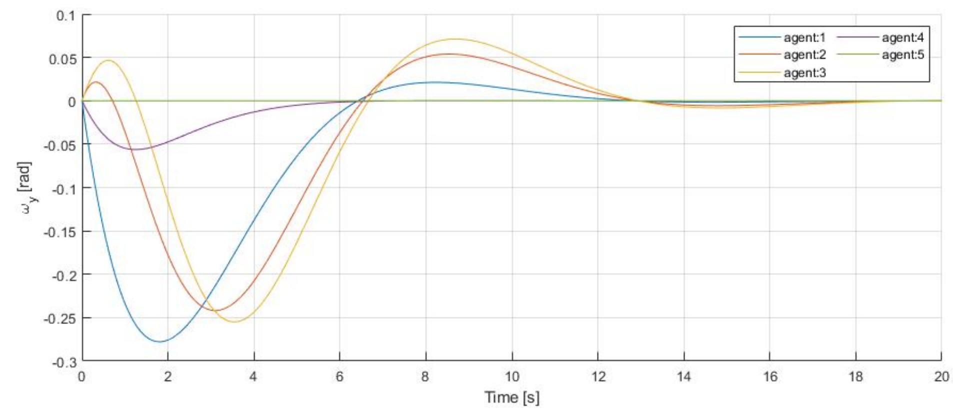


Figure 33: y-axis angular velocity fo rigid dynamics

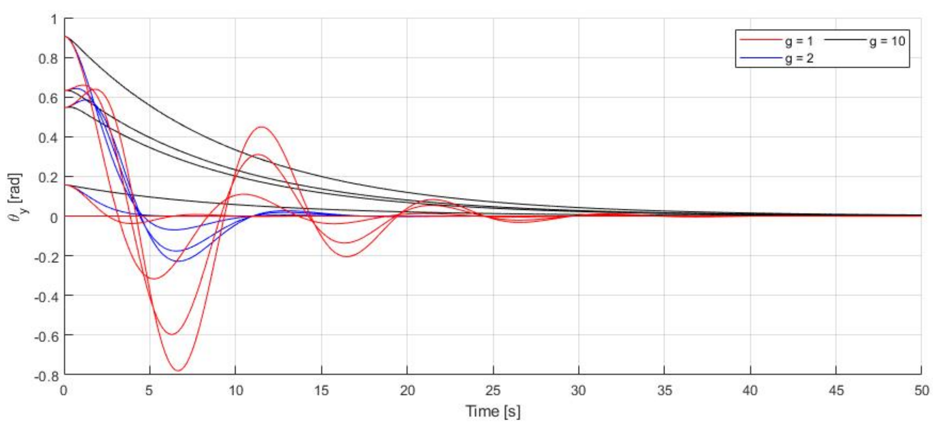


Figure 34: g values comparison

We can notice that as g increases, the oscillatory behaviour decreases. For very small values of g we get a long transient with great oscillations. Let‘s now perform the same simulation with the Leader‘s initial velocity different from 0, we expect the final value of the angular position of all the agents to diverge as a ramp with a slope equal to the Leader‘s initial velocity.

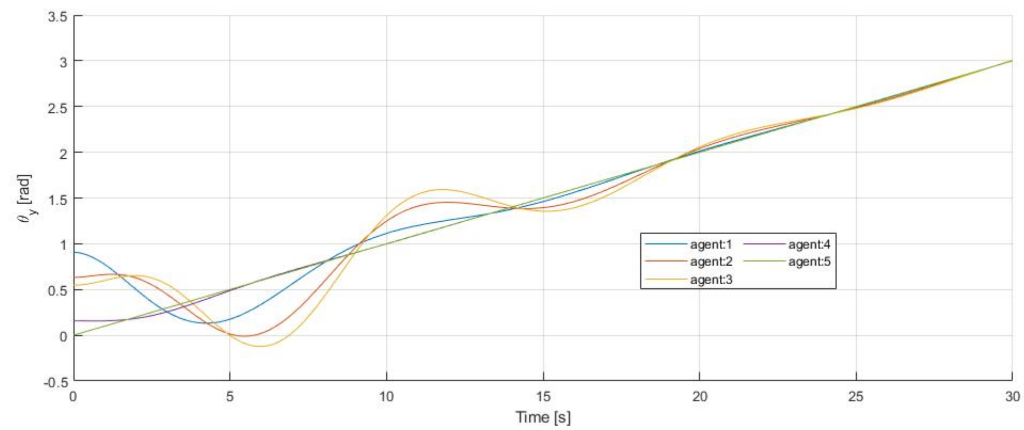


Figure 35: Leader's initial velocity different from 0

As expected the position of all the agents goes to infinity trying to follow the leader's one. In this case the initial condition $\dot{\theta}_5(0) = 0.1 rad/s$. Let's now analyze more complicated topologies, as the example in the next figure:

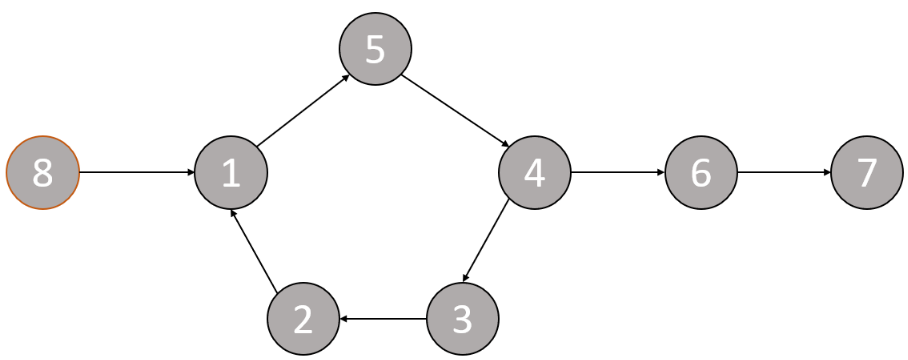


Figure 36: Graph topology, agent 8 (red) is the leader

In this case, to get a more realistic simulation, an input saturation is added to the model, that for small satellites can be around 0.02 Nm, so $u_{max} = 0.02$, $u_{min} = -0.02$. The initial conditions and physical parameters are the same as the previous simulation, obviously adding 3 agents.

The generalized laplacian $L_1 = D_{in} - A^T$ in this case is:

$$L_1 = 200 \begin{bmatrix} 2 & -1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Since the eigenvalues are not all real, the lower bound of g is: $g_{min} = 0.0065$. Above this value, we can tune g to tune the transient, the following simulation was performed with $g = 2.5$.

The following Figure(37) shows the transient of the rotations around the y-axis:

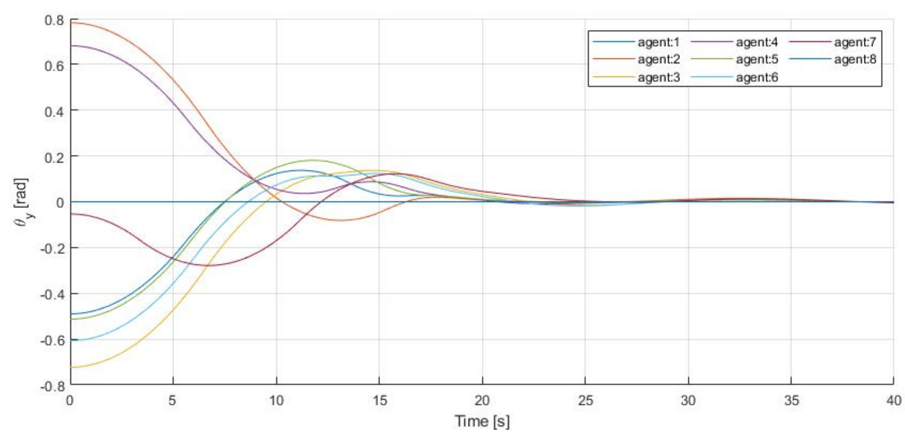


Figure 37: y-axis rotations for agents from 1 to 8, rigid dynamics

Consensus is reached in a reasonable time also in the presence of strong limitations on the input saturation. Now a different way to tune the transient is shown. Until now, the entire Laplacian matrix has been multiplied by a scalar factor ϵ, giving the same weight to all the arcs in the graph. However, the weights of the arcs can be tuned arbitrarely, so the idea is to lower the value of ϵ and giving a weight to each arc that is proportional to the distance in the graph between the node that it reaches and the Leader. Taking as an example the same topology of the last simulation (Figure 25) we can obtain the same performances in terms of settling time and control input energy giving to the arcs different weights as in the following figure:

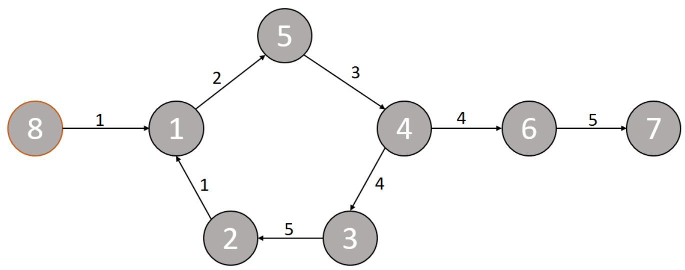


Figure 38: Graph topology with modified weights

So now the graph Laplacian can be written as:

$$L_1 = \epsilon D_w(D_{in} - A^T); \quad D_w = diag(d); \quad d = \begin{bmatrix} 1 & 5 & 4 & 3 & 2 & 4 & 5 & 0 \end{bmatrix}$$

The diagonal matrix D_w depends on the vector d that contains the distances as said before. The previous case can be seen as a subcase where $d = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$ and $\epsilon = 200$, to get similar performance in the two cases, we can set the quantity $||d||_1\epsilon$ to be the similar, so since in the first case $||d||_1 = 7$, ϵ can be set to 58 in this second case. The following Figure(39) shows the comparison between the two cases:

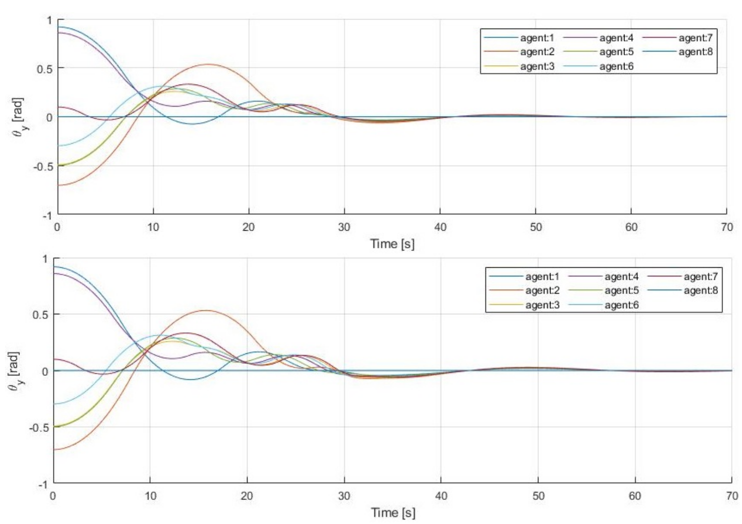


Figure 39: Case 1 above, Case 2 below

4.3 Rigid dynamics, discrete time

In this section the discrete time equivalent of the system in section 4.2 is studied. This can be useful to compute an estimate of the angular velocity in case it is not measurable. Let's start considering:

$$\dot{x}(t) = \begin{bmatrix} \dot{\theta}(t) \\ \ddot{\theta}(t) \end{bmatrix} = \begin{bmatrix} 0 & I \\ -gJ^{-1}L & -J^{-1}L \end{bmatrix} \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \end{bmatrix} = Ax(t)$$

We can use the Euler's method to discretize $\dot{x}(t) = Ax(t)$:

$$\dot{x}(t) \approx \frac{x(t+k) - x(t)}{k}$$
$$x(t+k) \approx (kA + I)x(t)$$
$$kA + I = \begin{bmatrix} I & kI \\ -kgJ^{-1}L & -kJ^{-1}L + I \end{bmatrix}$$

So we can say that:

$$\theta(t+k) \approx \theta(t) + k\dot{\theta}(t)$$
$$\dot{\theta}(t) \approx \frac{\theta(t+k) - \theta(t)}{k}$$

This means that the velocity is approximated with the average velocity between two time instants, t and $t+k$. Clearly $\lim_{k\to 0} \frac{\theta(t+k)-\theta(t)}{k} = \dot{\theta}(t)$ so the approximation is consistent. Now an upper bound for k must be computed to guarantee the convergence of the solution. Recalling that for a discrete LTI system the eigenvalues of the associated matrix must be in the unit circle, we can write:

$$eig(kA + I) = 1 - k\frac{g\lambda_i \pm \sqrt{g^2\lambda_i^2 - 4\lambda_i}}{2}$$

where λ_i is an eigenvalue of $J^{-1}L$. Calling $s_j = \frac{g\lambda_i \pm \sqrt{g^2\lambda_i^2 - 4\lambda_i}}{2}$ and recalling (4.1) that if $g^2 > \frac{Im^2(\lambda_i)}{Re(\lambda_i)||\lambda_i||^2}$, $Re(s_j) > 0$ for all non zero eigenvalues. Let's call Re_{min} the minumum real part among the real parts of all the non zero s_j, all the non zero eigenvaluea of A lie in a circular segment:

$$s_j \in \Theta = \{X : X = a + ib;\;\; a^2 + b^2 \le k||s_j||_{max};\;\; a \ge kRe_{min}\}$$

So $1 - ks_j$ for $k = 1$ lies in a circular segment that is flipped with respect to Θ and translated by 1, in the following Figure (40) represented by the green area:

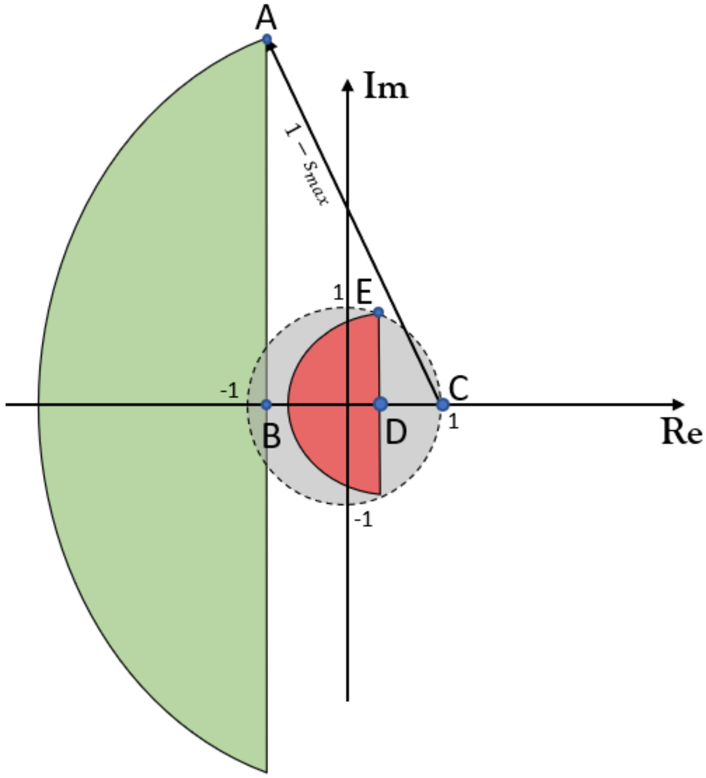


Figure 40: Visual proof

In this Figure (40) $s_{max} = ||s_j||_{max}$ is the maximum modulus among all the eigenvalues. The distance BC is $BC = 1 - Re_{min}$. Reducing k both the radius of Θ and the distance from point C = (1,0) are reduced. To ensure $\Theta \subset U$ where U is the unit circle (grey circle with dashed edge), segment AB for a generic k must be less or equal to segment ED.

$$AB = \sqrt{k^2 s_{max}^2 - (1 - kRe_{min})^2}, \quad ED = \sqrt{1 - (1 - kRe_{min})^2}$$

$$AB \leq ED \;\rightarrow\; k^2 s_{max}^2 - (1 - kRe_{min})^2 \leq 1 - (1 - kRe_{min})^2$$

$$k \leq \frac{1}{s_{max}} \;\leftrightarrow\; k \leq \frac{1}{||s_j||} \;\forall\; j : s_j \neq 0$$

Clearly, if $s_j = 0$ the correspondent eigenvalue is 1, but the behaviour of the zero eigenvalues of A was already analyzed in 4.1, so this discrete time equivalent diverge in the same way, for non zero leader‘s initial velocities.

Following Figures (41-42) show two simulations where the control law uses the estimated velocity, $s_{max} = 5.2513$ so $k_{min} = 0.1904$, in the fisrst simulation $k = 0.01$, in the second $k = 0.2$.

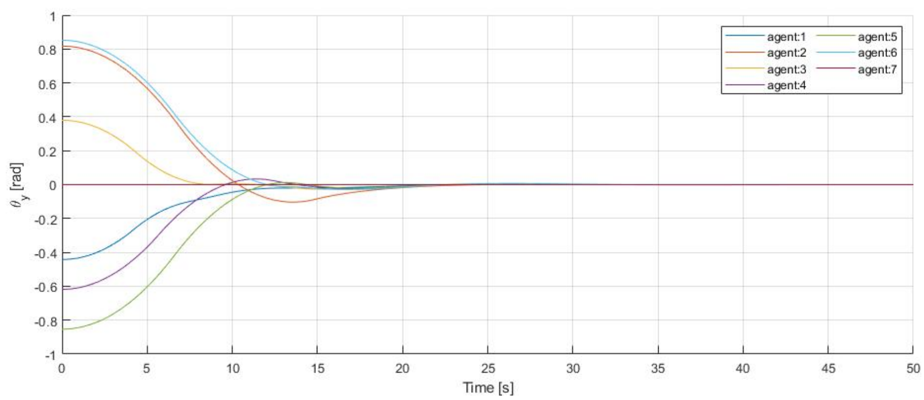


Figure 41: First simulation, k = 0.01

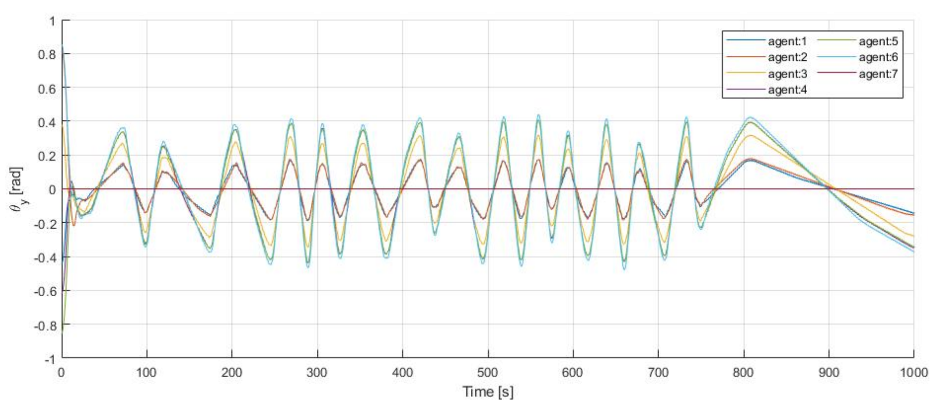


Figure 42: Second simulation, k = 0.2

4.4 Flexible dynamics, continuous time

In this section flexible dynamics are added to the model in order to get higher accuracy. As in section 3, the "rigid" equation is coupled with a second order equation that models the effects of a flexible appendage on the motion of the whole body:

$$\begin{cases} J\ddot{\theta}(t) + \Delta^T\ddot{\eta}(t) = u(t) \\ \Delta\ddot{\theta}(t) + \ddot{\eta}(t) + C\dot{\eta}(t) + K\eta(t) = 0 \\ u(t) = -gL\dot{\theta}(t) - L\theta(t) \end{cases}$$

Also in this case, considering N agents (N-1 followers and 1 leader) $J = diag(J_i) > 0$, $C = diag(C_i) > 0$, $K = diag(K_i) > 0$, $\theta(t) \in \mathbb{R}^{3N}$angular displacements vector, $\eta(t) \in \mathbb{R}^{m_{tot}N}$, where m_{tot} is the sum of the number of modes of the flexible parts of all the agents, is the modal coordinate vector. $\Delta = diag(\delta_i)$ is the coupling matrix from the space of θ to the space of η.

We can reduce the equations to a single first order equation:

$$A\dot{x}(t) + Bx(t) = 0$$

$$A = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & J & 0 & \Delta^T \\ 0 & 0 & I & 0 \\ 0 & \Delta & 0 & I \end{bmatrix}; \quad B = \begin{bmatrix} 0 & -I & 0 & 0 \\ gL & L & 0 & 0 \\ 0 & 0 & 0 & -I \\ 0 & 0 & K & C \end{bmatrix}; \quad x(t) = \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \\ \eta(t) \\ \dot{\eta}(t) \end{bmatrix}$$

A is a symmetric matrix, so $eig(A)$ are all real. Moreover, taking the Schur's complement of A:

$$S(A) = \begin{bmatrix} I & 0 \\ 0 & J \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ 0 & \Delta^T \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & \Delta \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & J - \Delta^T\Delta \end{bmatrix}$$

If $J - \Delta^T\Delta > 0$ also $A > 0$. So we assume:

$$\begin{bmatrix} J & \Delta^T \\ \Delta & I \end{bmatrix} > 0 \;\leftrightarrow\; J - \Delta^T\Delta > 0 \;\rightarrow\; A > 0$$

Hence A is invertible. In particular:

$$A^{-1} = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & (J - \Delta^T\Delta)^{-1} & 0 & -(J - \Delta^T\Delta)^{-1}\Delta^T \\ 0 & 0 & I & 0 \\ 0 & -\Delta(J - \Delta^T\Delta)^{-1} & 0 & \Delta(J - \Delta^T\Delta)^{-1}\Delta^T \end{bmatrix}$$

Since $A > 0$ also $A^{-1} > 0$.
B can be written as $B = \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix}$. The block B_2 has all positive eigenvalues, this can be saied since $C = C^T > 0$, $K = K^T > 0$, so the associated system is stable, moreover $det(B_2) = det(C)det(C^{-1}K) \neq 0$ so there is no zero eigenvalue.

The block B_1 was already studied in section 4.1. This means that the product $A^{-1}B$ does not change the sign of the eigenvalues of B, that are all non negative, moreover the zero eigenvalues of $A^{-1}B$ are the same with the same multiplicity of B_1. So $e^{-A^{-1}B}$ can be written in the same way as in section 4.1:

$$e^{-A^{-1}Bt} = Te^{-Jt}T^{-1} = T \begin{bmatrix} e^{J_0 t} & 0 \\ 0 & e^{Ft} \end{bmatrix} T^{-1}$$

Since e^{Ft} converges (eigenvalues of F are strictly negative):

$$\lim_{t\to\infty} \eta(t) = 0; \ \ \lim_{t\to\infty} \dot{\eta}(t) = 0$$

Now some simulations are presented with different conditions.

Simulation 1

In this first simulation the same topology of Figure 36 (N = 8) is considered. The considered parameters are:

$$\theta_0 = \begin{bmatrix} rand(3(N-1), 1) \\ \theta_f \end{bmatrix}; \ \ \dot{\theta}_0 = \begin{bmatrix} rand(3(N-1), 1) \\ 0 \\ 0 \\ 0 \end{bmatrix} \ \ ;$$

$$\eta_0 = rand(4N, 1); \ \ \dot{\eta}_0 = rand(4N, 1); \ \ \theta_f = \begin{bmatrix} 0.7684 & 0 & 0.224 \end{bmatrix}$$

J,C,K and Δ are defined randomly as in section 3.

$$g = 3; \ \ L = L_1 \otimes I; \ \ L_1 = \epsilon(D_{in} - A^T);$$

$$\epsilon = 200 \ \ D_w = diag(\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix})$$

The following figure shows the transient of the rotation around the $y-axis$:

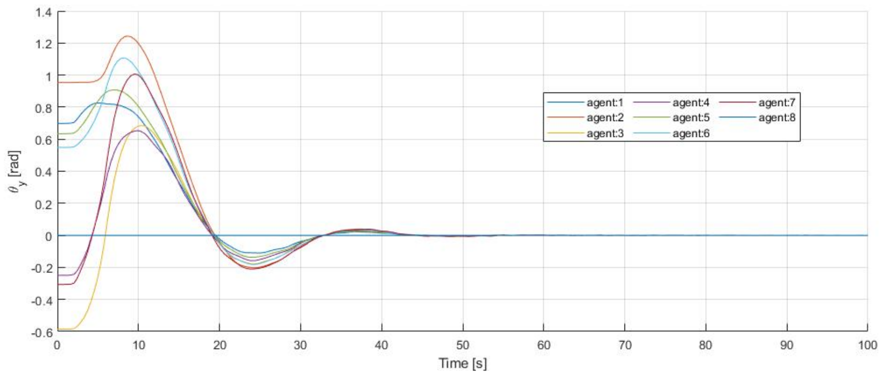


Figure 43: Rotation around y-axis for agents 1 to 8, flexible dynamics

Simulation 2

In this second simulation two cases are presented, in the first case, an undirected topology as in section 3 is used, then the matrix D_w is used to transform it in an undirected one to see if better performances can be achieved. The next figure shows the undirected topology:

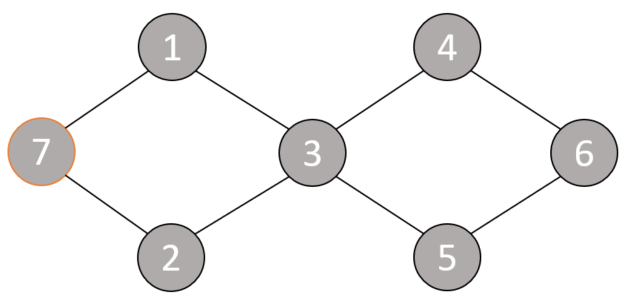


Figure 44: Undirected topology, 7(red) is the leader

The considered parameters are (N = 7):

$$\theta_0 = \begin{bmatrix} rand(3(N-1),1) \\ \theta_f \end{bmatrix}; \quad \dot{\theta}_0 = \begin{bmatrix} rand(3(N-1),1) \\ 0 \\ 0 \\ 0 \end{bmatrix};$$

$$\eta_0 = rand(4N,1); \quad \dot{\eta}_0 = rand(4N,1); \quad \theta_f = \begin{bmatrix} 0.7684 & 0 & 0.224 \end{bmatrix}$$

J,C,K and Δ are defined randomly as in section 3.

$$g = 3; \;\; L = L_1 \otimes I; \;\; L_1 = \epsilon(D_{in} - A^T);$$

$$\epsilon = 200 \;\; D_w = diag(\begin{bmatrix}1 & 1 & 1 & 1 & 1 & 1 & 0\end{bmatrix})$$

The next figures show the transient of the rotation around $y - axis$ and the transient of the flexible modes of all the agents:

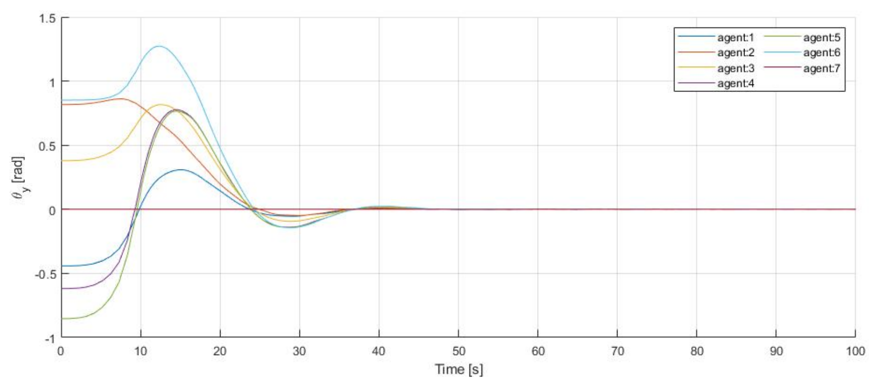


Figure 45: Case 1, y-axis rotations for agents 1 to 7

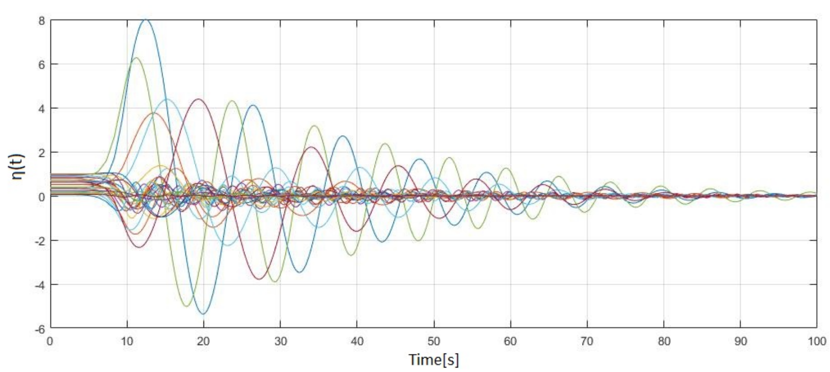


Figure 46: Modal coordinates for all the agents, all components

As expected from theory, angular displacements converge to the Leader, while modal coordinates converge to 0. A measure of the control input signal energy is computed as $u_e = ||u_a||_2$, where u_a is the vector that contains the signal norms of all the input torques for all the agents. In this first case $u_e = 92.4589$. Let's now consider the case where $D_w = diag(\begin{bmatrix}1 & 1 & 2 & 3 & 3 & 4 & 0\end{bmatrix})$ and $\epsilon = 58$:

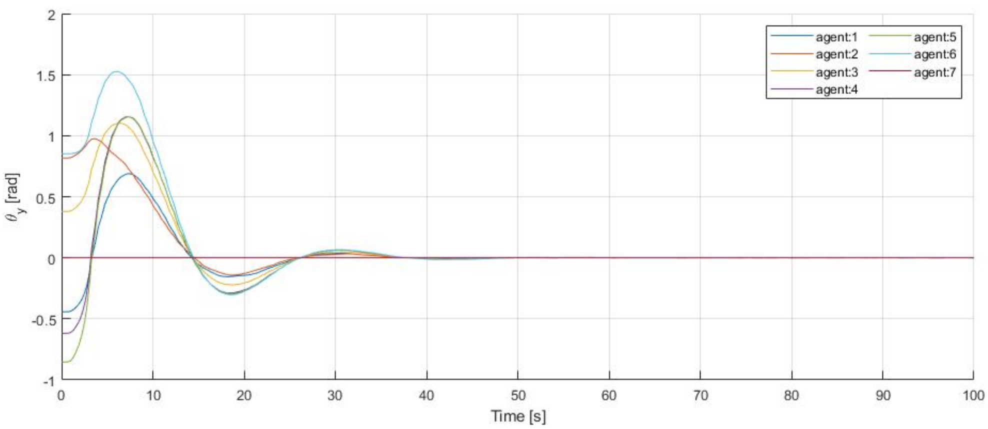


Figure 47: Case 2, y-axis rotations for agents 1 to 7

In this case performances in terms of setlling time are slightly better, but substantially the same, for example considering a tolerance band of ± 0.01 this case is already settled at 40 seconds, while the first case was not. However there is a big difference in terms of control input, since in this case $u_e = 32.8235$, so almost a third of the previous. This can be seen also from the following comparison between the control inputs on the $y - axis$ in the two cases:

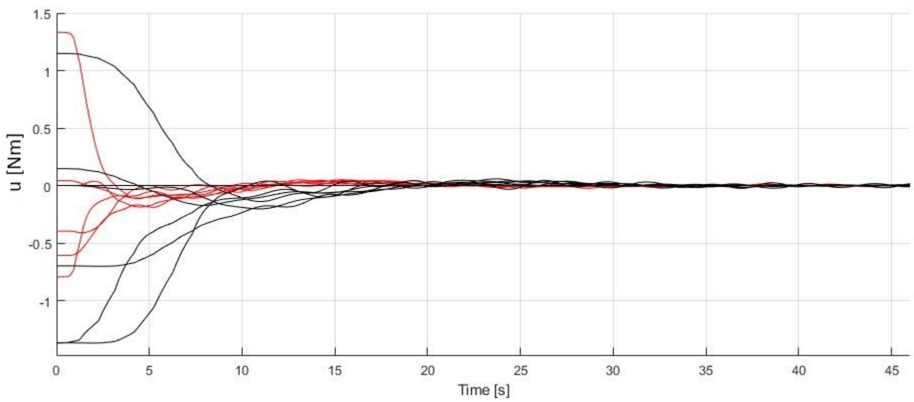


Figure 48: Black: Case 1, Red: Case 2

Notice that the product $D_w L_1$ is the Laplacian associated to a directed graph, in particular it is obtained by splitting the arcs of the previous undirected one and giving each of them a different weight depening on the entries of D_w:

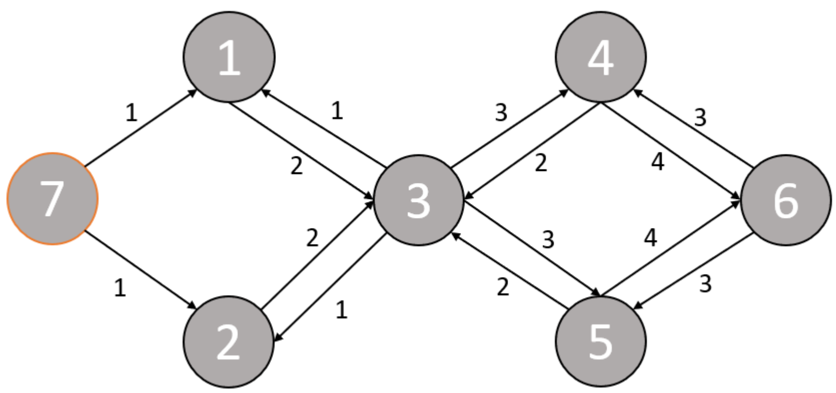


Figure 49: Graph associated to $D_w L_1$

4.5 External disturbances

In this section, a model for external disturbances is shown and added to the simulations. The external torques can be splitted in four different contributions:

$$J\ddot{\theta}(t) + \Delta^T \ddot{\eta}(t) = u(t) - T_{aer} - T_{mag} - T_{rad} - T_{grav}$$
$$\ddot{\theta}(t) = J^{-1}u(t) - J^{-1}\Delta^T \ddot{\eta}(t) - J^{-1}(T_{aer} + T_{mag} + T_{rad} + T_{grav})$$

- **Aerodynamic friction torque**:

  The orbit is considered to be at 300Km, the air density is computed using JB2008. The aerodynamic force is computed as $F_a = \frac{1}{2}\rho S v^2 C_d$ and the corresponding torque as $T_{aer} = r \wedge F_a$ where r is the distance vector between the center of pressure and the center of mass. To get the worst case scenario the center of pressure is set at the border of the spacecraft. To compute v a simulation of the spacrafts‘ orbit is added to the model. For sake of simplicity the orbit is considered the same for all the spacrafts, circular.

- **Gravity gradient torque**:

  The gravity gradient provoke a net torque around the center of mass. This torque can be computed as:

  $$T_{grav} = 3n^2 \begin{bmatrix} (I_{zz} - I_{yy})\phi \\ (I_{zz} - I_{xx})\theta \\ 0 \end{bmatrix}$$

  where $n = \sqrt{\frac{\mu}{a^3}}$, μ: gravitational parameter, a: orbital parameter, I_{xx}, I_{yy}, I_{zz}: Inertia matrix diagonal entries.

- **Solar radiation pressure**:

  Light rays from all celestial bosies carry some momentum, all of them, except the sun, are completely negligible. This produces a net force on the radiation center of pressure, that, if is not coincident with the center of mass, produces a torque. The force can be evaluated as:

  $$F_{rad} = (1 + K)P_s S$$

where K is the reflectivity of the body $(0\div1)$,$P_s = \frac{I_s}{c}$, $I_s = 1400W/m^2$, c = speed of light, S = frontal area. The corresponding torque is:

$$T_{rad} = r \wedge F_{rad}$$

where r is the distance vector between the center of mass and the center of pressure.

- **Magnetic torque**:

  The current loops in the spacecraft interact with the earth's magnetic field and produce a torque, since it is very complex to build an accurate model of the magnetic field, we just consider this disturbance as 10% of all the others. So:

$$T_{mag} = 0.1(T_{aer} + T_{rad} + T_{grav})$$

The **orbit** is considered to be the same for all the agents for sake of simplicity. The employed model is the free 2 body equation:

$$\dot{v} + \frac{\mu}{||r||^3}r = 0$$

where r is the reciprocal position vector (Earth-Spacecraft), μ is the Earth's gravitational parameter and $\dot{v}$ is the linear acceleration. In the following Figures (50-51) some of the time evolutions of these external torques are analyzed, starting from the aerodynamic torque with $C_d = 2.2$:

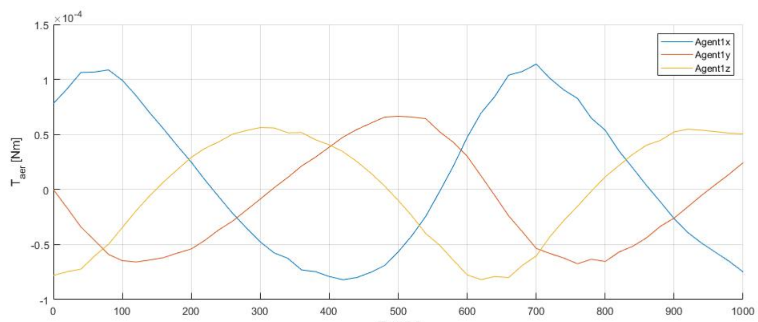


Figure 50: Aerodynamic torque on Agent 1

The aerodynamic torque follows the the coordinates of the orbit with an oscillatory behaviour.

The following Figure (51) shows two different cases of gravity gradient torque, with different Inertia matrices, clearly since the final position of the body has 0 y component, the gravity gradient torque on y converges to 0:

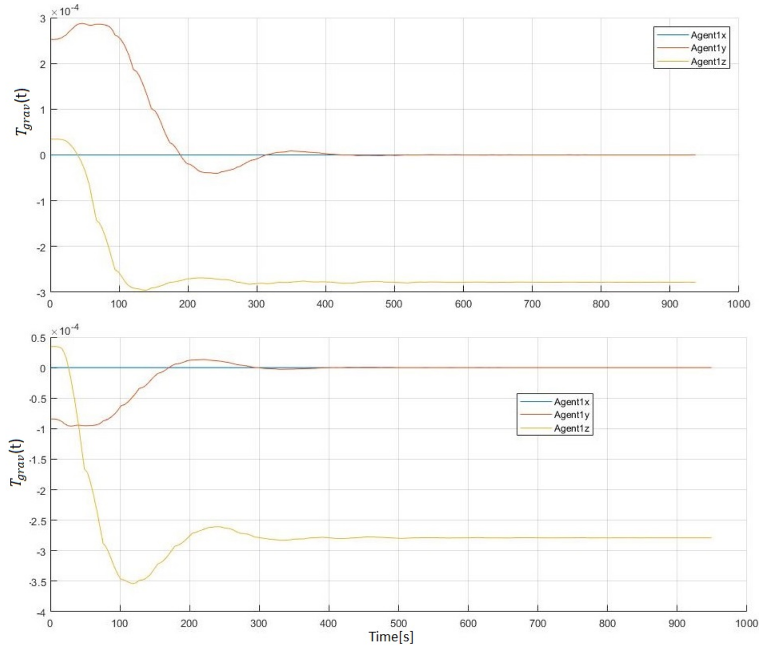


Figure 51: Gravity gradient torque, above $J_{xx} < J_{yy} < J_{zz}$, below $J_{yy} < J_{xx} < J_{zz}$

The solar radiation is in the order of 10^{-7} even in the worst case scenario (maximum distance from the center of pressure), so it is considered completely negligible. Let‘s now analyze the effect on the final angular positions. The following Figure(52) shows the comparison between two cases after a long time, the below with external torques, the above without:

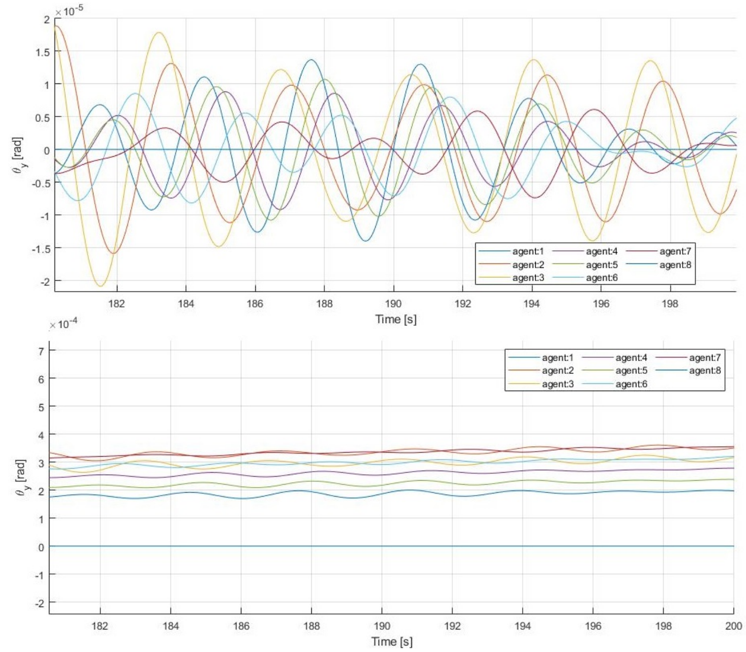


Figure 52: Positions settlings

In the second case the position trend is to settle to a final position around 10^{-4}, while in the first oscillations are still decreasing. So in the presence of external torques the control law is not able to guarantee an arbitrary precision on the position just waiting enough for it to settle. This problem is an example of possible future works.

5 Failure Management

In this section possible failures of one or more of the agents is taken in consideration, more precisely a failure that affects the capability of communication and, as a consequence, the graph topology. Let‘s consider the following example where 7 is the leader:

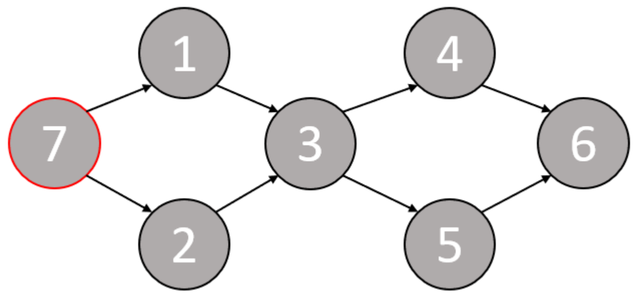


Figure 53: Topology example

If agent 3 stops to communicate with the others, the graph splits in 3 subgraphs each of them independent from the others. So we expect each of the subgraphs to evolve with initial conditions equal to the state that they reached at the instant of failure. The following Figure (55) shows the modified topology and the tansient of rotation angles around the $z-axis$ if agent 3 fails after 20 seconds:

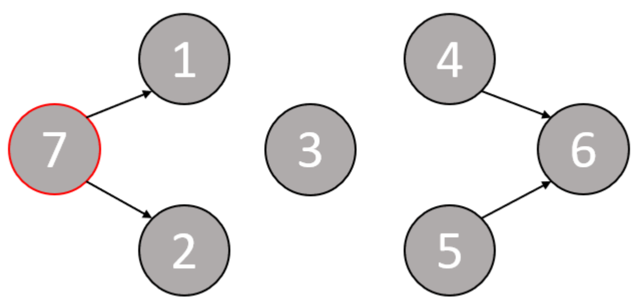


Figure 54: Modified topology

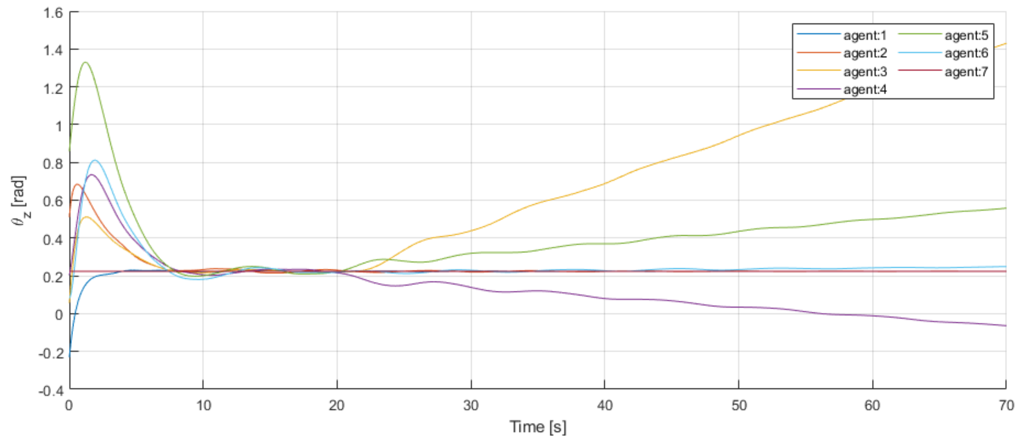


Figure 55: Transient with agent 3 failing at 20s

As expected, after 20 seconds, agents 1 and 2 still reach consensus, the others will not since there is no direct path starting from the leader. In particular agent 3 is completely disconnected, so it will follow its own dynamics, while agents 4,5 and 6 create a subgraph with two sources and agent 6 trend is to get an average position between 4 and 5.

These kind of situation can be faced with different strategies depending on how new communications can be established among the working agents (adding new edges to the graph). A first approach can be to recompute thetopology among the agents that still work in a way that there exist a directed path from the leader to any agent adding the least number of edges. A possible algorithm (1) could be:

1. Recompute the graph Laplacian removing the broken agent(s).
2. Check if $rank(L) = 1$, if so then a directed path exists and the algorithm stops, otherwise go to step 3.
3. Modify a 0-row that is not the last (corresponding to the leader) so that the diagonal element $L(i,i) = 1$ and $L(i, i \pm 1) = -1$. To chose between $i \pm 1$, select the one such that the $j - th$ agent with $j = i \pm 1$ belongs to a different subgraph with respect to the $i - th$, if possible.
4. Repeat from 2

In this way the resulting Laplacian corresponds to a new topology that guarantees consensus (in wich the broken agent(s) can be reinserted or not). Taking as example the case in figure 42 and 43, we have the original Laplacian L_0 and the Laplacian after the failure L_1 that are:

$$L_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 \\ -1 & -1 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; \quad L_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Applying the algorithm, we get a new Laplacian that is (reinserting agent 3 after the modifications):

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

that corresponds to:

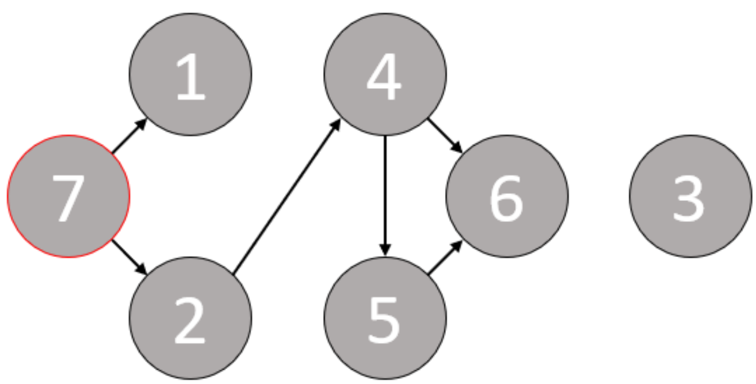


Figure 56: Recomputed topology, algorithm 1

A simulation is now shown in which at 20 seconds agent 3 breaks, and at 25 seconds the modified topology is replaced:

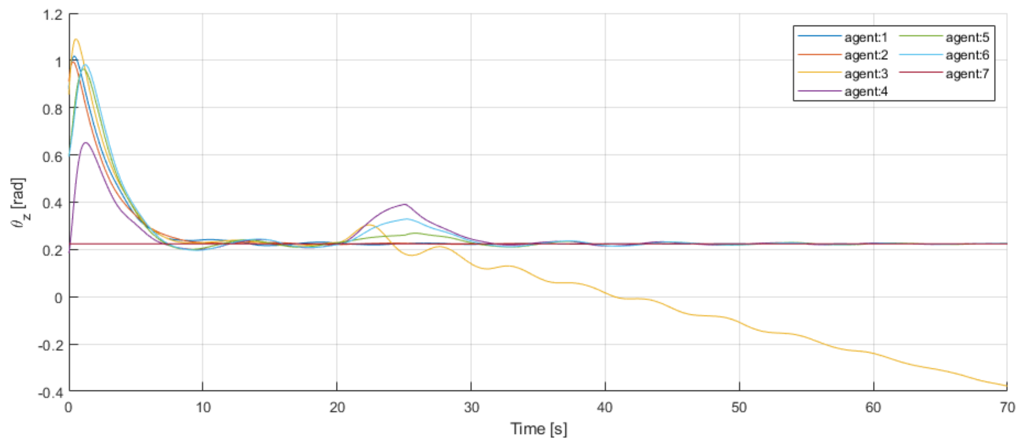


Figure 57: Angular displacement around $z - axis$, replaced topology at 25s

Agent 3 diverge as a ramp with a slope equal to the instantaneous velocity at 20 secods (the remaining oscillations are due to the flexible parts), while other agents reach consensus.

A second approach can be to set new leaders so that each subgraph has one. In particular each subgraph can set as leader the agent that was connected with an agent now belonging to a different subgraph. In case this happens for more than one, select the one that has the minimum gap in therms of position and/or velocity using the last information available. So a possible algorithm (2) can be:

1. Identify all the independent subgraphs
2. Check if a subgraph has a leader, if yes, move to point 4, if not move to point 3
3. Among the agent(s) that were connected to an agent that now belongs to a different subgraph, choose the one that has the minimum displacement and/or velocity gap (in terms of norm) with respect to the neighbours and set it as a leader.

4. Check if the subgraph topology is capable of reaching consensus, modify it as in algorithm 1 (obviously not allowing edges across the subgraphs).

5. Step to another subgraph if present, otherwise stop

For example, referring to the same starting topology (figure 43), we have 3 independent subgraphs, in the left one agent 7 is the original leader and the topology ha a directed path to agents 1 and 2. The subgraph on the right has two agents previously connected to agent 3, but agent 4 have a smaller position gap at the failure instant ($||\theta_4(20) - \theta_3(20)||$), so it is set as a leader. Now algorithm 1 is used to guarantee the existence of a directed path from agent 4 to 5 and 6. The modified subgraph Laplacian is:

$$L_{sub3} = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ -1 & -1 & 2 \end{bmatrix} \quad or \quad L_{sub3} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ -1 & -1 & 2 \end{bmatrix}$$

Obviously the subgraph containing only agent 3, it becomes the leader of itself. The new topology is the following:

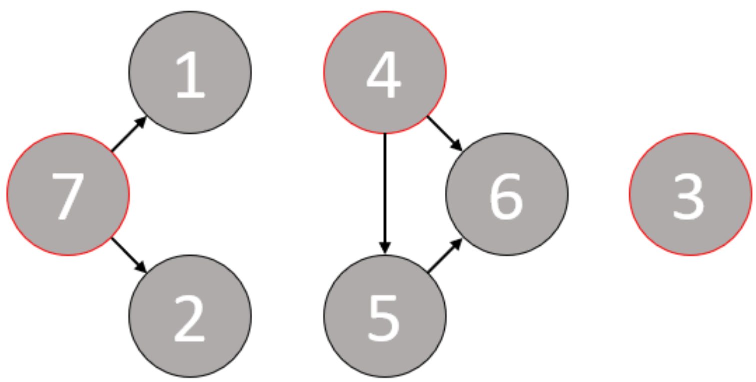


Figure 58: Recomputed topology, algorithm 2

In the next figure the angular displacement transient is shown:

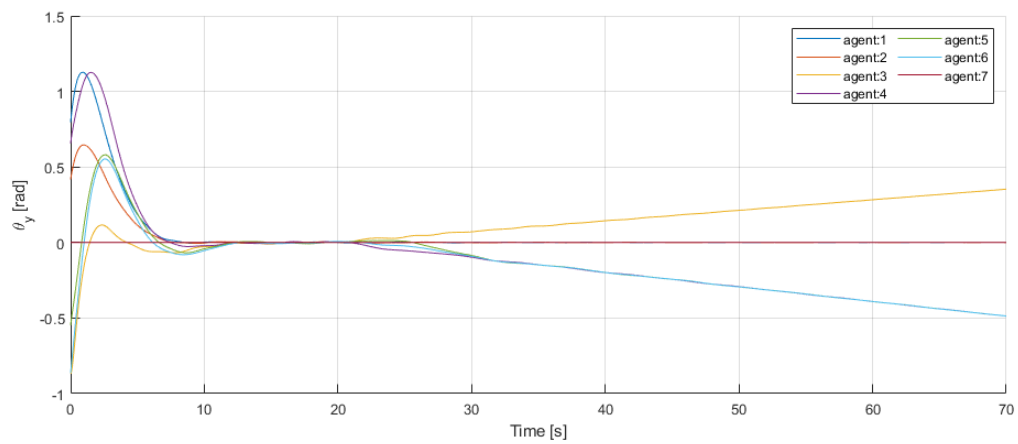


Figure 59: Rotations transient, topology switched at 25s, algorithm 2

Clearly in this latter case, the subgraph of 4,5 and 6 diverges following the velocity of 4, so it is less useful. However a third approach can be used with a slight modification to the second one:

1. Use algorithm 2

2. Put an edge from the original leader to each of the new ones, except for the broken agents.

In this way the topology is divided in groups of agents, and the leader of each subgroup is connected to the original leader. It is clear that these strategies must be chosen depending on which communications can be established between the agents, if any communication is possible, then there always exist a simple topology that connects the leader with each agent with a directed edge. So in general the strategy that brings the smallest number of changes in the graph is preferred. In this latter algorithm, the new subgroups leaders should be chosen also depending on the capability of receiving information from the original one.

The following two Figures (60-61) show the modified topology in this last case and the correspondent transient:

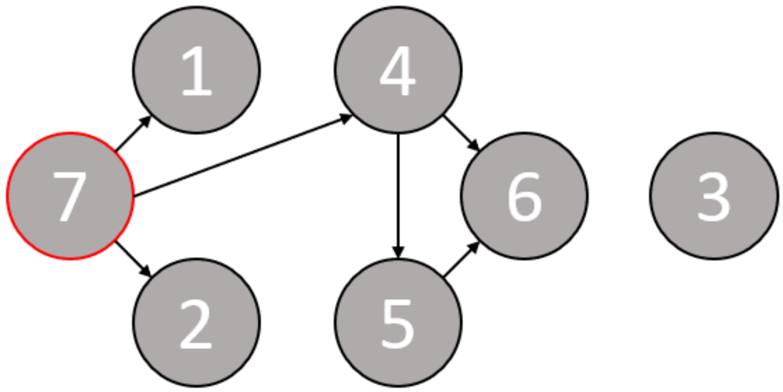


Figure 60: Recomputed topology, algorithm 3

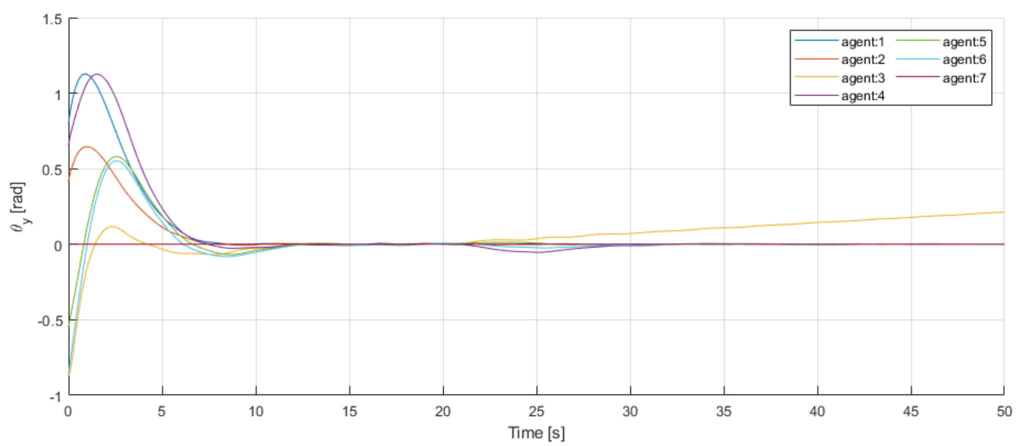


Figure 61: Rotations transient, topology switched at 25s, algorithm 3

These three strategies can be chosen also depending on how the graph is splitted after the failure and on subobjectives of the MAS, for example to minimize the number of communications.

6 Conclusions

The objective of this work was to study and improve the problem of controlling a Multi-Agent System in the aerospace context. After the introductive chapters, section 3 is a resume of some control strategies over undirected communications topologies with some numerical simulations. In particular a set of satellites with flexible appendages, considered both able or not to measure angular velocity, was considered taking consensus as final objective.

In section 4 the control laws in section 3 are extended to the directed graph topologies, in order to reduce the number of communications needed to reach cnsensus or to improve control performances. Sufficient parameters conditions to reach consensus over directed connected topologies with a source (leader) are given and some different ways to tune them are shown. Moreover, a way to transform undirected topologies into directed ones to get a strong improvement in terms of control input energy, leaving unaltered the transient charachteristics, is discussed. To get more precise simulations, a model of the most relevant external disturbances in Low-Earth Orbit is proposed and included. At last some sufficient conditions for the discrete time equivalent of the system to converge are provided in order to implement a differentiator to estimate the angular velocity if not measurable.

In section 5 the effects of the possible failure of an agent are disucssed with some algorithms to overcome the changes, trying to minimize the difference between the new and the original topology. Then some simulations with switching graphs are given. Hence, the overall objective of the thesis is reached.

As possible future works many aspects can be studied, for example find a control law that does not need a velocity measure or estimate to guarantee consensus over directed topologies, generalize section 4 to an arbirtary number of leaders, study the beahviour of the system in large scales or get an higher number of parameters comparisons to find the best combinations in different real cases.

7 References

[1] *"Robust Consensus of Second-Order Heterogeneous Multi-Agent Systems via Dynamic Interaction"*, Capello, Fujisaki, 2020.

[2] *"Second-Order Multi-Agent Consensus via Dynamic Displacement Interaction"*,Capello, Fujisaki.

[3] *"Dynamical group consensus of heterogenous multi-agent systems with input time delays"*,Guoguang Wen, Yongguang Yu, Zhaoxia Peng, Hu Wang, 2015.

[4] *"Multi-Agent Cooperative Control Consensus"*, Gulzar, Rizvi, Javed, Munir, 2018.

[5] *"Second-order consensus for nonlinear leader-following multi-agent systems via dynamic output feedback control"*,Haibo Du, Yingying Cheng, Yigang He, Ruting Jia, 2015.

[6] *"On the Role of the Eigenprojector of the Laplacian Matrix for Reaching Consensus in Multiagent Second-Order Systems"*, R. P. Agaev, 2019.

[7] *"The Forest Consensus Theorem"*, Pavel Chebotarev, Rafig Agaev, 2014.

[8] *"Some necessary and sufficient conditions for second-order consensus in multi-agent dynamical systems"*,W. Yu, G. Chen, M. Cao, 2010.

[9] *"Network Systems"* Francesco Bullo, 2019.

[10] *"Graphs and Matrices"*, R.B.Bapat, 2010.

[11] *"The Matrix Reference Manual"*,Mike Brookes, 1998-2021

[12] *"Laplacian Eigenvectors of Graphs"*,Türker BiyikoğuJosef Leydold-Peter F. Stadler, 2007.

[13] *"Sistemi Robotici Distribuiti, cap. 2"*, L. Pallottino, 2015.

[14] *"Satellite swarm survey and new conceptual design for Earth observation applications"*,A. Farrag, Saed Othman, Tarek Mahmoud, Ahmed Y. EL Raffiei, 2021.

[15] *"New Chattering Attenuation Sliding Mode Control of a Flexible Spacecraft"*, Hamidreza Nemat, Shinji Hokamoto, 2013.

[16] *"Output Stabilization of Flexible Spacecraft with Active Vibration Suppression"*, S. Di Gennaro, 2003.

[17] *"Active damping control of flexible appendages for spacecraft"*,Yizhe Zhang Xin Guan, 2018.

[18] *"Flexible Spacecraft Dynamics, Control and Guidance"*,Giovanni Campolo, 2015

[19] *"Spacecraft dynamics snd control: an introduction"*,Anton H.J. de Ruiter, Christopher J. Damaren and James R. Forbes, 2013.

[20] *"Orbit Perturbations"*, Walter U, 2018, In: Astronautics. Springer.

[21] *"Spacecraft Dynamics and Control"*, Marcel Sidi, 2011.

[22] *An Introduction to MultiAgent Systems*, Michael Wooldridge, 2001.

[23] *Multi-Agent Systems: A Survey*, A.Dorri, S. Kanhere, R. Jurdak, 2018.

[24] *A taxonomy for multi-agent robotics*,G. Dudek, M. R. Jenkin, E. Milios and D. Wilkes, Auto. Robot., vol. 3, no. 4, pp. 375-397, 1996.

[25] *A cooperative multi-agent robotics system: Design and modelling*,C. G. Cena, P. F. Cardenas, R. S. Pazmino, L. Puglisi and R. A. Santonja, Expert Syst. Appl., vol. 40, no. 12, pp. 4737-4748, 2013.

[26] *Consensus of linear multi-agent systems subject to communication delays and switching networks*, Maobin Lu Lu Liu, 2017.

[27] *Bounded Consensus Algorithms for Multi-Agent Systems in Directed Networks*, Haibo Du Shihua Li Shihong Ding, 2012.

[28] *Leader-following second-order consensus in multi-agent systems with sampled data via pinning control*, Bo Zhou, Xiaofeng Liao, 2014.

[29] *Leader-Following Coordination of Heterogeneous Multi-Agent Systems via Displacement Feedback*, Rodolfo Pietrasanta, Elisa Capello, Yasumasa Fujisaki, 2021.

7.1 Figures

[1] Starling1 Swarm Technology Demonstration (Sanchez et al., 2018)

[2] JPL - Mars Cube One (MarCO) at: www.jpl.nasa.gov