Advanced methodologies of Automatic Control in biomedical applications: two notable cases

Candidate

Maria Laura Aceto
ID number 1227722

Thesis Advisor

Prof. Paolo Di Giamberardino

March 2021

Thesis defended on 13/07/2021
in front of a Board of Examiners composed by:

Giuseppe Baselli
Stefano Panzieri
Fabio Tardella (chairman)

Advanced methodologies of Automatic Control in biomedical applications: two notable cases
Ph.D. thesis. Sapienza – University of Rome

This thesis has been typeset by LaTeX and the Sapthesis class.

Author's email: marialaura.aceto@uniroma1.it

Abstract

Interesting and useful applications of Automatic Control can be found/discovered in many sectors: one of them is the medical, or better the biomedical area.

This work focuses on two notable cases of applications of Automatic Control to biomedical field: the first one is linked to application of multi-agents systems study to epidemiological networks, while the second one deals with the use of a real micro-robotic device.

The interest in epidemiological networks is linked to the spread of Covid-19 in Italy and around the world in 2020. The research, carried out during PhD, was already focused on the study of dynamical networks systems but this event led to a focus on applications to real cases of epidemiological networks.

The second application arises from the possibility of contributing to find a solution to a real problem, that, if solved, can lead to relevant applications, in biomedical field. Indeed, introduction of robotics, and even more micro-robotics, has led to huge innovations in the medical field, that can be improved and integrated over time.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This Thesis reports an overview on research activities carried out by the author within the XXXII PhD program in ABRO (Automatic Control, Bioengineering and Operation Research), Curriculum Automatic Control, undertaken at the University of Rome "La Sapienza", Department of Computer, Control and Management Engineering "Antonio Ruberti".

1.1 Motivation and Contributions

In last century, advances in studies, have upset the techniques used in medicine and they have contributed to the birth of new branches of study and research, i.e. biomedical engineering. Advances in science and technology are also reflected in medical fields: biomedical applications are used for diagnosis, surgery, monitoring of parameters and vital signs, medical cares, ... In last decades, biomedical applications have gained an increased importance. Methodologies from different engineering and scientific fields are applied to biomedical studies and applications; in particular, methodologies from Automatic Control are used to advance in sectors above mentioned ([1], [2], [3], [4]). There are still many applications on which studies can be done and there are many techniques, not strictly thought out to biomedical field, that can be explored in the future research.

In different medical areas, there can be found/discovered interesting and useful applications of Automatic Control; this work focuses on two remarkable ones: the first one is linked to application of multi-agents systems study to epidemiological networks, while the second one deals with the use of a real micro-robotic device.

The interest in epidemiological networks is linked to the spread of Covid-19 in Italy and around the world in 2020. The research, carried out during PhD, was already focused on the study of dynamical networks systems but this event led to a focus on applications to real cases of epidemiological networks.

The second application arises from the possibility of contributing to find a solution to a real problem, that, if solved, can lead to relevant applications, in biomedical field. Indeed, introduction of robotics, and even more micro-robotics, has led to huge innovations in the medical field, that can be improved and integrated over time.

Some of the results of the research, done within the PhD program and exposed in this thesis, can be found in:

- P. Di Giamberardino, M. L. Aceto, O. Giannini, and M. Verotti, *"Recursive least squares filtering algorithms for on-line viscoelastic characterization of biosamples"*, in Actuators, vol. 7, no. 4. Multidisciplinary Digital Publishing Institute, 2018, p. 74

- P. Di Giamberardino, M. L. Aceto, O. Giannini, and M. Verotti, *"Dynamic estimation of visco-elastic mechanical characteristics of biological samples under micro manipulation"** in ICINCO (2), 2018, pp. 513–520

- M.L. Aceto, P. Di Giamberardino, *"Multiagent dynamical networks for virus spread modelling*, submitted to ICINCO 2021

*This work has been awarded as "Best Short Paper/Poster Award", in Area of Robotics and Automation, at ICINCO 2018 (International Conference on Informatics in Control, Automation and Robotics).

1.2   Outline of the Thesis

The thesis is organized as follows:

- Chapter 1 provides this brief introduction;

- Chapter 2 gives an overview of multi-agent systems and new techniques for topological variations detection;

- Chapter 3 is focused on multi-agent epidemiological networks;

- Chapter 4 describes the real micro-manipulator and its modelling;

- Chapter 5 exposes the results obtained with the estimation technique applied to the micro-gripper.

Chapter 2

Multi-agent systems

This Chapter gives an overview of multi-agent systems and new techniques for topological variations detection.

2.1 Introduction

Network/multi-agent systems study has gained a great relevance in many fields of recent scientific research, including that of control theory. The increased interest is due to large number of application fields of this kind of systems. Applications can be found in power grids, telecommunication, security, medicine, transportation, biology, Industry 4.0, home automation, Internet of Thinks [5, 6, 7, 8]. A classical application in biomedical fields can be found in studies and use of sensors networks [9] for diagnostic or prevention.

One of the branches of mathematics, from which it is started to study and to model the connections between objects, is the graph theory, firstly introduce by L. Euler in [10] (for further details refer to [11]). A network/multi-agent system can be viewed as a graph in which each node is an agent, with its own dynamics, and the interaction between two nodes is represented as a link/edge. An interaction can represent a connection between nodes, exchange of data, flows of information, materials and so on, and it can be mono-directional or bi-directional. In general, according to persistence of its edges, a network can be static (the edges are fixed a priori), dynamic (the edges can vary, in a deterministic way, according to the status of some network agents) or random (the edges can vary according to a probability distribution). The connections between nodes and links is the so-called topology of the network, and studying topology variations is one of the topics to be addressed when dealing with dynamical network systems. This aspect can helps in many research fields, among which we can find consensus networks [12], formation control [13], flocking control [14].

Changes in topology can be caused both by failure, in nodes or link,s and by intentional disconnections, in some cases, aimed to isolate one ore more elements in the network itself. Usually topological variations, caused by failures, are not desirables and can lead to a misbehaviour of the network (i.e. in power networks [15]). Intentionally disconnections can be introduced for several reasons, i.e. in order to isolate one or more nodes, block the communication between agents, in case it's needed to

split the network in two or more subnetworks (i.e in epidemic control [16]).

In literature, they can be found different approaches and algorithm related to detectability of topology variations; i.e, in [17] it's proposed a method based on the observation of jump discontinuities in the output derivatives, in [18] it's proposed a method based on Kalman Filter, in [19] the detection is done considering cross power spectral densities changes.

2.2   Discernibility

In [20], [21] and [22], the authors propose an approach for detection of topological variations in dynamical network systems based on analysis tools from switching systems theory, in particular the concept of discernibility/ distinguishability/ mode-observability [23]. This approach aims to identify if it's possible to detect a network topology variation, even before the construction of an identification algorithm.

The basic idea is that the network can be viewed as a multi-mode/ switching system: a network with different topology, caused by links/nodes disconnections, can be viewed as a system in different operational mode. When it is in different operational mode, the network can rise to different dynamics (in this case the operational modes are said to be discernible) or to same dynamics (in this case the operational modes are said to be indiscernible). The works deal with networks composed by homogeneous continuous-time linear systems, with a weighted and undirected communication graph, and they identify necessary and sufficient conditions, on the eigenspace components related to the nominal network topology, for discernibility for this kind of multi-agent networks.

In [24], [25] and [26], this approach is extended focusing on consensus networks. In [24], the author extends the analysis to edge disconnection problem in case of discrete-time multi-agent consensus networks. The work shows that, if the communication graph is connected, edge disconnections, not affecting connectedness of the network, do not impact the final consensus value reached by the agents. In addition, it identifies necessary and sufficient conditions for discernibility on the original state matrix and on its eigenvectors; these conditions are equivalent if the state of some or all agents are known. Unlike, in case of node disconnections, final consensus value changes [25], but it can be restored with a distributed control action. In [26], in addition to edge disconnection problem in a discrete-time multi-agent consensus network, the authors also investigate how to identify, in a finite number of steps, the exact edge that got disconnected. They derive necessary and sufficient conditions for these problems, whether the states of some and all agents are known.

The aforementioned works, on detection of topological variations, deal with multi-agent networks, in which the systems connected are homogeneous and all their parameters as well as the weights among each nodes are known.

In [27], the study is extended to networks of not homogeneous nodes, with dynamics that obey to linear Differential–Algebraic Equations (DAEs) instead of Ordinary-Differential Equations (ODEs). The authors show that the properties of the nominal network configuration can be used to identify possible existence of indiscernible topological changes.

In [28], the study is focused on multi-agent networks, in which the weights among

each nodes are not known. The authors derive necessary and sufficient conditions on the graph for generic detectability and isolation.

In this work, we extend the results of detecting topological variations to general case of multi-agent networks in which:

- the dynamics of each node is linear;

- the nodes are not homogeneous;

- the graph is directed (it can exists edge from not i and j, without edge from node j and i);

- the connection between nodes are modelled as incoming/ outcoming flows.

Here, we briefly recall the model and approach defined in [21]. Consider a multi-agent network, composed by N nodes, each one represented by linear system. As stated previously, the network can be viewed as a graph $G = (V, E)$, with a predefined topology, where the set of nodes is represented by $V := \{1, 2, ..., N\}$ and the set of link is represented by $E \subseteq (V \times V)$.

The dynamics of the ith node is the following:

$$\begin{cases} \dot{x}_i & = Ax_i + B \sum\limits_{j \in N_i} w_{ij}(x_j - x_i) \\ y_i & = Cx_i \end{cases} \quad \text{i = 1, 2, ..., N} \quad (2.1)$$

where:

- $x_i/x_j \in \mathbb{R}^n$ are, respectively, the state vector of node i and j;

- $y_i \in \mathbb{R}^m$ is the output vector of node i;

- the matrices $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times n}$ and $C \in \mathbb{R}^{n \times m}$ are state transition, input and output matrices of node i;

- $w_{ij} \in \mathbb{R}^+ \cup \{0\}$ represents the coupling strength between node i and j, $w_{ij} \neq 0$ if node j is in the set N_i of neighbours of node i, so if $(j, i) \in E$. Note that, being the graph undirected, all edges are bidirectional, and, consequently, $w_{ij} = w_{ji}$.

Recalling the definition of Laplacian matrix L of the graph G:

$$L = [l_{ij}] \in \mathbb{R}^{N \times N} \quad (2.2)$$

where

$$l_{ij} := \begin{cases} -w_{ij} & j \neq i \\ \sum\limits_{k \in N_i} w_{ik} & j = i \end{cases}$$

the whole network state and output dynamics can be written as:

$$\begin{cases} \dot{x} & = \Phi x \\ y & = Hx \end{cases} \quad (2.3)$$

where:

- $x \in \mathbb{R}^{Nn}$ is the aggregated state vector of network;

- $y \in \mathbb{R}^{Nm}$ is the aggregated output vector of the network;

- $\Phi \in \mathbb{R}^{Nn \times Nn}$ is the transition matrix of the network defined as

$$\Phi := (I_N \otimes A) - (L \otimes B)$$

- $H \in \mathbb{R}^{Nm \times Nn}$ is the output matrix of the network, defined as $H := (I_N \otimes C)$.

Note that, in case of undirected graph, the Laplacian matrix (2.2) is symmetric.
The network, in nominal conditions, namely in predefined topology, can be fully characterized by the pair (G, Φ). In case of topological variations, a new graph $\bar{G}$ representing the updated status of the connections, a new set of edges $\bar{E}$ and Laplacian matrix $\bar{L}$ shall be considered. If the variation is caused by link (i, j) disconnection, new set of edges will be $\bar{E} = E \setminus (i, j)$, while if it's caused by node i disconnection $\bar{E} = E \setminus (i, j); \forall j \in N_i$. In both cases, whole network state dynamics becomes:

$$\begin{cases} \dot{\bar{x}} & = \bar{\Phi}\bar{x} \\ \bar{y} & = \bar{H}\bar{x} \end{cases} \tag{2.4}$$

where $\bar{H}$ can be equal or different from H.
The modified network can, therefore, be characterized by the pair $(\bar{G}, \bar{\Phi})$.
The basic idea is to consider the system obtained from the parallel interconnection of nominal and modified network:

$$\begin{cases} \dot{\tilde{x}} & = \Delta \tilde{x} \\ \tilde{y} & = \Gamma \tilde{x} \end{cases} \tag{2.5}$$

where:

- $\tilde{x} \in \mathbb{R}^{2Nn}$ is the state vector of the interconnected system;

- $\tilde{y} \in \mathbb{R}^{2Nm}$ is the output vector of the interconnected system;

- $\Delta \in \mathbb{R}^{2Nn \times 2Nn}$ is the state-transition matrix of the interconnected system defined as $\Delta = diag(\Phi, \bar{\Phi})$;

- $\Gamma \in \mathbb{R}^{2Nm \times 2Nm}$ is the output matrix of the interconnected system defined as $\Gamma = [I \quad -I]$ with $I \in \mathbb{R}^{Nm \times Nm}$.

The following definition should be recalled: two LTI systems of order n, with state transition matrices Φ and $\bar{\Phi}$, are said to be *indiscernible* with the respect to state $x_0 \in \mathbb{C}^n$, if $x(t) = \bar{x}(t)$ for all $t \geq 0$. In linear case, this can be specified as $e^{\Phi t} x_0 = e^{\bar{\Phi} t} x_0$ for all $t \geq 0$.
Taking into account definition of indiscernible state, and the interconnected system dynamics (2.5), the discernibility of state x and $\bar{x}$ can be considered equivalent to the observability of the pair (Δ, Γ).
The main results in [21], not strictly related to particular node dynamics in (2.1), are the followings:

- it's sufficient to verify if common eigenvalues to Φ and $\bar{\Phi}$, with common eigenvectors, exist, in order to verify the existence of not-null indiscernible states [ref to Corollary 1 in [21]]. If Φ is diagonalizable, the set of all complex indiscernible states is:

$$I_{\mathbb{C}} = \{0\} \oplus (V(\lambda) \cap \bar{V}(\lambda)); \;\; \lambda \in (spec(\Phi) \cap \bar{\Phi})$$

where $V(\lambda)$ and $V(\lambda)$ are, respectively, the eigenspace of $\lambda \in spec(\Phi)$ and the eigenspace of $\lambda \in spec(\bar{\Phi})$.

- the generalized eigenvectors and the corresponding Jordan chains shall be considered in order to fully determine the set of indiscernible states, if Φ is not diagonalizable [ref to Proposition 1 in [21]]. In this case, the set of all complex indiscernible states is:

$$I_{\mathbb{C}} = \{0\} \oplus span(W(\lambda); \;\; \lambda \in (spec(\Phi) \cap \bar{\Phi})$$

where, for each $\lambda \in (spec(\Phi) \cap \bar{\Phi})$, $W(\lambda)$ is a matrix whose columns form an orthonormal basis of the set of all the generalized eigenvalues $x \in V_{\nu(\lambda)}(\lambda)$ satisfying the condition $(\Phi - \lambda I)^i x = (\bar{\Phi} - \lambda I)^i x$, for $i = 1, ..., max(\nu(\lambda), \bar{\nu}(\lambda))$, with $\nu(\lambda)$ corresponds to the size of the largest Jordan block associated with λ, common eigenvalue to Φ and $\bar{\Phi}$.

The other results of interest (Proposition 2/Theorems 1 - 2) show the conditions under which a node or link disconnection can be detected. They refer to a specific structure of $\Phi - \bar{\Phi}$ and shall be customized, i.e. in case nodes with different dynamics are considered (refers to Section 3.C of just mentioned work for generalizations of these results). It shall be noted that, taking into account the specific node dynamics, if controllability of the pair (A, B) holds, the existence of not-null indiscernible states via common eigenvalues to Φ and $\bar{\Phi}$, with common eigenvectors, is equivalent to verification of common eigenvalues to L and $\bar{L}$, with common eigenvectors (that can be checked only by looking at the eigenspace of nominal Laplacian L).

2.3 Multi-agent dynamical networks with flows

The dynamics of node i in (2.1) takes into account the coupling between node i itself and its neighbours, considering that its dynamics is influenced by the difference of its neighbours states with respect to its state. We now consider a dynamical network in which each node dynamics is influenced by the flows between itself and its neighbours.

Consider N nodes, represented as linear systems, with the following dynamics:

$$\begin{cases} \dot{x}_i & = A_i x_i + B_i u_i \\ y_i & = C_i x_i \end{cases} \quad \text{i} = 1, 2, ..., \text{N} \tag{2.6}$$

where:

- $x_i / x_j \in \mathbb{R}^n$ are, respectively, the state vector of node i and j;

- $u_i \in \mathbb{R}^n$ is the input vector of node i;

- $y_i \in \mathbb{R}^m$ is the output vector of node i;

- $A_i \in \mathbb{R}^{n \times n}$ is the state-transition matrix of node i;

- matrices $B_i \in \mathbb{R}^{n \times n}$ and $C_i \in \mathbb{R}^{m \times n}$ are, respectively, input and output matrices of node i.

We now connect these nodes to each other, considering the possibility to have flows between them: each node can have input/ output flows from /to its neighbours. For each node i, we model these flows as the input acting on it:

$$u_i = \underbrace{\sum_{j \in N_i} w_{ij} x_j}_{\text{incoming flows}} - \underbrace{\sum_{j \in N_i} w_{ji} x_i}_{\text{outcoming flows}} \tag{2.7}$$

where the first part of (2.7) model the incoming flows from all nodes $j \in N_i$ to node i, while the latter the outcoming ones from node i to nodes $j \in N_i$.
In (2.7), the value w_{ij}:

- is a non negative real value $w_{ij} \in \mathbb{R}^+ \cup \{0\}$;

- represents the weight of the flow from node j to node i;

- $w_{ij} = 0$ in case there is no flow from node j to node i;

- in general, in case of directed graph, w_{ij} is different from w_{ji}: you can have a flow from node j to node i without flow from node i and node j and viceversa.

In Figure 2.1, it's depicted the connection of node i and j with the corresponding flows.
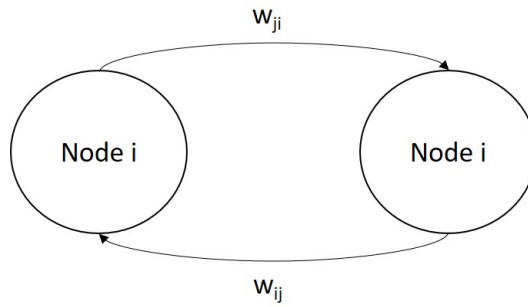

Figure 2.1. Flows between nodes i and j

Considering the input (2.7), the dynamics of each node i becomes:

$$\begin{cases} \dot{x}_i &= A_i x_i + B_i \sum_{j \in N_i} w_{ij} x_j - B_i \sum_{j \in N_i} w_{ji} x_i \\ y_i &= C_i x_i \end{cases} \quad \text{i} = 1, 2, ..., \text{N} \tag{2.8}$$

Taking into account that weights of the flows are modelled with values w_{ij} and dynamics of each component of state x_i is influenced by flows coming from the same component of states of its neighbours, it is reasonable to consider matrix B_i as identity matrix of dimension n.

The state dynamics in (2.8) can be rewritten in the same form of (2.1) as:

$$\begin{cases} \dot{x}_i & = \bar{A}_i x_i + B_i \sum_{j \in N_i} w_{ij}(x_j - x_i) \\ y_i & = C_i x_i \end{cases} \quad \text{i} = 1, 2, ..., \text{N} \quad (2.9)$$

where:

$$\bar{A}_i = A_i + B_i \sum_{j \in N_i} w_{ij} - B_i \sum_{j \in N_i} w_{ji}$$

Considering the node dynamics in (2.8), the whole network state and output dynamics, as in (2.3), can be written as:

$$\begin{cases} \dot{x} & = \Delta x \\ y & = Hx \end{cases} \quad (2.10)$$

where:

- $x \in \mathbb{R}^{Nn}$ is the aggregated state vector of network;

- $y \in \mathbb{R}^{Nm}$ is the aggregated output vector of the network;

- $\Delta \in \mathbb{R}^{Nn \times Nn}$ is the aggregated transition matrix defined as $\Delta = \Phi + \Psi$;

- $\Phi \in \mathbb{R}^{Nn \times Nn}$ is a block diagonal matrix defined as

$$\Phi = diag\{A_i\} \quad (2.11)$$

- $\Psi \in \mathbb{R}^{Nn \times Nn}$ is a block matrix defined as

$$\Psi = [\psi_{ik}] \quad (2.12)$$

where:

$$\psi_{ik} := \begin{cases} -B_i \sum_{j \in N_i} w_{ji} & i = k \\ B_i w_{ik} & i \neq k \end{cases} \quad (2.13)$$

in a more extended form:

$$\Psi = \begin{pmatrix} -B_1 \sum_{j \in N_1} w_{ji} & \cdots & B_1 w_{1,N} \\ B_2 w_{2,1} & \cdots & B_2 w_{2,N} \\ \vdots & \ddots & \vdots \\ B_N w_{N,1} & \cdots & -B_N \sum_{j \in N_N} w_{jN} \end{pmatrix} \quad (2.14)$$

The structure of matrix Ψ (2.14) can help to identify if one node can be reached starting from one of the other nodes, namely if there is a path π_{ij} (a sequence of distinct edge for which there is a sequence of distinct vertices) from node i and j. If all ψ_{ik} are not null, the graph is complete and each node is connected to other nodes in the network.

Proposition 1. *Consider a dynamical network* (G, Φ, Ψ), *where the graph is directed, one node* i *is not reachable if* $\psi_{ik} = \underline{0}$ $\forall k \neq i \in V$. *Contrariwise, there are no outcoming flows from node* i *if* $\psi_{ii} = \underline{0}$. *If the graph is undirected, there are no flows linked to node* i, *if* $\psi_{ik} = \underline{0}$ $\forall k \in V$.

Proof. Node i cannot be reached if it has no incoming flows, namely $w_{ij} = \underline{0}$ $\forall j \in N_i$. In this case, ψ_{ik} defined in (2.13), becomes:

$$\psi_{ik} := \begin{cases} -B_i \sum\limits_{j \in N_i} w_{ji} & i = k \\ \underline{0} & i \neq k \end{cases}$$

The null outcoming flows from node i imply $w_{ji} = 0$, ψ_{ik} defined in (2.13), becomes:

$$\psi_{ik} := \begin{cases} \underline{0} & i = k \\ B_i w_{ik} & i \neq k \end{cases}$$

You can easily see that, in case of undirected graph, $w_{ij} = w_{ji}$, we have $\psi_{ik} = \underline{0}$, $\forall k \in V$. $\qquad\square$

As in (2.4), the dynamics, after node/link becomes:

$$\begin{cases} \dot{x} & = \bar{\Delta} x \\ y & = Hx \end{cases} \tag{2.15}$$

where $\bar{\Delta} = \Phi + \bar{\Psi}$: all the modification in the dynamics of the network, due to disconnections, can be found in $\bar{\Psi}$.
The following result holds in case of disconnections.

Proposition 2. *Consider two dynamical networks* (G, Φ, Ψ) (2.10) *and* $(\bar{G}, \Phi, \bar{\Psi})$ (2.15)*, with the latter resulting from a disconnection. Consider any eigenvalue–eigenvector pair* (α, v) *of* Δ. *There exist non-null indiscernible states if and only if matrices* Ψ *and* $\bar{\Psi}$ *corresponding to* (G, Φ, Ψ) *and* $(\bar{G}, \Phi, \bar{\Psi})$ *have a common eigenvalue–eigenvector pair* (λ, v), *with* $\lambda \neq \alpha$.

Proof. From *Corollary 1* in [21], we know that, if Δ and $\bar{\Delta}$ has no eigenvalue-eigenvector pair in common, there are no not-null indiscernible states and, in addition, that an eigenvector $v \in V(\alpha)$ of $\bar{\Delta}$ is indiscernible if and only if $\Delta v = \bar{\Delta} v = \alpha v$. Considering the structure of Δ and $\bar{\Delta}$, we have:

$$\alpha v = \Delta v = \bar{\Delta} v =$$
$$= \Phi v + \Psi v = \Phi v + \bar{\Psi} v$$

and $\Phi v + \Psi v = \Phi v + \bar{\Psi} v$ if and only if $\Psi v = \bar{\Psi} v = \lambda v$. $\qquad\square$

Taking into account previous Proposition, disconnections can be detected taking into account only common eigenvalue-eigenvector pairs of Ψ and $\bar{\Psi}$.
In next Subsection, the consequences of link/node disconnections are exploited. Consider the multi-agent network with dynamics (2.10), characterized by the tuple

(G, Φ, Ψ). In case of disconnection of link (i, j), modified network is described by updated tuple $(\bar{G}, \Phi, \bar{\Psi})$ (2.15).

If graph G is directed, matrix $\bar{\Psi}$ is modified as follows:

$$\bar{\Psi} = \Psi - w_{ij}[(e_i \otimes I_n)B_i - (e_j \otimes I_n)B_j](e_j \otimes I_n)' \qquad (2.16)$$

where

- e_i and e_j are, respectively, the i-th and j-th unit vector of $\mathbb{R}^N$;

- I_n is the identity matrix of dimension n.

The following result holds in case of link disconnection for directed graph.

Theorem 1. *Consider two dynamical networks (G, Φ, Ψ) (2.10) and $(\bar{G}, \Phi, \bar{\Psi})$ (2.15), with the latter resulting from a disconnection of link (i, j). Consider any eigenvalue–eigenvector pair (λ, v) of Ψ. Then, if G is a directed graph, (λ, v) is also an eigenvalue–eigenvector pair of $\bar{\Psi}$, if and only if $B_i v^{(j)} = 0$ and $B_j v^{(j)} = 0$, where $v^{(j)}$ is the j-th component of v.*

Proof. Consider any eigenvalue–eigenvector pair (λ, v) of Ψ. From (2.16),

$$\begin{aligned}\bar{\Psi}v &= \Psi v - w_{ij}\left\{[(e_i \otimes I_n)B_i - (e_j \otimes I_n)B_j](e_j \otimes I_n)'\right\}v \\ &= \lambda v - w_{ij}\left\{[(e_i \otimes I_n)B_i - (e_j \otimes I_n)B_j](e_j \otimes I_n)'\right\}v\end{aligned} \qquad (2.17)$$

Considering that the Kronecher Product: $(e_i \otimes I_n) = E_i$ is a block matrix defined as:

$$E_i = [\epsilon_{ik}] \in \mathbb{R}^{Nn \times n} \qquad (2.18)$$

where:

$$\epsilon_{ik} := \begin{cases} I_i & i = k \\ 0 & i \neq k \end{cases}$$

Equation (2.17) becomes:

$$\begin{aligned}\bar{\Psi}v &= \lambda v - w_{ij}[E_i B_i - E_j B_j]E_j' v \\ &= \lambda v - w_{ij}[E_i B_i - E_j B_j]v^{(j)}\end{aligned} \qquad (2.19)$$

So eigenvalue–eigenvector pair (λ, v) of Ψ is also an eigenvalue–eigenvector pair $\bar{\Psi}$ if and only if $[E_i B_i - E_j B_j]v^{(j)} = 0$. Considering that $E_p B_p = EB_p$ is a block matrix

$$EB_p = [\epsilon_{ik}] \in \mathbb{R}^{Nn \times n} \qquad (2.20)$$

where:

$$\epsilon_p := \begin{cases} B_p & i = k = p \\ 0 & otherwise \end{cases}$$

So we have, $[E_i B_i - E_j B_j]v^{(j)} = EB_i - EB_j = 0$, if and only if $B_i v^{(j)} = 0$ and $B_j v^{(j)} = 0$. $\qquad \square$

In case of undirected graph G, matrix $\bar{\Psi}$ is modified as follows:

$$
\begin{aligned}
\bar{\Psi} = \Psi &- w_{ij}[(e_i \otimes I_n)B_i - (e_j \otimes I_n)B_j](e_j \otimes I_n)' \\
&- w_{ij}[(e_j \otimes I_n)B_j - (e_i \otimes I_n)B_i](e_i \otimes I_n)'
\end{aligned}
\tag{2.21}
$$

where

- e_i and e_j are, respectively, the i-th and j-th unit vector of $\mathbb{R}^N$;

- I_n is the identity matrix of dimension n.

The following result holds in case of link disconnection for undirected graph.

Theorem 2. *Consider two dynamical networks (G, Φ, Ψ) (2.10) and $(\bar{G}, \Phi, \bar{\Psi})$ (2.15), with the latter resulting from a disconnection of link (i, j). Consider any eigenvalue–eigenvector pair (λ, v) of Ψ. Then, if G is an undirected graph, (λ, v) is also an eigenvalue–eigenvector pair of $\bar{\Psi}$, if and only if $B_i v^{(i)} = B_j v^{(j)}$ and $B_j v^{(i)} = B_i v^{(j)}$, where $v^{(i)}$ and $v^{(j)}$ are, respectively, the i-th and the j-th component of v.*

Proof. Consider any eigenvalue–eigenvector pair (λ, v) of Ψ. From (2.21),

$$
\begin{aligned}
\bar{\Psi}v = \Psi v &- w_{ij}\left\{[(e_i \otimes I_n)B_i - (e_j \otimes I_n)B_j](e_j \otimes I_n)'\right\}v \\
&- w_{ij}\left\{[(e_j \otimes I_n)B_j - (e_i \otimes I_n)B_i](e_i \otimes I_n)'\right\}v \\
= \lambda v &- w_{ij}\left\{[(e_i \otimes I_n)B_i - (e_j \otimes I_n)B_j](e_j \otimes I_n)'\right\}v \\
&- w_{ij}\left\{[(e_j \otimes I_n)B_j - (e_i \otimes I_n)B_i](e_i \otimes I_n)'\right\}v
\end{aligned}
\tag{2.22}
$$

Recalling the Kronecher Product defined in (2.18), equation (2.22) becomes:

$$
\begin{aligned}
\bar{\Psi}v &= \lambda v - w_{ij}[E_iB_i - E_jB_j]E_j'v - w_{ij}[E_jB_j - E_iB_i]E_i'v \\
&= \lambda v - w_{ij}[E_iB_i - E_jB_j]v^{(j)} - w_{ij}[E_jB_j - E_iB_i]v^{(i)}
\end{aligned}
\tag{2.23}
$$

Recalling (2.20), we have $w_{ij}[E_iB_i - E_jB_j]v^{(j)} - w_{ij}[E_jB_j - E_iB_i]v^{(i)} = 0$, and consequently eigenvalue–eigenvector pair (λ, v) of Ψ is also an eigenvalue–eigenvector pair $\bar{\Psi}$, if and only if $B_i v^{(i)} = B_j v^{(j)}$ and $B_j v^{(i)} = B_i v^{(j)}$. $\qquad\square$

In case of disconnection of node i, there are no more incoming and outcoming flows to/from node i; disconnecting node i can be viewed as disconnection of all edges linked to i: links (i, j) and (j, i) for all $j \in N_i$. The modified network is described by updated tuple $(\bar{G}, \Phi, \bar{\Psi})$ (2.15).
Matrix $\bar{\Psi}$ is modified as follows:

$$
\begin{aligned}
\bar{\Psi} = \Psi &- \sum_{j \in N_i}\left\{w_{ij}[(e_i \otimes I_n)B_i - (e_j \otimes I_n)B_j](e_j \otimes I_n)'\right\} \\
&- \sum_{j \in N_i}\left\{w_{ji}[(e_j \otimes I_n)B_j - (e_i \otimes I_n)B_i](e_i \otimes I_n)'\right\}
\end{aligned}
\tag{2.24}
$$

The following result holds in case of node disconnection for directed and undirected graph.

Theorem 3. *Consider two dynamical networks $(G, \Phi, \Psi$ (2.10) and $(\bar{G}, \Phi, \bar{\Psi})$ (2.15), with the latter resulting from a disconnection of node i. Consider any eigenvalue–eigenvector pair (λ, υ) of Ψ. Then, (λ, υ) is also an eigenvalue–eigenvector pair of $\bar{\Psi}$, if and only if the following conditions hold:*

$$\sum_{j \in N_i} \left\{ w_{ij} B_i \upsilon^{(j)} \right\} = \sum_{j \in N_i} \left\{ w_{ji} B_i \upsilon^{(i)} \right\} = 0$$

and

$$w_{ij} B_j \upsilon^{(i)} = w_{ji} B_j \upsilon^{(j)}, \, \forall j \in N_i$$

where $\upsilon^{(j)} \, \upsilon^{(i)}$ are, respectively, the j-th and i-th component of υ.
If the graph G is undirected, then, (λ, υ) is also an eigenvalue–eigenvector pair of $\bar{\Psi}$, if and only if the following conditions hold:

$$\sum_{j \in N_i} \left\{ B_i \upsilon^{(j)} \right\} = \sum_{j \in N_i} \left\{ B_i \upsilon^{(i)} \right\} = 0$$

and

$$B_j \upsilon^{(i)} = B_j \upsilon^{(j)}, \, \forall j \in N_i$$

where $\upsilon^{(j)} \, \upsilon^{(i)}$ are, respectively, the j-th and i-th component of υ.

Proof. Consider any eigenvalue–eigenvector pair (λ, υ) of Ψ. From (2.24),

$$\begin{aligned}
\bar{\Psi}\upsilon &= \Psi\upsilon - \sum_{j \in N_i} \left\{ w_{ij} [(e_i \otimes I_n) B_i - (e_j \otimes I_n) B_j](e_j \otimes I_n)' \right\} \upsilon \\
&\quad - \sum_{j \in N_i} \left\{ w_{ji} [(e_j \otimes I_n) B_j - (e_i \otimes I_n) B_i](e_i \otimes I_n)' \right\} \upsilon \\
&= \lambda\upsilon - \sum_{j \in N_i} \left\{ w_{ij} [(e_i \otimes I_n) B_i - (e_j \otimes I_n) B_j](e_j \otimes I_n)' \right\} \upsilon \\
&\quad - \sum_{j \in N_i} \left\{ w_{ji} [(e_j \otimes I_n) B_j - (e_i \otimes I_n) B_i](e_i \otimes I_n)' \right\} \upsilon
\end{aligned} \qquad (2.25)$$

Considering (2.18), equation (2.25) becomes:

$$\begin{aligned}
\bar{\Psi}\upsilon &= \lambda\upsilon - \sum_{j \in N_i} \left\{ w_{ij} [E_i B_i - E_j B_j] E_j' \right\} \upsilon - \sum_{j \in N_i} \left\{ w_{ji} [E_j B_j - E_i B_i] E_i' \right\} \upsilon \\
&= \lambda\upsilon - \sum_{j \in N_i} \left\{ w_{ij} [E_i B_i - E_j B_j] \upsilon^{(j)} \right\} - \sum_{j \in N_i} \left\{ w_{ji} [E_j B_j - E_i B_i] \upsilon^{(i)} \right\}
\end{aligned} \qquad (2.26)$$

So eigenvalue–eigenvector pair (λ, υ) of Ψ is also an eigenvalue–eigenvector pair $\bar{\Psi}$ if and only if

$$\sum_{j \in N_i} \left\{ w_{ij} [E_i B_i - E_j B_j] \upsilon^{(j)} \right\} - \sum_{j \in N_i} \left\{ w_{ji} [E_j B_j - E_i B_i] \upsilon^{(i)} \right\} = 0 \qquad (2.27)$$

Equation (2.27) is satisfied if and only if:

$$\sum_{j \in N_i} \left\{ w_{ij} B_i \upsilon^{(j)} \right\} = \sum_{j \in N_i} \left\{ w_{ji} B_i \upsilon^{(i)} \right\}$$

and

$$w_{ij} B_j v^{(i)} = w_{ji} B_j v^{(j)}, \forall j \in N_i$$

If the graph is undirected, $w_{ij} = w_{j,i}$, so the conditions become:

$$\sum_{j \in N_i} \left\{ B_i v^{(j)} \right\} = \sum_{j \in N_i} \left\{ B_i v^{(i)} \right\}$$

and

$$B_j v^{(i)} = B_j v^{(j)}, \forall j \in N_i$$

$\square$

All results about discernibility, have been applied to case of linear node dynamics. Nevertheless, it can be noted that Proposition 2 (and subsequent results) can be extended to the case in which the internal dynamics of each node $i - th$ is non-linear, as:

$$\begin{cases} \dot{x}_i &= f_i(x_i) + B_i \sum_{j \in N_i} w_{ij} x_j - B_i \sum_{j \in N_i} w_{ji} x_i \\ y_i &= C_i x_i \end{cases} \qquad \text{i} = 1, 2, ..., \text{N} \qquad (2.28)$$

where $f_i(x_i)$ is a non linear function.
Considering the node dynamics in (2.28), the whole network state and output dynamics, as in (2.3), can be written as:

$$\begin{cases} \dot{x} &= \Delta(x) \\ y &= Hx \end{cases} \qquad (2.29)$$

where:

- $x \in \mathbb{R}^{Nn}$ is the aggregated state vector of network;

- $y \in \mathbb{R}^{Nm}$ is the aggregated output vector of the network;

- $\Delta(\cdot)$ is the aggregated non linear function $\Delta(\cdot) = \Phi(\cdot) + \Psi$;

- $\Phi(\cdot)$ is a block diagonal matrix defined as

$$\Phi(\cdot) = diag\left\{ f_i(\cdot) \right\} \qquad (2.30)$$

- $\Psi \in \mathbb{R}^{Nn \times Nn}$ is a block matrix defined as (2.12)

Proposition 3. *Consider two dynamical networks* (G, Φ, Ψ) *(2.29) and* $(\bar{G}, \Phi, \bar{\Psi})$ *(2.15), with the latter resulting from a disconnection. There exist non-null indiscernible states if and only if matrices* Ψ *and* $\bar{\Psi}$ *corresponding to* (G, Φ, Ψ) *and* $(\bar{G}, \Phi, \bar{\Psi})$ *have a common eigenvalue–eigenvector pair* (λ, v).

Proof. Consider the eigenvalue–eigenvector pair (λ, v) of Ψ and considering the structure of $\Delta(\cdot)$ and $\bar{\Delta}(\cdot)$, we have:

$$\Delta(v) = \bar{\Delta}(v) =$$
$$= \Phi(v) + \Psi v = \Phi(v) + \bar{\Psi} v$$

and $\Phi(v) + \Psi v = \Phi(v) + \bar{\Psi} v$ if and only if $\Psi v = \bar{\Psi} v = \lambda v$. $\square$

It can be seen that matrix Ψ represents the connections between nodes, so it's impacted from any disconnections acting on the network, namely from topological changes. Understanding if it's possible to detect network topology variations, it's useful in case of problems of Fault-Detection-Isolation or if the disconnections are intentionally introduced as control actions to be performed on network, i.e. in control of epidemics.

Chapter 3

Multi-agent epidemiological networks

In this Chapter, we apply the results of Chapter 2 to a multi-agent network, in which each node models a SIR comportamental system (firstly introduced by Kermack and McKendrick in [29]).

Applications of this epidemiological model can be found in different areas [30], including medicine (epidemics), informatics (P2P networks, computer viruses), economics (financial network contagion).

3.1 SIR Epidemiological model

Epidemic models are mathematical representation of diseases, characterized by variables representing different status of the individuals. Common used status for individuals are:

- Susceptible for individual who can be infected;

- Infected;

- Removed for those who have already been infected;

- Exposed for infected not yet infectious;

- Deceased due to disease.

Among the different models [31, 32, 33, 34], in this work we deal with SIR (Susceptible-Infected-Removed) model [29].

The dynamics of an epidemic, with classical SIR model can be represented by the following system of ODEs:

$$\begin{cases} \dot{S}(t) & = -\beta S(t)I(t) \\ \dot{I}(t) & = \beta S(t)I(t) - \gamma I(t) \\ \dot{R}(t) & = \gamma I(t) \end{cases} \quad (3.1)$$

where:

- $S(t)$: number of Susceptible individuals at time t, they can remain susceptible or move to the Infected population;

- $I(t)$: number of Infected individuals at time t, they can infect the Susceptible individuals and move to the Recovered population to recover or die;

- $R(t)$: number of Removed individuals at time t, they have been infected and are healed/immune or died;

- constant $\beta \in \mathbb{R}^+ \cup \{0\}$ is the transmission rate from an infected person; at time t, each Infected generates $\beta S(t)$ new Infected per day;

- constant $\gamma \in \mathbb{R}^+ \cup \{0\}$ is the removal rate; at time t, γ fraction of Infected per day becomes Recovered. It's inversely proportional to the average infectious period $\gamma = \frac{1}{\tau}$;

- the initial conditions are: initial number of Susceptible individuals not null $S(0) > 0$, while initial numbers of Infected and Recovered are greater or equal to zero, $I(0) \geq 0$ and $R(0) \geq 0$.

The properties of (3.1) [35, 36] are the following:

- changing the sign of t, it's possible to evolve back in time;

- $N(t)$ number of total population is conserved, with $N(t) = S(t) + I(t) + R(t)$;

- the number of Infected grows from a small initial number to an huge one, if $\dot{I}(0) \geq 0$, that is if $\beta S(0) > \gamma$;

- transitions from S directly to R and viceversa and from I to S are not allowed.

The transitions among Susceptible, Infected and Recovered are depicted in Figure 3.1, where demographic factors (birth and death rates) are not represented.
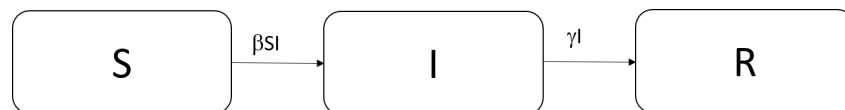

Figure 3.1. SIR transitions

SIR model has been extensively studied and applied, in some cases with modifications, to different outbreaks over the years (i.e. for HIV [37], for H1N1 [38], for Ebola [39] and, in last two years, for Covid-19 [40, 41]).
In Figure 3.2, it's depicted a classical evolution of Susceptible, Infected and Recovered individuals over time, in 'free running', without any interventions aimed to reduce the spread of the epidemic. At the beginning, most of the individuals are Susceptible, there are no Recovered and a small number of Infected is injected in the system (hundreds of Infected with respect to million of Susceptible individuals). In the first phase, the number of Infected has a rapid growth, while the number of Susceptible falls and the number of Recovered grows, both less rapidly than the Infected one. After the peak, the number of Infected starts to decrease, while the

Susceptible individuals continue to decrease and the Recovered ones continue to increase. At the end, there are no more Infected and Susceptible because all the individuals are Recovered.
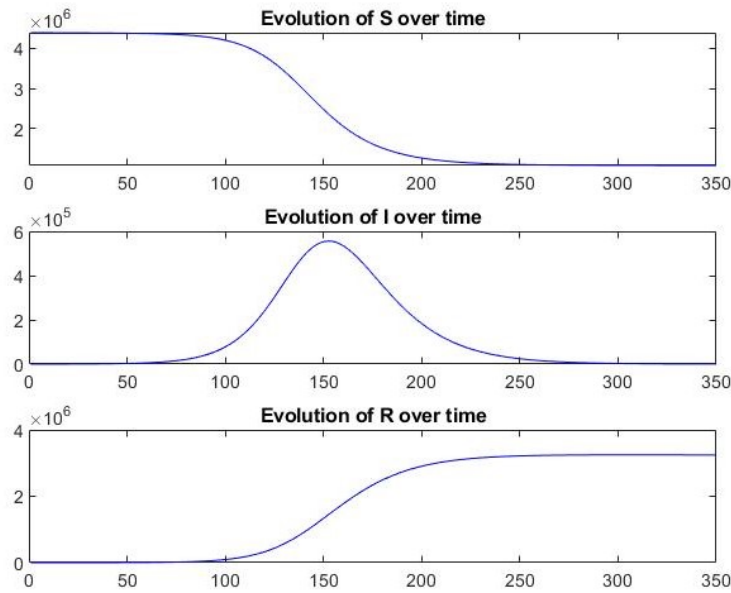

Figure 3.2. Classical SIR evolution

In Figures 3.3 and 3.4, the dependence of SIR model on parameters is shown. In both Figures, blue lines represent same evolution shown in Fig 3.2, while green lines show the modified evolutions in case of modifies parameters β (Fig. 3.3) or γ (Fig. 3.4). The initial conditions ($S(0) \approx 4.3 \cdot 10^6$ individuals, $R(0) = 0$ individuals, $I(0) = 100$ individuals) are the same for all the evolutions. In the baseline evolution (blue line), we have $\beta = 3.369 \cdot 10^{-8} \ day^{-1}$ and $\gamma = 0.079 \ day^{-1}$.

In Figure 3.3a and 3.3b, it's depicted, in green, the evolution of SIR model with, respectively, increased ($\beta = 6.739 \cdot 10^{-8} \ day^{-1}$) and decreased ($\beta = 3.032 \cdot 10^{-8} \ day^{-1}$) value of β, with respect to evolution in blue. It can be see that value β has effect to speed and force of epidemics: the larger the value of β, the faster the number of infected grows and reaches a higher peak.

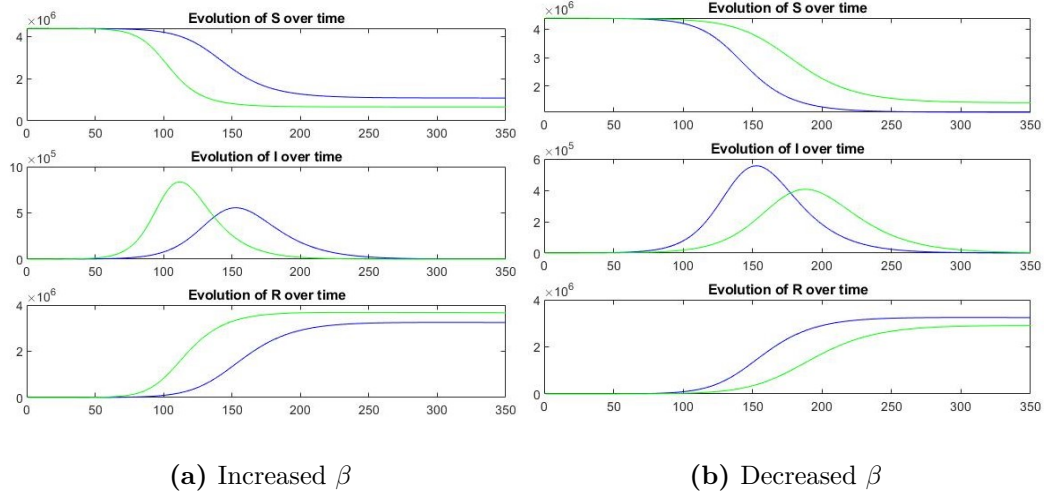**(a)** Increased β **(b)** Decreased β

Figure 3.3. SIR evolution with different β

In Figure 3.4a and 3.4b, it's depicted, in green, the evolution of SIR model with, respectively, increased ($\gamma = 0.095 \; day^{-1}$) and decreased ($\gamma = 0.071 \; day^{-1}$) value of γ, with respect to evolution in blue. It can be seen that value γ has an infect inverse effect with respect to effect of β.

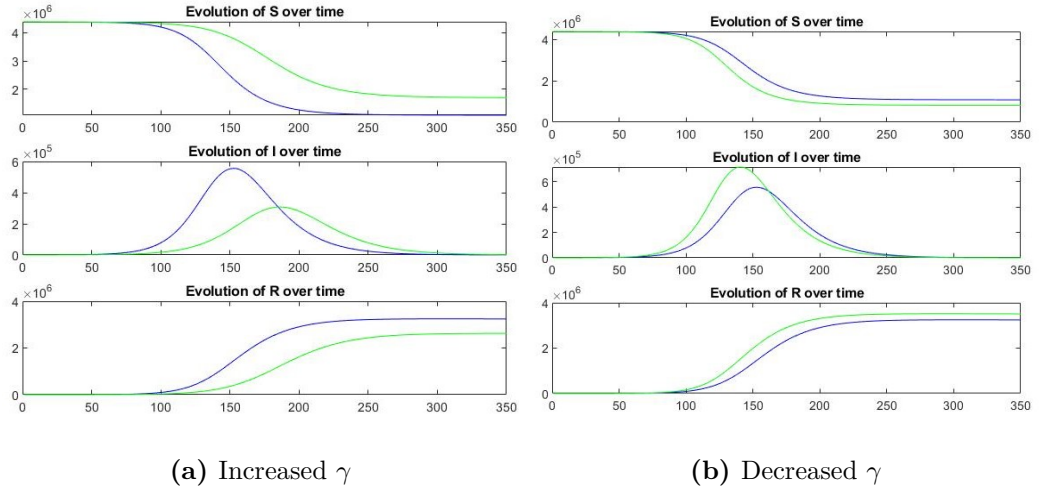**(a)** Increased γ **(b)** Decreased γ

Figure 3.4. SIR evolution with different γ

If we add demographic factors ([30, 42]) to (3.1), the dynamical model becomes:

$$\begin{cases} \dot{S}(t) &= \delta N(t) - \beta S(t)I(t) - \mu_S S(t) \\ \dot{I}(t) &= \beta S(t)I(t) - \gamma I(t) - \mu_I I(t) \\ \dot{R}(t) &= \gamma I(t) - \mu_R R(t) \end{cases} \quad (3.2)$$

where:

- constant $\delta \in \mathbb{R}^+ \cup \{0\}$ is the birth rate of the population;

- constants $\mu_S, \mu_I, \mu_R \in \mathbb{R}^+ \cup \{0\}$ are, respectively, the natural death rate for Susceptible, Infected and Recovered (independent from the disease).

The dynamics of the infection depends on the reproduction number $R_0(t)$ defined as $R_0(t) = \beta S(t)/\gamma$: if $R_0(t) < 1$ the number of infected will decrease, otherwise it will increases resulting in the spread of the infection. If the number of deaths from the virus is small compared with the living population, births and deaths (other than deaths caused by the infection) can be neglected and removed from the model (3.1). In case of small variation of susceptible ($S(t)$ almost constant over time), the product $\beta S(t)$ can be considered constant, $\beta S(t) = \beta_S$.

Variations on SIR lead to different models, in which, for example, cases of epidemics without immunity, with incubation periods, where infants can be born with immunity, where immunity lasts only for a short period of time, are considered.

Just as there are many studies on the evolution of epidemics, there are also different studies in the literature about automatic control technique applied to epidemics.

One problem studied is the design and stability analysis of observers for epidemiological models [43, 44]. Various relevant branches of research have focused on control of the outbreak spread: i.e. effects of different control strategies for Covid-19 has been investigated (preventive quarantine, isolation, antiviral therapy, therapy against complications, number of tests on population) in [45], optimal control studies for epidemics can be found in [46, 47].

3.2 Multi-agent SIR network

Most of the works found in the literature deal with a single SIR model, while we are interested in interactions between different models. In [48] and [49], the authors make use of exact and approximated Markov chain for modelling of epidemic spread over a given complex network, composed by interconnected SIS/SIRS models. In [50], a Lyapunov function structure for a two-population SEIR model has been introduced. In [51], it's proposed a novel control strategy which ensures that all agents in a SIS network, exponentially (respectively asymptotically), converge to the disease-free equilibrium (DFE). In [52], a Markov Decision Process models the control of an epidemic, with application of different planning algorithms to arrive at targeted control strategies.

In this section, we introduce the problem of interconnections among different populations, each of them modelled as a SIR model with its own parameters.

Consider a multi-agent systems composed by N node, the state of each node i (taking into account (3.1)) can be expressed as:

$$\dot{x}_i = \begin{pmatrix} \dot{x}_{1,i} \\ \dot{x}_{2,i} \\ \dot{x}_{3,i} \end{pmatrix} = \begin{pmatrix} \dot{S}_i \\ \dot{I}_i \\ \dot{R}_i \end{pmatrix} \in \mathbb{R}^3 \tag{3.3}$$

Note that from (3.3) to follow, we remove the dependence on t from the formulas, in order to improve readability.

From (3.3) and (3.1), the dynamics of node i, belonging to a multi-agent network,

can be expressed as:

$$\dot{x}_i = f_i(x_i) + B_i \sum_{j \in N_i} w_{ij} x_j - B_i \sum_{j \in N_i} w_{ji} x_i \qquad (3.4)$$

where

- $x_i/x_j \in \mathbb{R}^n$ are, respectively, the state vector of node i and j;

- matrix $B_i \in \mathbb{R}^{n \times n}$ is the input matrix of node i;

- $w_{ij} \in \mathbb{R}^+ \cup \{0\}$ represents the weight of the flow from node j to node i and it is equal to zero in case of absence of the flow;

- the non linear function $f_i(x_i)$ representing the proper dynamics of node i is defined as:

$$f_i(x_i) = \begin{pmatrix} \beta_i x_{1,i} x_{2,i} - \mu_{Si} x_{1,i} \\ \beta_i x_{1,i} x_{2,i} - \gamma x_{2,i} - \mu_{Ii} x_{2,i} \\ \gamma x_{2,i} - \mu_{Ri} x_{3,i} \end{pmatrix} + \underline{\delta_i} N_i \qquad (3.5)$$

  with

$$\underline{\delta_i} = \begin{pmatrix} \delta_i \\ 0 \\ 0 \end{pmatrix}$$

In case of small variation of susceptible ($S_i(t)$ almost constant over time), as already mentioned, we can consider $\beta_i S_i(t) = \beta_{Si}$. With this assumption, (3.5) becomes linear in x_i and, without demographic factor ($\delta_i, \mu_{Si}, \mu_{Ii}, \mu_{Ri}$), it can be expressed as:

$$f_i(x_i) = A_i x_i = \begin{pmatrix} 0 & -\beta_{Si} & 0 \\ 0 & \beta_{Si} - \gamma_i & 0 \\ 0 & \gamma_i & 0 \end{pmatrix} \begin{pmatrix} x_{1,i} \\ x_{2,i} \\ x_{3,i} \end{pmatrix} \qquad (3.6)$$

It's easy to see that matrix A_i in (3.6) has two eigenvalues ($\lambda = 0$ with multiplicity 2 and $\lambda = \beta_{Si} - \gamma_i$) and the equilibrium points of the system are the disease free $x_{e,i} = (x_{1,i}, 0, x_{3,i})^T$ and the endemic $x_{e,i} = (x_{1,i}, x_{2,i}, x_{3,i})^T$ with $\beta_{Si} = \gamma_i$.

It should be noted that this approximation can be used only in case $\beta_{Si} < \gamma_i$. In Figure 3.5, we can find the evolution of linearised SIR model with different values of β_{Si}. The initial conditions ($S_i(0) \approx 4.3 \cdot 10^6$ individuals, $R_i(0) = 0$ individuals, $I_i(0) = 100$ individuals) and the value of $\gamma_i = 0.0795 \ day^{-1}$ are the same. In Fig. 3.5a, we have the evolution with $\beta_{Si} = 0.0148 \ day^{-1}$, while in Fig. 3.5b, we have the evolution with $\beta_{Si} = 0.0895 \ day^{-1}$. In the first case, we can see the decrease of the Infected over time, while in the latter case we can see the number of Infected increasing and the number of Susceptible decreasing (it could become negative over a longer period of time or if initial value $I_i(0)$ is greater).
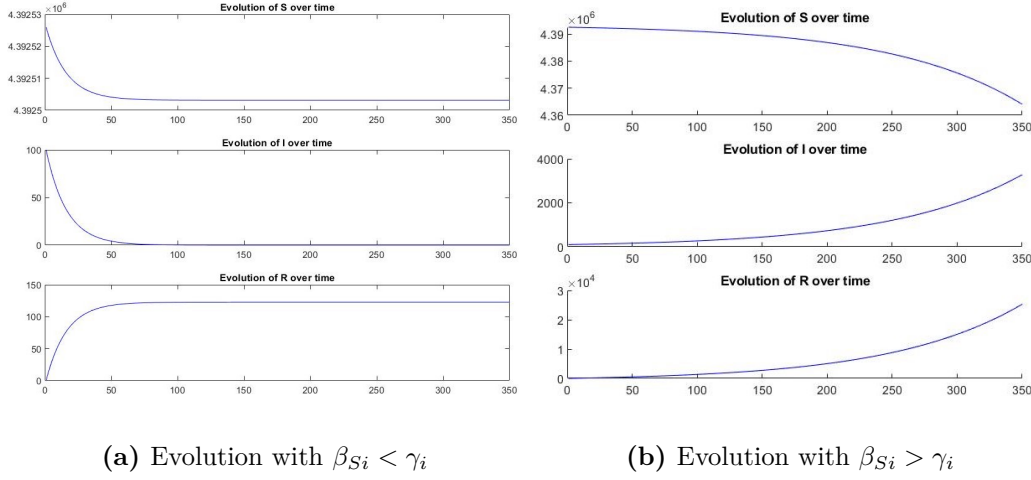
(a) Evolution with $\beta_{Si} < \gamma_i$

(b) Evolution with $\beta_{Si} > \gamma_i$

Figure 3.5. Linear SIR evolution with different β_{Si} wrt γ_i

If $\beta_{Si} = \gamma_i$ the number of Infected remains constant $I_i(t) = I_i(0)$ for all $t \geq 0$, while the number of Susceptible decreases linearly with respect to $I_i(0)$ in β_{Si} and the number of Recovered increases linearly with respect to $I_i(0)$ in γ_i. The dynamics of node i (3.4), with the assumption of small variations of S_i over time and without considering demographic factor, can be rewritten in same form of (2.8), as:

$$\dot{x}_i = A_i x_i + B_i \sum_{j \in N_i} w_{ij} x_j - B_i \sum_{j \in N_i} w_{ji} x_i \tag{3.7}$$

3.3 Simulations

When studying the evolution of an epidemics in a population, effects of migrations and travelling shall be taken into account. The 'internal' dynamics of each SIR model changes if there are flows of incoming or outcoming individuals. These effects shall be considered, for example, when some control strategies should be designed in order to reduce the spread of the epidemics.

In last years, most researches, in different scientific areas, has been focused on Covid-19 from different points of view. Works as [53], [54], [55], [56] corroborate, with real data analysis, the statement that travelling and/ or measure act to reduce travels have impacts of global and local evolution of epidemics.

In this Chapter, we expose the results of numerical simulations of the evolution of an epidemics in Italy, performed using Matlab®.

We consider a multi-agent network composed by $N = 20$ nodes, where each node is a region of Italy, whose dynamics is expressed by (3.7) and (3.6). The whole network dynamics, according to (2.10) is:

$$\begin{cases} \dot{x} &= \Delta x = (\Phi + \Psi)x \\ y &= Hx \end{cases} \tag{3.8}$$

where:

- number of nodes is $N = 20$;

- state and output dimensions of each node are, respectively, $n = 3$ and $m = 3$;

- $x \in \mathbb{R}^{60}$ is the aggregated state vector of network;

- $y \in \mathbb{R}^{60}$ is the aggregated output vector of the network;

- $\Delta \in \mathbb{R}^{60 \times 60}$ is the aggregated transition matrix, with Φ and Ψ defined in (2.11) and (2.12);

- $H \in \mathbb{R}^{60 \times 60}$ is the aggregated output matrix.

The dynamics of each node can be expressed with:

$$\begin{cases} \dot{x}_i & = A_i x_i + B_i \sum_{j \in N_i} w_{ij} x_j - B_i \sum_{j \in N_i} w_{ji} x_i \\ y_i & = C_i x_i \end{cases} \tag{3.9}$$

with the following assumptions:

- there are small variations of S_i, hence the approximated linear dynamics;

- all the node outputs are known, thus $C_i = I_3$, $\forall i$;

- the input matrix of each node is the identity matrix of dimension equal to each node state dimension, $B_i = I_3$, $\forall i$. The matrices B_i have no values different from 0 out of the diagonal, because we consider that dynamics of Susceptible, Infected and Recovered of each node are impacted, respectively, only from flows of Susceptible, Infected and Recovered to/ from its neighbouring nodes.

Parameters β_i and γ_i in A_i are assumed different for each node and they are listed in Table 3.1.

Region	**i**	**Par.** $\beta_{Si}\,[day^{-}1]$	**Par.** $\gamma_i\,[day^{-}1]$
Abruzzo	1	0.0040935318	0.0440041908
Basilicata	2	0.0018282933	0.0033003300
Calabria	3	0.0057069471	0.0221650504
Campania	4	0.0150376410	0.1626349926
Emilia-Romagna	5	0.0137731242	0.0415926265
Friuli Venezia Giulia	6	0.0039372300	0.0429280397
Lazio	7	0.0159976514	0.0325700529
Liguria	8	0.0060039173	0.0836653386
Lombardia	9	0.0274487836	0.0329042563
Marche	10	0.0048459267	0.0276870995
Molise	11	0.0010291597	0.0475737392
Piemonte	12	0.0148022109	0.0795896975
Puglia	13	0.0110226003	0.0224210314
Sardegna	14	0.0047555938	0.0185849555
Sicilia	15	0.0148235777	0.0197086003
Toscana	16	0.0120988374	0.0626676420
Trentino-Alto Adige	17	0.0025916312	0.0283706747
Umbria	18	0.0028737300	0.0293032786
Valle D'Aosta	19	0.0003858634	0.0876132930
Veneto	20	0.0134452849	0.0968906720

Table 3.1. Regional parameters

For the connections, we assume, that there are flows between two regions if they are neighbouring (i.e no flows for regions that are islands) and that the weights of all flows are constant $w_{ij} = 0.001$.

In Table 3.2, the connections for each node are shown. Different simulations are performed with different initial values $I_i(0)$, while the initial Recovered populations are assumed to be null in all regions and the initial Susceptible populations (official data of the resident population in each region in 2019 [57]) are listed in Table 3.2.

Region	i	Neighbouring regions	S(0) [p]	R(0) [p]
Abruzzo	1	7 10 11	1322247	0
Basilicata	2	3 4 13	570365	0
Calabria	3	2	1965128	0
Campania	4	2 7 11 13	5839084	0
Emilia-Romagna	5	8 9 10 12 16 20	4448841	0
Friuli Venezia Giulia	6	20	1217872	0
Lazio	7	1 4 10 11 16 18	5898124	0
Liguria	8	5 12 16	1565307	0
Lombardia	9	5 12 17 20	10018806	0
Marche	10	1 5 7 16 18	1538055	0
Molise	11	1 4 7 13	310449	0
Piemonte	12	5 8 9 19	4392526	0
Puglia	13	2 4 11	4063888	0
Sardegna	14		1653135	0
Sicilia	15		5056641	0
Toscana	16	5 7 8 10 18	3742437	0
Trentino-Alto Adige	17	9 20	1062860	0
Umbria	18	7 10 16	888908	0
Valle D'Aosta	19	12	126883	0
Veneto	20	5 6 9 17	4907529	0

Table 3.2. Regional connections and initial values

Figure 3.6 shows the multi-agent network used for simulation, in which each node correspond to one of region labelled according to column i in Table 3.2. Note that for readability, a bidirectional flow between neighbouring regions is represented, instead of two different flows.
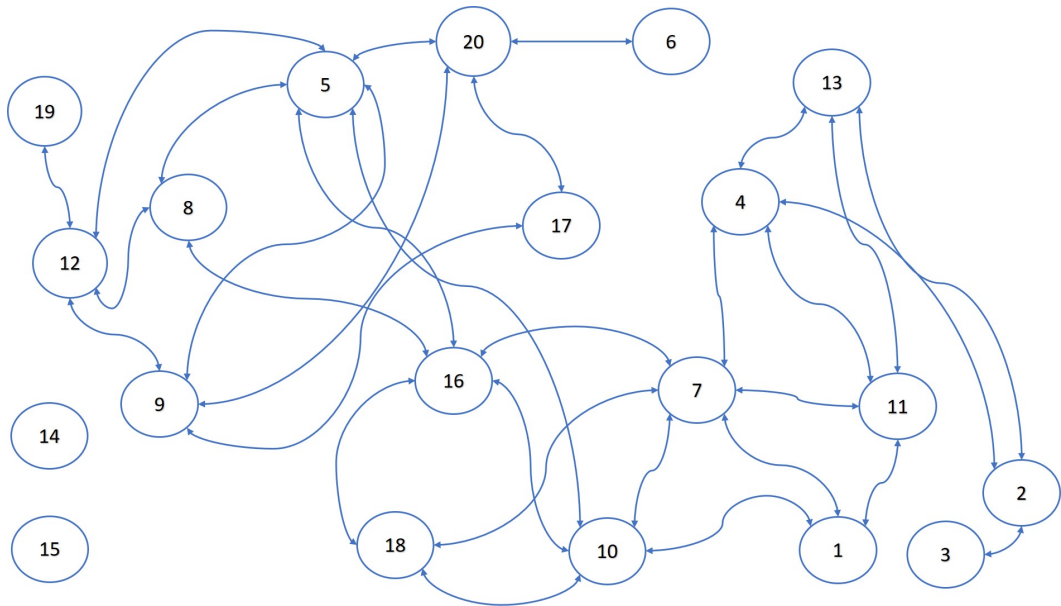

Figure 3.6. Epidemiological network

All the simulations, in linearised case, helps to understand the effect of epidemics propagation in the network in a short time interval. For the long term evolution, it shall be considered the case of non linear SIR models.

3.3.1 Simulation in linear case

In first simulation, we compare the evolution of each node dynamics (S depicted in Figure 3.7 and I depicted in Figure 3.8) in case 1000 infected individuals are injected in the system in node number 9 ($I_i(0) = 1000$ if $i = 9$ and $I_i(0) = 0$ otherwise), if there are (green lines) or not (blue lines) connections between regions.
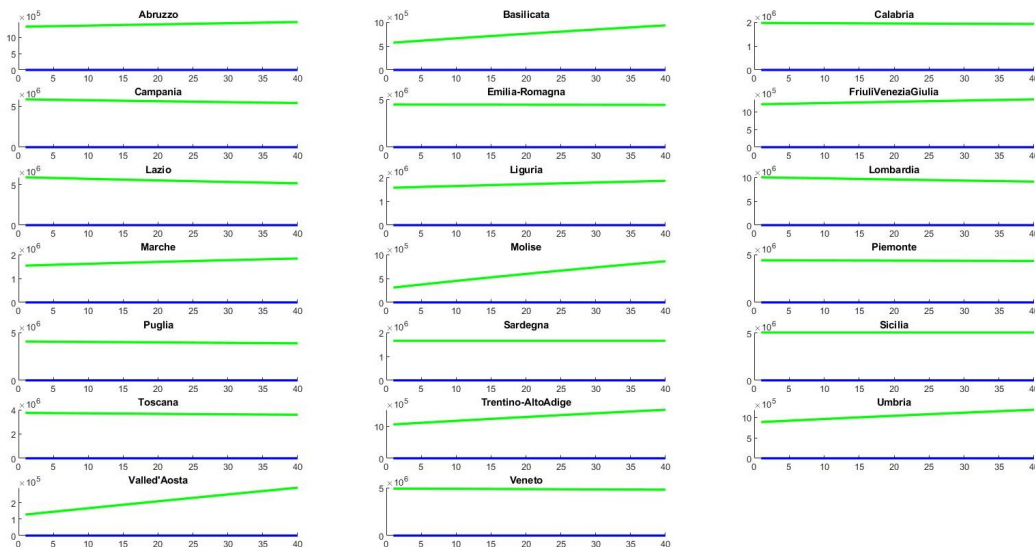
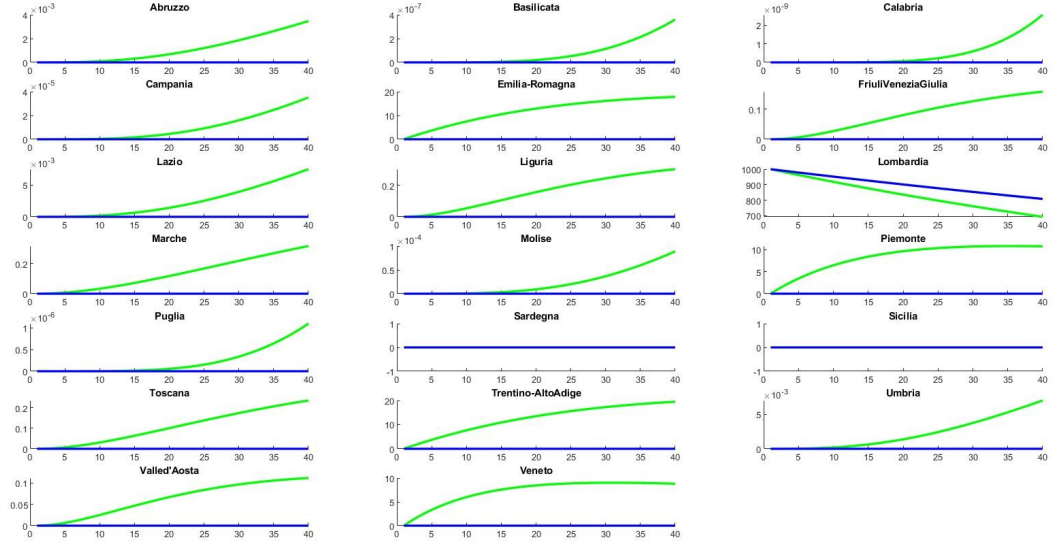**Figure 3.7.** Evolution of S_i without/with connections

Figure 3.8. Evolution of I_i without/with connections

From the analysis of Figures 3.7 and 3.8, it's easy to see the effects of connections between regions (green lines) with respect to evolutions without them (blue lines). Note that the evolutions of population of islands (here considered as isolated nodes) are the same in both cases.

It's interesting to see how number of Infected for regions with flows change in the different scenarios. In Lombardia (region number 9), number of Infected decreases faster in case of flows, because it's assumed that some of them move in other regions over time. As the Infected number in Lombardia is reduced, same quantities increase for all the connected regions. However, it can be noted that the speed of the epidemics is different from one region to another. Of course, each region has it's own parameters β_{Si} and γ_i, that lead its epidemics dynamics but also the distance with respect to Lombardia impacts the evolutions. We can see that, regions with shorter path from it, has a faster increase of Infected population with respect to time (see, for example, Veneto and Piemonte with respect to Lazio).

Taking into account previous considerations, different control actions, related to connections, can be performed on the network in order to reduce the spread of the epidemics. These actions imply only disconnections in the network and they can be added to different control action acting on one or all populations (i.e. local lockdown). The dynamics of each node, taking into account a generic control action u_i, can be expressed with:

$$\begin{cases} \dot{x}_i & = A_i x_i + B_i \sum_{j \in N_i} w_{ij} x_j - B_i \sum_{j \in N_i} w_{ji} x_i + u_i \\ y_i & = C_i x_i \end{cases} \tag{3.10}$$

Firstly, we consider that, at certain time t_c, when the epidemics starts to spread, a control action is taken, which consists in prohibiting all travels between regions,

namely in (3.10):

$$u_i = \theta(t_c) \left\{ -B_i \sum_{j \in N_i} w_{ij} x_j + B_i \sum_{j \in N_i} w_{ji} x_i \right\} \tag{3.11}$$

where $\theta(\cdot)$ is the unit step function, different from 0 for $t \geq t_c$.

Theorem 3 gives the necessary and sufficient conditions under which the disconnection of one node is not discernible, but in this case, we disconnect all nodes in the network. Instead of checking the conditions for all the node, we start checking them for node $i = 19$ (it has only one neighbouring node). We let the tool compute the eigenvalues and eigenvectors and we, easily, discover that $v^{(19)} \neq 0$, so the disconnection of node 19 is discernible. Considering that this disconnection is discernible, it derives that the applied control action leads to a discernible state of the overall network.

In Figure 3.9, we can see the evolutions of Infected in all regions: the green lines depict the evolutions in nominal connections, while the red lines depict the evolutions after disconnections at $t_c = 10$ day.
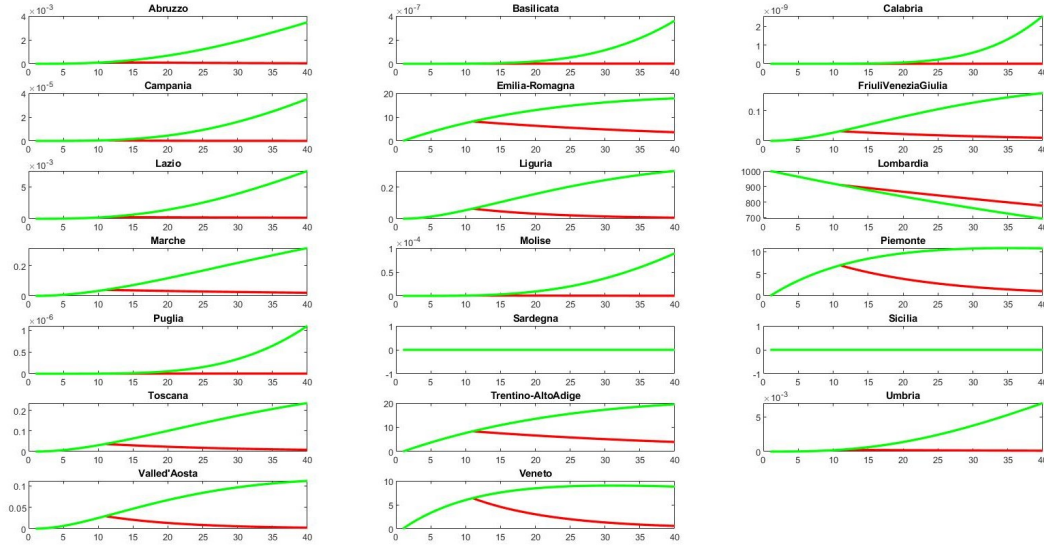

Figure 3.9. Evolution of I_i with disconnections

The evolution of overall Infected in the network, without control action (green line) and with control action (red line) is depicted in Figure (3.10): in Fig. 3.10a we can see that the overall number of Infected decreases after the control action thanks to the blocking of propagation in many regions; in Fig 3.10b, we can see the overall Infected evolution without I_9, from which it can be seen that the number of infected in all other nodes decreases rapidly.

(a) With I_9 **(b)** Without I_9
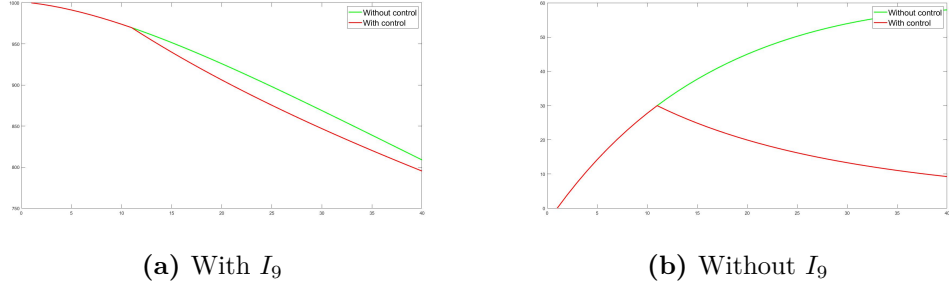
Figure 3.10. Total number of I in the network

One of the most impacting factor, it's the time of application of control action: the earlier it is applied, the less there is propagation on the network. In Figure 3.11, we can see the evolutions of Infected in all regions (without I_9), if the control action 3.11 is applied at different t_c.
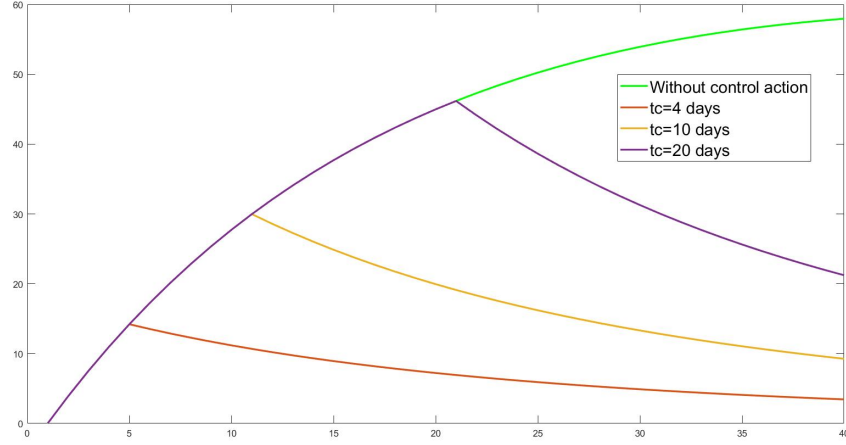

Figure 3.11. Evolution of I_i with disconnections at different t_c

Taking into account this analysis, we modify the control action, which now consists in prohibiting all travels to and from node 9 (node with higher number of Infected with respect to other regions). The control action in (3.10), shall be applied only to node 9 and it becomes:

$$
u_i = \begin{cases} \theta(t_c)[-B_9 \sum\limits_{j \in N_9} w_{9j}x_j + B_9 \sum\limits_{j \in N_9} w_{j9}x_9] & i = 9 \\ \theta(t_c)[-B_i w_{i9}x_9 + B_i w_{9i}x_i] & i \in N_9 \\ 0 & otherwise \end{cases}
$$

Theorem 3 gives the necessary and sufficient conditions under which the disconnection of one node is not discernible. As before, we let the tool compute the eigenvalues and eigenvectors and we, easily, discover that $v^{(9)} \neq 0$, so the disconnection of node 9 is discernible, hence the applied control action leads to a discernible state of the overall network.

In Figure 3.12, we can see the evolutions of Infected in all regions: the green lines depict the evolutions in nominal connections, while the red lines depict the evolutions after disconnections of node 9 at $t_c = 10$ day.
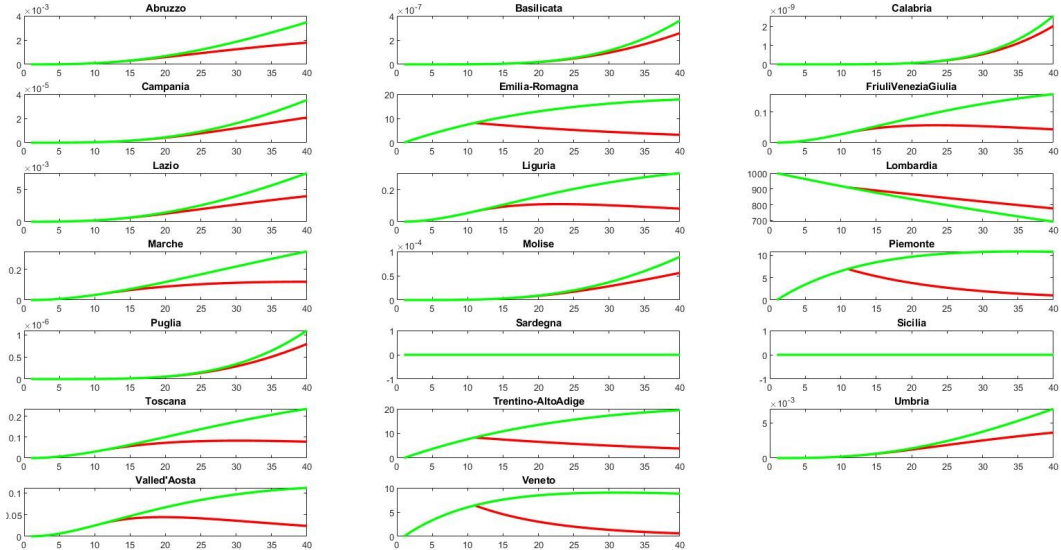

Figure 3.12. Evolution of I_i with disconnections of node 9

The evolution of overall Infected in the network, without control action (green line) and with control action (red line) is depicted in Figure (3.13): in Fig. 3.13a we can see that the overall number of Infected decreases after the control action thanks to the blocking of propagation in many regions; in Fig 3.13b, we can see the overall Infected evolution without I_9, from which it can be seen that the number of infected in all other nodes decreases rapidly. So this control action, with impacts only on a restricted number of nodes, has quite the same effects of overall Infected in the network.
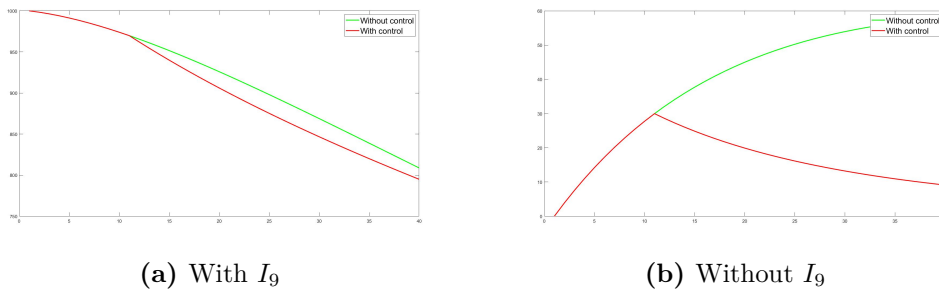

(a) With I_9

(b) Without I_9

Figure 3.13. Total number of I in the network with disconnection of node 9

As seen in Figure 3.11, the impact on epidemic propagation, due to control actions, depends on time t_c on which we disconnect the node. In real cases, we have some constraint on this t_c, i.e. this time cannot be less than the number of days it

takes to notice the initial spread of the pandemic. A problem can be which control action should be taken at fixed time t_c in order to reduce the propagation.

Consider the case in which 1000 infected individuals are injected in the system in node number 3 and 9 ($I_i(0) = 1000$ if $i = 3, 9$ and $I_i(0) = 0$ otherwise) and a control action should be applied at time $t_c = 10$ days in order to stop the propagation of the epidemics. In this case, the weights of all flows are constant $w_{ij} = 0.01$. We don't know the initial state of each node, but the rounded number of Infected in each region at t_c, listed in Table 3.3.

Region	Number of Infected [p]
Abruzzo	1
Basilicata	75
Calabria	771
Campania	2
Emilia-Romagna	57
Friuli VeneziGiulia	3
Lazio	1
Liguria	5
Lombardia	646
Marche	3
Molise	1
Piemonte	51
Puglia	4
Sardegna	0
Sicilia	0
Toscana	3
Trentino-Alto Adige	66
Umbria	1
Valle D'Aosta	2
Veneto	49

Table 3.3. Number of Infected in each region at $t_c = 10$ days

A different control action can be applied in this case, which now consists in splitting the network in sub-group, prohibiting all travels between some regions. In order to avoid propagation, in regions with longer paths from node with higher number of Infected, we disconnect the links, namely stop flows between nodes 5 and 16, 5 and 10, 8 and 16, 2 and 13, 2 and 4. The split network is depicted in Figure 3.14, refers to Figure 3.6 for original topology. Note that the dotted lines corresponds to the edge disconnected.
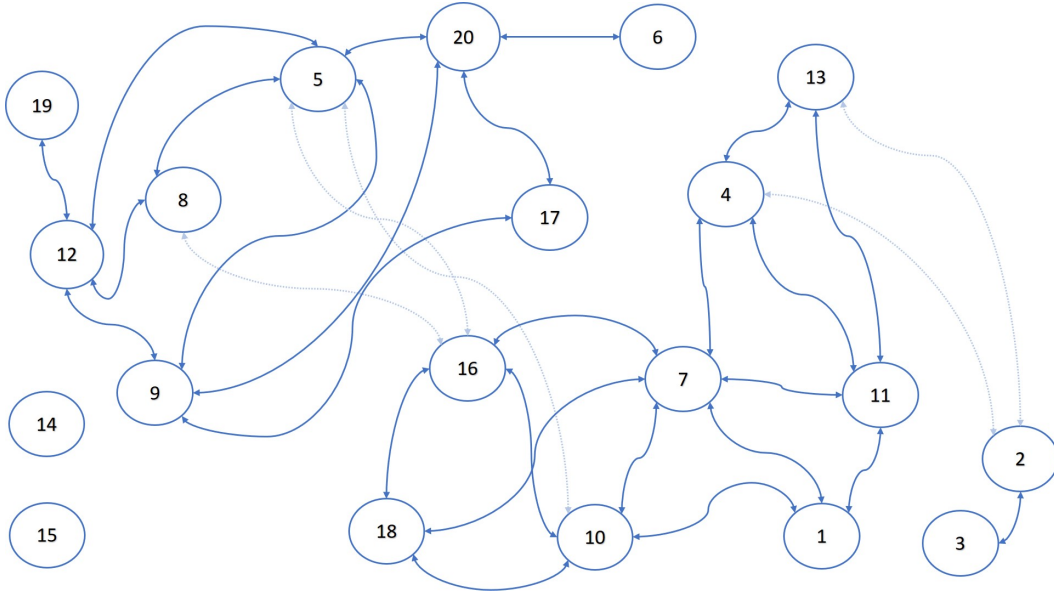
Figure 3.14. Epidemiological 'split' network

The control action in (3.10), shall be modified as follows:

$$
u_i = \begin{cases}
\theta(t_c)[-B_2(w_{2,4}x_4 + w_{2,13}x_{13}) + B_2(w_{4,2} + w_{13,2})x_2] & i = 2 \\
\theta(t_c)[-B_4(w_{4,2}x_2) + B_4(w_{2,4})x_4] & i = 4 \\
\theta(t_c)[-B_5(w_{5,16}x_{16} + w_{5,10}x_{10}) + B_5(w_{16,5} + w_{10,5})x_5] & i = 5 \\
\theta(t_c)[-B_8(w_{8,16}x_{16}) + B_8(w_{16,8})x_8] & i = 8 \\
\theta(t_c)[-B_{10}(w_{10,5}x_5) + B_{10}(w_{5,10})x_{10}] & i = 10 \\
\theta(t_c)[-B_{13}(w_{13,2}x_2) + B_{13}(w_{2,13})x_{13}] & i = 13 \\
\theta(t_c)[-B_{16}(w_{16,5}x_5 + w_{16,8}x_8) + B_{16}(w_{5,16} + w_{8,16})x_{16}] & i = 16 \\
0 & otherwise
\end{cases}
\tag{3.12}
$$

Theorem 2 gives the necessary and sufficient conditions under which the disconnection of one link is not discernible. As the graph corresponding to our network is undirected, all the weights are equal and for all nodes $B_i = I_3$, we know that the control action has a not discernible effect if an only if $v^i = v^j$ where i, j correspond to edge (i, j) disconnected. As previous cases, we let the tool compute the eigenvalues and eigenvectors. We verify that previous condition does not hold, i. e. , for link $(2, 4)$, so the control action leas to a discernible state.

In Figure 3.15, we can see the evolutions of Infected in all regions: the green lines depict the evolutions in nominal connections, while the red lines depict the evolutions after disconnections at $t_c = 10$ day. It can be noted that, applying this control action, the spread of epidemics, in part of the network, can be avoided.
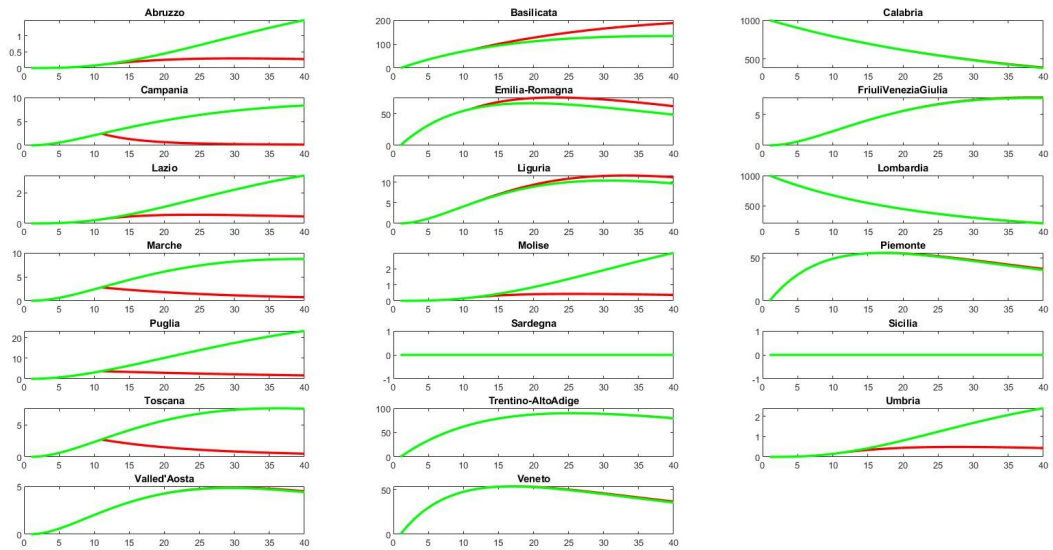
Figure 3.15. Evolution of I_i with network split

We can apply the control action (3.12) to the network in case of w_{ij} with different values (in this simulation, we set w_{ij} as random values from 0.01 and 0.1). In Figure 3.16, we can see the evolutions of Infected in all regions: the red lines depict the evolutions after disconnections at $t_c = 10$ day with w_{ij} equal for all i and j, while cyan lines depict the evolutions with different w_{ij}. It can be noted that, the evolutions of Infected differ in two cases.
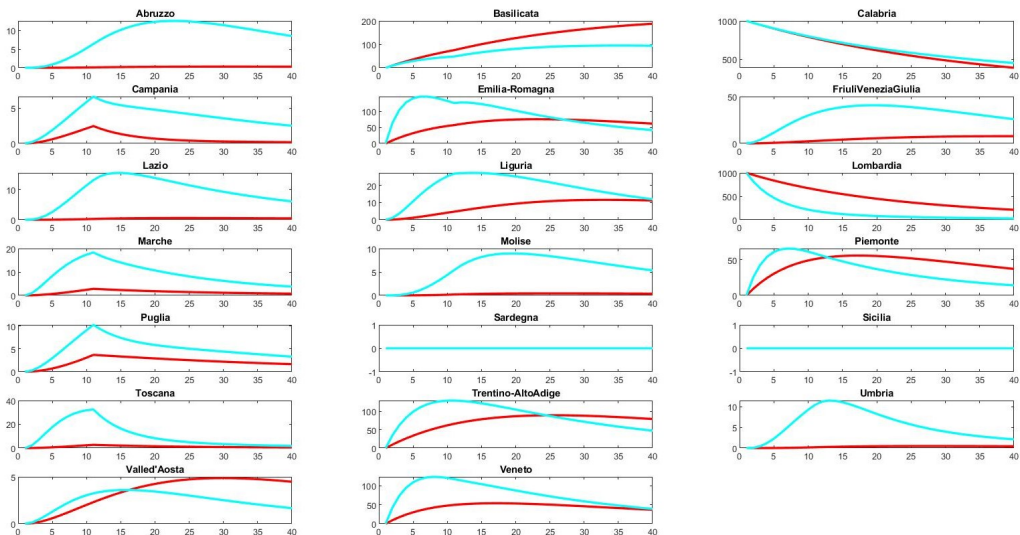

Figure 3.16. Evolution of I_i with network split and different w_{ij}

In last simulation, we simulate what happened, in Italy, at the beginning of 2020, when Covid-19 started to spread in northern Italy, and more slowly in central and southern Italy. We show the propagation of an epidemic in the network in case 30 (and then 100) infected individuals are injected in the system in Lombardia, node number 9. We consider the flows between northern regions to be greater with respect to central and southern ones ($w_{ij} = 0.1$ if $i/j = 5, 6, 9, 12, 17, 19, 20$ and $w_{ij} = 0.01$ otherwise). In Figures 3.17 and 3.18, we can see the evolution of Infected, in all node, at the beginning of spread of epidemic. It can be seen that, in both cases, the epidemics spreads rapidly in northern region and, in simulation time (30 days), starts to spread in central ones while it does not start to spread in southern ones. In the second case, Figure 3.18, we see number of Infected greater in some regions with respect to first case, Figure 3.17.
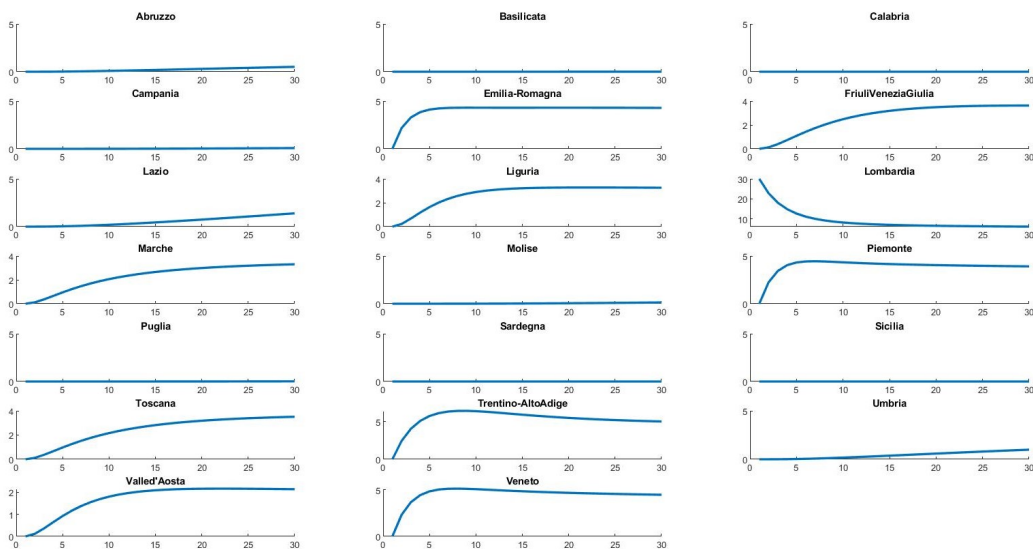

Figure 3.17. Spread of an epidemic in Italy starting from 30 I in Lombardia
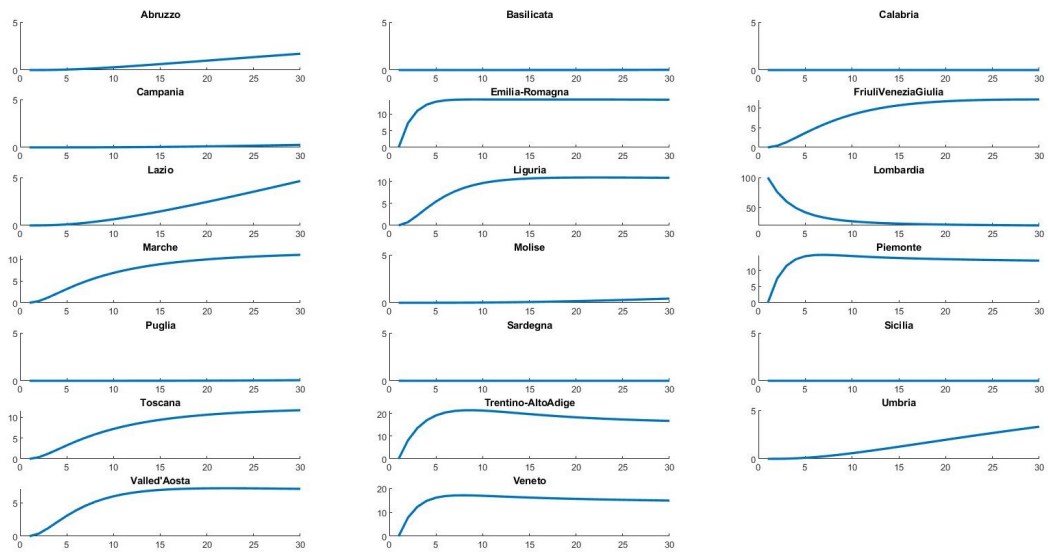
Figure 3.18. Spread of an epidemic in Italy starting from 100 I in Lombardia

3.4   Conclusions

In Chapter 3.2, the evolution of a disease in a multi-agent epidemiological network
is exploited, in which the state of each node represents the evolution of the epidemic
in its population and it is considered affected by flows of incoming and outcoming
people from its neighbouring nodes. The effects of some control actions, acting on
the connections between nodes, are also exploited, checking also the discernibility
of network state obtained after different disconnections.

Chapter 4

Micro-gripper model

This Chapter describes the real micro-manipulator, including its modelling, subject of some studies for tissue parameter estimation.

4.1 Introduction

Applications of robotics in medical area are more and more frequent and they are becoming standard techniques, in particular in some branch of surgery (references to some of these applications and studies can be found in [58, 59, 60, 61, 62]). In some cases, the robot in use is of human scale, but in several cases the dimensions of the robotic devices shall be very small (i.e. mini invasive interventions).
The main differences, between working with macroscopic or microscopic scale in gripping [63], are:

- dominance of adhesion forces at microscale (whereas at macroscale, the dominating force is gravitational force) caused by the scaling effect and contact hysteresis;

- risk of damaging objects manipulated and end-effectors;

- importance of visual feedback, for which microscopes are needed as a part of manipulation system, but they can cause problems due to restrict space and complexity of visual information processing;

- very high positioning accuracy required.

Just as the difficulty in manufacturing, actuating and using micro and nano devices, increases [64], so does the possibility of applying this type of device in different areas.
In this work, it's proposed the use of a real micro-gripper for estimation of the mechanical characteristics of the samples under examination, specifically their visco-elastic properties.

4.2   Experimental micro-gripper

A novel micro-gripper has been studied and fabricated as described in [65]. The fabrication method adopted makes use of Deep Reactive-Ion Etching applied on Silicon on Insulator wafer. The micro-device is depicted in Figure 4.1.

Figure 4.1. The real device

     The device is fabricated as a silicon monolithic structure, arranged with a comb drive at the anchor of each jaw. The comb actuators exert the input torques that, through the deflections of a flexure hinge [66, 67], move the jaws during the gripping tasks [68].

An operative scenario is drawn in Figure 4.2. In this Figure we can see the gripper pinching a tissue sample, kept between the jaws. The points B and C are the contact points between the jaws and the sample, while the points A and D represent the hinges/actuators.
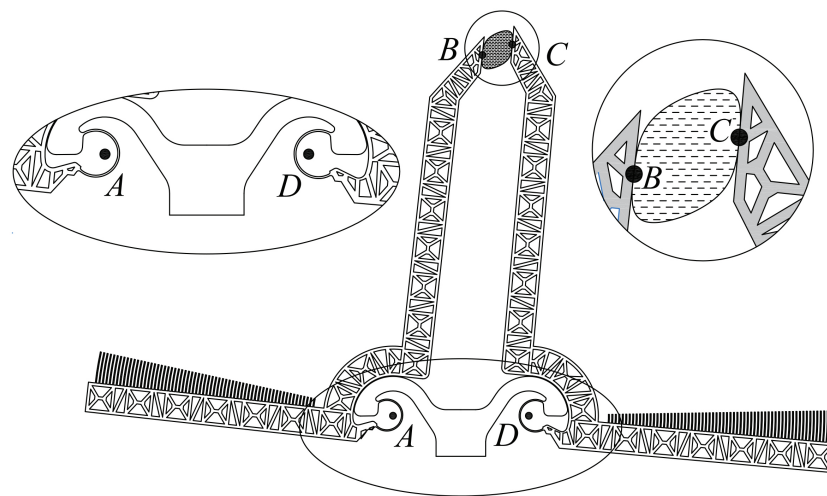

Figure 4.2. The gripping system

The problem has been already described and investigated in [69] and [70].
In the initial configuration, the gripper pinches the sample, so the two jaws are coupled by the presence of the handled sample. Therefore, this system can be modelled by introducing an equivalent 4-bar linkage as in Figure 4.3 (a), where link BC has lost the rigidity property. Considering the revolute centers and the tip contact positions, the two jaw lengths of AB and DC are assumed to be equal to l. The frame link length AD corresponds to the distance between the centers of the CSFHs and it can be assumed to be equal to d. Finally, the length BC is not fixed and it can be obtained as the sum of the initially undeformed length plus a variable deformation.
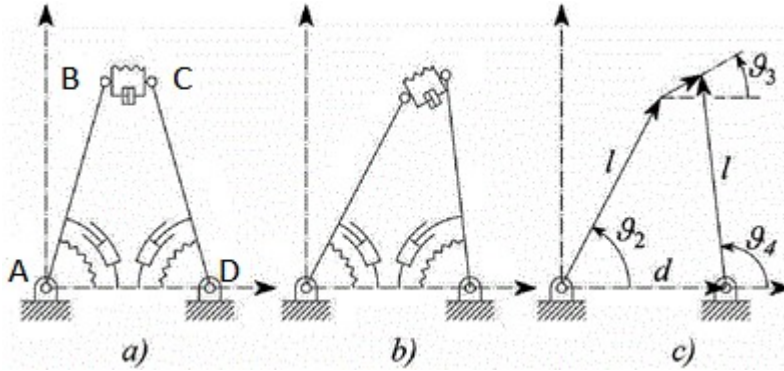

Figure 4.3. The gripping schema

In Figure 4.3 (b),(c), the gripper is represented in an operative condition different from the initial one. In Figure 4.3 (c), following the notation introduced in [70], the angles θ_i represent the relative angular displacements of the two links from their neutral configuration, with $i = 2$ for the left link AB and $i = 4$ for the right one DC. With reference to Figure 4.3 (c), θ_3 represents the relative angular displacements of the link BC from its neutral configuration. The angles are defined according to the counter clockwise rule.

4.3 Gripper mathematical model

Note that, in this work, each length/ orientation in its final configuration is expressed with tilde ˜, while the reference value by hat ˆ. Following this notation, the actual angles $\tilde{\theta}_2$, $\tilde{\theta}_3$ and $\tilde{\theta}_4$ can be expressed as sum of their reference values plus their deformation:

$$\tilde{\theta}_2 = \theta_2 + \hat{\theta}_2$$
$$\tilde{\theta}_4 = \theta_4 + \hat{\theta}_4$$
$$\tilde{\theta}_3 = \theta_3 + \hat{\theta}_3$$

Simple geometric considerations give $\hat{\theta}_2 = \pi - \hat{\theta}_4$.
The reference value for the distance BC, corresponding to zero elastic reaction force of the sample, is denoted by $\hat{u}$, and its actual value by $\tilde{u}$. Therefore, the deformation is equal to $u = \tilde{u} - \hat{u}$. The values of the variables in the neutral condition are given in Table 4.1.

Variable name	Numerical value
$\hat{\theta}_2$	1.44 rad
$\hat{\theta}_4$	1.70 rad
$\hat{\theta}_3$	0 rad
$\hat{u}$	$150 \cdot 10^{-6}$ m

Table 4.1. Constants

The following parameters are defined:

- l is the common length of the two links which constitutes the jaws, i.e. the distances AB and CD;

- d is the distance between the hinges (AD);

- k_2, k_4 and k are the torsional stiffness of the two jaws and the stiffness coefficient of the tissue sample, respectively;

- r_2, r_4 and r (or c_2, c_4 and c) are the viscous damping coefficients of the two jaws and of the sample, respectively;

- I_2 and I_4 are the two jaws moments of inertia around A and D;

- τ_2 and τ_4 are the input torques generated by the comb drives.

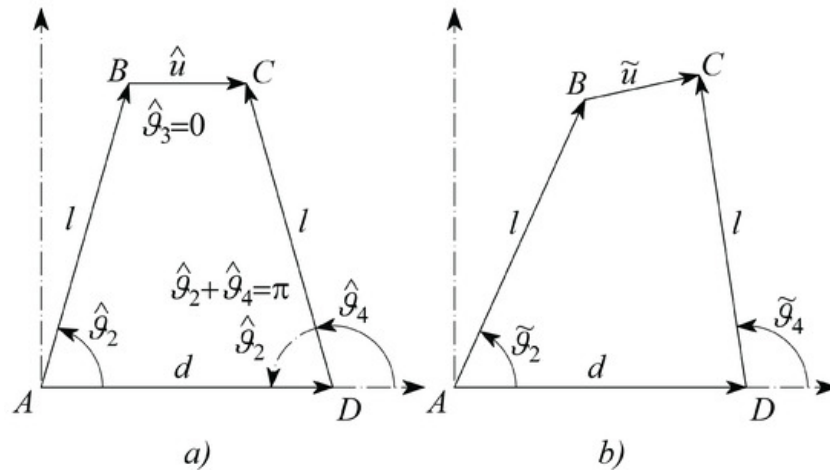Figure 4.4 clarifies all the listed parameters: in initial (a) and operational (b) conditions.

Figure 4.4. Detailed gripping schema

Numerical values of the parameters are given in Table 4.2.

Parameter	Numerical value
d	$5.47 \cdot 10^{-4}$m
l	$1.496 \cdot 10^{-3}$m
I_2, I_4	$1.25 \cdot 10^{-14}$ kg m^2
k_2, k_4	$0.30 \cdot 10^{-6}$ (Kg m^2) / (s^2 rad)
r_2, r_4	$1.24 \cdot 10^{-12}$ (Kg m^2) / (s rad)

Table 4.2. Numerical values of the parameters

With reference to 4.3, with $i = \sqrt{-1}$, it can be write the following closed loop vector equation:

$$le^{i\theta_2} + \vec{BC}e^{i\theta_3} - d - le^{i\theta_4} = 0 \tag{4.1}$$

From 4.1, $\vec{BC}$, expressed as complex number $x = \vec{BC}e^{i\theta_3}$, is

$$x = d - le^{i\theta_2} + le^{i\theta_4}$$
$$= \vec{BC}e^{i\theta_3}$$
$$= \vec{BC}(\cos\theta_3 + \imath\sin\theta_3)$$

whose real and imaginary part are, respectively:

$$\Re x = d - l\cos\tilde{\theta}_2 + l\cos\tilde{\theta}_4$$
$$\Im x = -l\sin\tilde{\theta}_2 + l\sin\tilde{\theta}_4 \tag{4.2}$$

From 4.2, the following expressions can be derived:

$$\theta_3 = \arctan\frac{\Im x}{\Re x} \tag{4.3}$$

$$\tilde{u} = \sqrt{(\Im x)^2 + (\Re x)^2} - \hat{u} \tag{4.4}$$

By simple mathematical operations, the velocity $\dot{u}$ can be computed:

$$\dot{u} = \dot{\theta}_2 l\sin\left(\tilde{\theta}_2 - \theta_3\right) - \dot{\theta}_4 l\sin\left(\tilde{\theta}_4 - \theta_3\right). \tag{4.5}$$

It follows the angular dynamical model of each of the two links.
For the first joint, from the torque balance condition and assuming the inertia of the sample negligible, one has:

$$I_2\ddot{\theta}_2 = -r_2\dot{\theta}_2 - k_2\theta_2 - rl\sin\left(\tilde{\theta}_2 - \theta_3\right)\dot{u} - kl\sin\left(\tilde{\theta}_2 - \theta_3\right)u + \tau_2 \tag{4.6}$$

while for the second one, the corresponding expression is :

$$I_4\ddot{\theta}_4 = -r_4\dot{\theta}_4 - k_4\theta_4 + rl\sin\left(\tilde{\theta}_4 - \theta_3\right)\dot{u} + kl\sin\left(\tilde{\theta}_4 - \theta_3\right)u + \tau_4 \tag{4.7}$$

It should be noted that computation of $\dot{\theta}_3$ is not needed.
In next Chapter, different techniques are applied for tissue parameter estimations with the described micro-gripper.

Chapter 5

Micro-gripper parameters estimation

This Chapter exposes the results obtained with the estimation technique applied to the micro-gripper.

In [69], a solution is proposed, for the measurement of the elastic coefficient of a sample by gripping it and evaluating the reaction elastic force generated. The technique proposed makes use of the actuation of one of the two gripper jaws, leaving the second one passive, and controlling the angular displacement of the passive jaw, by means of a feedback control scheme, in order to keep the variation of the angle at a certain, sufficiently small, value which may assure that only a small force acts on the sample tissue. In [70], the solution is expanded in order to allow to obtain both the viscosity and the damping coefficients of the sample. The solutions proposed in [69] and [70] need a particular input signal waveforms to perform such measurements, that is compatible with clinical diagnostic tests applications, but it is not suitable for a tissue manipulations during surgery operations. The necessity of works with different inputs and in the most heterogeneous operating conditions possible, has driven the research of an online parameters estimator for the microgripper. In the following sections, different solutions are exploited and applied to the system described in previous Chapter.

5.1 Recursive Least Square Filters

The first approach, for system parameters identification [71], is based on recursive least square methods. It makes use of numerical solution and it can be successfully applied under the condition that the set of parameters to be estimated appear linearly in the dynamics. RLS algorithms are adaptive filtering algorithms that recursively finds the coefficients that minimize a weighted linear least squares cost function relating to the input signals. For convergence analysis of RLS estimation algorithms refers to [72].

The micro-gripper model is non-linear but it can be rearranged in the form of a linear time varying systems in which the parameters to be identified are the unknowns, and all the other terms are function of the state and of the output variables, supposed

measurable. This technique has been fruitfully used in several applications, as for example in [73, 74, 75, 76]. The application of these technique to the device has also been described in two works [77, 78].

Equations 4.6 and 4.7 can be rewritten in the following linear parametric form (linear with respect to the unknown parameters):

$$y_i(t) = M_i(t)\omega_i(t), \quad i \in [1, \dots, m], \tag{5.1}$$

where m are the degrees of freedom of the system; in our case, $m = 2$. The term $\omega_i(t)$ is one of the of unknown parameters, whereas $y_i(t)$ and $M_i(t)$ are known quantities coming from the dynamics expressions.

All the quantities in (5.1) are time varying, since they are computed during the dynamics evolution; this means that also $\omega_i(t)$, despite it is referred as the parameters vector, is a function of time because the estimated values change at each update of the procedure, converging to the constant values of the parameters. Equations 4.6 and 4.7, in form of 5.1, become:

$$y_i(t) = I_i\ddot{\theta}_i + r_i\dot{\theta}_i + k_i\theta_i - \tau_i \qquad \text{i} = 1, 2 \tag{5.2}$$

where:

$$M_i(t) = \mp lsin(\theta_i + \hat{\theta}_i - \theta_3)(\dot{u} \;\; u) \qquad \text{i} = 1, 2 \tag{5.3}$$

and

$$\omega_i(t) = \begin{pmatrix} r(t) \\ k(t) \end{pmatrix} \qquad \text{i} = 1, 2 \tag{5.4}$$

With reference to (5.2),(5.3) and (5.4), the general expressions to be defined for a generic recursive least squares (RLS) filtering [79] algorithm, for $i = 1, 2$, are:

$$
\begin{aligned}
\hat{\omega}_i(t) &= \hat{\omega}_i(t-1) + K_i(t)\epsilon_i(t) \\
\epsilon_i(t) &= y_i(t) - \hat{y}_i(t) \\
\hat{y}_i(t) &= \phi_i^T(t)\hat{\omega}_i(t-1) \\
K_i(t) &= Q_i(t)\phi_i(t)
\end{aligned}
\tag{5.5}
$$

where

- $\hat{\omega}_i(t)$ are the current estimation values of $\omega_i(t)$;

- $\hat{y}_i(t)$ are the current estimation values of $y_i(t)$;

- $\epsilon_i(t)$ is the current prediction error;

- the gain $K_i(t)$ determines how much the prediction error affects the update in the parameters estimation;

- $\phi_i(t)$ represents the gradient of the predicted model output with respect to $\omega_i(t)$.

According to 5.5, the current estimation values $\hat{\omega}_i(t)$ are updated recursively as new data are available online.

Two different recursive last squares (RLS) filtering algorithms are adopted for the

on–line identification of the viscous damping and the stiffness coefficient of the tissue sample. The symmetric structure of the dynamics implies that, under a full state measurement, the results obtained choosing $m = 1$ are the same as the case $m = 2$. After having proved the equivalence and the effectiveness of both the cases, the choice in real applications should be driven by the simplicity of the measurement. The difference in the two approaches, with respect to the formulation in equations (5.5), consists in the choice of $Q_i(t)$. The first of the two cases addressed is designed and simulated for $i = 2$ and the second one with $i = 4$.

The first method used, is a forgetting factor based RLS algorithm, that implies, in (5.5) (for $i = 1, 2$), the following choice:

$$Q_i(t) = \frac{P_i(t-1)}{\lambda + \phi_i^T(t) P_i(t-1) \phi_i(t)} \tag{5.6}$$

where the covariance matrix $P_i(t)$ is:

$$P_i(t) = \frac{1}{\lambda} \left(P_i(t-1) - R_i(t) \right) \tag{5.7}$$

with:

$$R_i(t) = \frac{P_i(t-1) \phi_i(t) \phi_i^T(t) P_i(t-1)}{\lambda + \phi_i^T(t) P_i(t-1) \phi_i(t)}$$

The value $\lambda \in R$ is the so-called *forgetting factor*.

According to previous equations, estimation values $\hat{\omega}_i(t)$ are computed in order to minimize the cost function, selected as the sum of residuals squares, weighted by the forgetting factor:

$$\hat{\omega}_i(t) = \arg \min_\theta \sum_{k=1}^{t} \lambda^{t-k} \epsilon_i^2(t) \tag{5.8}$$

It's assumed that the residual $\epsilon_i(t)$, the difference between the estimated and the measured value for $y_i(t)$, is affected by a white noise with covariance equal to 1.

The forgetting factor ($0 < \lambda \leq 1$) is introduced in the cost function 5.8 in order to weight the time sequence of the errors $\epsilon_i(t)$, according to an exponentially decreasing weight. The smaller λ is, the smaller is the contribution of previous samples to the covariance matrix in 5.7. This choice is effective in case of time varying parameters. When dealing with constant parameters, the choice $\lambda = 1$ is usually adopted. Note that the performance of RLS is often sensitive to the chosen forgetting factor value and some extensions are, also, proposed for a variable value of λ [80]. The forgetting factor method is a particular case of the Kalman filter.

The second method used, is a normalized gradient based RLS algorithm, that implies, in equations (5.5) (for $i = 1, 2$), the following choice:

$$Q_i(t) = \frac{\gamma}{|\phi_i(t)|^2 + \beta}, \tag{5.9}$$

where γ is the adaptation gain scaled by the gradient $\phi_i(t)$. The bias β is added to the square norm of the gradient vector in the denominator of the previous expression, in order to prevent critical situations in the values estimation in case $\phi_i(t)$ is close to zero. This algorithm requires only the initialization for the values of the parameters

to be estimated.

It should be noted that RLS filtering algorithm are for static/slowly time varying data: the estimation is more accurate with the increasing number of data.

The two, above described algorithms, are applied in order to validate the approach. The forgetting factor based RLS [77, 78] is applied to the case of $i = 2$, while the normalized gradient RLS [78] is applied to the case of $i = 4$.

5.1.1 Numerical simulations

Numerical simulations, using Matlab® and Simulink® tools, have been performed in order to show effectiveness, benefits and differences of the proposed estimation methods [78].

Three cases have been exploited and simulated:

A. a realistic case with elastic and damping coefficient much greater than the ones of the mechanical structure, with $r = 8.4 \cdot 10^{-6}$ N m s/rad and $k = 2.5 \cdot 10^{-3}$ N m/rad. Note that,in this case, the elastic coefficient is greater than the damping one;

B. a realistic case with elastic and damping coefficient much greater than the ones of the mechanical structure, with $r = 8.4 \cdot 10^{-3}$ N m s/rad and $k = 2.5 \cdot 10^{-6}$ N m/rad. Note that, unlike the previous case, the damping coefficient is greater than the elastic one: this choice has been made in order to check, by comparison, the dependency of the algorithms convergence from the two different mechanical characteristics;

C. a case used for testing the behaviour of the RLS algorithms in when dealing with a very poorly damped factor, with $r = 8.4 \cdot 10^{-11}$ N m s/rad while $k = 2.5 \cdot 10^{-5}$ N m/rad.

For all the simulations and for both the algorithms, the initial coefficients values have been chosen as $r(0) = 10^{-9}$ N m s/rad and $k(0) = 10^{-7}$ N m /rad.

The following additional choices has been made:

- For the forgetting factor based RLS algorithm (5.5), (5.6), the square covariance matrix P has been set as a 2×2 diagonal matrix, with both the diagonal elements equal to 10^{20}, while the forgetting factor λ has been chosen as $\lambda = 0.99$;

- For the normalized gradient based RLS algorithm (5.5), (5.9), the adaptation gain γ has been set as $\gamma = 0.9$ and the normalization bias has been chosen as $\beta = 2.2 \cdot 10^{-16}$.

In Figures 5.1, 5.2, 5.3, 5.4, 5.5, 5.6, the solid line refers to the estimation evolution with the first algorithm (5.5) - (5.6), whereas the dashed one is referred to the second algorithm (5.5) - (5.9); the dotted line corresponds to the true values of the parameters, plotted as a reference.

Simulation results obtained for the first case ($k = 2.5 \cdot 10^{-3}$ Nm/rad and $r = 8.4 \cdot 10^{-6}$ Nms/rad) are depicted in Figure 5.1. As expected, both the algorithms converges in a very small time, but the first faster than the second one.
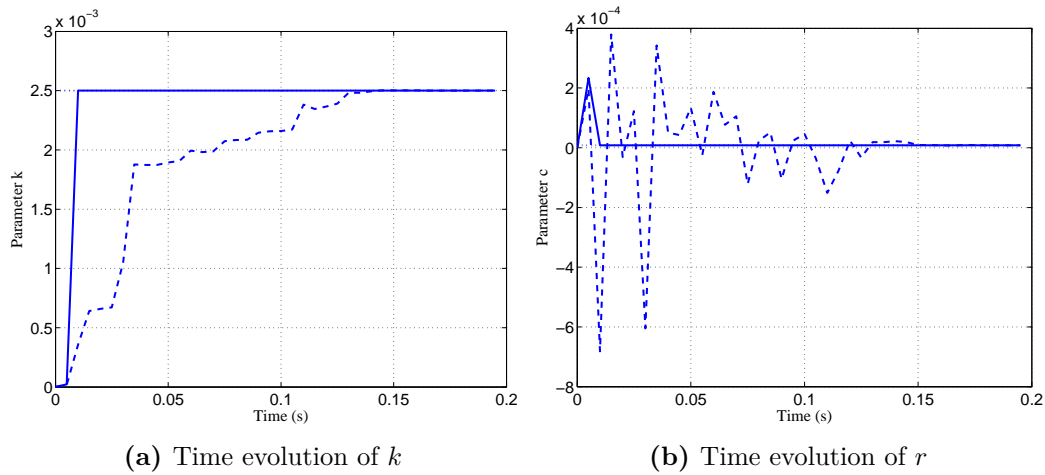
(a) Time evolution of k

(b) Time evolution of r

Figure 5.1. Time evolution of the estimated values, first case, $k = 2.5 \cdot 10^{-3}$ Nm/rad and $r = 8.4 \cdot 10^{-6}$ Nms/rad

However, from Figure 5.2 that depicts the time update estimation of parameter r after the transient, it is possible to observe that the normalized gradient based algorithm (dashed), despite the slower convergence to the true value, is more uniform that the forgetting factor based one (solid).
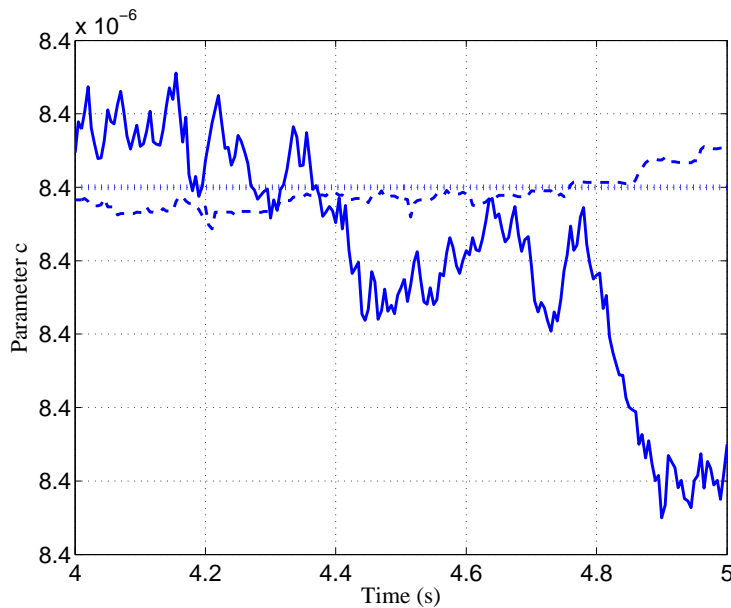

Figure 5.2. r after the transient in first case

Simulation results obtained for the second case ($k = 2.5 \cdot 10^{-6}$ Nm/rad and $r = 8.4 \cdot 10^{-3}$ Nms/rad) are depicted in Figure 5.3. As expected, both the algorithms converges in a very small time, but the first faster than the second one. The difference in the convergence rate for the two approaches is confirmed.
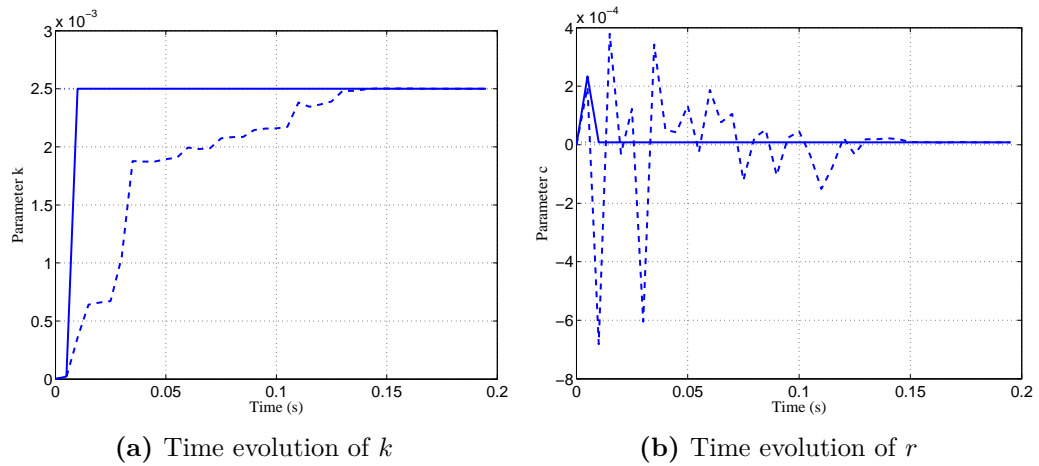
(a) Time evolution of k

(b) Time evolution of r

Figure 5.3. Time evolution of the estimated values, second case, $k = 2.5 \cdot 10^{-6}$ Nm/rad and $r = 8.4 \cdot 10^{-3}$ Nms/rad

In this case the higher uniformity of the result obtained by the forgetting factor based algorithm (solid line) with respect of the normalized gradient based one (dashed line) is evidenced by Figure 5.4, where the evolution of the estimations of k with the two algorithms after the transient is reported.
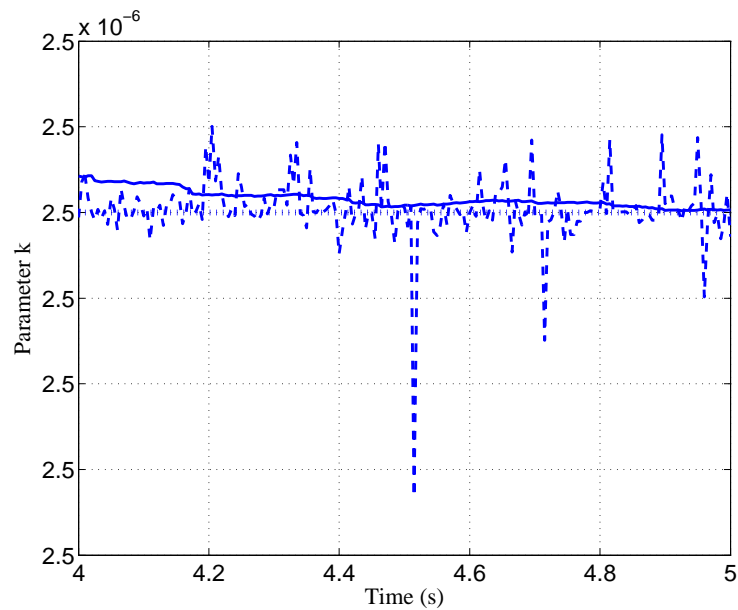

Figure 5.4. r after the transient in first case

Simulation results obtained for the third case ($k = 2.5 \cdot 10^{-5}$ Nm/rad and $r = 8.4 \cdot 10^{-11}$ Nms/rad) are depicted in Figure 5.5. As expected, both the algorithms converges in a very small time, but the first faster than the second one.
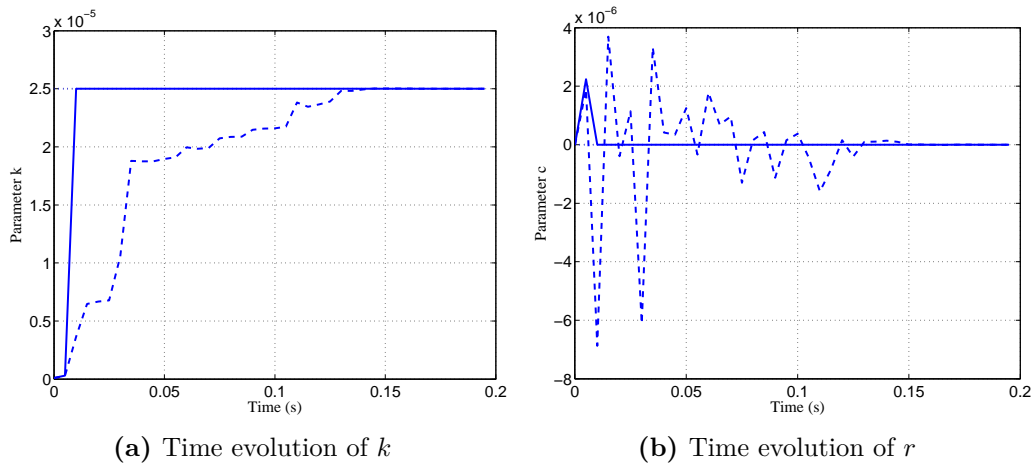
(a) Time evolution of k **(b)** Time evolution of r

Figure 5.5. Time evolution of the estimated values, third case, $k = 2.5 \cdot 10^{-5}$ Nm/rad and $r = 8.4 \cdot 10^{-11}$ Nms/rad

As in the first case, the higher uniformity of the result obtained by the forgetting factor based algorithm (solid line) with respect of the normalized gradient based one (dashed line) is confirmed and evidenced by Figure 5.6, where the evolution of the estimations of k with the two algorithms after the transient is reported.
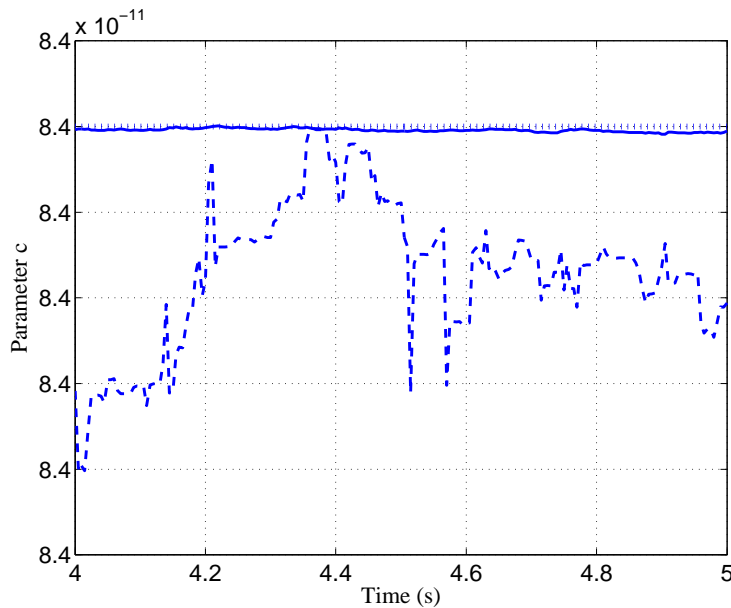

Figure 5.6. r after the transient in third case

The above described simulations, confirm the possibility of using the classical recursive least square algorithms for parameters estimation with the micro-gripper. However, these techniques are not robust with respect of the presence of noise and model parameters uncertainties. For this reason, the possibility of using a Kalman Filter is exploited in next Section.

5.2   Extended Kalman Filter

In last decades, the Kalman Filter (firstly introduced in [81]) becomes a standard in the area of state estimation. It is an optimal linear estimator for linear systems, in presence of additive white noise in the process and in the measurements. An extension of Kalman Filter to non linear systems is the so-called Extended Kalman Filter [82], based on linearisation of state equations at each time step. It follows a description of EKF algorithm [82, 83].

Consider a non-linear discrete-time system:

$$\begin{cases} x(k+1) & = f(x(k), u(k+1)) + w(k+1) \\ y(k+1) & = h(x(k+1)) + v(k+1) \end{cases} \tag{5.10}$$

where

- $x(k+1) \in \mathbb{R}^n$ is the state vector at step $k+1$;

- $y(k+1) \in \mathbb{R}^m$ is the output vector at step $k+1$;

- $u(k+1) \in \mathbb{R}^p$ is the input vector at step $k+1$;

- $w(k+1) \in \mathbb{R}^n$ is the process noise vector at step $k+1$, assumed to be Gaussian with covariance matrix $Q(k+1)$;

- $v(k+1) \in \mathbb{R}^m$ is the measurement noise vector at step $k+1$, assumed to be Gaussian with covariance matrix $R(k+1)$.

The Extended Kalman Filter computes, recursively, the estimation of state, denoted by $\hat{x}(k+1)$, upon observations y up to time k. It can be divided in two phases: Predict phase, in which the estimation of state at current step is computed from previous state estimation, and Update phase, in which the estimation of step computed is refined and updated taking into account observations at current step. Such recursive form of this algorithm allows fast estimation in real-time.

EKF algorithm can be summarized as follows:

- Predict phase

    - Compute the predicted state estimation of x at step $k+1$ with observations up to step k, by applying the function f to previous step state estimation and taking into account current step inputs:

    $$\hat{x}(k+1|k) = f(\hat{x}(k|k), u(k+1))$$

    - Compute the covariance matrix of the predicted state estimation, at step $k+1$ with observations up to step k, as:

    $$P(k+1|k) = F(k+1)P(k|k)F'(k+1) + Q(k+1)$$

    where the state transition matrix is defined as the Jacobian of function f, computed in previous step state estimation:

    $$F(k+1) = \left. \frac{\partial f}{\partial x} \right|_{\hat{x}(k|k), u(k+1)}$$

- Update phase

  - Compute the output prediction error, the so-called innovation:

  $$\tilde{y}(k+1) = y(k+1) - h(\hat{x}(k+1|k))$$

  - Compute the Kalman gain:

  $$K(k+1) = P(k+1|k)H'(k)[H(k+1)P(k+1|k)H'(k+1) + R(k+1)]'$$

  where the output matrix is defined as the Jacobian of function h, computed in previous step state estimation:

  $$H(k+1) = \left.\frac{\partial h}{\partial x}\right|_{\hat{x}(k|k)}$$

  - Update covariance matrix:

  $$P(k+1|k+1) = [I - K(k+1)H(k+1)]P(k+1|k)$$

  This update is susceptible to numerical errors, which can cause loss of positive definiteness of covariance matrix $P(k+1|k+1)$. In order to avoid this issue, P can be computed in the so-called *Joseph form* [84, 85] as:

  $$P(k+1|k+1) = [I - K(k+1)H(k+1)]P(k+1|k)[I - K(k+1)H(k+1)]' + K(k+1)R(k+1)K'(k+1)$$

  - Update state estimation, taking into account current step observation:

  $$\hat{x}(k+1|k+1) = \hat{x}(k+1|k) + K(k+1)\tilde{y}(k+1)$$

It should be noted that EKF has two main drawbacks:

- linearisation can lead to unstable filters if there is no the condition of local linearity;

- Jacobian matrix derivation can be not trivial in practical applications.

The choice of initial values $P(0)$ and $\hat{x}(0)$ has a fundamental role in the convergence of the estimation ([86, 87]). If the user is confident that the initial estimates $\hat{x}(0)$ are close to the true values $x(0)$, low values for elements in $P(0)$ should be chosen. If there are uncertainty on the initial estimation, the elements in $P(0)$ should be high, in order to reflect the lack of confidence in $\hat{x}(0)$. A proposal for $P(0)$ initialisation, made in [88], if true initial value $x(0)$ is known, is:

$$P(0) = diag((\hat{x}(0) - x(0))'(\hat{x}(0) - x(0)))$$

If true initial value $x(0)$ is not known, often reasonably accurate lower ($x_l(0)$) and upper ($x_u(0)$) bounds for it can be used to approximate $\hat{x}(0)$ as $\hat{x}(0) = 0.5(x_l(0) + x_u(0))$.

The dynamics (4.6) and (4.7) of the micro-gripper system, in order to be reconducted to form in (5.10), can be rewritten as:

$$\begin{cases} \dot{x}(t) & = f(t, x) + Gu_\tau(t) \\ y(t) & = Hx(t) \end{cases} \tag{5.11}$$

where

- the state vector $x \in \mathbb{R}^6$ is defined as follows:

$$x(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \\ x_5(t) \\ x_6(t) \end{pmatrix} = \begin{pmatrix} \theta_2(t) \\ \dot{\theta}_2(t) \\ \theta_4(t) \\ \dot{\theta}_4(t) \\ r(t) \\ k(t) \end{pmatrix} \tag{5.12}$$

Note that the two unknown parameters r and k are considered as states of the system itself;

- the input vector $u_\tau \in \mathbb{R}^2$ is composed by the torque:

$$u_\tau(t) = \begin{pmatrix} \tau_2(t) \\ \tau_4(t) \end{pmatrix} \tag{5.13}$$

- the displacement angles θ_2 and θ_4 compose the output vector $y \in \mathbb{R}^2$:

$$y(t) = \begin{pmatrix} \theta_2(t) \\ \theta_4(t) \end{pmatrix} \tag{5.14}$$

- the function $f(t, x)$ is:

$$f(t, x) = \begin{cases} x_2(t) \\ \frac{1}{I2}(-l \sin \hat{\theta}_2 + x_1(t) - \theta_3(t)(x_5(t)\dot{u}(t) + x_6(t)u(t)) - r_2 x_2(t) - k_2 x_1(t)) \\ x_4(t) \\ \frac{1}{I4}(l \sin \hat{\theta}_4(t) + x_3(t) - \theta_3(t)(x_5(t)\dot{u}(t) + x_6(t)u(t)) - r_4 x_4(t) - k_4 x_3(t)) \\ 0 \\ 0 \end{cases}$$

$$\tag{5.15}$$

- the input matrix $G \in \mathbb{R}^{6\times2}$ is:

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

- $H \in \mathbb{R}^{m\times6}$ is the output matrix to be defined according to the test to be performed and the m state variables available.

As the EKF is a discrete filter, the system (5.11) has been discretized by means of Euler method, obtaining the following expressions (to which the noises can be added to be in same form of (5.10)):

$$\begin{cases} x(k+1) &= x(k) + \delta t(f(k, x(k)) + Gu_\tau(k+1)) \\ y(k+1) &= Hx(k+1) \end{cases} \tag{5.16}$$

where δt is the step size. An Extended Kalman Filter algorithm can be applied to system (5.16) in order to estimate its state variables taking into account the measurements available.

5.2.1 Numerical simulations

Numerical simulations, using Matlab® tools, have been performed in order to show the estimation of EKF to micro-gripper system.

In real cases, it could be useful to estimate one of the viscoelastic parameter of the issue, assuming that other one is known. This estimation can find applications when you want to distinguish a healthy tissue from a diseased one: a diseased tissue differs from the healthy one in different characteristics, often the difference relies in viscoelastic properties [89], i.e. in stiffness or viscous damping coefficient. Since we simulate the evolution of the system, all the errors and noise affecting the system are modelled as Gaussian process and measurement noises. A sinusoidal wave torque τ_2 is applied to the first joint, while no torque τ_4 is applied to second one.

In the following simulations, we assume in first case the damping coefficient known with some uncertainties modelled as noise, while in the latter case the stiffness known with some uncertainties modelled as noise.

In first simulation, it's assumed to have measurements of displacements θ_2 and θ_4, velocities $\dot{\theta}_2$ and $\dot{\theta}_4$ and the viscous damping coefficient of the sample r. The matrix $H \in \mathbb{R}^{2 \times 6}$, in this case, is defined as follows:

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

The following values have been set for the simulation:

- real values of parameters $r = 1 \cdot 10^{-8}$ Nms/rad and $k = 0.5 \cdot 10^{-6}$ Nm/rad ;

- initial estimated state $\hat{x}(0) = [\hat{\theta}_2, 0, \hat{\theta}_4, 0, 0, 0]'$;

- initial value of covariance matrix of the predicted state estimation $P(0) = diag([1, 100, 1, 100, 1 \cdot 10^{-8}, 1])$;

- Noise covariance matrices:

$$Q = \begin{bmatrix} 4 \cdot 10^{-6} & 2 \cdot 10^{-5} & 2 \cdot 10^{-6} & 4 \cdot 10^{-5} & 0 & 0 \\ 2 \cdot 10^{-5} & 1 \cdot 10^{-4} & 1 \cdot 10^{-5} & 2 \cdot 10^{-4} & 0 & 0 \\ 2 \cdot 10^{-6} & 1 \cdot 10^{-5} & 1 \cdot 10^{-6} & 2 \cdot 10^{-5} & 0 & 0 \\ 4 \cdot 10^{-5} & 2 \cdot 10^{-4} & 2 \cdot 10^{-5} & 2 \cdot 10^{-4} & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 \cdot 10^{-10} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \cdot 10^{-9} \end{bmatrix}$$

$$R = diag([2 \cdot 10^{-5}, 1 \cdot 10^{-4}, 1 \cdot 10^{-5}, 1 \cdot 10^{-8}])$$

In Figure 5.7 and 5.8 the blue lines correspond to the true values of the parameters, plotted as a reference, while the red lines represents the estimated values.
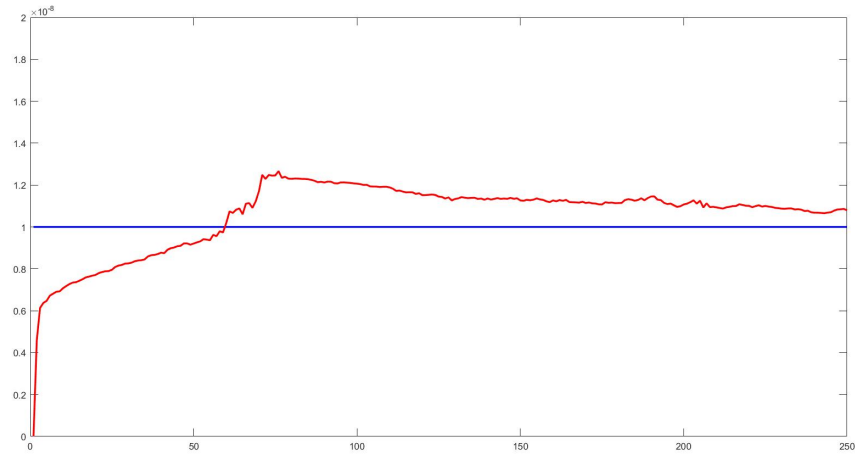
Figure 5.7. Time evolution of the estimated damping coefficient - simulation 1
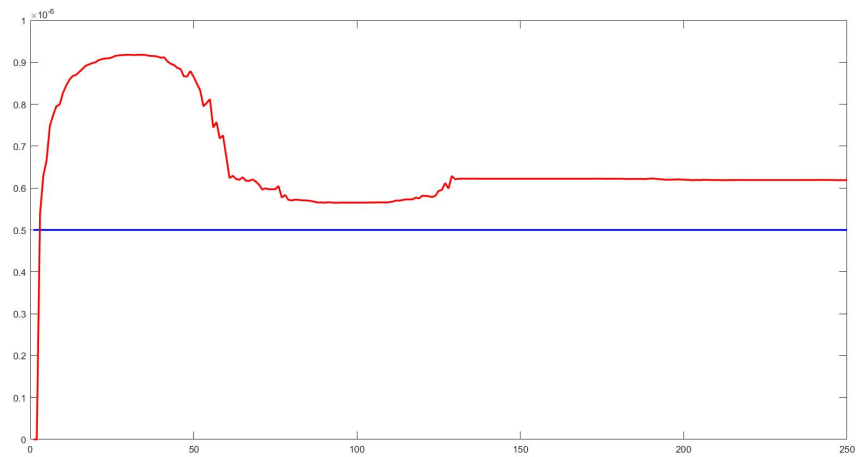

Figure 5.8. Time evolution of the estimated stiffness coefficient - simulation 1

In Figures 5.9 and 5.10, they are depicted the mean squared errors (MSE) of all estimated states.
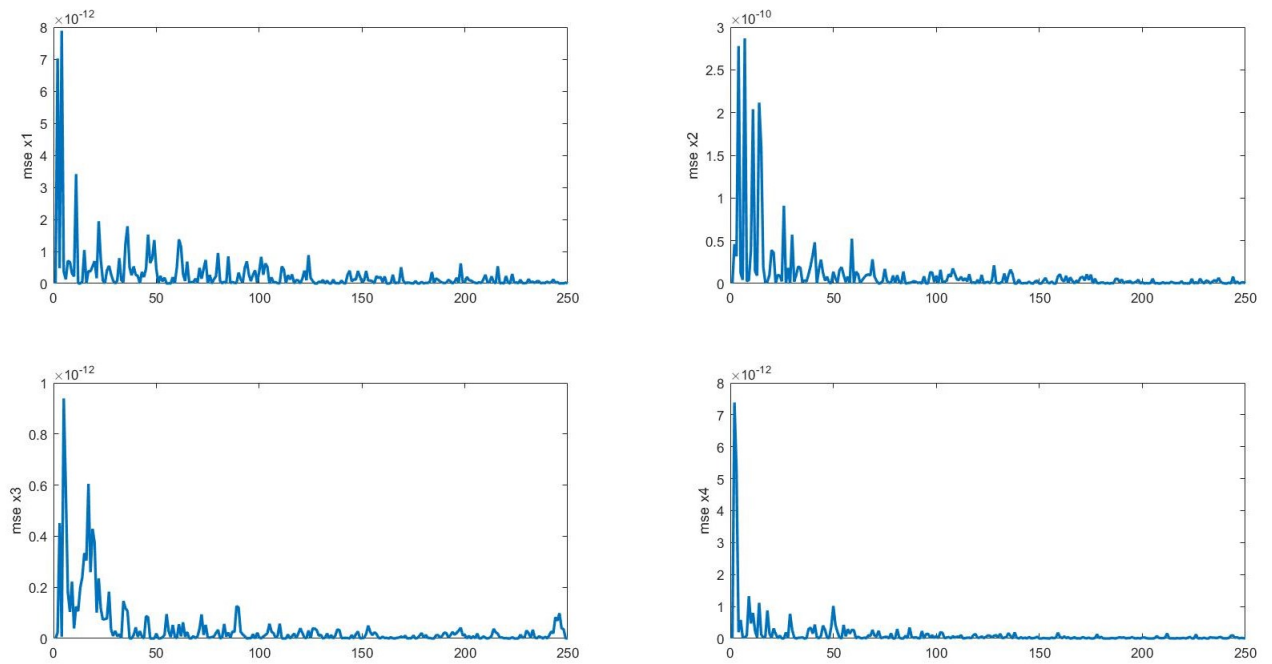
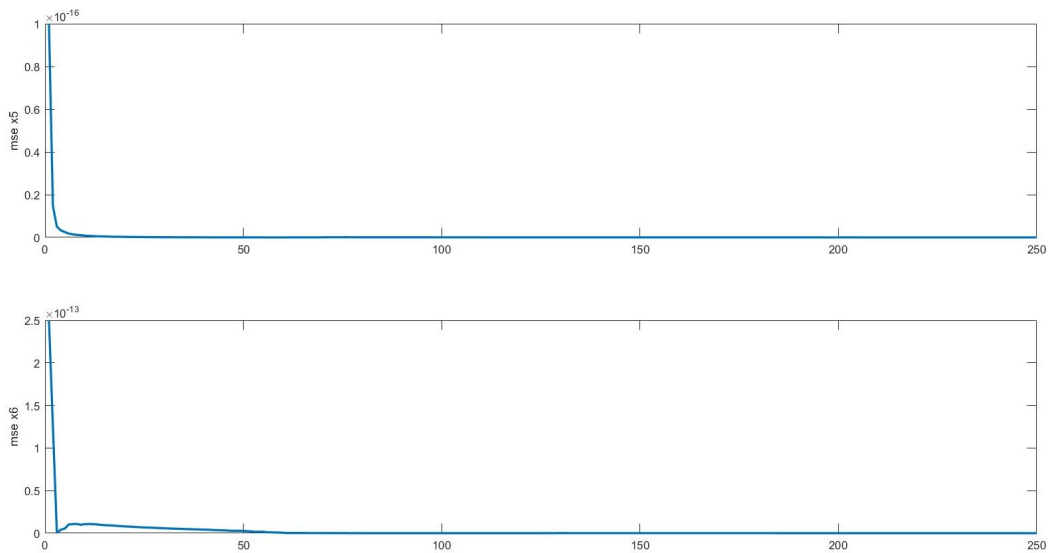**Figure 5.9.** Time evolution of mse of angles and velocities - simulation 1

Figure 5.10. Time evolution of mse of parameters - simulation 1

In second simulation, it's assumed to have measurements of displacements θ_2 and θ_4, velocities $\dot{\theta}_2$ and $\dot{\theta}_4$ and the stiffness coefficient of the sample k. The matrix

$H \in \mathbb{R}^{2 \times 6}$, in this case, is defined as follows:

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

The following values have been set for the simulation:

- real values of parameters $r = 0.6 \cdot 10^{-9}$ Nms/rad and $k = 1 \cdot 10^{-6}$ Nm/rad ;

- initial estimated state $\hat{x}(0) = [\hat{\theta}_2, 0, \hat{\theta}_4, 0, 0, 0]'$;

- initial value of covariance matrix of the predicted state estimation $P(0) = diag([1, 100, 1, 100, 1, 1 \cdot 10^{-8}])$;

- Noise covariance matrices:

$$Q = \begin{bmatrix} 4 \cdot 10^{-6} & 2 \cdot 10^{-5} & 2 \cdot 10^{-6} & 4 \cdot 10^{-5} & 0 & 0 \\ 2 \cdot 10^{-5} & 1 \cdot 10^{-4} & 1 \cdot 10^{-5} & 2 \cdot 10^{-4} & 0 & 0 \\ 2 \cdot 10^{-6} & 1 \cdot 10^{-5} & 1 \cdot 10^{-6} & 2 \cdot 10^{-5} & 0 & 0 \\ 4 \cdot 10^{-5} & 2 \cdot 10^{-4} & 2 \cdot 10^{-5} & 2 \cdot 10^{-4} & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 \cdot 10^{-10} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \cdot 10^{-9} \end{bmatrix}$$
$$R = diag([2 \cdot 10^{-5}, 1 \cdot 10^{-4}, 1 \cdot 10^{-5}, 1 \cdot 10^{-8}])$$

In Figure 5.11 and 5.12 the blue lines correspond to the true values of the parameters, plotted as a reference, while the red lines represents the estimated values.
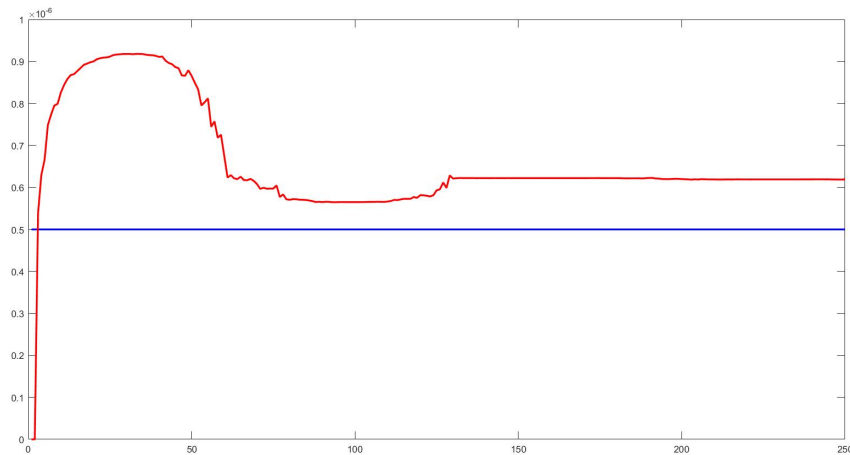

Figure 5.11. Time evolution of the estimated damping coefficient - simulation 2
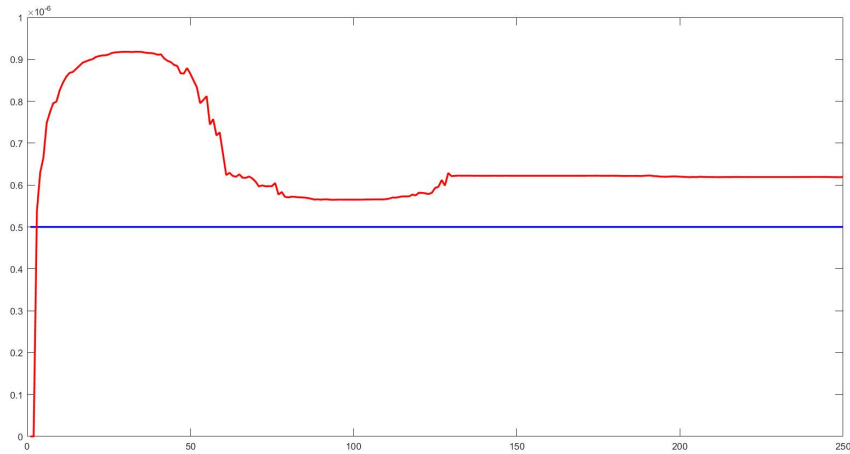
Figure 5.12. Time evolution of the estimated stiffness coefficient - simulation 2

In Figures 5.13 and 5.14, they are depicted the mean squared errors (MSE) of all estimated states.
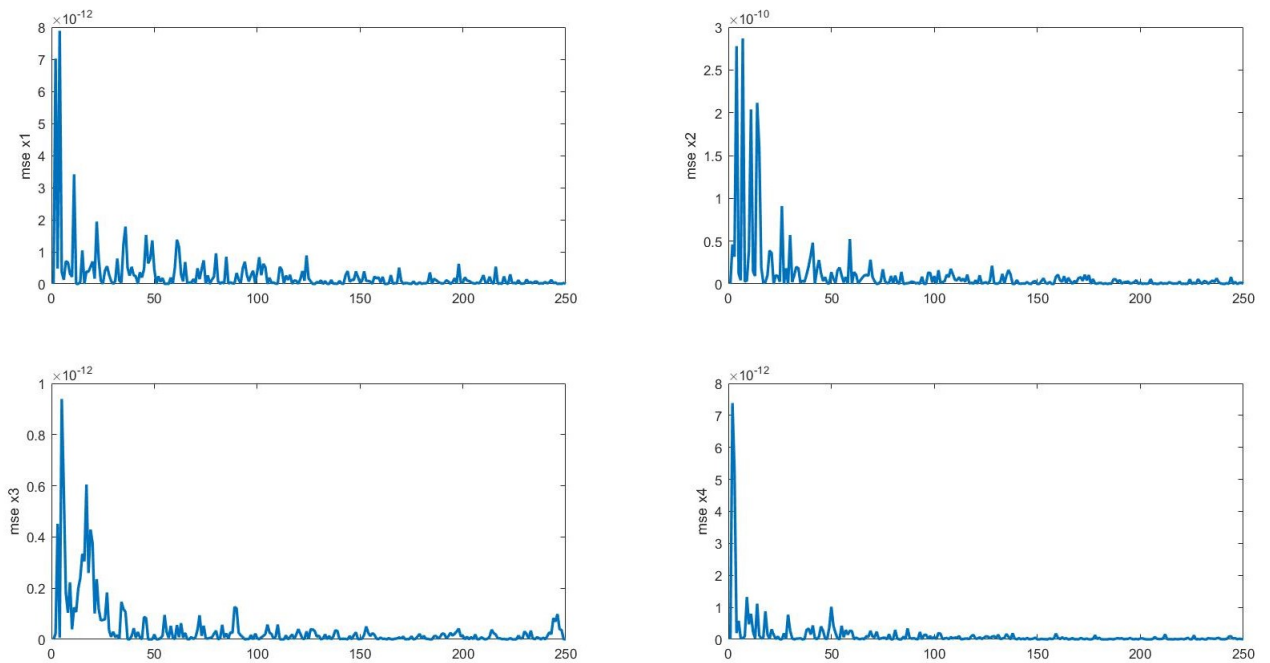

Figure 5.13. Time evolution of mse of angles and velocities - simulation 2
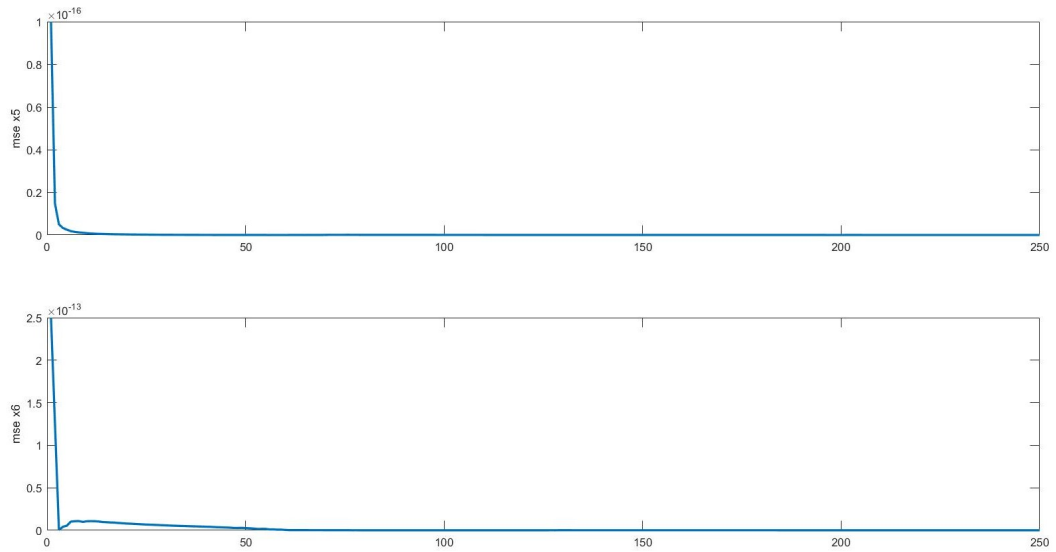
Figure 5.14. Time evolution of mse of parameters - simulation 2

5.3 Conclusions

In this Chapter, the possibility of using a recently constructed micro gripper device for the estimation of the elastic and the damping coefficients of a sample element by pinching it is proved. In Section 5.1, two classical recursive least squares algorithms for the parameter estimation is proposed, while in Section 5.2 an Extended Kalman Filter algorithm for state estimation is proposed. The simulations in 5.1.1 and 5.2.1 show how it is possible to obtain the values of the parameters without the necessity of a specific testing operation, but also during any operative conditions for the gripper.

Next step should be the application of estimation technique to real data, instead of simulated ones, in order to validate the approach also in a real environment. Then, the EKF algorithm applied could be improved, in order to have a parallel estimation of both viscoelastic parameters, exploiting the possibility of applying a dual EKF ([90, 91]) or an Unscented KF ([92]).

Bibliography

[1] S. S. Hacısalihzade, *Biomedical Applications of Control Engineering*, 1st ed. Springer-Verlag Berlin Heidelberg, 2013.

[2] J. B. A. L. J. Fernandez de Canete, C. Galindo, *Automatic Control Systems in Biomedical Engineering*, 1st ed. Springer International Publishing, 2018.

[3] A. T. Azar, *Control Applications for Biomedical Engineering Systems*, 1st ed. Elsevier, 2020.

[4] H. T. Banks, *Modeling and control in the biomedical sciences.* Springer Science & Business Media, 2013, vol. 6.

[5] D. K. Molzahn, F. Dörfler, H. Sandberg, S. H. Low, S. Chakrabarti, R. Baldick, and J. Lavaei, "A survey of distributed optimization and control algorithms for electric power systems," *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 2941–2962, 2017.

[6] X.-M. Zhang, Q.-L. Han, X. Ge, D. Ding, L. Ding, D. Yue, and C. Peng, "Networked control systems: a survey of trends and techniques," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 1, pp. 1–17, 2019.

[7] P. M. Kebria, H. Abdi, M. M. Dalvand, A. Khosravi, and S. Nahavandi, "Control methods for internet-based teleoperation systems: A review," *IEEE Transactions on Human-Machine Systems*, vol. 49, no. 1, pp. 32–46, 2018.

[8] D. Ivanov, S. Sethi, A. Dolgui, and B. Sokolov, "A survey on control theory applications to operational systems, supply chain management, and industry 4.0," *Annual Reviews in Control*, vol. 46, pp. 134–147, 2018.

[9] S. Ahmadzadeh, J. Luo, and R. Wiffen, "Review on biomedical sensors, technologies and algorithms for diagnosis of sleep disordered breathing: Comprehensive survey," *IEEE Reviews in Biomedical Engineering*, pp. 1–1, 2020.

[10] L. Euler, "The seven bridges of königsberg," *The world of mathematics*, vol. 1, pp. 573–580, 1956.

[11] J. L. Gross and J. Yellen, *Handbook of graph theory.* CRC press, 2003.

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

[13] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424–440, 2015.

[14] A. Barve and M. J. Nene, "Survey of flocking algorithms in multi-agent systems," *International Journal of Computer Science Issues (IJCSI)*, vol. 10, no. 6, p. 110, 2013.

[15] H. Guo, C. Zheng, H. H.-C. Iu, and T. Fernando, "A critical review of cascading failure analysis and modeling of power system," *Renewable and Sustainable Energy Reviews*, vol. 80, pp. 9–22, 2017.

[16] Z. Tao, F. Zhongqian, and W. Binghong, "Epidemic dynamics on complex networks," *Progress in Natural Science*, vol. 16, no. 5, pp. 452–457, 2006.

[17] M. A. Rahimian and V. M. Preciado, "Detection and isolation of failures in linear multi-agent networks," *arXiv preprint arXiv:1309.5540*, 2013.

[18] J. A. W. B. Costanzo, D. Materassi, and B. Sinopoli, "Using viterbi and kalman to detect topological changes in dynamic networks," in *2017 American Control Conference (ACC)*, 2017, pp. 5410–5415.

[19] J. A. Costanzo, D. Materassi, and B. Sinopoli, "Inferring link changes in dynamic networks through power spectral density variations," in *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*.   IEEE, 2017, pp. 220–227.

[20] G. Battistelli and P. Tesi, "Detecting topology variations in dynamical networks," in *2015 54th IEEE Conference on Decision and Control (CDC)*.   IEEE, 2015, pp. 3349–3354.

[21] ——, "Detecting topology variations in networks of linear dynamical systems," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 3, pp. 1287–1299, 2017.

[22] S. Roy, M. Xue, G. Battistelli, and P. Tesi, "Comment on "detecting topology variations in networks of linear dynamical systems"," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 1, pp. 187–188, 2019.

[23] F. Küsters and S. Trenn, "Switch observability for switched linear systems," *Automatica*, vol. 87, pp. 121–127, 2018.

[24] M. E. Valcher, "Consensus in the presence of communication faults," in *2019 18th European Control Conference (ECC)*.   IEEE, 2019, pp. 1062–1067.

[25] M. E. Valcher and G. Parlangeli, "On the effects of communication failures in a multi-agent consensus network," in *2019 23rd International Conference on System Theory, Control and Computing (ICSTCC)*.   IEEE, 2019, pp. 709–720.

[26] G. Parlangeli and M. E. Valcher, "On the detection and identification of edge disconnections in a multi-agent consensus network," *arXiv preprint arXiv:2101.06728*, 2021.

[27] D. Patil, P. Tesi, and S. Trenn, "Indiscernible topological variations in dae networks," *Automatica*, 2019.

[28] Y. Zhang, Y. Xia, J. Zhang, and J. Shang, "Generic detectability and isolability of topology failures in networked linear systems," *ArXiv*, vol. abs/2005.04687, 2020.

[29] W. O. Kermack and A. G. McKendrick, "A contribution to the mathematical theory of epidemics," *Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character*, vol. 115, no. 772, pp. 700–721, 1927.

[30] H. S. Rodrigues, "Application of sir epidemiological model: new trends," *arXiv preprint arXiv:1611.02565*, 2016.

[31] M. Y. Li and J. S. Muldowney, "Global stability for the seir model in epidemiology," *Mathematical biosciences*, vol. 125, no. 2, pp. 155–164, 1995.

[32] K. L. Cooke and P. Van Den Driessche, "Analysis of an seirs epidemic model with two delays," *Journal of Mathematical Biology*, vol. 35, no. 2, pp. 240–260, 1996.

[33] P. Di Giamberardino, D. Iacoviello, F. Papa, and C. Sinisgalli, "Dynamical evolution of covid-19 in italy with an evaluation of the size of the asymptomatic infective population," *IEEE Journal of Biomedical and Health Informatics*, 2020.

[34] H. W. Hethcote and P. van den Driessche, "An sis epidemic model with variable population size and a delay," *Journal of mathematical biology*, vol. 34, no. 2, pp. 177–194, 1995.

[35] J. D. Murray, *Mathematical biology: I. An introduction.* Springer Science & Business Media, 2007, vol. 17.

[36] J. Satsuma, R. Willox, A. Ramani, B. Grammaticos, and A. Carstea, "Extending the sir epidemic model," *Physica A: Statistical Mechanics and its Applications*, vol. 336, no. 3-4, pp. 369–375, 2004.

[37] O. Zakary, M. Rachik, and I. Elmouki, "On the impact of awareness programs in hiv/aids prevention: an sir model with optimal control," *Int. J. Comput. Appl*, vol. 133, no. 9, pp. 1–6, 2016.

[38] D. Furushima, S. Kawano, Y. Ohno, and M. Kakehashi, "Estimation of the basic reproduction number of novel influenza a (h1n1) pdm09 in elementary schools using the sir model," *The open nursing journal*, vol. 11, p. 64, 2017.

[39] M. T. Hossain, M. M. Miah, M. B. Hossain *et al.*, "Numerical study of kermack-mckendrik sir model to predict the outbreak of ebola virus diseases using euler and fourth order runge-kutta methods," *American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)*, vol. 37, no. 1, pp. 1–21, 2017.

[40] I. Cooper, A. Mondal, and C. G. Antonopoulos, "A sir model assumption for the spread of covid-19 in different communities," *Chaos, Solitons & Fractals*, vol. 139, p. 110057, 2020.

[41] G. Ellison, "Implications of heterogeneous sir models for analyses of covid-19," National Bureau of Economic Research, Tech. Rep., 2020.

[42] D. Chen, "On the integrability of the sir epidemic model with vital dynamics," *Advances in Mathematical Physics*, vol. 2020, 2020.

[43] A. Ibeas, M. de la Sen, S. Alonso-Quesada, I. Zamani, and M. Shafiee, "Observer design for seir discrete-time epidemic models," in *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*. IEEE, 2014, pp. 1321–1326.

[44] K. H. Degue and J. Le Ny, "An interval observer for discrete-time seir epidemic models," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 5934–5939.

[45] P. Di Giamberardino and D. Iacoviello, "Evaluation of the effect of different policies in the containment of epidemic spreads for the covid-19 case," *Biomedical signal processing and control*, vol. 65, p. 102325, 2021.

[46] H. Behncke, "Optimal control of deterministic epidemics," *Optimal control applications and methods*, vol. 21, no. 6, pp. 269–285, 2000.

[47] D. H. Morris, F. W. Rossine, J. B. Plotkin, and S. A. Levin, "Optimal, near-optimal, and robust epidemic control," *arXiv preprint arXiv:2004.02209*, 2020.

[48] H. J. Ahn and B. Hassibi, "On the mixing time of the sis markov chain model for epidemic spread," in *53rd IEEE Conference on Decision and Control*, 2014, pp. 6221–6227.

[49] N. A. Ruhi and B. Hassibi, "Sirs epidemics on complex networks: Concurrence of exact markov chain and approximated models," in *2015 54th IEEE Conference on Decision and Control (CDC)*, 2015, pp. 2919–2926.

[50] M. Newton and A. Papachristodoulou, "Network lyapunov functions for epidemic models," in *2020 59th IEEE Conference on Decision and Control (CDC)*. IEEE, 2020, pp. 1798–1803.

[51] S. Gracy, P. E. Paré, H. Sandberg, and K. H. Johansson, "Analysis and distributed control of periodic epidemic processes," *IEEE Transactions on Control of Network Systems*, vol. 8, no. 1, pp. 123–134, 2021.

[52] C. Ho, M. J. Kochenderfer, V. Mehta, and R. S. Caceres, "Control of epidemics on graphs," in *2015 54th IEEE Conference on Decision and Control (CDC)*. IEEE, 2015, pp. 4202–4207.

[53] M. Chinazzi, J. T. Davis, M. Ajelli, C. Gioannini, M. Litvinova, S. Merler, A. P. y Piontti, K. Mu, L. Rossi, K. Sun *et al.*, "The effect of travel restrictions

on the spread of the 2019 novel coronavirus (covid-19) outbreak," *Science*, vol. 368, no. 6489, pp. 395–400, 2020.

[54] S. Lai, N. W. Ruktanonchai, L. Zhou, O. Prosper, W. Luo, J. R. Floyd, A. Wesolowski, M. Santillana, C. Zhang, X. Du *et al.*, "Effect of non-pharmaceutical interventions to contain covid-19 in china," *Nature*, vol. 585, no. 7825, pp. 410–413, 2020.

[55] M. Gatto, E. Bertuzzo, L. Mari, S. Miccoli, L. Carraro, R. Casagrandi, and A. Rinaldo, "Spread and dynamics of the covid-19 epidemic in italy: Effects of emergency containment measures," *Proceedings of the National Academy of Sciences*, vol. 117, no. 19, pp. 10 484–10 491, 2020.

[56] M. U. Kraemer, C.-H. Yang, B. Gutierrez, C.-H. Wu, B. Klein, D. M. Pigott, L. Du Plessis, N. R. Faria, R. Li, W. P. Hanage *et al.*, "The effect of human mobility and control measures on the covid-19 epidemic in china," *Science*, vol. 368, no. 6490, pp. 493–497, 2020.

[57] ISTAT, "www.istat.it/it/ - istituto nazionale di statistica website."

[58] W. Chen, H. Khamis, I. Birznieks, N. F. Lepora, and S. J. Redmond, "Tactile sensors for friction estimation and incipient slip detection—toward dexterous robotic manipulation: A review," *IEEE Sensors Journal*, vol. 18, no. 22, pp. 9049–9064, 2018.

[59] C. Yang, Y. Xie, S. Liu, and D. Sun, "Force modeling, identification, and feedback control of robot-assisted needle insertion: a survey of the literature," *Sensors*, vol. 18, no. 2, p. 561, 2018.

[60] A. Khan and Y. Anwar, "Robots in healthcare: A survey," in *Science and Information Conference*. Springer, 2019, pp. 280–292.

[61] K. Tai, A.-R. El-Sayed, M. Shahriari, M. Biglarbegian, and S. Mahmud, "State of the art robotic grippers and applications," *Robotics*, vol. 5, no. 2, p. 11, 2016.

[62] M. G. Antonelli, W. D'Ambrogio, and F. Durante, "Development of a pneumatic soft actuator as a hand finger for a collaborative robot," in *Proceedings of the 2018 2nd International Conference on Mechatronics Systems and Control Engineering*, 2018, pp. 67–71.

[63] M. Savia and H. N. Koivo, "Contact micromanipulation—survey of strategies," *IEEE/ASME Transactions on Mechatronics*, vol. 14, no. 4, pp. 504–514, 2009.

[64] N. P. Belfiore, "Micromanipulation: A challenge for actuation," in *Actuators*, vol. 7, no. 4. Multidisciplinary Digital Publishing Institute, 2018, p. 85.

[65] A. Bagolini, S. Ronchin, P. Bellutti, M. Chistè, M. Verotti, and N. P. Belfiore, "Fabrication of novel mems microgrippers by deep reactive ion etching with metal hard mask," *Journal of Microelectromechanical Systems*, vol. 26, no. 4, pp. 926–934, 2017.

[66] M. Verotti, R. Crescenzi, M. Balucani, and N. P. Belfiore, "Mems-based conjugate surfaces flexure hinge," *Journal of Mechanical Design*, vol. 137, no. 1, p. 012301, 2015.

[67] M. Verotti, A. Dochshanov, and N. P. Belfiore, "Compliance synthesis of csfh mems-based microgrippers," *Journal of Mechanical Design*, vol. 139, no. 2, p. 022301, 2017.

[68] R. Cecchi, M. Verotti, R. Capata, A. Dochshanov, G. B. Broggiato, R. Crescenzi, M. Balucani, S. Natali, G. Razzano, F. Lucchese *et al.*, "Development of micro-grippers for tissue and cell manipulation with direct morphological comparison," *Micromachines*, vol. 6, no. 11, pp. 1710–1728, 2015.

[69] A. Bagolini, P. Bellutti, P. Di Giamberardino, V. Rudas, I.J.and D'Andrea, M. Verotti, A. Dochshanov, and N. Belfiore, "Stiffness characterization of biological tissues by means of mems-technology based micro grippers under position control," in *Mechanisms and Machine Science*, vol. 49, 2018, pp. 939 – 947.

[70] P. Di Giamberardino, A. Bagolini, P. Bellutti, I. J. Rudas, M. Verotti, F. Botta, and N. P. Belfiore, "New mems tweezers for the viscoelastic characterization of soft materials at the microscale," *Micromachines*, vol. 9, no. 1, p. 15, 2018.

[71] A. Astrom and B. Wittenmark, *Adaptive Control*, 2nd ed. Addison Wesley, 199.

[72] E. Eweda and O. Macchi, "Convergence of the rls and lms adaptive filters," *IEEE Transactions on Circuits and Systems*, vol. 34, no. 7, pp. 799–803, 1987.

[73] F. Flacco, A. De Luca, I. Sardellitti, and N. G. Tsagarakis, "Robust estimation of variable stiffness in flexible joints," in *in Proc. of 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011.

[74] C. Lundquist and T. B. Schön, "Recursive identification of cornering stiffness parameters for an enhanced single track model," *IFAC Proceedings Volumes*, vol. 42, no. 10, pp. 1726 – 1731, 2009, 15th IFAC Symposium on System Identification.

[75] A. Vahidi, A. Stefanopoulou, and H. Peng, "Recursive least squares with forgetting for online estimation of vehicle mass and road grade: Theory and experiments," *Vehicle System Dynamics*, vol. 43, no. 1, pp. 31–55, 1 2005.

[76] S. D. Lee and S. Jung, "A recursive least square approach to a disturbance observer design for balancing control of a single-wheel robot system," *2016 IEEE International Conference on Information and Automation (ICIA)*, pp. 1878–1881, 2016.

[77] P. Di Giamberardino, M. L. Aceto, O. Giannini, and M. Verotti, "Dynamic estimation of visco-elastic mechanical characteristics of biological samples under micro manipulation." in *ICINCO (2)*, 2018, pp. 513–520.

[78] ——, "Recursive least squares filtering algorithms for on-line viscoelastic characterization of biosamples," in *Actuators*, vol. 7, no. 4. Multidisciplinary Digital Publishing Institute, 2018, p. 74.

[79] L. Ljung, *System Identification: Theory for the User*, 2nd ed. Prentice Hall, 1999.

[80] A. L. Bruce, A. Goel, and D. S. Bernstein, "Convergence and consistency of recursive least squares with variable-rate forgetting," *Automatica*, vol. 119, p. 109052, 2020.

[81] R. E. Kalman, "A new approach to linear filtering and prediction problems," 1960.

[82] A. Gelb, *Applied optimal estimation.* MIT press, 1974.

[83] L. Ljung, "Asymptotic behavior of the extended kalman filter as a parameter estimator for linear systems," *IEEE Transactions on Automatic Control*, vol. 24, no. 1, pp. 36–50, 1979.

[84] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software.* John Wiley & Sons, 2004.

[85] F. De Vivo, M. Battipede, P. Gili, A. Brandl, and W. Press, "Ill-conditioned problems improvement adapting joseph covariance formula to non-linear bayesian filters," *WSEAS Transactions on Electronics*, vol. 7, pp. 18–25, 2016.

[86] P. Vachhani, S. Narasimhan, and R. Rengaswamy, "Recursive state estimation in nonlinear processes," in *Proceedings of the 2004 American Control Conference*, vol. 1. IEEE, 2004, pp. 200–204.

[87] E. L. Haseltine and J. B. Rawlings, "Critical evaluation of extended kalman filtering and moving-horizon estimation," *Industrial & engineering chemistry research*, vol. 44, no. 8, pp. 2451–2460, 2005.

[88] R. Schneider and C. Georgakis, "How to not make the extended kalman filter fail," *Industrial & Engineering Chemistry Research*, vol. 52, no. 9, pp. 3354–3362, 2013.

[89] C. Rianna and M. Radmacher, "Comparison of viscoelastic properties of cancer and normal thyroid cells on different stiffness substrates," *European Biophysics Journal*, vol. 46, no. 4, pp. 309–324, 2017.

[90] J. Van Lint, S. P. Hoogendoorn, and A. Hegyi, "Dual ekf state and parameter estimation in multi-class first-order traffic flow models," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 14 078–14 083, 2008.

[91] H. Khodadadi and H. Jazayeri-Rad, "Applying a dual extended kalman filter for the nonlinear state and parameter estimations of a continuous stirred tank reactor," *Computers & chemical engineering*, vol. 35, no. 11, pp. 2426–2436, 2011.

[92] E. A. Wan, R. Van Der Merwe, and S. Haykin, "The unscented kalman filter," *Kalman filtering and neural networks*, vol. 5, no. 2007, pp. 221–280, 2001.