UNIVERSITÀ
DEGLI STUDI
DI PADOVA

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

Distributed Optimization Strategies for Mobile Multi-Agent Systems

Ph.D. candidate
Marco Fabris

Advisor
prof. Angelo Cenedese

Director & Coordinator
prof. Andrea Neviani

Ph.D. School in
Information Engineering

Department of
Information Engineering
University of Padova
2019

University of Padova
Department of Information Engineering

Università
degli Studi
di Padova

Ph.D. course in: Information Engineering
Curriculum: Information Science and Technology

Distributed Optimization Strategies for Mobile Multi-Agent Systems

Director: Prof. Andrea Neviani
Advisor: Prof. Angelo Cenedese

Ph.D. candidate: Marco Fabris
Cycle: 32nd (2016-2019)

Year: 2019

"Neo: But we control these machines, they don't control us.
Councillor Hamann: Of course not, how could they? The idea's pure nonsense, but... it does make one wonder just... what is control?
Neo: If we wanted, we could shut these machines down.
Councillor Hamann: Of course... that's it. You hit it! That's control, isn't it? If we wanted, we could smash them to bits. Although if we did, we'd have to consider what would happen to our lights, our heat, our air."

Abstract

In the recent years, the field of automation has witnessed a tendency to employ networked control for groups of agents to fulfill complex tasks arising from robotic surveillance to smart-grid production, from autonomous vehicle servicing to sensor network monitoring and a variety of space-based applications. Such mobile systems are referred to as multi-agent systems (MASs). In this thesis, the main research interest concerns how to design the effective decentralized coordination among autonomous agents, by means of a limited local sensing, to perform common tasks, aiming at reaching a high-quality overall performances. This approach has had a transformative impact in several application domains and, indeed, MASs possess well-known capabilities for autonomous and intelligent operation based on planning, reactivity, learning, proactivity, mobility, adaptivity and reasoning. Furthermore, such elements collaborate to seek a solution to problems that are out of reach for the single entity.
What has led to this rapid progress in the last two decades is a combination of technological advances in price, scale of the platforms themselves, computational performance and a novel breakthrough of how the mobile robots should be arranged algorithmically. At the light of these increasing trends of investigation, it is crucial to reinforce the latest knowledge to keep up with the research demand.

The main objective of this thesis is firstly represented by the investigation of distributed strategies to optimize the design of MASs. In particular, numerous theoretical methodologies borrowed from Optimization Theory, Combinatorial Graph Theory, Dynamic Systems and Control are applied to realistic multi-agent-based models in order to offer new perspectives exploring efficient solutions and consolidate the vast paradigm of MASs. To fully present the potential of the optimization approach applied to MASs, four major research studies are here considered and deeply examined in the theoretical, algorithmic and application point of view.

1. The development of a distributed algorithm to perform dynamic robotic coverage and focus on event with limited sensing capabilities.

2. The design of optimal control laws to accomplish time-invariant robotic formation tracking that also take into account energy consumption aspects.

3. The conceptualization, analysis and optimization of a general regularized distributed linear model to perform a state estimation from relative measurements.

4. The research of spectral properties connecting circulant algebraic entities to the Dirichlet kernel.

Sommario

Recentemente, nel campo dell'automazione, si ha assistito a una tendenza data dell'impiego di controlli su reti formate da gruppi di agenti per realizzare complessi compiti, emergenti dalla sorveglianza robotica alla produzione di smart grids, dai servizi per veicoli autonomi al monitoraggio attuato grazie a reti di sensori e una vasta gamma di applicazioni spaziali. A tali sistemi mobili si fa riferimento con i cosiddetti sistemi multi-agente (MASs). In questa tesi, il principale interesse di ricerca verte sull'indagine di un efficace coordinamento decentralizzato attraverso agenti autonomi, per mezzo di un utilizzo di sensori limitati a misurazioni locali, con lo scopo di svolgere compiti comuni, mirando al raggiungimento di prestazioni globali ad elevata qualità. Questo approccio ha avuto un impatto trasformativo in vari domini applicativi e, infatti, i MASs possiedono note capacità per operazioni intelligenti autonome basate sulla pianificazione, la reattività, l'apprendimento, la proattività, la mobilità, l'adattabilità e il ragionamento. Inoltre, tali elementi collaborano alla ricerca di soluzioni a problemi al di fuori della portata per la singola entità.
Ciò che ha condotto a questo rapido progresso negli ultimi due decenni è la combinazione di avanzamenti tecnologici nel prezzo, nelle dimensioni delle piattaforme stesse, nella capacità computazionale e in un'innovativa svolta sul come i robot mobili dovrebbero organizzarsi metodicamente. Alla luce di questo crescente andamento di investigazione, è cruciale rinforzare le conoscenze recenti per stare al passo con le esigenze di ricerca.

Il principale obiettivo di questa tesi è in primo luogo rappresentato dall'investigazione di strategie distribuite per ottimizzare la progettazione di MASs. In particolare, numerose metodologie teoriche prese in prestito dalla teoria dell'ottimizzazione, dalla teoria combinatoria dei grafi, dai sistemi dinamici e dal controllo sono applicate a realistici modelli multi-agente al fine di offrire nuove prospettive in grado di esplorare soluzioni efficienti e consolidare il vasto paradigma dei MASs. Allo scopo di presentare appieno il potenziale dell'approccio di ottimizzazione applicato ai MASs, quattro studi predominanti vengono qui considerati e largamente esaminati da un punto di vista teorico, algoritmico e applicativo.

1. Lo sviluppo di un algoritmo distribuito per eseguire copertura dinamica con agenti robotici e focalizzazione attorno a un evento mediante sensori a capacità limitata.

2. La progettazione di leggi ottimali per governare un gruppo di robot e realizzare un inseguimento tempo-invariante in formazione tenendo conto del consumo energetico.

3. Il concepimento, l'analisi e l'ottimizzazione di modelli lineari distribuiti e regolarizzati per portare a termine una stima dello stato a partire da misure relative.

4. La ricerca di proprietà spettrali che connettono entità algebriche dotate di struttura circolante al nucleo di Dirichlet.

Acknowledgments

First of all, I would like to thank my advisor prof. Angelo Cenedese for his supervision which started four years ago, during my master thesis and continued during the exciting period of my PhD. I express all my thankfulness for his patience and for having been a precious guide while discovering a new world made of intriguing and challenging problems.

I would also like to deeply thank prof. John Hauser for having supervised my stay at CU in Boulder, who had been a mentor together with prof. Cenedese. His help has been indispensable while moving to the USA and to broaden my knowledge.

A thank goes to each of the professors and researchers in the University of Padova, since, during these eight years as an academic student they all have forged my mind for the worst case scenarios and contributed to grow me as a cultured scholar, aiming for an advanced education.

Another special thank goes to prof. Gian Antonio Susto, prof. Daniel Zelazo, prof. Ming Cao and prof. Subhrakanti Dey, which I had the pleasure to meet and know in Padova, for their availability, time for consultation and the opportunities they offered me.

To conclude the research-side and begin the personal-side thanks, I must mention my colleagues, especially the SPARCS members (Angelo, Giulia, Luca, Riccardo, Alessandra, Nicola), all the PhD students and post-docs I've known for having shared together and overcome most of the difficulties in these three years.

All my friends scattered around the world (Europe, USA, Japan, Brazil and Turkey), with a special mention to my fellow Andrea, the most astonishing guy I've ever met, and the buddies of Colorado for having accepted me as part of their circle and given all the necessary assistance in hard circumstances. Life without them would be much more empty and sad.

A special thanks goes to my parents, Luciano and Anna, and the rest of my family, which have provided support so far, despite of the limited possibilities. I've always been taught to be tolerant, friendly, easy-going, determined and well-mannered

Finally, I'd like to thank all the people who have believed in me and most important, those who have criticized me, directly or indirectly, since they have been giving me the tools – and the grit – to improve and become a better person.

Acronyms & Abbreviations

2D	two-dimensional
3D	three-dimensional
AACC	American Automatic Control Council
Alg.	Algorithm
APF	Attractive Potential Function
AUV	Autonomous Underwater Vehicle
CGT	Combinatorial Graph Theory
Chap.	Chapter
CO	Combinatorial Optimization
Conj.	Conjecture
Cor.	Corollary
CU	University of Colorado
CV	Calculus of the Variations
DCS	Distributed Control Systems
DE	Distributed Estimation
Def.	Definition
DO	Distributed Optimization
DS&C	Dynamic Systems and Control
e.g.	exempli gratia
ESR	Essential Spectral Radius
Ex.	example
FC	Formation Control
FG	Formation Graph
Fig.	Figure
FIR	Finite Impulse Response
FT	Formation Tracking
GCD	Greatest Common Divisor
GPS	Global Positioning System
HBVP	Hamiltonian Boundary-Value Problem
i.e.	id est
IEEE	Institute of Electrical and Electronics Engineers
IFAC	International Federation of Automatic Control
Lem.	Lemma
l.h.s.	left hand side

LQ	Linear Quadratic
MAS	Multi-Agent System
MSE	mean squared error
NCS	Networked Control System
NO4MAS	Networked Optimization for Multi-Agent Systems
NLP	Nonlinear Programming problem
OC	Optimal Control
OIFT	Optimal Time-Invariant Formation Tracking
OT	Optimization Theory
PD	Proportional Derivative (controller)
Ph.D./PhD	Philosophiæ Doctor
PP	Proximal Point (algorithm)
PRONTO	Projection Operator based Newton's method for Trajectory Optimization
Prop.	Proposition
RM	Relative Measurements
RT	Rigidity Theory
r.h.s.	right hand side
Sec.	Section
Subsec.	Subsection
TC	Tracking Control
QoS	Quality of Service
RPF	Repulsive Potential Function
SPARCS	Space and Aerial Control Systems
Thm.	Theorem
TILQ	Time-Invariant Linear Quadratic
TO	Trajectory Optimization
UAV	Unmanned Aerial Vehicle
USA	United States of America
w.l.o.g.	without loss of generality
w.r.t.	with respect to

Notation

The majority of symbols adopted for this manuscript is defined in the appendix: the interested reader is thus addressed to Sec. B. All the statements belonging to the latter have to be intended of global usage for the whole document. In the following lines, the most used symbols together with further basic notation are reported as a preliminary.

Generic entities

M	dimension of Euclidean spaces (positive natural number)
N	degrees of freedom/ generic dimension/ generic natural number
Ω	generic set
ω	generic element of a set (usually a real or complex number)
$\mathbf{\Omega}$	generic matrix, note: all matrices are bold uppercase
$\boldsymbol{\omega}$	generic vector, note: all vectors are bold lowercase

Numerical constants

π	pi
e	Neper number
$\mathbf{i}$	imaginary unit
$\mathbf{0}_N$	zero vector of dimension N
$\mathbb{1}_N$	vector of dimension N whose component are equal to 1
$\mathbf{Z}_{N_1 \times N_2}$	zero matrix of dimensions $N_1 \times N_2$
$\mathbf{Z}_N$	zero matrix of dimensions $N \times N$
$\mathbf{I}_N$	identity matrix of dimension N

Set operations and relations

$\subset$	proper subset of
$\subseteq$	improper subset of
$\overline{\Omega}$	complement subset of set Ω
$\in$	belongs to
$\notin$	does not belongs to
$\Omega_1 \cup \Omega_2$	union of set Ω_1 with set Ω_2
$\Omega_1 \cap \Omega_2$	intersection between set Ω_1 with set Ω_2
$\Omega_2 \setminus \Omega_1$	elements of Ω_2 which are not in Ω_1
$\bigcup$	union operator
$\bigcap$	intersection operator
$\lvert\Omega\rvert$	cardinality of finite set Ω

Generic relations

$=$	equality relation
$\simeq$	approximate equal to
$>$	greater than/ positive (for vectors, matrices)
$\geq$	equal or greater than / nonnegative (for vectors, matrices)/ subgroup relation
$\succ$	positive definite (for matrices)
$\succeq$	positive semi-definite (for matrices)
$<$	less than
$\leq$	equal or less than
$\prec$	negative definite (for matrices)
$\preceq$	negative semi-definite (for matrices)
$\forall$	for all
$\exists$	there exists
$:=$	assignment of a quantity
$\leftarrow$	stored variable
$\rightarrow$	function relation
$\mapsto$	maps to

Well-known sets and classes

$\varnothing$	empty set
$\mathbb{N}$	set of the natural numbers
$\mathbb{Z}$	set of the integer numbers
$\mathbb{R}$	set of the real numbers
$\mathbb{C}$	set of the complex numbers
$\mathbb{S}^1$	unit radius circle
$\mathscr{C}^i$	class of continuous functions up to the i-th derivative
$< \infty$	finite values/ convergent sequence
$\mathrm{stoc}(\mathbb{R})$	class of stochastic matrices
$\mathrm{stoc}^2(\mathbb{R})$	class of doubly stochastic matrices
$\mathcal{X}$	topological space

Mathematical quantities

$\lambda_i^{\mathbf{\Omega}}$	i-th eigenvalue of matrix $\mathbf{\Omega}$
$\boldsymbol{\varpi}_i^{\mathbf{\Omega}}$	i-th eigenvector of matrix $\mathbf{\Omega}$
$\boldsymbol{\upsilon}_i^{\mathbf{\Omega}}$	i-th left eigenvector of matrix $\mathbf{\Omega}$
h	cost functional
L	maximum length (natural number)
$\mathbf{\Gamma}$	curve in a space
z	generalized frequency (variable for the Z-transform)

Standard numeric operators

χ_Ω	characteristic function over set Ω
$\max(\Omega)$	maximum element in Ω
$\min(\Omega)$	minumum element in Ω
$\arg\min\ h$	argument that minimizes the cost functional h
$\exp(\cdot)$	exponential function
$\operatorname{sech}(\cdot)$	hyperbolic secant
$\operatorname{arcosh}(\cdot)$	inverse hyperbolic cosine
$\lfloor\cdot\rfloor$	floor operator
$\lceil\cdot\rceil$	ceiling operator
$\Re$	real part of a complex number
$\Im$	imaginary part of a complex number
$\lvert\omega\rvert$	modulus of $\omega \in \mathbb{C}$
Δ	scalar variation operator, e.g. $\Delta t = t_{final} - t_{initial}$
$\sum$	summation symbol
$\prod$	product symbol
$\frac{d}{d\omega}$	derivative w.r.t. ω
$\frac{\partial}{\partial\omega}$	partial derivative w.r.t. ω
$\int l(\omega)d\omega$	integral of function $l(\omega)$ w.r.t. ω
$\nabla_{\mathbf{x}}$	gradient w.r.t. $\mathbf{x}$
$\mathcal{H}_{\mathbf{xx}}$	Hessian w.r.t. $\mathbf{x}$

Physical quantities, Control Theory

t	discrete time instant
T	final time horizon (real and positive)
T_s	sampling time
τ	continuous time instant
$\mathbf{p}_i$	position of the i-th agent such that $\mathbf{p}_i \in \mathbb{R}^M$
$\dot{\mathbf{p}}_i$	velocity of the i-th agent such that $\dot{\mathbf{p}}_i \in \mathbb{R}^M$
$\mathbf{p}_c$	centroid position, $\mathbf{p}_c \in \mathbb{R}^M$
$\dot{\mathbf{p}}_c$	centroid velocity, $\dot{\mathbf{p}}_c \in \mathbb{R}^M$
$\mathbf{x}$	state, variable or unknown vector
$\mathbf{u}$	control input vector
U	generic space for a control input
X	generic space for the state/ variable
$\mathcal{T}$	trajectory manifold
$\mathcal{P}$	PRONTO map
ϑ	generic parameter

Algebraic operators

$[\boldsymbol{\omega}]_i$	entry i of vector $\boldsymbol{\omega}$
$[\boldsymbol{\Omega}]_{ij}$	entry (i,j) of matrix $\boldsymbol{\Omega}$
$\boldsymbol{\Omega}^{-1}$	inverse of matrix $\boldsymbol{\Omega}$
$\boldsymbol{\Omega}^{\top}$	transpose of matrix $\boldsymbol{\Omega}$
$\boldsymbol{\Omega}^{*}$	conjugate transpose of matrix $\boldsymbol{\Omega}$
$\boldsymbol{\Omega}^{-\top}$	inverse and transpose of matrix $\boldsymbol{\Omega}$
$\boldsymbol{\Omega}^{\dagger}$	pseudo-inverse of $\boldsymbol{\Omega}$
$\text{span}\{\boldsymbol{\omega}\}$	space spanned by vector $\boldsymbol{\omega}$
$\text{im}\,\{\boldsymbol{\Omega}\}$	range of matrix $\boldsymbol{\Omega}$
$\ker\{\boldsymbol{\Omega}\}$	kernel of matrix $\boldsymbol{\Omega}$
$\det(\boldsymbol{\Omega})$	determinant of the matrix $\boldsymbol{\Omega}$
$\text{adj}(\boldsymbol{\Omega})$	adjugate matrix of $\boldsymbol{\Omega}$
$\text{rank}(\boldsymbol{\Omega})$	rank of the matrix $\boldsymbol{\Omega}$
$\text{tr}(\boldsymbol{\Omega})$	trace of the matrix $\boldsymbol{\Omega}$
$\langle \boldsymbol{\omega}_i, \boldsymbol{\omega}_j \rangle_{\boldsymbol{\Omega}}$	scalar product between vectors $\boldsymbol{\omega}_i, \boldsymbol{\omega}_j$ according to matrix $\boldsymbol{\Omega}$
$\|\boldsymbol{\omega}\|_{\boldsymbol{\Omega}}$	norm of vector $\boldsymbol{\omega}$ weighted by matrix $\boldsymbol{\Omega}$
$\|\boldsymbol{\omega}\|_N$	N-norm of vector $\boldsymbol{\omega}$
$\text{Diag}(\boldsymbol{\omega})$	diagonal matrix whose diagonal elements are specified by vector $\boldsymbol{\omega}$
$\text{diag}(\boldsymbol{\Omega})$	vector given by the diagonal of matrix $\boldsymbol{\Omega}$
$\otimes$	Kronecker product
$\Lambda(\boldsymbol{\Omega})$	spectrum of matrix $\boldsymbol{\Omega}$
$\mathscr{R}(\cdot,\cdot)$	Rayleigh quotient

Graph Theory

$\mathcal{G}$	(undirected) graph
$\mathcal{V}$	vertex set
v_i	i-th vertex
n	number of vertices in a graph
$\mathcal{E}$	edges set
e_{ij}	edge connecting vertex v_i to vertex v_j
E	number of edges in a graph
$\mathcal{N}_i$	neighborhood of the i-th vertex
$\deg(v_i)$	degree of the i-th vertex
$\mathbf{A}$	adjacency matrix
$\mathbf{D}$	degree matrix
$\mathbf{E}$	incidence matrix
$\mathbf{L}$	graph Laplacian matrix
$\boldsymbol{\mathcal{L}}$	normalized graph Laplacian matrix
$\mathscr{R}$	Randić matrix

Contents

1

Introduction

"It's the question that brought you here."
Trinity

Contents

1.1 Technological trends in the contemporary networked era

According to Goldberg (2019), the original robots relied on collaboration. In the play that coined the word 'robot' (R.U.R., 1920, by the Czech writer Karel Čapek) robot workers acted cooperating to rebel against unfair working conditions. And the first real robots, designed during World War II to handle radioactive materials, moved their mechanical limbs under the close supervision of human 'tele-operators' who utilized levers behind protected walls.
Since then, almost all the members of the Robotics community, had assumed that robots must carry their own power supply, memory and computing circuitry and be self-contained. This assumption imposed rigid design constraints, limiting the capability of robots to manage uncertainty and adapt to changing conditions.
Nevertheless, over the past decade robots have begun to collaborate again, accelerated by progress in cloud and networking computing. Contemporary robots are immersed in a networking ecosystem that involve massive remote data hubs, sensors, data streams, distributed computing and a myriad of human inputs. Robots can download software and data on demand, and accomplish stochastic motion planning and learning remotely both online and offline. This novel generation of robots will be able to deal better with unpredictable scenarios and environments, and integrate safely and usefully in our world.

In this wide-ranging overview, some of the most fascinating questions and challenges for analyzing, designing and governing networked control systems are introduced by focusing on the engineering constraints induced by the geometric, combinatorial, algebraic and analytic properties of the information-exchange mechanism.

1.1.1 Networked Control Systems

Authors in Mesbahi and Egerstedt (2010) affirm that Network Science has emerged as a powerful conceptual paradigm in Engineering and Science. Phenomena and constructs such as random and small-world networks, interconnected networks, and phase transition nowadays arise in a large variety of research literature, ranging across statistical physics, social networks, economics, sensor networks and of course multi-agent control and coordination. The reason for this uncommon attention to Network Science is twofold. On the one hand, in a number of disciplines – particularly in material and biological sciences – it has become vital to obtain a deeper understanding of the role that inter-elemental interactions play in the collective functionality of multilayered systems. On the other hand, technological advances have facilitated an ability to synthesize networked engineering systems – such as those found in sensor networks, multi-vehicle systems and nanostructures – that resemble, sometimes remotely, their natural counterparts in terms of their operational and functional complexity.

As stated in Zhang, Han, Ge, Ding, Ding, Yue, and Peng (2019b), the rapid development of network technologies has led much variability to people's life. Modern communication networks can provide reliable and swift communication between any two or more physical plants located in different places. Such outstanding features make communication networks extensively utilized to connect control components within a control loop, bringing to the so-called Networked Control Systems (NCSs). NCSs have been applied in several areas, such as industrial automation, environments, space exploration, robots, remote diagnostics and troubleshooting, aircraft, manufacturing plant monitoring, automobiles and teleoperations.

An NCS inherits both advantages and disadvantages from communication networks. On the one hand, communication networks allow data packets to be shared among control components, which means that some traditional point-to-point wiring in the installation of a control system may be avoided. As a result, control cost incurred from installation and maintenance can be reduced significantly. Besides, the well-used communication protocols can ensure data packets to be successfully transmitted between control components, which renders an NCS of high reliability. On the other hand, communication networks transmit data packets only in the form of digital signals rather than continuous signals. This means that signals from physical plants should be sampled and quantized before being transmitted. Furthermore, since the network bandwidth is limited, network traffic congestion is unavoidable, usually leading to some constraints such as packet dropouts and network-induced delays.

Two of the key research directions for NCSs are represented by '*control of networks*' and '*control over networks*' Murray (2003). 'Control of networks' aims at providing a good quality of service (QoS) of communication networks such that NCSs can fulfill satisfactory control performance. In this framework, network resources can be utilized efficiently and network data flows perform very well. 'Control over networks' aims at devising suitable control strategies to reduce the effects of network imperfections on desired control performance. Clearly, 'control of networks' focuses on how to improve the QoS of networks while 'control over networks' on how to enhance control robustness

against network constraints. The latter falls into the scope of control systems while the former is within the field of computer science. Among the results reported by literature in control over networks, five issues have gained much attention: quantization control, sampled-data control, networked control, security control and event-triggered control. Nevertheless, in this thesis, the emphasis is placed in a third sub-field of networked systems, historically represented by the so-called *multi-agent systems* Zampieri (2008).

1.2 Multi-agent systems

Multi-agent systems (MASs) cope with the study of how network interactions between network components and its architecture influence global control targets. More precisely, the problem here is to understand how local laws describing the behavior of individual agents influence the global behavior of the networked system.

Nature has created a large number of multi-agent systems, where local interaction rules/mechanisms are exploited at different levels by groups of agents to attain a common group goal Chen and Ren (2019). Flocks of birds and schools of fish are typical examples of MASs, which have fascinated scientists from diverse disciplines, such as Biology, Computer Science and Physics. Thanks to the parallel characteristics, MASs can be employed to solve engineering problems that are difficult or impossible for an individual agent to accomplish. For instance, the sub-field of Swarm Robotics aims to integrate principles and theoretical background of swarm behaviors with theory and methodology describing principles of self-organizing adaptation and cooperative localization of autonomous robots leading to a flexible stand-alone system, as depicted in Fig. 1.1.
MASs are more robust – the malfunction of the single agent or a small portion of the group typically will not affect the functionality of the system; MASs are scalable – no matter what the size of the system is, the communication and computation costs of MASs are kept at a reasonably low level.

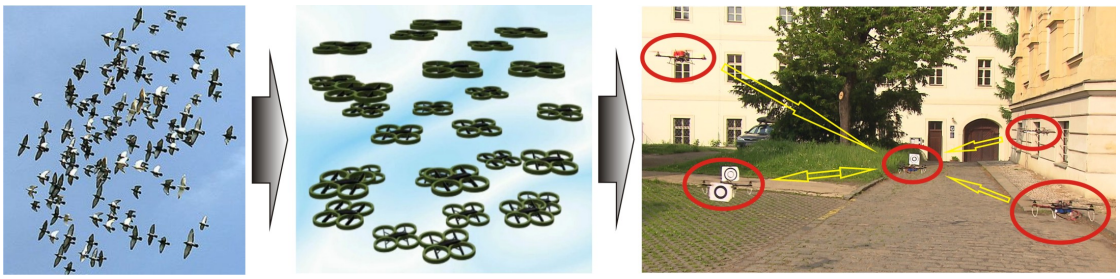


Figure 1.1. Micro aerial vehicles stabilized relatively to their neighbors within a formation or a swarm. [*Credits* : http://mrs.felk.cvut.cz/research/swarm-robotics]

Although research in the field of MASs can be traced back long ago, it was only at the beginning of the 21st century that MASs emerged as a separate subject. In 2005, the paper "Coordination of Groups of Mobile Autonomous Agents Using Nearest Neighbor Rule" won the prestigious George S. Axelby outstanding paper award from IEEE Transactions on Automatic Control, a top-notched journal in systems and control. Then, the International Federation of Automatic Control (IFAC) and American Automatic Control Council (AACC) organized a series of workshops and conferences focusing on MASs. IFAC

established several technical committees having close ties with MASs. New journals with a major interest in MAS or network systems are established, e.g., IEEE Transactions on Network Sciences and Engineering and IEEE Transactions on Control of Network Systems.

As a critical enabler of several research fields, MASs are quintessentially multidisciplinary. In the search for applications and theories, biologists, computer scientists, physicists, and others have all contributed to the development of MASs. In 1987, Craig Reynold proposed three rules that lead to simulated flocking: velocity matching, collision avoidance and flock centering Reynolds (1987). On the other perspective, the study on MASs also provides insights for the related fields, leading to innovative developments in the latter. A cornerstone of MAS control is the mathematical rigor where stability and convergence analysis plays a central role, maybe an inheritance from Control Theory.
Although MAS Theory shares some analogies with other branches of natural sciences, there exists some fundamental diversity. In natural sciences, methodological reductionism plays a key role and has achieved great success, which attempts to explain entire systems in terms of their elements. Reductionism assumes that the interactions among system components (subsystems) are not essential and their effect is negligible. However, the assumption failed to work for MASs. The goal of MAS study is to understand and exploit the local interaction rules among its elements, from which a global behavior can emerge. As a result, methodological reductionism becomes less useful for MAS study. Compared with single agent control where there already exists interaction among several systems components, e.g., control component and sensing component, MASs add another layer of interactions at a higher level. How to understand and exploit these interactions is the key to the success of MASs.

In many engineering fields, it is not unusual to see that multiple agents work cooperatively to accomplish a complex task. The examples include distributed reconfigurable sensor networks, smart grids, space-based interferometers, distributed machine learning and surveillance, combat, and reconnaissance systems. Although the problems arise from different application domains, they share some fundamental characteristics. Firstly, agents have simple communication, sensing, and computation capabilities and function in a fully autonomously manner; secondly, there is not a central decision coordinator or maker, and each agent makes its own decision by its local information, i.e., the system is distributed. Today, the research scope of MASs is still expanding.

1.2.1 Examples of networked systems

Few examples extracted from Mesbahi and Egerstedt (2010) of such distributed and networked systems in which the paradigm of MAS is crucial are reported hereafter.

Formation Flight

Distributed aerospace systems, such as fleets of autonomous rovers, formations of unmanned aerial vehicles and multiple spacecraft, have been identified as a novel paradigm for a large variety of applications. It is envisioned that distributed aerospace technologies will allow for the implementation of a spatially distributed network of vehicles that cooperate toward a common collective scientific, military, or civilian target. These systems are

of great interest since their distributed architecture promises a significant cost reduction in their design, operation and manufacturing. Additionally, distributed aerospace systems lead to higher degrees of scalability and adaptability in response to changes in the system capabilities and mission specifications. An example of a multiple platform aerospace system is space-borne optical interferometry. Space interferometers are distinguished by their operational environment and composition. They are composed of separated optical instruments, leading to a so-called sparse aperture. Although optical interferometers can, in principle, function on the earth's surface, there are many advantages in operating them in space. Space-borne interferometers have greater optical resolution and sensitivity, wider field of view, and greater detection capability. The resolution of these interferometers, compared with space telescopes (e.g. Hubble), depends on the separation between the light collecting elements rather than their size. Consequently, as the achievable imaging resolution of a space telescope is dictated by the size of the launch vehicle and the complex deployment mechanism, advanced manufacturing techniques, the capability of a space-borne optical interferometer is limited by how accurately the operation of separated optical elements can be coordinated. These space-borne optical interferometers can be mounted on a single large space structure, composed of rigid or semirigid trusses or even inflatable membranes. In this case, the structural dynamics of the spacecraft plays a crucial role in the operation and the success of the mission. Another strategy is to fly the interferometer on multiple physically separated spacecraft, namely, a distributed space system.
Another important set of applications of networked aerospace systems falls in the area of unmanned aerial vehicles (UAVs) of various capabilities and scales. These vehicle systems provide unique capabilities for a number of mission objectives, including surveillance, mapping, synthetic aperture imaging, environmental monitoring and target detection.

Sensor Networks

A wireless sensor network consists of spatially distributed autonomous devices that cooperatively monitor environmental or physical conditions, such as vibration, sound, pressure or temperature. Each node in a sensor network is equipped with a wireless communication device as well as an energy source – such as a battery – that requires to be efficiently used. The cost, size, and fidelity of a single sensor node can vary greatly, often in direct correspondence with its computational speed, energy usage and the ease by which it can be integrated within the network. Each sensor exchanges information on its local measurements with other network nodes in order to reach an accurate estimate of the environmental or physical variable of interest. It is remarkable to notice that the efficiency requirement on the utilization of the energy source for each sensor often dictates a geometric pattern on the inter-node communication for the sensor network.

Nanosystems

In the last twenty years, there has been a surge of interest by material scientists in organic compounds that are interconvertible via chemical reactions; this process is often addressed as tautomerization. These chemical reactions can be used for building molecular switches, where a molecule is governed between two or more stable states in a controlled fashion. Other electronic components such as transistors and diodes can be made that

rely on similar induced transitions between structural isomers. Such molecular devices can then be put together, leading to the possibility of designing molecular networks, circuits and, generally, molecular dynamic systems. For instance, a molecular switch is a hydrogen tautomerization employed to manipulate and probe a naphthalocyanine molecule via low-temperature scanning tunneling microscopy. The functionality and properties of the corresponding molecular networks and machines are highly dependent on the inter-molecular bonds that can generally be shaped by techniques such as electron beam lithography and molecular beam epitaxy.

Social Networks

Social networks are comprised of social entities, such as organizations and individuals, with a given set of interdependencies. The interaction between these entities can assume a multitude of relations, such as informational, financial and social. Such networks are of significant interest in a variety of areas, including organizational studies, theoretical sociology and sociolinguistics. In fact, the structure of social networks has always been of fundamental importance for understanding these networks. Recently, the notion of manipulating the network architecture has been contemplated as a viable means of altering the network behavior. For instance, the concept of a change agent refers to a network entity that indirectly or intentionally accelerates or causes social, behavioral or cultural change in the network.

Energy Networks

Large-scale complex energy systems, delivering mechanical and electrical energy from generators to loads via an intricate distribution network, are among the most useful engineered networked dynamic systems. These systems often consist of a heterogeneous set of dynamic subsystems, such as switching logics and power electronics, that evolve over multiple timescales. Dynamics, control and stability of individual power system elements (e.g., synchronous machines) or their interconnections (e.g., multi-machine models) have widely been examined in the literature. Nevertheless, as the need for more efficient utilization and generation of energy has become prevalent, distributed network structures, such as the smart grids, have gained particular relevance.

1.3 Structure of the manuscript

The remainder and comprehensive structure of this manuscript is organized as follows. In Chap. 2, the general background, the theoretical methodologies and the contributions of this work as well as the common thread in it are provided. Chapters 3, 4, 5, 6 represent the thesis core since they report the predominant research studies carried out during the PhD activity: further details about them are given in Sec. 2.3. Chap. 7 draws the general conclusions and presents potential future works for the contents of this dissertation. Finally, further supplementary material and a detailed appendix are reported in Chap. A and Chap. B respectively.

2

Networked Optimization for Multi-Agent Systems

"Believe me when I say we have a difficult time ahead of us. But if we are to be prepared for it, we must first shed our fear of it."
Morpheus

Contents

2.1 General background

In the recent decades, researchers have focused their attention in distributed multi-agent networks composed by a large number of smart devices that can collaborate to achieve a common target and are capable of taking local decisions independently, without any supervisory framework. Although complex large-scale monitoring and control systems are not new, as, for example, smart grids applications Gao, Xiao, Liu, Liang, and Chen (2012) or urban traffic control Wang, Djahel, and McManis (2014), novel architectural models are still emerging, as UAVs for multimedia delivery in industry Al-Turjman and Alturjman (2018). The choice of multi-agent systems is motivated by the fact that they can be supported by hierarchical developments, which has the considerable

advantage to be relatively easy to be designed maintaining safety guarantees. However, these systems require reliable actuators and sensors that are, generally, very expensive. In addition, the potential adoption of a centralized approach has been extensively proven to be a limitation for flexibility, scalability, adaptivity and maintenance Ge, Yang, and Han (2017). The recent trend, indeed, encourages the substitution of costly actuators, sensors and communication mechanisms with a larger number of devices that can autonomously compensate potential failures and a high computational burden through smart communication protocols and strong cooperation.

2.1.1 Foundations

As computer network become increasingly complex, the problem of allocating resources within such networks, in a distributed fashion, will turn more and more into of a design and implementation concern Brooks (2008). This is especially true where the allocation involves distributed collections of resources with several degrees of utility that can satisfy the desired allocation, and where this allocation process must be done in soft real-time. The field of MASs has developed a large variety of methodologies for distributed resource allocation. One application that highlights new issues in distributed resource allocation is that involving the management of adaptive distributed sensor networks. Indeed, distributed sensor networks are drawing growing attention within computer science and beyond. In adaptive sensor networks, the communication resources, processing and allocation of sensing, is dynamically regulated based on the phenomena that are occurring in the environment so as to most effectively interpret those phenomena. For instance, in such networks there may be diverse types of sensors, and sensor capabilities may be dynamically adjustable, e.g., in terms of parameters such as what aspects of the environment are sensed, the frequency and precision of sensing, and the nature of whatever local processing is carried out prior to transmitting information. This distributed resource allocation also has to resolve conflicting resource assignments that may occur when there are multiple phenomena happening in the environment that require to be interpreted currently.
To summarize, the foundations of multi-agent systems are deeply rooted in the focused use of distributed architectures and advanced theoretical tools allowing their analysis, design and implementation.

Centralized vs distributed architectures

Differently from centralized networks (see Fig. 2.1), distributed networks are not the easiest to maintain since they have multiple point of failure. This is not the case of the centralized architectures, which, in theory, are more prone to maintenance. On the other hand, this is also its major drawback: the centralized architectures are very unstable, since any problem that affects the central server can generate chaos throughout the system. Nonetheless, the distributed architectures are more stable, by storing the totality of the system information in a large number of nodes that maintain equal conditions with each other. This same aspect is what gives distributed networks a higher level of security, since to perform malicious attacks a large number of nodes should be struck at the same time. As the information is distributed among the nodes of the network, if a

legitimate change is made, this will be reflected in the rest of the nodes of the system that will accept and verify the new information. On the other hand, if some illegitimate variation is made, the rest of the nodes will be able to detect it and will not validate this information. This consensus between nodes protects the network from deliberate attacks or accidental changes of information. In addition, distributed systems have an advantage over centralized systems in terms of network speed, since the information is not stored in a central location a bottleneck is less likely. Indeed, whenever a large number of people attempts to access a centralized server, waiting times slow down the system. Also, centralized systems tend to exhibit scalability problems, since the capacity of the server is limited and cannot support infinite traffic. Distributed systems have greater scalability, due to the large number of nodes that support the network. Finally, in a distributed network the extraction of any of the nodes would not disconnect from the network to any other. All the nodes are connected to each other without necessarily having to pass through one or several local centers. In this type of networks the center/periphery division disappears and therefore the filtering capability over the information flowing through it makes it a practical and efficient system.

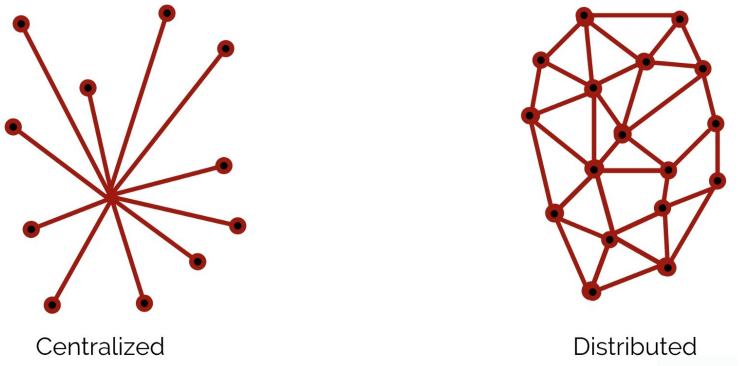


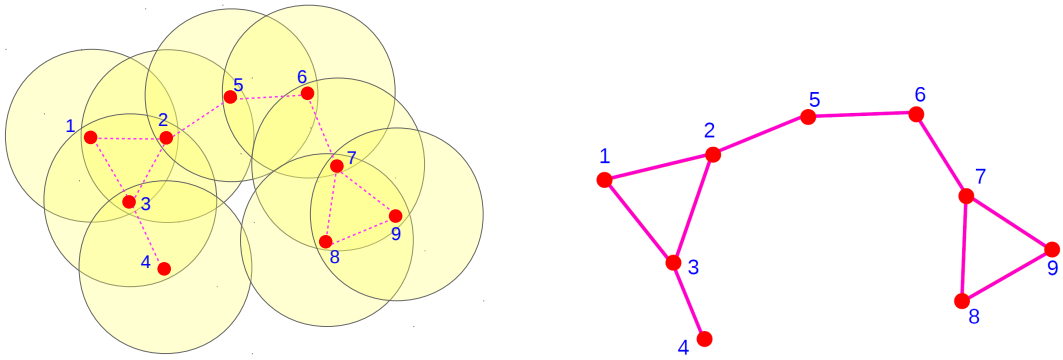
Figure 2.1. Schematization of centralized and ditributed architectures.

Mobile multi-agent systems: combinatorial modeling

Distributed Coordination of multi-agent networks introduces problems, models, and issues such as collective periodic motion coordination, collective tracking with a dynamic leader, and containment control with multiple leaders, and explores ideas for their solution. Solving these problems extends the existing application domains of multi-agent networks; for example, collective periodic motion coordination is appropriate for applications involving repetitive movements, collective tracking guarantees tracking of a dynamic leader by multiple followers in the presence of reduced interaction and partial measurements, and containment control enables maneuvering of multiple followers by multiple leaders.

According to the existing literature, combinatorial tools are imperative to cope with multi-agent systems, e.g. Graph Theory and Rigidity Theory. An n-agent system can be modeled through a *graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ so that each element in the *nodes set* $\mathcal{V} = \{v_1 \dots v_n\}$ corresponds to an agent in the group, while the *edge set* $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ describes the agents interactions (see also Sec. B.3 for further notions of Graph Theory). In the rest of the thesis, it is assumed that $\mathcal{G}$ is undirected and that the set $\mathcal{E}$ depicts both agents

visibility and communication capabilities, meaning that there exists $e_{ij} = (v_i, v_j) \in \mathcal{E}$ if and only if the i-th and j-th agents can sense each other and are able to reciprocally exchange information according to some predetermined communication protocol, as illustrated in Fig. 2.2. The agents interplays are generally represented by the *adjacency matrix* $\mathbf{A}$. For each node v_i in $\mathcal{G}$, the *neighborhood* $\mathcal{N}_i$ represents the set of agents interacting with the i-th agent. By convention, it holds that $v_i \notin \mathcal{N}_i$ and, whenever $\mathcal{N}_i$ possesses any order, $v_j = [\mathcal{N}_i]_j$ is the j-th neighboring vertex of v_i, for $j = 1, ..., |\mathcal{N}_i|$. The cardinality of $\mathcal{N}_i$ is the *degree*, $\deg(v_i)$, of the i-th agent. This corresponds to the i-th element in the main diagonal of the *degree matrix* $\mathbf{D} = \text{diag}(\mathbf{A}\mathbb{1}_n) \in \mathbb{R}^{n \times n}$, where $\mathbb{1}_n$ indicates a n-dimensional (column) vectors whose entries are all ones. The matrix $\mathbf{D}$ in turn contributes to the definition of the *Laplacian matrix*, $\mathbf{L} = \mathbf{D} - \mathbf{A} \in \mathbb{R}^{n \times n}$. Other remarkable matrices, related to $\mathcal{G}$, are represented by the *normalized Laplacian* $\boldsymbol{\mathcal{L}} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ and the *Randić matrix* $\boldsymbol{\mathscr{R}} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$. Lastly, depending on the context, vertices and edges can be both weighted by positive real *weights*.



(a) Agent interactions (b) Corresponding graph representation

Figure 2.2. A network of agents equipped with omnidirectional range sensor can be viewed as a graph, with nodes corresponding to agents and edges to interconnections. [*Credits* : Ramaithitima et al. (2015)]

2.1.2 A quick overview on optimization problems

Nowadays, the concept of optimization is now well rooted as a principle underlying the analysis of many complex allocation or decision problems. It offers a certain degree of philosophical elegance that is hard to dispute, and it often provides an indispensable degree of operational simplicity. Using this optimization philosophy, one approaches a complex decision problem, involving the selection of values for a number of interrelated variables, by focusing attention on a single objective designed to quantify performance and measure the quality of the decision. Such objective is minimized (or maximized, depending on the context) subject to the constraints that may limit the selection of decision variable values. If a suitable single aspect of a problem can be characterized by an objective and isolated, be it speed or distance in a physical problem, loss or profit in a business setting, social welfare in the context of government planning, or expected return in the environment of risky investments, optimization may provide a suitable framework for analysis.

In this paragraph, few elements for the understanding of the ideas that are necessary to state and formalize interesting engineering problems for networked multi-agent systems are provided. Classical paradigms of Mathematical Programming, such as Convex Optimization or Combinatorial Optimization are introduced as well as Optimal Control problems, in order to serve as a preliminary for the forthcoming multi-agent optimization problems in the following chapters of this thesis.

Problem formalization in Mathematical Programming

Mathematical Programming is the selection of a best element (with regard to some criterion) from some set of available alternatives. Optimization problems of sorts arise in all quantitative disciplines from Computer Science and Engineering to Operations Research and Economics, and the development of solution methods has been of interest in mathematics for centuries. Many optimization problems can be classified, yet the general Mathematical Programming problem, according to Luenberger and Ye (2008), can be stated as

$$\begin{aligned} \min_{\mathbf{x}} \quad & h(\mathbf{x}) \\ \text{subject to} \quad & c_i^I(\mathbf{x}) \leq 0, \quad i = 1, \dots, n_I \\ & c_i^E(\mathbf{x}) = 0, \quad i = 1, \dots, n_E \\ & \mathbf{x} \in X \end{aligned} \tag{2.1}$$

where $h(\mathbf{x})$ is a cost functional to be minimized in the variable $\mathbf{x}$, $c_i^I(\mathbf{x})$ is the i-th inequality constraint, $c_i^E(\mathbf{x})$ is the i-equality constraint and X is the feasibility set for $\mathbf{x}$. Whenever $n_I = 0$ and $n_E = 0$, i.e. no constraint is present, and $X = \mathbb{R}^N$, problem (2.1) is said to be *unconstrained*.

Example 2.1.1 (Convex optimization problems)**.**
Convex optimization problems can be characterized similarly to problem (2.1), assuming that $h(\mathbf{x})$, $c_1^I(\mathbf{x}), \dots, c_{n_I}^I(\mathbf{x})$ are convex functions, $c_1^E(\mathbf{x}), \dots, c_{n_E}^E(\mathbf{x})$ are linear affine functions in $\mathbf{x}$ and X a convex set in $\mathbb{R}^N$.

Example 2.1.2 (Quadratic optimization problems)**.**
Quadratic optimization problems can be characterized similarly to problem (2.1), assigning

$$h(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{Q}\mathbf{x} + \mathbf{a}^\top \mathbf{x}, \qquad \mathbf{Q} \in \mathbb{R}^N, \ \mathbf{a} \in \mathbb{R}^N, \tag{2.2}$$

assuming functions $c_1^I(\mathbf{x}), \dots, c_{n_I}^I(\mathbf{x})$ and $c_1^E(\mathbf{x}), \dots, c_{n_E}^E(\mathbf{x})$ are linear affine in $\mathbf{x}$ and $X = \mathbb{R}^N$.

Example 2.1.3 (Combinatorial optimization problems)**.**
Combinatorial optimization problems can be characterized similarly to problem (2.1), assuming $h(\mathbf{x}) = h(\mathbf{x}_0, \mathbf{x})$ for a given $\mathbf{x}_0$ belonging to the set of instances $I \subseteq \mathbb{R}^N$, and $\mathbf{x} = \mathbf{x}(\mathbf{x}_0) \in X(\mathbf{x}_0)$ where $X(\mathbf{x}_0)$ is the finite feasibility set induced by instance $\mathbf{x}_0$. Frequently, $X(\mathbf{x}_0)$ is a discrete set, e.g. it is a subset of $\mathbb{Z}^N$ or $\mathbb{N}^N$. Function $h(\mathbf{x}_0, \mathbf{x})$ is then optimized over $\mathbf{x}$.
For each problem of this kind, there is a corresponding decision problem that asks whether

there is a feasible solution for some particular measure $h(\mathbf{x}_0, \mathbf{x})$. For instance, let $\mathcal{G}$ be a graph which contains vertices v_i and v_j. A combinatorial optimization problem might be "find a path from v_i to v_j that uses the fewest edges".

Optimal Control problems

Optimal Control problems are particular optimization problems in which the main focus is to find a control law for a dynamic system over a period of time such that an objective function is optimized. It has numerous applications in both science and engineering. For example, the dynamic system might be a spacecraft with controls corresponding to rocket thrusters, and the objective might be to reach the moon with minimum fuel expenditure Luenberger (1979). Or the dynamic system could be a nation's economy, with the objective to minimize unemployment; the controls in this case could be fiscal and monetary policy. In this paragraph, a concise formalization of classical Optimal control problems is provided. Further details can be found in Fornasini (2013); Vinter (2010).

Firstly, the *dynamics* of the generic system with *state* $\mathbf{x} \in \mathbb{R}^{N_S}$ and *input* $\mathbf{u} \in \mathbb{R}^{N_I}$

$$\dot{\mathbf{x}}(\tau) = \mathbf{f}(\mathbf{x}(\tau), \mathbf{u}(\tau)) \tag{2.3}$$

is considered, together with its *initial condition*

$$\mathbf{x}(0) = \mathbf{x}_0. \tag{2.4}$$

It is assumed that $\mathbf{f}$ is sufficiently smooth, e.g. $\mathscr{C}^2$ in $(\mathbf{x}, \mathbf{u})$ and $\mathscr{C}^0$ in τ, and instantaneous values of the input $\mathbf{u}$ belong, by the nature of the problem or by conditions that have to be satisfied by the control variables, to an assigned set U of $\mathbb{R}^{N_I}$:

$$\mathbf{u} \in U \subset \mathbb{R}^{N_I}, \quad \forall \tau. \tag{2.5}$$

Secondly, assuming that there exists an input $\mathbf{u}(\cdot)$ satisfying (2.5), the control problem under analysis is to seek $\mathbf{u}(\cdot)$, acting on the time interval $[0, T]$, such that it minimizes[1] the *objective functional*

$$h(\mathbf{x}(\cdot), \mathbf{u}(\cdot)) = m(\mathbf{x}(T)) + \int_0^T l(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau) d\tau. \tag{2.6}$$

Functional (2.6) identifies:

- a specific trajectory of the state $\mathbf{x}(\cdot)$, that substituted along with $\mathbf{u}(\cdot)$ in the arguments of the *instantaneous cost* l, determines the incremental values and the amount of the integral;
- a specific *final state* $\mathbf{x}(T)$ and, with it, the *final cost* m.

To conclude this brief premise, cost (2.6) can be interpreted as a functional defined on the set of admissible input functions $\mathbf{u}(\cdot)$ and the set of the state trajectories $\mathbf{x}(\cdot)$. Therefore,

[1] Analogously, one can consider the opposite problem, i.e. the objective maximization. This is solved by means of the same techniques here illustrated, either changing sign conventions while defining (2.6) or using the antithetical Pontryagin's Minimum Principle.

this can be seen as the minimization problem of functional (2.6) subject to the condition that $\mathbf{x}(\cdot)$ and $\mathbf{u}(\cdot)$ are constrained by (2.3) and (2.4).

2.1.3 Motivations, application examples and new perspectives

According to Xie and Liu (2017), modeling and computation tasks are becoming much more complex as their size continues to increase. As a result, it is laborious and difficult to handle them using centralized methods. Although motivations to apply multi-agent systems for researchers from various disciplines are different, the major advantages of using multi-agent technologies include:

1. individuals take into account the application-specific nature and environment;
2. local interactions between individuals can be modeled and investigated;
3. difficulties in modeling and computation are organized as sublayers and/or components.

Therefore, multi-agent systems provide a good solution to distributed control as a computational paradigm.

In the forthcoming discussion, some of the most classic applications of multi-agent systems to cooperative tasks are provided and analyzed, with a point of view focused on Optimization Theory. In particular, well established topics as Voronoi partitions for Robotic Coverage Control, graph theoretic methods related to agreement protocols, linear consensus algorithms for Distributed Estimation and design of artificial-potential control laws for agent formations are introduced together with relevant application examples, allowing for a mathematical formalism based on optimization problems. To this end, the key elements to achieve such cooperative tasks are properly synthesized into constrained or unconstrained cost functionals obtained by the specific requirements of the problem under analysis. Also, few strategies to tackle the minimization of these task-oriented costs are proposed and debated.

Robotic Coverage Control

The coverage problem concerns ensuring that a collection of mobile robots are placed in such a way that the area under consideration is completely covered by sensors, that is, such that each location in the area is seen by at least one sensor. Applications that demand this task include exploring a confined space for search and rescue Dornhege, Kleiner, Hertle, and Kolling (2016); Vijay Kumar, Rus, and Sanjiv Singh (2004), deploying wireless coverage Liu, Chin, Yang, and He (2019), and scouting for urban combat Mclurkin and Smith (2004). Once distributed, the swarm may be largely stationary, as in the cases above, or may attempt to move dynamically as an aggregate. Coverage techniques are generally relevant for exchanging information in the presence of controlled interconnected dynamic systems Antonelli (2013). Typically, approaches to this problem require local coordinate information, such as distance and angle to neighbors or relative heading (see, for example, Cortes, Martinez, Karatas, and Bullo (2004) and Olfati-Saber (2006)). In the forthcoming lines, the concept of Voronoi tessellations is introduced along with a procedure that leads to the accomplishment of an optimal surface subdivision, known

as Lloyd's algorithm (see also Du, Faber, and Gunzburger (1999) and Mesbahi and Egerstedt (2010) for further details).

Given a closed set $\Omega \subseteq \mathbb{R}^2$, the collection of closed sets $\{\Omega_i\}_{i=1}^n$ is called a *tessellation* of Ω if $\Omega_i \cap \Omega_j = \varnothing$ for $i \neq j$ and $\bigcup_{i=1}^n \Omega_i = \Omega$. The interpretation here is that Ω_i is the region in Ω that agent i is responsible for, addressed as the i-agent *dominance region*. Let $\|\cdot\|_2$ denote the Euclidean norm on $\mathbb{R}^2$. Given a set of points $\{\mathbf{p}_i\}_{i=1}^n$ belonging to Ω and denoting with Ω the tessellation $\{\Omega_i\}_{i=1}^n$, the following local cost *coverage functional* can be defined:

$$h_\Omega(\mathbf{p}_1, \ldots, \mathbf{p}_n, \Omega) = \sum_{i=1}^{n} \int_{\Omega_i} \|\boldsymbol{\omega} - \mathbf{p}_i\|_2^2 \, d\boldsymbol{\omega} = \int_{\Omega} \|\boldsymbol{\omega} - \mathbf{p}_i\|_2^2 d\boldsymbol{\omega}. \tag{2.7}$$

The interpretation of functional (2.7) is that it divides the space Ω into the dominance regions Ω_i, and then parametrizes how well these regions are covered by the agents, with the coverage quality (how well agent in $\mathbf{p}_i$ can sense the point $\boldsymbol{\omega}$) degrading quadratically as a function of $\|\boldsymbol{\omega} - \mathbf{p}_i\|_2$.

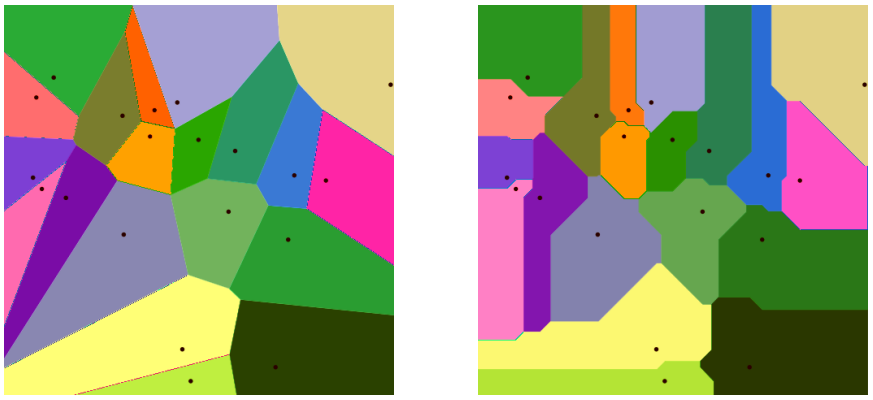
Intuitively, it makes sense to simplify the problem of minimizing (2.7) over points $\mathbf{p}_i$, $i = 1, \ldots, n$, and $\hat{\Omega}$ by assuming that $\hat{\Omega}$ is the *Voronoi partition* of Ω, that is, $\hat{\Omega} = \{\hat{\Omega}_i\}_{i=1}^n$, where the i-th *Voronoi tessellation* $\hat{\Omega}_i$ corresponding to the point $\mathbf{p}_i$ is defined by

$$\hat{\Omega}_i = \left\{ \boldsymbol{\omega} \in \Omega \mid \|\boldsymbol{\omega} - \mathbf{p}_i\|_2 \leq \|\boldsymbol{\omega} - \mathbf{p}_j\|_2 , \text{ for } j = 1, ..., n, \ j \neq i \right\}. \tag{2.8}$$

In this formulation, the *locational optimization problem* is in the form of minimizing

$$h_{\hat{\Omega}}(\mathbf{p}_1, \ldots, \mathbf{p}_n) = \int_{\Omega} \min_{i \in \{1,\ldots,n\}} \|\boldsymbol{\omega} - \mathbf{p}_i\|_2^2 \, d\boldsymbol{\omega} \tag{2.9}$$

Depending on the application, there exist many different names for Voronoi regions, including, for instance, Dirichlet regions, area of influence polygons, Meijering cells, Thiessen polygons, and S-mosaics (see Fig. 2.3).



(a) Euclidean-based partition (b) Manhattan-based partition

Figure 2.3. Two Voronoi diagrams attained by partitioning the space Ω with different criteria: (a) Voronoi partition obtained with functional 2.9, using the Euclidean norm $\|\cdot\|_2$. (b) Similar Voronoi partition obtained by substituting the Manhattan norm $\|\cdot\|_1$ to $\|\cdot\|_2$ in the definition of 2.9.

Given a density function $f_\rho : \Omega \to \mathbb{R}$ defined in Ω, the mass centroid $\mathbf{p}_{c,i}$ of Ω_i is

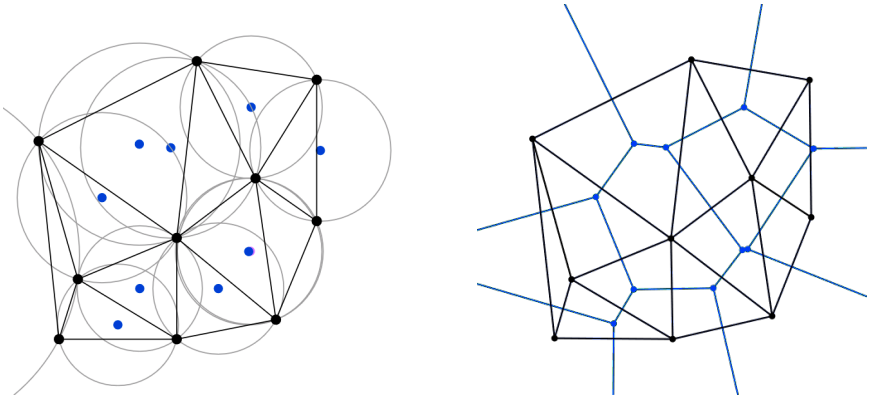
defined by

$$\mathbf{p}_{c,i} = \frac{\int_{\Omega_i} \boldsymbol{\omega} f_\rho(\boldsymbol{\omega}) d\boldsymbol{\omega}}{\int_{\Omega_i} f_\rho(\boldsymbol{\omega}) d\boldsymbol{\omega}}. \tag{2.10}$$

Thus, for the n given points $\{\mathbf{p}_i\}_{i=1}^n$, it is possible to construct their associated Voronoi tessellations $\{\hat{\Omega}_i\}_{i=1}^n$. On the other hand, given the tessellations $\{\hat{\Omega}_i\}_{i=1}^n$, it is possible to define their mass centroids $\{\mathbf{p}_{c,i}\}_{i=1}^n$. Problem (2.9) suggests, indeed, that the optimal solution for the Voronoi partition is achieved when condition

$$\mathbf{p}_i = \mathbf{p}_{c,i}, \quad i = 1, ..., n \tag{2.11}$$

holds, i.e., the points $\mathbf{p}_i$ that serve as *generators* for the Voronoi tessellations $\hat{\Omega}_i$, are themselves the mass centroids of those regions. This situation is quite special since, in general, arbitrarily chosen points in $\mathbb{R}^2$ are not the centroids of their associated Voronoi tessellations. As shown in Fig. 2.4, these particular regions are, in a general framework, *polytopes* (see Subsec. B.1.1) and possess correspondent *dual tessellations* (in $\mathbb{R}^2$, the Delaunay triangulations).



(a) Delaunay triangulation (b) Voronoi partition

Figure 2.4. (a) The Delaunay triangulation with all the circumcircles and their centers, in blue. (b) Connecting the centers of the circumcircles produces the Voronoi partition, in blue.

One may ask, how to obtain Voronoi tessellations. Consider the following problem: given

- a two-dimensional Euclidean closed space $\Omega \subset \mathbb{R}^2$,
- n generators $\mathbf{p}_i \in \Omega_i \subseteq \Omega$ representing the n positions of the agents to be deployed,
- the density function $f_\rho : \Omega \to \mathbb{R}$,

find

- n regions $\hat{\Omega}_i$ that tessellate Ω, such that $\{\hat{\Omega}_i\}_{i=1}^n$ are the correspondent Voronoi tessellation for $\{\mathbf{p}_i\}_{i=1}^n$,
- n points $\mathbf{p}_{c,i} \in \Omega$, such that $\{\mathbf{p}_{c,i}\}_{i=1}^n$ are the mass centroid of $\{\hat{\Omega}_i\}_{i=1}^n$.

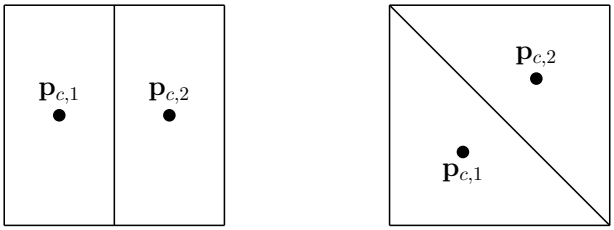


Figure 2.5. Two centroidal Voronoi tessellations of a square. The points $\mathbf{p}_{c,1}$ and $\mathbf{p}_{c,2}$ are the centroids of the rectangles on the left or of the triangles on the right.

The solution of this problem is in general not unique. For example, consider the case of $n = 2$, with $\Omega \subset \mathbb{R}^2$ a square, and $f_{\rho,i} = 1$, for all $i = 1, \ldots, n$. Two solutions are depicted in Fig. 2.5; others may be obtained through rotations.

Now, a deterministic approach for the determination of centroidal Voronoi tessellations of a given set, known as Lloyd's algorithm, is briefly illustrated and discussed. Given a set Ω, a positive integer n, and a probability density function f_ρ defined on Ω,

1. select an initial set of n points $\{\mathbf{p}_i\}_{i=1}^n$, e.g., by using a Monte Carlo method;
2. construct the Voronoi tessellation $\{\hat{\Omega}_i\}_{i=1}^n$ of Ω associated with the points $\{\mathbf{p}_i\}_{i=1}^n$;
3. compute the mass centroids $\{\mathbf{p}_{c,i}\}_{i=1}^n$ of the Voronoi regions $\{\hat{\Omega}_i\}_{i=1}^n$ found in step 2; these centroids $\{\mathbf{p}_{c,i}\}_{i=1}^n$ are the new set of points w.r.t. $\{\mathbf{p}_i\}_{i=1}^n$.
4. If this new set of points meets some convergence criterion, terminate; otherwise, return to step 2.

Defining $\mathbf{p}_c = \left[\mathbf{p}_{c,1}^\top, \ldots, \mathbf{p}_{c,n}^\top\right]^\top$, Lloyd's algorithm can be seen as an equilibrium for the algebraic differential equation

$$\mathbf{p}_{c,i}(t+1) = \frac{\int_{\Omega_i(\mathbf{p}_c(t))} \boldsymbol{\omega} f_\rho(\boldsymbol{\omega}) d\boldsymbol{\omega}}{\int_{\Omega_i(\mathbf{p}_c(t))} f_\rho(\boldsymbol{\omega}) d\boldsymbol{\omega}}. \tag{2.12}$$

Agreement protocols

According to Mesbahi and Egerstedt (2010), *agreement* is one of the fundamental protocol problems in multi-agent coordination, where a group of agents are to agree on a joint state value. Agreement among the processes in a distributed system is a fundamental requirement for a wide range of applications: whether to commit a transaction to a database, agreeing on the identity of a leader, atomic broadcasts and state machine replication. The real world applications include state estimation Li, Wang, Wei, Ma, Hu, and Ding (2015), clock synchronization Swain and Hansdah (2015), PageRank Brin and Page (2017), opinion formation Kouvaros and Lomuscio (2016); Parsegov, Proskurnikov, Tempo, and Friedkin (2015), smart power grids Camarinha-Matos (2016), control of UAVs (and multiple robots/agents in general) Chmaj and Selvaraj (2015), load balancing Amelina, Fradkov, Jiang, and Vergados (2015), blockchain Zheng, Xie, Dai, Chen, and Wang (2017) and others. Many forms of coordination require the processes to exchange information to negotiate with one another and eventually reach a common understanding

or agreement, before taking application-specific actions.
Consider a situation where a group of sensors are to measure the temperature of a given area. Although the temperature measured by each sensor will vary according to its location, it is required that the sensor group – using an information sharing network – agree on a single value that represents the temperature of the surface. For this, the collection of sensors needs a *protocol* over the network, allowing it to achieve consensus on what the common sensor measurement value should be.
The importance of the agreement protocol is twofold. On one hand, agreement has a close relation to a host of multi-agent problems such as swarming, attitude alignment, rendezvous, flocking and distributed estimation. On the other hand, this protocol provides a concise formalism for examining means by which the network topology dictates properties of the dynamic process evolving over it.

The agreement protocol involves n dynamic units, labeled $1, 2, \ldots, n$, interconnected via relative information-exchange links. The rate of change of each unit's state is assumed to be governed by the sum of its relative states w.r.t. a subset of other (neighboring) units. In the time-invariant continuous linear case, denoting the scalar state of unit i as $x_i \in \mathbb{R}$, one has

$$\dot{x}_i(\tau) = \sum_{j \in \mathcal{N}_i} (x_j(\tau) - x_i(\tau)), \quad i = 1, \ldots, n, \tag{2.13}$$

where $\mathcal{N}_i$ is the set of units adjacent to, or neighboring, unit i in the network. When the adopted notion of adjacency is symmetric (see Subsec. B.3.1), the overall system can be represented by

$$\dot{\mathbf{x}}(\tau) = -\mathbf{L}\mathbf{x}(\tau) \tag{2.14}$$

where the positive semidefinite matrix $\mathbf{L}$ is the Laplacian of the agents' interaction network $\mathcal{G}$ and $\mathbf{x}(\tau) = \left[x_1(\tau), \ldots, x_n(\tau)\right]^\top \in \mathbb{R}^n$. Equation (2.14) is addressed as the *agreement dynamics*. If $\mathbf{x}_i$ is a vector in $\mathbb{R}^M$, with $M > 1$, one can still obtain a compact description of (2.14), which is provided by

$$\dot{\mathbf{x}}(\tau) = -(\mathbf{I}_M \otimes \mathbf{L})\mathbf{x}(\tau) \tag{2.15}$$

where $\mathbf{x}(\tau) = \left[\mathbf{x}_1(\tau)^\top, \ldots, \mathbf{x}_n(\tau)^\top\right]^\top \in \mathbb{R}^N$, with $N = Mn$, $\mathbf{I}_M$ being the identity matrix of dimension M and $\otimes$ denoting the Kronecker product operator[2]. Therefore, in this paragraph, it is assumed, w.l.o.g., that $M = 1$.
As such a *state of agreement* is of great interest, a formal definition of this notion is yielded by

Definition 2.1.4 (Agreement set)**.**
The *agreement set* $\mathcal{A} \subseteq \mathbb{R}^n$ is the subspace span$\{\mathbb{1}_n\}$, that is,

$$\mathcal{A} = \{\mathbf{x} \in \mathbb{R}^n | x_i = x_j, \text{ for all } i, j\}. \tag{2.16}$$

It is worth to notice that a vector $\mathbf{x}^\star$ belongs to the agreement set $\mathcal{A}$ if and only if $\mathbf{x}^\star$

[2]Recall that the Kronecker product has a lower priority w.r.t. the matrix or vector multiplication.

is a minimizer of the cost function

$$h_{\mathcal{A}}(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \mathbf{L}\mathbf{x} = \frac{1}{4}\sum_{i=1}^{n}\sum_{j\in\mathcal{N}_i}(x_i - x_j)^2. \tag{2.17}$$

This, undoubtedly, provides an important optimization viewpoint of the agreement problem. Furthermore, another relevant observation can be made, since the agreement set $\mathcal{A}$ also represents the set of equilibria for the *consensus dynamics* described by (2.14). Indeed, all these crucial aspects belonging to the agreement protocol connect Optimization issues to concepts related to System Theory, Control Theory and Graph Theory, forming and indissoluble bond. The most remarkable fact confirming the previous statement is expressed in the following theorem.

Theorem 2.1.5 (Convergence of the consensus dynamics)**.**
For an undirected connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, the state trajectory generated by dynamics in (2.14) with initial condition $\mathbf{x}(0) = \mathbf{x}_0 \in \mathbb{R}^n$ satisfies

$$\lim_{\tau\to+\infty}\mathbf{x}(\tau) = \frac{\mathbb{1}_n\mathbb{1}_n^\top}{\mathbb{1}_n^\top\mathbb{1}_n}\mathbf{x}_0 = \mathbf{x}^\star \tag{2.18}$$

with a rate of convergence dictated by the algebraic connectivity $\lambda_1^{\mathbf{L}}$ of $\mathcal{G}$, i.e. the smallest nonzero eigenvalue of the Laplacian $\mathbf{L}$ associated to $\mathcal{G}$ (see also Subsec. B.3.2).

Nevertheless, it is imperative to recall that Theorem 2.1.5 actually represents just a little piece of a bigger picture: in general, agreement protocols can be applied to a larger class of problems, e.g. involving a discrete-time dynamics, a nonlinear dynamics, time-variant weighted graphs, directed graphs or nondeterministic models. In conclusion, to justify this approach to reality, it is provided a well-known example in which the synchronization of several coupled oscillators takes place.

Example 2.1.6 (Coupled oscillators and the Kuramoto model)**.**
According to Dörfler and Bullo (2014), a network of n oscillators (see Fig. 2.6) can be represented by a weighted, undirected, and connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ and $\mathcal{E}$ denote the set of oscillators (nodes) and edges (inter-agent forces), respectively. The weighted adjacency matrix $\mathbf{A}$ of $\mathcal{G}$ can be exploited to capture the forces $a_{ij} = [\mathbf{A}]_{ij} \geq 0$ between each couple of oscillators (i, j). Let $\mathbb{S}^1$ be the unit radius circle. The i-th oscillator phase is denoted by the angle $\theta_i \in \mathbb{S}^1$, whose dynamics evolve as a heterogeneous system:

$$\dot{\theta}_i = \omega_{n,i} - \sum_{j=1}^{n} a_{ij}\sin(\theta_i - \theta_j), \quad i = 1, \ldots, n. \tag{2.19}$$

Assuming that all inter-agent forces a_{ij} are uniform, namely $a_{ij} = K_{sync}/n$ for all i, j, Kuramoto showed that synchronization occurs in model (2.19) if the coupling gain K_{sync} exceeds a certain threshold K_{critical} function of the distribution of the natural frequencies $\omega_{n,i}$ (Kuramoto (1975)). Dynamics (2.19) with uniform weights are nowadays known as the Kuramoto model of coupled oscillators, and Kuramoto's original work initiated a broad stream of research.

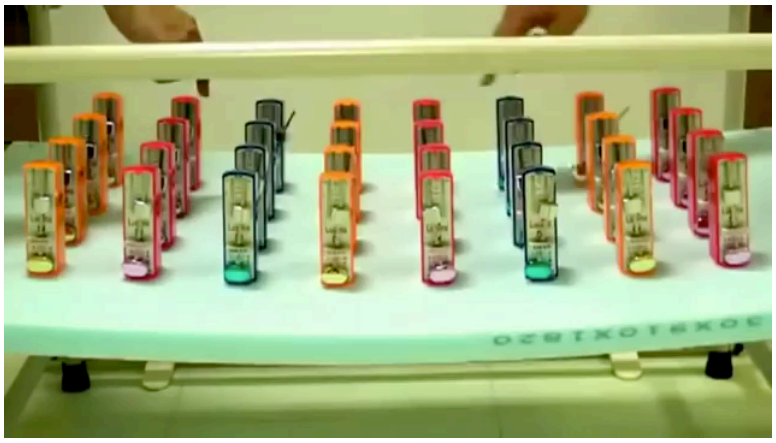


Figure 2.6. 32 Metronome Synchronization
[*Credits* : https://www.youtube.com/watch?v=5v5eBf2KwF8]

Distributed Estimation via linear consensus algorithms

Estimation theory is a truth-seeking endeavor; it is the scientific means of designing processes by which a static or dynamic variable of interest can be uncovered by processing noisy signal processing and control.

According to Garin and Schenato (2010), in the past decades, it has been witnessed the growth of engineering systems composed by a large number of devices that can communicate and cooperate to achieve a common goal. Although complex large-scale monitoring and control systems are not new, as for example nuclear plants and air traffic control, a new architectural paradigm is emerging, mainly due to the adoption of smart agents. In fact, traditional large-scale systems have a centralized or at best a hierarchical architecture, which has the advantage to be relatively easy to be designed and has safety guarantees. However, these systems require very reliable sensors and actuators, are generally very expensive, and do not scale well due to communication and computation limitations. The recent trend to avoid these problems is to substitute costly sensors, actuators and communication systems with a larger number of devices that can autonomously compensate potential failures and computation limitations through communication and cooperation. Although very promising, this new paradigm brings new problems into the picture. In particular, there are only few tools for predicting the global behavior of the system as a whole starting from the local sensing and control rules adopted by the smart sensors and actuators. As a consequence, there has been a strong effort in past years by many engineering as to develop such tools. One of the most promising tools are the linear consensus algorithms, which are simple distributed algorithms which require only minimal computation, communication and synchronization to compute averages of local quantities that reside in each device. These algorithms have their roots in the analysis of Markov chains Seneta (2006) and have been deeply studied within the computer science community for load balancing Muthukrishnan, Ghosh, and Schultz (1998); Tsitsiklis and Athans (1984) and within the linear algebra community for the asynchronous solution of linear systems Frommer and Szyld (2000); Strikwerda (2002).

Let us consider the following linear and discrete time consensus protocol:

$$\mathbf{x}(t+1) = \mathbf{F}(t)\mathbf{x}(t), \qquad \mathbf{x}(0) = \mathbf{x}_0 \tag{2.20}$$

where $\mathbf{x}(t) = \begin{bmatrix} x_1(t) & \cdots & x_n(t) \end{bmatrix}^\top \in \mathbb{R}^n$ and, for all t, $\mathbf{F}(t) \in \mathbb{R}^{n\times n}$ is a *stochastic* matrix, i.e. $[\mathbf{F}(t)]_{ij} = \mathrm{f}_{ij}(t) \geq 0$ and $\sum_{j=1}^n \mathrm{f}_{ij}(t) = 1, \forall i, \forall t$, i.e. each row sums to unity. For some applications, it may be also required for $\mathbf{F}(t)$ to be a *doubly stochastic* matrix, namely, both $\mathbf{F}(t)$ and $\mathbf{F}(t)^\top$ be stochastic.
As in the previous paragraph, it is worth to note that this model involves scalar states x_i, for $i = 1, \ldots, n$, that can potentially be extended to their vector versions $\mathbf{x}_i \in \mathbb{R}^M$, such that the total dimension of the whole state $\mathbf{x}(t) = \begin{bmatrix} \mathbf{x}_1(t)^\top & \cdots & \mathbf{x}_n(t)^\top \end{bmatrix}^\top \in \mathbb{R}^N$ be $N = Mn$. Let also $\mathrm{f}_{ij} \neq 0$ only if the i-the vertex $v_i \in \mathcal{V}$ of an undirected or directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ can receive information from the j-the vertex $v_j \in \mathcal{V}$, that is, the edge e_{ij} belongs to $\mathcal{E}$. Suppose that all the agents communicate synchronously[3] and that the transmissions among them are reliable.
Assuming all entries f_{ij} to be constant over the time, the standard consensus algorithm works as follows.

1. At time $t = 0$ node v_i initializes the state $x_i(0)$ to its observation $x_{0,i}$.
2. At iteration t each node v_i receives the estimate $x_j(t)$ from all its neighbors $v_j \in \mathcal{N}_i$.
3. Based on the received information, node v_i updates state x_i as

$$x_i(t+1) = \mathrm{f}_{ii}x_i(t) + \sum_{v_j \in \mathcal{N}_i} \mathrm{f}_{ij}x_j(t). \tag{2.21}$$

To analyze the properties of (2.20), the interested reader is referred to the insightful preliminaries given in Sec. B.2, for which the following theorem represents a theoretical endpoint.

Theorem 2.1.7 (Consensus dynamics)**.**
If a constant row-stochastic matrix $\mathbf{F} \in \mathbb{R}^{n\times n}$ has an associated graph that is connected and aperiodic, i.e. $\mathbf{F}$ is primitive, then, for $t \in \mathbb{N}$, one has

(i) $\lim_{t\to+\infty} \mathbf{F}^t = \boldsymbol{v}_0^{\mathbf{F}} \mathbb{1}_n^\top$, *where* $\boldsymbol{v}_0^{\mathbf{F}}$ *is positive and* $\mathbf{F}^\top \boldsymbol{v}_0^{\mathbf{F}} = \boldsymbol{v}_0^{\mathbf{F}}$, $\mathbb{1}_n^\top \boldsymbol{v}_0^{\mathbf{F}} = 1$;

(ii) the solution to (2.20), *with* $\mathbf{F}(t)$ *constant, satisfies*

$$\lim_{t\to+\infty} x(t) = (\boldsymbol{v}_0^{\mathbf{F}} \mathbb{1}_n^\top)\mathbf{x}_0; \tag{2.22}$$

and the rate of semi-convergence is yielded by the essential spectral radius $\lambda^{\mathbf{F}} = \max_{|\lambda_i^{\mathbf{F}}|<1} |\lambda_i^{\mathbf{F}}|$ *of* $\mathbf{F}$, *namely the second largest eigenvalue (in modulus) of* $\mathbf{F}$ *belonging to the interior of the unit circle.*

If additionally $\mathbf{F} \in \mathbb{R}^{n\times n}$ is doubly stochastic, then

[3] However, in practice, it may be very expensive (in terms of communication overhead) or even impossible (with very large networks) to guarantee synchrony. To overcome these issues, many asynchronous consensus protocols have been developed, e.g. the broadcast asymmetric protocol or the gossip symmetric protocol Aysal, Yildiz, Sarwate, and Scaglione (2009).

(i) $\lim_{t\to+\infty} \mathbf{F}^t = \boldsymbol{v}_0^{\mathbf{F}} \mathbb{1}_n^\top = \dfrac{\mathbb{1}_n \mathbb{1}_n^\top}{\mathbb{1}_n^\top \mathbb{1}_n}$;

(ii) the solution to (2.20)*, with* $\mathbf{F}(t)$ *constant, satisfies*

$$\lim_{t\to+\infty} x(t) = (\boldsymbol{v}_0^{\mathbf{F}} \mathbb{1}_n^\top)\mathbf{x}_0 = \frac{\mathbb{1}_n \mathbb{1}_n^\top}{\mathbb{1}_n^\top \mathbb{1}_n}\mathbf{x}_0 = \mathrm{avg}(\mathbf{x}_0)\mathbb{1}_n; \tag{2.23}$$

where agv *is the average operator that computes the mean of all the components of its vector argument.*

It is worth to note that the gist of stochasticity plays a crucial role in Theorem 2.1.7, since all the convergence results depend on this peculiar property. Moreover, doubly stochastic matrices are the key to fulfill the so-called *average consensus*, i.e. the convergence of the state toward the average of the given initial condition. To achieve the latter configuration for $\mathbf{F}$ several techniques are available, e.g. the well-known design of $\mathbf{F}$ based on Metropolis-Hastings weights defined as

$$\begin{cases} \mathrm{f}_{ij} = \dfrac{1}{1+\max(\deg(v_i),\deg(v_j))}, & \text{for } i \neq j \\ \mathrm{f}_{ii} = 1 - \sum\limits_{j\in\mathcal{N}_i} \mathrm{f}_{ij}, & \text{otherwise.} \end{cases} \tag{2.24}$$

It is also remarkable to highlight the profound connection between that the agreement protocol in (2.14) and the time evolution given by (2.20). It is possible to monitor the progress of the agreement dynamics (2.14) at $T_s > 0$ time intervals,

$$\mathbf{x}(t+T_s) = \mathrm{e}^{-T_s\mathbf{L}}\mathbf{x}(t), \qquad t = kT_s, \quad k \in \mathbb{N}. \tag{2.25}$$

To motivate this statement, the following proposition from Mesbahi and Egerstedt (2010) is reported.

Proposition 2.1.8.
For all graphs $\mathcal{G}$ *with* n *nodes and sampling intervals* T_s*, one has*

$$\mathrm{e}^{-T_s\mathbf{L}}\mathbb{1}_n = \mathbb{1}_n \qquad \textit{and} \qquad \mathrm{e}^{-T_s\mathbf{L}} \geq 0; \tag{2.26}$$

that is, for all $\mathcal{G}$ *and* $T_s > 0$, $\mathrm{e}^{-T_s\mathbf{L}}$ *is a stochastic matrix. In fact, the right and left eigenvectors of* $\mathrm{e}^{-T_s\mathbf{L}}$ *are those of* $\mathbf{L} \in \mathbb{R}^{n\times n}$*, respectively, associated with eigenvalues* $\mathrm{e}^{T_s\lambda_i^{\mathbf{L}}}$*, for* $i = 0, \ldots, n-1$.

Lastly, to give a rough idea of the importance of Consensus Theory, a final example of network design based on agreement dynamics is proposed in the following lines.

Example 2.1.9 (Camera network calibration)**.**
Consider a distributed planar camera network of $n = 3$ digital cameras, each measuring a bearing angle $x_i(\tau)$, for $i = 1, \ldots, n$, and forming a connected undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{W})$ weighted as in Fig. 2.7 by the real constants $\bar{\mathrm{f}}_{12}, \bar{\mathrm{f}}_{13} > 0$.

In order to accomplish an initial balanced calibration among cameras, the agreement protocol

$$\dot{x}_i(\tau) = \sum_{v_j\in\mathcal{N}_i} \bar{\mathrm{f}}_{ij}(x_j(\tau) - x_i(\tau)), \quad \forall i, j = 1, \ldots, n \tag{2.27}$$

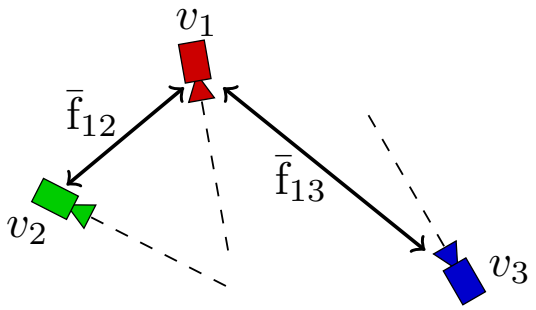


Figure 2.7. Patrolling system made of 3 cameras $v_1, v_2, v_3 \in \mathcal{V}$. Thick arrows depict the information exchange between a pair of cameras, while dashed lines denote the reference for bearing angles for each element of the system.

can, in principle, be exploited to compute the distributed average of the current angles. However, since the measurements are taken digitally, it is reasonable to assume that the evolution of the i-th consensus is yielded by the discrete-time model

$$x_i(t+T_s) = x_i(t) + T_s \sum_{v_j \in \mathcal{N}_i} \bar{\mathrm{f}}_{ij}(x_j(t) - x_i(t)) \tag{2.28}$$

for sufficiently small sample intervals $T_s > 0$. Rewriting the dynamics in (2.28), one obtains

$$\mathbf{x}(t+T_s) = \mathbf{F}\mathbf{x}(t), \quad \mathbf{F} = \begin{bmatrix} 1 - T_s\bar{\mathrm{f}}_{12} - T_s\bar{\mathrm{f}}_{13} & T_s\bar{\mathrm{f}}_{12} & T_s\bar{\mathrm{f}}_{13} \\ T_s\bar{\mathrm{f}}_{12} & 1 - T_s\bar{\mathrm{f}}_{12} & 0 \\ T_s\bar{\mathrm{f}}_{13} & 0 & 1 - T_s\bar{\mathrm{f}}_{13} \end{bmatrix}. \tag{2.29}$$

Therefore, weights $\bar{\mathrm{f}}_{12}, \bar{\mathrm{f}}_{13}$ and sampling interval T_s can be chosen, for instance, trying to maximize the rate of convergence of the relative dynamics, i.e. solving in practice

$$\begin{aligned} (\bar{\mathrm{f}}^\star_{12}, \bar{\mathrm{f}}^\star_{13}, T_s^\star) = \underset{\{\bar{\mathrm{f}}_{12}, \bar{\mathrm{f}}_{13}, T_s\}}{\arg\min} \ & \lambda^{\mathbf{F}} \\ \text{subject to } & \mathbf{F} \in \mathrm{stoc}(\mathbb{R}) \\ & T_s > 0, \end{aligned} \tag{2.30}$$

where $\lambda^{\mathbf{F}}$ is the essential spectral radius of $\mathbf{F}$ (see Sec. B.2). The solution to (2.30) is quite trivial: it yields the spectrum $\Lambda(\mathbf{F}) = \{-0.5, 0.5, 1\}$, and thus $\lambda^{\mathbf{F}} = 0.5$, for $\bar{\mathrm{f}}^\star_{12} = \bar{\mathrm{f}}^\star_{13} = (2T_s^\star)^{-1}$ and any sampling time $T_s^\star > 0$.
To conclude, it is worth to underline that optimization techniques do not only recur during the analysis phase, as mentioned in the previous paragraphs to describe coverage or agreement topics, but it also arises while design. Indeed, in the design phase, optimization can play a key role to achieve the best possible performances (as well as stability and robustness for the system), e.g. in problem (2.30).

Artificial-potential-based Formation Control

According to Hernández-Martínez and Aranda-Bricaire (2011), motion coordination, specifically formation control, is an important research area of mobile multi-agent systems

Yang Quan Chen and Zhongmin Wang (2005). The main goal is to coordinate a group of mobile agents or robots to achieve a desired formation pattern avoiding inter-agent collisions at the same time.
The formation strategies are decentralized because it is assumed that every agent measures the position of a certain subset of agents and, eventually, it detects the position of other agents when a minimal allowed distance is violated and collision danger appears. Thus, the main intention is to achieve desired global behaviors through local interactions. Also, the decentralized approaches offer greater autonomy for the robots, less computational load in control implementations and its applicability to large scale groups. Authors of Desai (2002); Muhammad and Egerstedt (2005) proposed that the possible inter-agent communications and the desired relative position of every agent w.r.t. the others can be represented by a Formation Graph (FG). The application of different FG's to the same group of robots produces different dynamics on the team behavior. In the literature, some special FG topologies are chosen and the convergence to the desired formation and noncollision is analyzed for any number of robots.
A decentralized formation strategy must comply with two fundamental requirements (Cao, Fukunaga, and Kahng (1997)):

1. global convergence to the desired formation;

2. inter-agent collision avoidance.

The standard methodology of Artificial Potential Functions consists in applying the negative gradient of a mixture of Attractive (APF) and Repulsive Potential Functions (RPF) as control inputs to satisfy the convergence and noncollision properties. The APFs are designed according the desired inter-agent distances and steer all agents to the desired formation. The RPFs are based on functions of the distance of a pair of agents. In a decentralized noncollision strategy, a local RPF tends to infinity when two agents collide and vanishes smoothly until the minimal allowed distance is reached. A control law based on APFs only guarantees the global convergence to the formation pattern. However inter-robot collision can occur. The addition of RPFs guarantees the noncollision.

Formation control schemes can be mainly classified into two categories. First, the behavior-based schemes come from the study of animal behaviors where the agents are formed following simple behavior rules, as maintaining a distance between neighbors, swarm intelligence and self-organization, aggregation, flocks, hunter-prey system. This scheme considers to all agents with the same sensing capacities and generally converges to formation patterns without a specific position for every agent (e.g. the Voronoi partition obtained in (2.11)). The second scheme is related to model-based behaviors or emergent behaviors on the context of FGs. Some tools of Combinatorial Theory and Linear Systems Theory are used to analyze the closed-loop system (see Sec. B.2 on Linear Spectral Theory, Sec. B.3 on Graph Theory and, in particular, Sec. B.4 on Rigidity Theory (RT)). Because of significant interest to this work, an example involving the latter approach about FGs is illustrated in the following lines.

Example 2.1.10 (Formation stabilization of a simple integrator)**.**
Given a framework $(\mathcal{G}, \mathbf{p})$ representing the formation graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of n agents with positions $\mathbf{p} = \begin{bmatrix} \mathbf{p}_1^\top & \dots & \mathbf{p}_n^\top \end{bmatrix}^\top \in \mathbb{R}^N$ such that $\mathbf{p}_1, \dots, \mathbf{p}_n \in \mathbb{R}^M$, it is required to achieve

a desired geometric configuration

$$\mathcal{A}_F = \left\{ \mathbf{p} \in \mathbb{R}^N : \|\mathbf{p}_i - \mathbf{p}_j\|_2 = d_{ij},\ \forall e_{ij} \in \mathcal{E} \right\} \tag{2.31}$$

in which d_{ij} represent the desired inter-agent distance between vertex i and vertex j. Set in (2.31) is expressed by means of the relative distances $\mathbf{p}_i - \mathbf{p}_j$, i.e. the displacement vectors $\mathbf{e}_k = \mathbf{e}_{ij} = \mathbf{p}_i - \mathbf{p}_j$ ordered as $k = 1, \dots, E = |\mathcal{E}|$, and represents an affine agreement set for the formation task. Let us define the *formation potential*

$$\mathcal{U}(\mathbf{p}) = \frac{1}{4}\sum_{k=1}^{E}(\|\mathbf{e}_k\|_2^2 - d_k^2)^2 = \frac{1}{4}\sum_{k=1}^{E}\sigma_k^2 = \left\| r_{\mathcal{G}}(\mathbf{p}) - \frac{1}{2}\mathbf{d}^2 \right\|_2^2, \tag{2.32}$$

where $r_{\mathcal{G}}(\mathbf{p})$ is the *rigidity function* (defined in Sec. B.4), $\sigma_k = \mathbf{e}_k^\top \mathbf{e}_k - d_k^2$ is the k-th additive potential and $\mathbf{d}^2$ is the ordered vector whose components are the squared desired distances d_k^2. In order to control the velocities of the n agents, the dynamics yielded by the *gradient system*

$$\dot{\mathbf{p}} = -\nabla \mathcal{U}(\mathbf{p}) \tag{2.33}$$

can be exploited Anderson (2011); Krick, Broucke, and Francis (2009) and, consequently, attain the *formation goal*

$$\lim_{\tau \to +\infty} \|\mathbf{p}_i - \mathbf{p}_j\|_2 = d_{ij}, \quad \forall e_{ij} \in \mathcal{E}. \tag{2.34}$$

It is worth to notice that the gradient dynamics in (2.33) can be rewritten as

$$\dot{\mathbf{p}} = -\frac{1}{2}\sum_{k=1}^{E}\frac{\partial \sigma_k}{\partial \mathbf{p}}\sigma_k = -\boldsymbol{\mathcal{R}}(\mathbf{p})^\top \sigma_k = -\boldsymbol{\mathcal{R}}(\mathbf{p})^\top \boldsymbol{\mathcal{R}}(\mathbf{p})\mathbf{p} - \boldsymbol{\mathcal{R}}(\mathbf{p})^\top \mathbf{d} \tag{2.35}$$

highlighting the fact that it shares many of the aspects belonging to a state-dependent weighted agreement protocol. Observation in (2.35) allows to prove the stability and the convergence of the gradient dynamics in (2.33), assuming that the formation potential in (2.32) is a Lyapunov function. Indeed, denoting with $\boldsymbol{\sigma} = \begin{bmatrix} \sigma_1 & \cdots & \sigma_E \end{bmatrix}^\top \in \mathbb{R}^E$ in which each component of $\boldsymbol{\sigma}$ preserves the same order for the displacement vectors, one obtains

$$\frac{d}{d\tau}\mathcal{U}(\mathbf{p}) = -\boldsymbol{\sigma}^\top \boldsymbol{\mathcal{R}}(\mathbf{p})^\top \boldsymbol{\mathcal{R}}(\mathbf{p})\boldsymbol{\sigma} \leq 0 \tag{2.36}$$

where $\boldsymbol{\mathcal{R}}(\mathbf{p}) \in \mathbb{R}^{E \times N}$ is the *rigidity matrix* (defined in Sec. B.4) associated to framework $(\mathcal{G}, \mathbf{p})$ and $N = Mn$. This results tells that if the rigidity matrix has full row rank then the distributed distance-based formation control law converges (locally[4]) to set (2.31). It is also remarkable to prove that the convergence is exponential, since, leveraging the Min-max Theorem (see Sec. B.2),

$$\frac{d}{d\tau}\mathcal{U}(\mathbf{p}) = -\frac{\boldsymbol{\sigma}^\top \boldsymbol{\mathcal{R}}(\mathbf{p})^\top \boldsymbol{\mathcal{R}}(\mathbf{p})\boldsymbol{\sigma}}{\frac{1}{4}\boldsymbol{\sigma}^\top \boldsymbol{\sigma}}\mathcal{U}(\mathbf{p}) \leq -4\lambda_{MIN}^{\boldsymbol{\mathcal{R}}} \boldsymbol{\mathcal{R}}(\mathbf{p})^\top \boldsymbol{\mathcal{R}}(\mathbf{p})\mathcal{U}(\mathbf{p}) \tag{2.37}$$

[4]See also Sec. B.4 for a short panoramic on the differences between infinitesimal rigidity, related to the local convergence of the formation, and global rigidity.

where $\lambda_{MIN}^{\mathcal{R}}$ is the smallest nonzero eigenvalue of $\mathcal{R}(\mathbf{p})$, also known as the *rigidity eigenvalue*. Again, in this example an optimization perspective can be given, since the formation goal can be, in principle, accomplished by the design of control law based on the potential (2.32), which represents the cost to be minimized.

Looking ahead

Research on multi-agent systems is mainly focused on system architecture, consensus algorithm, distributed optimization, and software tools for simulation and implementation. Agent technologies are crucial now and will continue to evolve in the future. To this end, some interesting contributions that are displayed in the forthcoming chapters can be, indeed, denoted by

1. the extension of distributed approaches to dynamic coverage problems in unknown environments and in a framework with limited sensing capabilities,
2. the application of theoretical methods for multi-agent systems and Trajectory Optimization to steer a swarm of robots taking into account various specifications that exhibit trade-off aspects,
3. the distributed estimation from noisy relative measurements provided by a network by means of highly performing iterative algorithms,
4. the deep mathematical investigation of specific case studies for a networked group of agents (e.g. a surveillance camera network designed with a specific topological pattern).

To conclude this section, it is worth to reaffirm that all the methodologies presented so far are composed of well established theoretical concepts, since they give the possibility to accomplish significant results in practice. On the other hand, in the following section, further methods that may sometimes seem apparently disconnected to the application examples illustrated will be introduced: new perspectives will be given in order to broaden the capability of investigation and fulfill novel challenging goals.

2.2 Theoretical methodologies

In this section, the general research strategy that outlines the way in which this research has been undertaken and identifies the methods used is expounded. Methods discussed in this methodology description define the means or modes of data collection and how a specific result is calculated. Specifically, the modality of application of the following methods is debated for the accomplishment of the thesis purposes:

- *analysis and synthesis of feedback systems*, used to capture, interpret, model and understand the general behavior of a group of elements and design the technology by which a process or procedure is performed with minimal human assistance;
- *Swarm-Robotics-oriented strategies*, employed when the main focus of a project is directed towards the coordination of multiple robots as a system which consists of large numbers of mostly simple physical robots;

- *graph-based motion planning and clustering*, adopted when a robot (or a group of entities that locate in close geographical proximity) has to determine its own position in its reference frame and then to plan a path towards some goal location;

- *iterative methods for Optimization*, exploited to generate a sequence of improving approximate solutions, starting from an initial guess, for a disparate class of optimization problems.

The complex of this techniques represents a large set of useful tools to cope with the mathematical issues and challenges explained in Sec. 2.1.

2.2.1 Analysis and synthesis of feedback systems

A *system* is a group of interacting or interrelated entities that form a unified whole. It is delineated by its spatial and temporal boundaries, surrounded and influenced by its environment, described by its structure and purpose and expressed in its functioning. A feedback system is a system based on *feedback loops*, i.e. actions that occur when the outputs of the system are routed back as inputs as part of a chain of cause-and-effect that forms a circuit. Feedback systems are designed to perform well under significant uncertainties in the system and environment for extended periods of time, and they must be able to compensate for significant failures without external intervention. Examples of feedback systems can be yielded by a group of controlled Unmanned Aerial Vehicles (UAVs), a driven mechanical pendulum or an electronic circuit.

In the following lines, some important preliminaries addressing the analysis and synthesis of feedback systems are provided.

A System Theory perspective

System Theory is the interdisciplinary study of systems. According to Fornasini (2013), in the *analysis* framework, the most immediate goal is the evaluation of the dynamic behavior of the system consequent to the choice of initial state and the input functions, or to the variation of one or more characteristic parameters of the adopted model. Whereas, once completed an analysis phase, the obtained results are utilized to proceed towards the input signal *synthesis*, which corresponds to a desired trajectory of the state and the output.

To translate this methodology into practice, the construction of *mathematical models* is essential, since these have the ability to provide a description of a system using mathematical language. The leanings to apply analytical techniques, not only in the study of natural phenomena or technological processes but also in subjects that have been historically tackled in a qualitative fashion, has broaden the range of problems that are studied leveraging mathematical methodologies.
The mathematical language is less rich than commonly spoken languages; however the recourse to abstract systems and their formalized language is crucial whenever one wants to effectuate an analysis leading to quantifiable results and whenever, in a synthesis framework, one wants to compare the effects of different control and management strategies. Furthermore, the construction process of a mathematical model does not only

have interest from the final-result point of view, but also in the intermediate stages of study and clarification for the phenomenon under examination.

Typically, the dynamics of autonomous systems is represented by means of state-space models. A *state-space representation* is a mathematical model of a physical system as a set of input, output and state variables related by first-order differential equations (for continuous quantities) or difference equations (for discrete quantities), e.g. system in (2.3)-(2.4). State variables are variables whose values evolve through time in a way that depends on the values they have at any given time and also depends on the externally imposed values of input variables. Output variables' values depend on the values of the state variables. To conclude, the "state space" is the Euclidean space in which the variables on the axes are the state variables. The state of the system can be represented as a vector within that space. To abstract from the number of inputs, outputs and states, these variables are expressed as vectors. Additionally, if the dynamic system is linear, time-invariant, and finite-dimensional, then the differential and algebraic equations may be written in matrix form.

State feedback and general control goals

Often, the design of input signals is based on the analysis of a phenomenon called *feedback*: the latter represents the capability of a dynamic system to take into account output results in order to modify features and attributes of the system itself. To implement feedback loops on an feedback system, the design of feedback controllers is crucial. According to Nerode and Kohn (1992), a *feedback controller* or *closed-loop controller* is a system that interacts with another system called *plant* corresponding to a pre-established configuration. Whereas, in an *open-loop controller*, also called a *non-feedback controller*, the control action from the controller is independent of the state of the plant (see Fig. 2.8). The function of any controller is to drive the overall system, composed of the plant and the controller, to a possibly time varying condition, called *the goal*, while exhibiting a behavior that satisfies stated requirements and constraints in the presence of environmental, structural, and knowledge uncertainties.

Every feedback controller is characterized by a mapping whose domain is the Cartesian product of the goal and sensor data spaces and whose range is the space of feasible actions. Here, the word space has a generic sense, implying a set of elements together with associated algebraic, topological and measure structures. This mapping is referred to as the *control law*, e.g. (2.5). The control law is thus a recipe for processing in real time sensor and goal data, and generating action commands to the plant so that the system satisfies the given requirements.
Usually, there are three fundamental aspects to be taken into account while designing a control law: stability, performances and robustness. A system is said to be *stable*, if its state and output are under control, i.e. if they can be driven accordingly to the given purpose without causing any ungovernable, risky or dangerous behavior. Obviously, stability represents the most important requirement for feedback systems; nevertheless, performances and robustness make a relevant difference in real life applications. Good performances are attained when all the control goals are achieved w.r.t. a certain physical quantity, e.g. when the settling time specification is respected. Whereas, robustness is obtained if the control law is able to achieve the given specifications even in case of

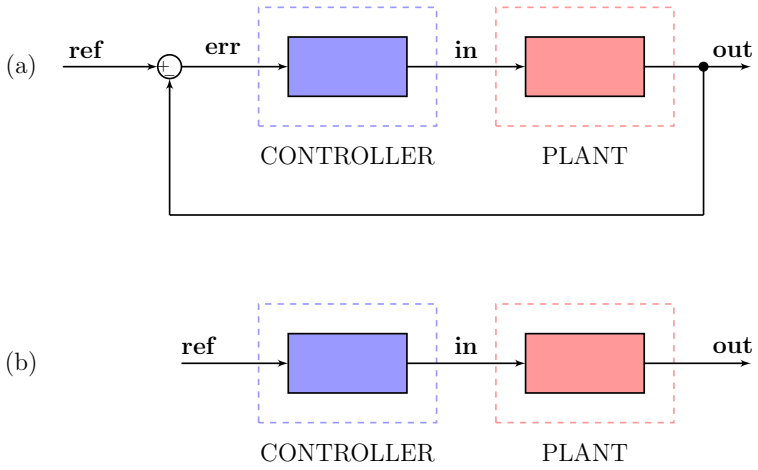


Figure 2.8. Different control interconnection strategies. (a) Closed-loop control. The output **out** is fed back and compared with the reference signal **ref** to obtain the error **err**. The latter is then used to compute the input **in** through the controller. Finally, the input governs the plant to achieve the goal. (b) Open-loop control. The reference signal is directly used to compute the input that governs the plant. No error is taken into consideration.

model uncertainties, presence of faults, usage of noisy sensor/actuator measurements or different operating conditions for the system. Finally, to give a concrete idea of these essential concepts, an illustrative example is proposed below.

Example 2.2.1 (Design of control laws for simple cases)**.**
Given the discrete linear system Σ_ϑ parametrized by $\vartheta \in (1, 2)$ with state $\mathbf{x} \in \mathbb{R}^2$, input $\mathbf{u} \in \mathbb{R}$, output $\mathbf{y} \in \mathbb{R}^2$

$$\Sigma_\vartheta : \begin{cases} \mathbf{x}(t+1) = \mathbf{A}_\vartheta \mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) \end{cases} = \begin{cases} \mathbf{x}(t+1) = \begin{bmatrix} -0.5 & 0 \\ 0 & \vartheta \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \mathbf{u}(t) \\ \mathbf{y}(t) = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \mathbf{x}(t) \end{cases} \tag{2.38}$$

and assuming familiarity with the System Theory, it is briefly shown how to design

(i) an output feedback control law $\mathbf{u}_R$ that stabilizes Σ_ϑ such that $\mathbf{u}_R$ guarantees robustness to variations in ϑ;

(ii) an output feedback control law $\mathbf{u}_P$ that stabilizes Σ_ϑ such that $\mathbf{u}_P$ ensures the best convergence performance, provided that ϑ is known.

A generic output feedback control law $\mathbf{u}$ for a linear system $\Sigma = (\mathbf{A}, \mathbf{B}, \mathbf{C})$, in which $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ are the update, input and output matrices respectively, is such that $\mathbf{u} = \mathbf{u}(\mathbf{x}) = \mathbf{K}\mathbf{y} = \mathbf{K}\mathbf{C}\mathbf{x}$, where $\mathbf{x}$ and $\mathbf{y}$ are the state and the output respectively. In order to solve both problems (i) and (ii), it is useful to compute the feedback update

matrix, i.e.

$$\mathbf{A}_\vartheta + \mathbf{BKC} = \begin{bmatrix} -0.5 + K_1 & -K_2 \\ K_1 & \vartheta - K_2 \end{bmatrix}, \qquad \mathbf{K} = \begin{bmatrix} K_1 & K_2 \end{bmatrix}. \tag{2.39}$$

Then, to accomplish task (i), whose goal is to achieve stability and robustness for the system, one may trivially choose $\mathbf{u}_R = \mathbf{K}_R\mathbf{y}$ such that

$$K_{R,1} = 0, \qquad K_{R,2} = 1.5, \tag{2.40}$$

from the information given by (2.39). On the other hand, to attain task (ii), whose goal is to achieve stability and optimal performances provided that ϑ is now known, the analysis of the *characteristic polynomial* of (2.39)

$$\Delta(\lambda) = \lambda^2 + (K_1 - K_2 + \vartheta - 0.5)\lambda + (\vartheta K_1 + 0.5K_2 - 0.5\vartheta) \tag{2.41}$$

leads to a deadbeat controller $\mathbf{u}_P = \mathbf{K}_P\mathbf{y}$ such that

$$K_{P,1} = \frac{0.25}{0.5 + \vartheta}, \qquad K_{P,2} = \frac{\vartheta^2}{0.5 + \vartheta}, \tag{2.42}$$

by imposing $\Delta(\lambda) = \lambda^2$.

In this basic example, the role of *spectral allocation* is highlighted. The leitmotiv of control law design for a (discrete[5]) time-invariant linear system $\Sigma = (\mathbf{A}, \mathbf{B}, \mathbf{C})$ is indeed the task-oriented placement of the zeros of the characteristic polynomial, which coincide with the poles of the (discrete) *transfer function*

$$\mathbf{W}(z) = \mathbf{C}(z\mathbf{I}_N - \mathbf{A})^{-1}\mathbf{B} \in \mathbb{R}^{N\times N}. \tag{2.43}$$

Sensitivity analysis to parameter variations

The controller parameters are typically matched to the plant characteristics and since the plant may change, it is important that they are chosen in such a way that the closed loop system is not sensitive to variations in the plant dynamics.
Denoting with $\mathbf{W}(z, \vartheta)$ the closed-loop transfer function of a generic[6] system Σ_ϑ that is parametrized by $\vartheta \in \mathbb{R}$, the sensitivity to parameter variations of Σ_ϑ w.r.t. ϑ can be quantified as

$$S_\vartheta(z) = \frac{\partial \ln\left(\det\left[\mathbf{W}(z,\vartheta)\right]\right)}{\partial \ln(\vartheta)}. \tag{2.44}$$

Applying differentiation rules and dividing by ϑ, definition (2.44) yields

$$\bar{S}_\vartheta(z) = \operatorname{tr}\left[\mathbf{W}(z,\vartheta)^{-\top}\frac{\partial \mathbf{W}(z,\vartheta)}{\partial \vartheta}\right], \tag{2.45}$$

where $\operatorname{tr}[\cdot]$ denotes the trace operator. Remarkably, (2.45) can be interpreted as the multivariate generalization of the classic relative sensitivity function J. and M. (2008) for

[5] These concepts also apply in the continuous framework, replacing the Z-transform by the Laplace transform and using common sense.
[6] Discrete and time-invariant, for the purposes of this thesis.

a scalar version $W(z,\vartheta)$ of transfer function $\mathbf{W}(z,\vartheta)$, since, for a small increment $\Delta\vartheta$ of parameter ϑ, one has the first order Taylor approximation

$$W(z,\vartheta+\Delta\vartheta) \simeq W(z,\vartheta) + \frac{\partial W(z,\vartheta)}{\partial\vartheta}\Delta\vartheta = W(z,\vartheta)\left(1+\bar{S}_\vartheta(z)\Delta\vartheta\right). \tag{2.46}$$

Hence, the modulus of (2.45) provides a normalized indicator that measures the robustness of system Σ_ϑ. In formulas, if $|\bar{S}_\vartheta(z)|$ tends to 0 for a fixed z then variations of ϑ do not influence the response of Σ_ϑ at that frequency. Furthermore, relative sensitivity in (2.45) is also related to the precision of the frequency response of Σ_ϑ at steady state conditions: $\bar{S}_\vartheta(1)=0$ holds[7] if and only if the perturbed system $\Sigma_{\vartheta+\Delta\vartheta}$ described by $W(z,\vartheta+\Delta\vartheta)$ maintains the same step response of the nominal system Σ_ϑ.

Derivation of optimal control laws

Optimal Control is an extension of the Calculus of Variations (CV), and is a mathematical optimization method for deriving control policies Sargent (2000), namely, it can be seen as a control strategy in Control Theory. The method is largely due to the work of Lev Pontryagin and Richard Bellman in the 1950s, after contributions to CV (see also Sec. B.5) by Edward J. McShane Bryson (1996). In the following lines, the importance of the Pontryagin's Minimum Principle and the role of Trajectory Optimization to derive control laws is highlighted and briefly discussed (more details can be found in Fornasini (2013); Vinter (2010) and Rao (2014), respectively).

Theorem 2.2.2 (Pontryagin's Minimum Principle - restricted version[8])**.**
Assume that the input function $\mathbf{u}(\cdot)$*, subject to condition* $\mathbf{u}(\tau)\in U\subseteq\mathbb{R}^{N_I}$*,* $\forall\tau\in[0,T]$*, represents the optimal control that minimizes the objective* (2.6) *for* (2.3)-(2.4)*. Let* $\mathbf{x}(\cdot)\in X\subseteq\mathbb{R}^{N_S}$ *be the correspondent state trajectory and* $\boldsymbol{\lambda}(\cdot):[0,T]\to\mathbb{R}^{N_S}$ *a function*[9] *such that* $\mathbf{u}(\cdot)$*,* $\mathbf{x}(\cdot)$ *and* $\boldsymbol{\lambda}(\cdot)$ *satisfy*

$$\frac{d\mathbf{x}(\tau)}{d\tau} = \mathbf{f}(\mathbf{x}(\tau),\mathbf{u}(\tau)), \tag{2.47}$$

$$\mathbf{x}(0) = \mathbf{x}_0; \tag{2.48}$$

$$-\dot{\boldsymbol{\lambda}}(\tau)^\top = \boldsymbol{\lambda}(\tau)^\top\frac{\partial\mathbf{f}(\mathbf{x}(\tau),\mathbf{u}(\tau))}{\partial\mathbf{x}} + \frac{\partial l(\mathbf{x}(\tau),\mathbf{u}(\tau),\tau)}{\partial\mathbf{x}}, \tag{2.49}$$

$$\boldsymbol{\lambda}(T)^\top = \frac{\partial m(\mathbf{x}(T))}{\partial\mathbf{x}}. \tag{2.50}$$

Therefore, the Hamiltonian function

$$\mathrm{H}(\boldsymbol{\lambda},\mathbf{x},\mathbf{u}) = \boldsymbol{\lambda}^\top\mathbf{f}(\mathbf{x},\mathbf{u}) + l(\mathbf{x},\mathbf{u},\tau) \tag{2.51}$$

satisfies, for all $\tau\in[0,T]$ *and for all* $\bar{\mathbf{u}}\in U$*, the inequality*

$$\mathrm{H}(\boldsymbol{\lambda}(\tau),\mathbf{x}(\tau),\bar{\mathbf{u}}) \leq \mathrm{H}(\boldsymbol{\lambda}(\tau),\mathbf{x}(\tau),\mathbf{u}(\tau)). \tag{2.52}$$

[7] $\bar{S}_\vartheta(0)=0$ for continuous-time systems.
[8] The most general version of Pontryagin's Minimum Principle applies to (almost) any possible objective functional and choice of constraints. However, a treatise on the general version would require a much longer discussion, falling outside the thesis purposes.
[9] Components of $\boldsymbol{\lambda}$ are usually called *generalized Lagrange multipliers*.

The analysis of Theorem 2.2.2 represents a valuable starting point to understand how Optimal Control problem can be generally tackled. The discussion begins from the definition of the Hamiltonian function in (2.51) and the optimization w.r.t. the *augmented objective*

$$\hat{h} = h(\mathbf{x}(\cdot), \mathbf{u}(\cdot)) - \int_0^T \boldsymbol{\lambda}(\tau)^\top \left[\dot{\mathbf{x}}(\tau) - \mathbf{f}(\mathbf{x}(\tau), \mathbf{u}(\tau))\right] d\tau \tag{2.53}$$

$$= m(\mathbf{x}(T)) + \int_0^T \left[\mathrm{H}(\boldsymbol{\lambda}(\tau), \mathbf{x}(\tau), \mathbf{u}(\tau)) - \boldsymbol{\lambda}(\tau)^\top \dot{\mathbf{x}}(\tau)\right] d\tau. \tag{2.54}$$

It is worth to note that $\hat{h} = h$ whenever $\mathbf{x}(\cdot)$ and $\mathbf{u}(\cdot)$ satisfy (2.3)-(2.4). By using a variational approach, the first order variation $\delta\hat{h}$ of (2.53)-(2.54) is given by

$$\delta\hat{h} = \delta m(\mathbf{x}(T)) + \int_0^T \delta\left[\mathrm{H}(\boldsymbol{\lambda}(\tau), \mathbf{x}(\tau), \mathbf{u}(\tau)) - \boldsymbol{\lambda}(\tau)^\top \dot{\mathbf{x}}(\tau)\right] d\tau \tag{2.55}$$

$$\simeq \left(\frac{\partial m(\mathbf{x}(T))}{\partial \mathbf{x}} - \boldsymbol{\lambda}(T)^\top\right) \delta\mathbf{x}(T) + \int_0^T \left[\dot{\boldsymbol{\lambda}}(\tau)^\top + \boldsymbol{\lambda}(\tau)^\top \frac{\partial \mathbf{f}(\mathbf{x}(\tau), \mathbf{u}(\tau))}{\partial \mathbf{x}}\right] \delta\mathbf{x}(\tau) d\tau \tag{2.56}$$

$$+ \boldsymbol{\lambda}(0)^\top \delta\mathbf{x}(0) + \int_0^T \left[\mathrm{H}(\boldsymbol{\lambda}(\tau), \mathbf{x}(\tau), \bar{\mathbf{u}}) - \mathrm{H}(\boldsymbol{\lambda}(\tau), \mathbf{x}(\tau), \mathbf{u}(\tau))\right] \delta\mathbf{x}(\tau) d\tau \tag{2.57}$$

where symbol $\simeq$ addresses the assumption that $\|\delta\mathbf{x}(\cdot)\|$ is sufficiently small. Hence, optimality condition $\delta\hat{h} = 0$ can be satisfied leveraging the *state equation* (2.47)-(2.48), which can be written as

$$\begin{cases} \dot{\mathbf{x}}(\tau) = \nabla_{\boldsymbol{\lambda}} \mathrm{H}(\boldsymbol{\lambda}(\tau), \mathbf{x}(\tau), \mathbf{u}(\tau)) \\ \mathbf{x}(0) = \mathbf{x}_0 \end{cases}, \tag{2.58}$$

the *adjoint (or costate) equation* (2.49)-(2.50), which can be written as

$$\begin{cases} \dot{\boldsymbol{\lambda}}(\tau) = -\nabla_{\mathbf{x}} \mathrm{H}(\boldsymbol{\lambda}(\tau), \mathbf{x}(\tau), \mathbf{u}(\tau)) \\ \boldsymbol{\lambda}(T) = \nabla_{\mathbf{x}} m(\mathbf{x}(T)) \end{cases}, \tag{2.59}$$

and, lastly, exploiting inequality (2.52). Variables $\mathbf{u}(\cdot)$, $\mathbf{x}(\cdot)$ and $\boldsymbol{\lambda}(\cdot)$ are considered unknown for the $2N_S$ scalar equations yielded by (2.58)-(2.59). Furthermore, inequality (2.52) leads to the remaining N_I algebraic equations obtained by imposing the *optimal control equation*

$$\mathbf{0}_n = \nabla_{\mathbf{u}} \mathrm{H}(\boldsymbol{\lambda}(\tau), \mathbf{x}(\tau), \mathbf{u}(\tau)), \tag{2.60}$$

which can be exploited if a solution $\bar{\mathbf{u}}(\cdot)$ belongs to the interior of U (see Subsec. B.1.1). Assuming that an optimal input $\bar{\mathbf{u}}(\cdot) \in \mathrm{int}(U)$ yielded by equation (2.60) satisfies

$$\frac{\partial^2 \mathrm{H}(\boldsymbol{\lambda}(\tau), \mathbf{x}(\tau), \bar{\mathbf{u}})}{\partial \mathbf{u}^2} \succ 0, \quad \text{for all } \tau \in [0, T], \tag{2.61}$$

is a sufficient condition to affirm that $\bar{\mathbf{u}}$ verifies the so-called *Minimum Principle*, i.e. $\bar{\mathbf{u}}$ minimizes the Hamiltonian function:

$$\min_{\mathbf{u} \in U} \mathrm{H}(\boldsymbol{\lambda}(\tau), \mathbf{x}(\tau), \mathbf{u}(\tau)) = \mathrm{H}(\boldsymbol{\lambda}(\tau), \mathbf{x}(\tau), \bar{\mathbf{u}}). \tag{2.62}$$

This can be justified by the fact that, if (2.62) holds, then the last term in (2.57) is negative, indicating that the variation $\delta\hat{h}$ is also negative[10], meaning that $\bar{\mathbf{u}}$ decreases the value of h. In other words, assuming by contradiction that $\bar{\mathbf{u}}$ imply

$$\mathrm{H}(\boldsymbol{\lambda}(\tau), \mathbf{x}(\tau), \bar{\mathbf{u}}) > \mathrm{H}(\boldsymbol{\lambda}(\tau), \mathbf{x}(\tau), \mathbf{u}(\tau)), \tag{2.63}$$

it follows that the values assumed by functional h for the optimal input $\bar{\mathbf{u}}$ are greater of those for a generic input $\mathbf{u}$, violating the hypothesis that $\bar{\mathbf{u}}$ minimizes h.

Example 2.2.3 (Time-Invariant Linear Quadratic Optimal Control)**.**
Time-Invariant Linear Quadratic (TILQ) optimal control problems can be described as an application of Theorem 2.2.2 to objectives defined in (2.6) whenever those are written as the quadratic costs

$$l(\mathbf{x}, \mathbf{u}) \quad = \mathbf{a}^\top \mathbf{x} + \mathbf{b}^\top \mathbf{u} + \frac{1}{2}\mathbf{x}^\top \mathbf{Q}\mathbf{x} + \frac{1}{2}\mathbf{u}^\top \mathbf{R}\mathbf{u} + \mathbf{x}^\top \mathbf{S}\mathbf{u}, \quad \mathbf{Q} \succeq 0,\ \mathbf{R} \succ 0; \tag{2.64}$$

$$m(\mathbf{x}(T)) = \mathbf{r}_1^\top \mathbf{x}(T) + \frac{1}{2}\mathbf{x}(T)^\top \mathbf{P}_1 \mathbf{x}(T), \qquad \mathbf{P}_1 \succeq 0; \tag{2.65}$$

and are subject to a linear dynamics

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \quad \mathbf{x}(0) = \mathbf{x}_0; \tag{2.66}$$

in which $\mathbf{x} \in \mathbb{R}^{N_S}$ and $\mathbf{u} \in \mathbb{R}^{N_I}$ represent the state and input, respectively.
The Hamiltonian related to this control problem is given by

$$\mathrm{H}(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{u}) = \boldsymbol{\lambda}^\top (\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}) + \mathbf{a}^\top \mathbf{x} + \mathbf{b}^\top \mathbf{u} + \frac{1}{2}\mathbf{x}^\top \mathbf{Q}\mathbf{x} + \frac{1}{2}\mathbf{u}^\top \mathbf{R}\mathbf{u} + \mathbf{x}^\top \mathbf{S}\mathbf{u}; \tag{2.67}$$

therefore, the optimal control equation in (2.60) yields an optimal input of the form

$$\bar{\mathbf{u}} = -\mathbf{R}^{-1}(\mathbf{S}^\top \mathbf{x} + \mathbf{B}^\top \boldsymbol{\lambda} + \mathbf{b}). \tag{2.68}$$

Also, resorting to the state and adjoint equations in (2.58)-(2.59) and assuming $\mathbf{u} = \bar{\mathbf{u}}$, one obtains

$$\begin{cases} \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\boldsymbol{\lambda}} \end{bmatrix} = \begin{bmatrix} \mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{S}^\top & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B} \\ \mathbf{S}\mathbf{R}^{-1}\mathbf{S}^\top - \mathbf{Q} & \mathbf{S}\mathbf{R}^{-1}\mathbf{B}^\top - \mathbf{A}^\top \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\lambda} \end{bmatrix} + \begin{bmatrix} -\mathbf{B}\mathbf{R}^{-1}\mathbf{b} \\ \mathbf{S}\mathbf{R}^{-1}\mathbf{b} - \mathbf{a} \end{bmatrix} \\ \begin{bmatrix} \mathbf{x}(0) \\ \boldsymbol{\lambda}(T) \end{bmatrix} = \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{P}_1 \mathbf{x}(T) + \mathbf{r}_1 \end{bmatrix} \end{cases} \tag{2.69}$$

Now, the solution to this problem rests upon the key idea that the overall Hamiltonian boundary-value problem (HBVP) in (2.69) can be tackled exploiting the linearity of the dynamics, that is solving backwards the following differential equation

$$\begin{bmatrix} \mathbf{x}(\tau) \\ \boldsymbol{\lambda}(\tau) \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{N_S} \\ \mathbf{P}(\tau) \end{bmatrix} \mathbf{x}(\tau) + \begin{bmatrix} \mathbf{0}_{N_S} \\ \mathbf{r}(\tau) \end{bmatrix}, \quad \mathbf{P}(T) = \mathbf{P}_1,\ \mathbf{r}(T) = \mathbf{r}_1 \tag{2.70}$$

for $\mathbf{x}(T)$ fixed, where $\mathbf{P} \in \mathbb{R}^{N_S \times N_S}$ is an unknown symmetric matrix and $\mathbf{r} \in \mathbb{R}^{N_S}$ is an

[10] $\delta\mathbf{x}(0) = 0$ since $\mathbf{x}(0)$ is fixed.

unknown affine vector. Combining the differential equations in (2.69) with those in (2.70) and assigning $\mathbf{K} := \mathbf{R}^{-1}(\mathbf{S}^\top + \mathbf{B}^\top\mathbf{P})$, it is possible to derive the first order optimality condition related to this problem

$$\forall \mathbf{x}: \quad \left[\dot{\mathbf{P}} + (\mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{S}^\top)^\top\mathbf{P} + \mathbf{P}(\mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{S}^\top) - \mathbf{K}^\top\mathbf{R}\mathbf{K} + \mathbf{Q}\right]\mathbf{x} + \\ \dot{\mathbf{r}} + (\mathbf{A} - \mathbf{B}\mathbf{K})^\top\mathbf{r} - \mathbf{K}^\top\mathbf{r} + \mathbf{a} = \mathbf{0}_{N_S}. \tag{2.71}$$

Remarkably, the two terms located at the different lines in (2.71) can be distinguished. In particular, for (2.71) to hold, it is sufficient that the term in the squared brackets be arranged into a Riccati equation, such that, with the same value for $\mathbf{K}$, both the differential equations

$$-\dot{\mathbf{P}} = (\mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{S}^\top)^\top\mathbf{P} + \mathbf{P}(\mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{S}^\top) - \mathbf{K}^\top\mathbf{R}\mathbf{K} + \mathbf{Q} \tag{2.72}$$

and

$$-\dot{\mathbf{r}} = (\mathbf{A} - \mathbf{B}\mathbf{K})^\top\mathbf{r} - \mathbf{K}^\top\mathbf{r} + \mathbf{a} \tag{2.73}$$

hold independently, since $\mathbf{r}$ is assumed to be affine w.r.t. $\mathbf{x}$.
In practice, one exploits optimality conditions (2.72)-(2.73) to obtain the optimal control input $\bar{\mathbf{u}}$ in the following steps:

1. find the solution[11] $\mathbf{P}_o(\cdot)$ that solves the Riccati equation in (2.72) backwards in time, assuming $\mathbf{P}_o(T) = \mathbf{P}_1$;
2. compute the *optimal feedback gain* $\mathbf{K}_o := \mathbf{R}^{-1}(\mathbf{S}^\top + \mathbf{B}^\top\mathbf{P}_o)$;
3. solve backward in time equation (2.73), assuming $\mathbf{r}(T) = \mathbf{r}_1$ and using $\mathbf{K}_o$, to find the solution $\mathbf{r}_o(\cdot)$;
4. lastly, from (2.68), the optimal control input is yielded by

$$\bar{\mathbf{u}} = -\mathbf{K}_o\mathbf{x} - \mathbf{R}^{-1}(\mathbf{B}^\top\mathbf{r}_o + \mathbf{b}). \tag{2.74}$$

To conclude, it is worth to notice that the second order sufficient condition in (2.61) is satisfied in this framework since

$$\forall(\mathbf{x}, \mathbf{u}): \quad \frac{\partial^2 \mathrm{H}}{\partial \mathbf{u}^2} = \mathbf{R} \succ 0; \tag{2.75}$$

hence, input $\bar{\mathbf{u}}$ has to be consider a minimizer for the correspondent objective.

On the other hand, if none of the solution $\bar{\mathbf{u}}$ provided by (2.60) and verifying (2.61) belongs to $\mathrm{int}(U)$, a procedure of general validity to satisfy (2.62) does not exist.

To face the latter issue, Trajectory Optimization may help when full closed-loop solutions are either impossible or impractical. *Trajectory Optimization* (TO) is a process where it is desired to determine the path and the corresponding input (control) to a dynamic system that meet specified constraints on the system while optimizing a specified performance index. Generally speaking, trajectory optimization is a technique for computing an open-loop solution to an optimal control problem. Although the idea

[11] It can be shown that this solution it exists unique and, as expected, is symmetric.

of trajectory optimization has been around for hundreds of years (CV, brachystochrone problem), it only became practical for real-world problems with the advent of the computer. Many of the original applications of TO were in the aerospace industry, computing rocket and missile launch trajectories. Because complexity of modern applications has increased tremendously as compared to applications of the past, methods for TO continue to evolve and the discipline is becoming increasingly relevant in a wide range of subject including virtually all branches of engineering, economics, and medicine.

As a result of the complexity of most applications, optimal trajectories are typically generated using numerical methods. Numerical methods for TO are divided into two major classes: *indirect methods* and *direct methods*. In an indirect method, the first-order optimality conditions from variational calculus are employed. The TO problem is then converted into a multiple-point HBVP, as shown in (2.69). The HBVP is then solved numerically to determine candidate optimal trajectories called extremals. Each extremal solution of the HBVP is then examined to see if it is a local minimum, maximum, or a saddle point, and the extremal with the lowest cost is chosen. In a direct method, the state and/or control of the original TO problem is approximated by parameterizing the state and/or the control and the TO problem is transcribed to a finite-dimensional nonlinear programming problem (NLP), as illustrated in (2.1). The NLP is then solved using well known optimization techniques.

2.2.2 Swarm-Robotics-oriented strategies

The concept of swarm intelligence is based on the collective social behaviour of decentralized body, either natural or artificial like ant, fish, bird, bee etc. Swarm intelligence has gained very high priority among the researchers from different fields like commerce, science and engineering. Multiple editions of swarm intelligence's techniques made it suitable for optimization problems.
Swarm intelligence, is composed by three main principles: evaluation, comparing and imitation SinghPal and Sharma (2013). Evaluation is able to identify what is positive or negative in nature. Learning won't happen unless beings are capable of evaluate the attractive and repulsive characteristics of the environment. In comparison, living beings compare themselves with other beings as results of these comparisons may become a motivation to learning and/or modification. Imitation is an effective form of learning.

Swarm-Robotics-oriented strategies can be exploited in order to accomplish complex tasks as coverage or formation flocking in a distributed fashion. In this paragraph, few tools to accomplish robotic coverage, as the hexagonal packing policy Chang and Wang (2010) and the Simplicial Complex Theory Hatcher (2002); Nakahara (2003), are introduced.

Hexagonal-packing-based policies

In Geometry, circle packing addresses the study of circles arrangement on the plane such that no overlapping occurs. This can be seen as the two-dimensional analog of Kepler's sphere packing problem proposed in 1611. A circle configuration referring to the centers of circles is a collection of points such that the distance between any two points in the collection is equal to or greater than twice the common radius. Imagine

filling a wide container with small circles with the same radius inside. The density of the arrangement is provided the ratio between the area of the container that is taken up by the circles and the area of the container itself. In order to maximize the number of circles in the container, one has to find a distribution with the highest possible density, so that the circles are packed together as closely as possible. Therefore, the density of a circle configuration is the asymptotic limit on density with the container getting larger and larger. In 1773, Lagrange showed that the minimal density is $\pi/\sqrt{12}$ under the hypothesis that the circle structures are lattices. Furthermore, in 1831, Gauss demonstrated that the minimal density of a 3-dimensional sphere packing is $\pi/\sqrt{18}$ under the hypothesis that the sphere structures are lattices. Without the lattice assumption, the first proof of circle packing problem was conceived by Axel Thue. Nevertheless, it is generally considered that Thue's original proof was incomplete and that the first exhaustive demonstration of this evidence was developed by L. F. Toth (1940). Later, several various proofs were also suggested by Segre and Mahler, Davenport, and Hsiang. On the other hand, leveraging Delaunay triangulations, it is possible to show that the density of a Delaunay triangle is less than or equal to $\pi/\sqrt{12}$. This implies that the density of the union of any finite Delaunay triangles in a saturated circle configuration is also less than or equal to $\pi/\sqrt{12}$. Therefore, a simple proof of Thue's theorem can be obtained as follows.

Theorem 2.2.4 (Thue's hexagonal packing)**.**
The hexagonal lattice pattern depicted in Fig. 2.9 is $\pi/\sqrt{12}$ and it is the densest of all possible circle packings.

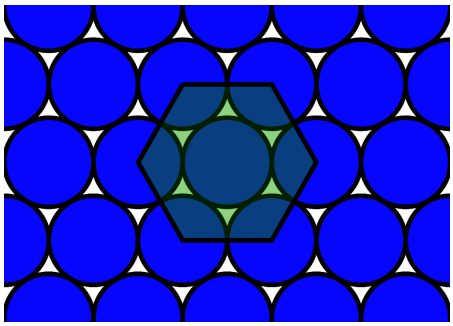


Figure 2.9. Identical circles in a hexagonal packing arrangement, the densest packing possible.

To conclude, Theorem 2.2.4 states that the regular hexagonal packing is the densest circle packing in the plane, yielding that the density of this circle configuration is almost 90.7%: this relevant fact can be adopted as a fundamental paradigm to develop efficient coverage algorithms in which the sensor capabilities of robots are established by maximum sensing radii.

Simplicial-complex-based tools

Simplices are an interesting topic of algebraic topology due to their combinatorial nature. Their main advantage is to provide a good way for representing topological structures in computers. As shown in Fig. 2.10, first few low dimensional simplices have their own names: 0-simplex, 1-simplex, 2-simplex and 3-simplex are also called vertex, edge, triangle and tetrahedron, respectively. Along with the concept of simplex, also the gist

of simplicial complex and abstract simplicial complex should be considered. A simplicial complex is defined as a collection of simplices; whereas, abstract simplicial complexes can be imagined as the highest generalization of simplicial complexes.

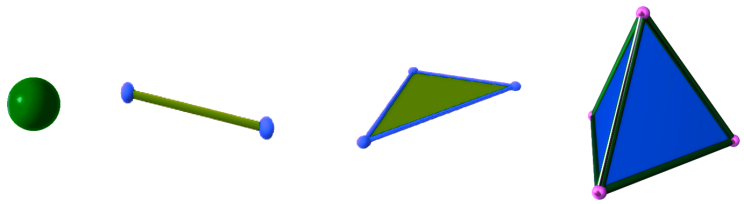


Figure 2.10. From left to right show a vertex, an edge, a triangle and a tetrahedron.

An important contribution to achieve an hexagonal packing in some coverage algorithm is undoubtedly yielded by the Vietoris-Rips complex, i.e. abstract simplicial complexes that take into account some metric allowing a maximum distance among its elements (see Subsec. B.1.2 for more formal definitions and technicalities). A Vietoris-Rips complex can be constructed from the relative abstract simplicial complex by using a paradigm called "Two-phase construction" Zomorodian (2010). The description of this method, illustrated only graphically in Fig. 2.11, lies outside the purpose of this introduction. However, to provide a useful preliminary for the following chapters, it is worth to underline the relation between a Vietoris-Rips complex and its corresponding 1-*skeleton* by means of the so-called *neighborhood graph*.

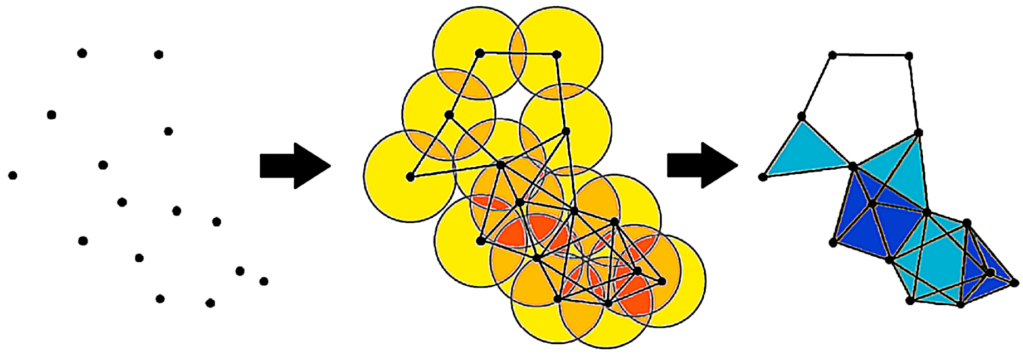


Figure 2.11. Construction of the Vietoris-Rips complex in a 3D space. In the first phase, a set of input points (on the left) is processed to obtain the neighborhood graph (in the middle). In the second phase, the 1-skeleton represented by the neighborhood graph is dimensionally expanded until it reaches the desired Vietoris-Rips complex (on the right).

The neighborhood graph $(\mathcal{G}, f_w)$ is characterized by the undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a weighting function $f_w : \mathcal{E} \longrightarrow \mathbb{R}$ defined on its edges. The 1-skeleton of the VR complex exactly coincides with its neighborhood graph. In particular, given an abstract simplicial complex $\mathcal{AS}^{(\kappa)}$ with n points $\mathcal{V}(\mathcal{AS}^{(\kappa)})$ belonging to $\mathcal{X} = \mathbb{R}^\kappa$ and the maximum diameter $\ell_\mathcal{X} > 0$, the 1-skeleton of the Vietoris-Rips complex on $\mathcal{AS}^{(\kappa)}$ is a neighborhood graph

$(\mathcal{G}_{\ell_\mathcal{X}}(\mathcal{AS}^{(\kappa)}), f_w)$, where $\mathcal{G}_{\ell_\mathcal{X}}(\mathcal{S}^{(\kappa)}) = (\mathcal{V}(\mathcal{AS}^{(\kappa)}), \mathcal{E}_{\ell_\mathcal{X}}(\mathcal{AS}^{(\kappa)}))$ is such that

$$\mathcal{E}_{\ell_\mathcal{X}}(\mathcal{AS}^{(\kappa)}) = \left\{ \{v_i, v_j\} \,|\mathrm{dist}(v_i, v_j) \leq \ell_\mathcal{X}, v_i \neq v_j \in \mathcal{V}(\mathcal{AS}^{(\kappa)}) \right\}; \tag{2.76}$$

$$f_w(\{v_i, v_j\}) = \mathrm{dist}(v_i, v_j), \quad \forall e_{ij} \in \mathcal{E}_{\ell_\mathcal{X}}(\mathcal{AS}^{(\kappa)}). \tag{2.77}$$

Lastly, for the second phase of the construction, Lem. B.3.46 on the path length given by the powers of the adjacency matrix can be used to pinpoint basic simplicial structures.

2.2.3 Graph-based motion planning and clustering

A topological map is a graph-based representation of the environment. Each node corresponds to a characteristic feature or zone of the environment, and can be associated with an action, such as turning, crossing a door, stopping, or going straight ahead. Usually, there are no absolute distances, nor references to any coordinate frame to measure space. This kind of maps are suitable for long distance qualitative navigation, and specially for path planning. In general, they do not explicitly represent free space so that obstacles must be detected and avoided on line by other means. Topological maps are simple and compact, take up less computer memory, and consequently speed up computational navigation processes Bonin-Font, Ortiz, and Oliver (2008). In addition, the selection of environmental features also plays a fundamental role since it helps in understanding data and reducing computation burden, improving the prediction performances Chandrashekar and Sahin (2014).

In the following lines, few concepts and tools to perform robotic navigation are introduced. Specifically, this paragraph gives a brief introduction on greedy algorithms for navigation Kamon, Rivlin, and Rimon (1996) and edge expansion techniques Chung (1997), providing also a nice insight on spectral clustering algorithms Peng, Sun, and Zanetti (2014).

Greedy algorithms for navigation

A greedy algorithm is a simple, intuitive algorithm that is used in optimization problems. In these kinds of algorithms the optimal choice is made at each step as the greedy procedure attempts to find the overall optimal way to solve the entire problem. Greedy algorithms take all of the data in a particular problem, and then set a rule for which elements to add to the solution at each step of the algorithm. If both of the properties below are true, a greedy algorithm can be used to solve the problem.

- Greedy choice property: a global (overall) optimal solution can be reached by choosing the optimal choice at each step.
- Optimal substructure: a problem has an optimal substructure if an optimal solution to the entire problem contains the optimal solutions to the sub-problems.

In other words, greedy algorithms work on problems for which it is true that, at every step, there is a choice that is optimal for the problem up to that step, and after the last step, the algorithm produces the optimal solution of the complete problem.

To make a greedy algorithm, it is required to identify an optimal substructure or sub-problem in the problem. Then, determine what the solution will include (for example, the largest sum). Finally, create some sort of iterative way to go through all of the sub-problems and build a solution.

In many circumstances, greedy algorithms can be useful to robotic navigation problems. To cite and give an insight of the most famous case, namely the search of a shortest path in a graph, the following example is provided.

Example 2.2.5 (Shortest path in a graph)**.**
Thinking of a mobile agent that is navigating in a Euclidean space, a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ may represent an appropriate tool to model the possible discretized locations and routes in which the robot can move. In order to navigate across two points in the space, i.e. two vertices v_i and v_j in $\mathcal{V}$, the agent has to walk along a path π_{ij} and hence, generally, one is interested at finding a shortest path $\pi_{ij}^\star$ between v_i and v_j, provided that each edge $e_k \in \mathcal{E}$ is endowed with a positive weight $|e_k|$, representing a distance. Therefore, indicating with $\ell_W(\pi_{ij})$ the total length of a path π_{ij} weighted once by $|e_k|$, for each e_k crossed, the following optimization problem can be set: compute $\pi_{ij}^\star$ such that

$$\pi_{ij}^\star = \min \ell_W(\pi_{ij}) = \min \sum_{e_k \in \pi_{ij}} |e_k|. \tag{2.78}$$

Problem (2.78) can be solved exploiting the well known Dijkstra's algorithm. An explanation of this procedure can be found in Goodrich and Tamassia (2011) but it is not proposed in the following lines, since it falls outside the thesis purposes. However, the key idea of *edge relaxation* in Dijkstra's algorithm is highlighted here to underline how a greedy paradigm acts. Assume that v_i and v_j are the source and destinations nodes respectively. Let also $\text{dist}(v_{k_0}, v_{k_1}) = \ell_W(\pi_{k_0 k_1})$ be the weighted distance between two adjacent vertices $v_{k_0} \in \mathcal{V}$ and $v_{k_1} \in \mathcal{V}$ and $\mathsf{D}_{ik}^{(t)}$ a variable storing the length of the best path from the source v_i to $v_k \in \mathcal{V}$ found until iteration t. The edge relaxation can be summarized as

$$\begin{aligned} &\textbf{if } \mathsf{D}_{ik_0}^{(t)} + \text{dist}(v_{k_0}, v_{k_1}) < \mathsf{D}_{ik_1}^{(t)} \textbf{ then} \\ &\quad \mathsf{D}_{ik_1}^{(t+1)} \leftarrow \mathsf{D}_{ik_0}^{(t)} + \text{dist}(v_{k_0}, v_{k_1}) \end{aligned}$$

It is worth to observe that the edge relaxation takes place only locally (among two adjacent vertices v_{k_0} and v_{k_1}), representing the core of the greedy procedure. Nevertheless, it can be proven that this leads to achieve an optimal solution for problem (2.78), meaning that a greedy approach is globally optimal in this case.

Edge expansion techniques for partitioning

Partitioning a graph into two or more pieces is one of the most fundamental problems in combinatorial optimization, and has comprehensive applications in various disciplines of computer science. One of the most studied graph partitioning problems is the edge expansion problem, i.e. finding a cut with few crossing edges normalized by the size of the smaller side of the cut.

Given a vertex-weighted graph $\mathcal{G}^v = (\mathcal{V}, \mathcal{E}, \mathcal{W}_v)$ and its subgraph $\mathcal{G}_S = (\mathcal{V}_S, \mathcal{E}_S)$,

consider the following isoperimetric quantity:

$$h_S(\mathcal{G}_S) = \frac{|\partial \mathcal{E}_S|}{\mathrm{vol}_v(\mathcal{G}_S)} \tag{2.79}$$

where $\partial \mathcal{E}_S$ and $\mathrm{vol}_v(\cdot)$ are the cut of $\mathcal{G}$ induced by $\mathcal{G}_S$ and the vertex-weighted volume respectively (see Subsec. B.3.1 for formal definitions and details). Intuitively, for a "good clustering" according to the balanced cut criterion one should look for a bipartition $\{\mathcal{G}_S, \mathcal{G}_{\overline{S}}\}$ of $\mathcal{G}^v$ such that

(i) $h_S(\mathcal{G}_S)$ and $h_{\overline{S}}(\mathcal{G}_{\overline{S}})$ are both small

(ii) $h_{\mathcal{G}}(\mathcal{G}_S) = h_S(\mathcal{G}_S) + h_{\overline{S}}(\mathcal{G}_{\overline{S}})$ is small

It is worth to note that: (i) can be formulated as the following optimization problem

$$\text{find } \mathcal{G}_S \text{ such that } \mathfrak{h}^{\partial}_{\mathcal{G}} = \min_{\mathcal{V}_S \subseteq \mathcal{V}} \max \ \{h_S(\mathcal{G}_S), h_{\overline{S}}(\mathcal{G}_{\overline{S}})\} = \min_{\mathcal{V}_S \subseteq \mathcal{V}} h^{Con}_{\mathcal{G}}(\mathcal{G}_S) \tag{2.80}$$

whereas (ii) can be formulated as the following one

$$\text{find } \mathcal{G}_S \text{ such that } \mathfrak{h}_{\mathcal{G}} = \min_{\mathcal{V}_S \subseteq \mathcal{V}} h_{\mathcal{G}}(\mathcal{G}_S). \tag{2.81}$$

These minimization problems are known to be NP-hard Matula and Shahrokhi (1990). However, one can show that it is possible to partition a graph in κ components exploiting the eigenvectors corresponding to the first κ eigenvalues of the Laplacian matrix Lee, Gharan, and Trevisan (2014). This result allows to obtain partitions that are almost optimal w.r.t. the balanced cut criterion. Alg. 1 provides a general procedure in order to accomplish an (almost) optimal partition of an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ into two subgraphs $\mathcal{G}_S = (\mathcal{V}_S, \mathcal{E}_S)$, $\mathcal{G}_{\overline{S}} = (\mathcal{V}_{\overline{S}}, \mathcal{E}_{\overline{S}})$.

Algorithm 1 Spectral method to partition a graph into two clusters
Input: Connected graph $\mathcal{G}$ and associated Laplacian $\mathbf{L}$
Output: Subgraphs $\{\mathcal{G}_S, \mathcal{G}_{\overline{S}}\}$

1: Choose the vertex weighting measure $\mathcal{W}_v$ for $\mathcal{G}$
2: Compute $\lambda^{\mathbf{L}}_1$ (the smallest non-zero eigenvalue of $\mathbf{L}$) and the associated eigenvector $\boldsymbol{\omega}$
3: Define $\mathcal{V}_S = \{v_i \in \mathcal{V} | [\boldsymbol{\omega}]_i \geq 0\}$
4: Partition the the vertex set $\mathcal{V}$ according to $\mathcal{V}_S$ and $\mathcal{V} \setminus \mathcal{V}_S$
5: Check the performance of the partition using Cheeger's inequalities (B.26)

Example 2.2.6 (Partition of a graph into two subgraphs)**.**
Consider the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ in Fig. 2.12. The eigenvectors of the Laplacian of $\mathcal{G}$ associated to $\lambda^{\mathbf{L}}_1 \simeq 0.3983$ are proportional to

$$\begin{bmatrix} 0.4929 & 0.2966 & 0.4929 & -0.3560 & -0.3560 & -0.3560 & -0.2142 \end{bmatrix}^{\top}. \tag{2.82}$$

Alg. 1 can be also implemented by means of the normalized Laplacian $\boldsymbol{\mathcal{L}} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$ instead of the Laplacian and checking inequalities (B.27) at the end of it. The eigenvectors

of the normalized Laplacian of $\mathcal{G}$ associated to $\lambda_1^{\mathcal{L}} \simeq 0.1597$ are proportional to

$$\begin{bmatrix} 0.4883 & 0.4071 & 0.4883 & -0.3260 & -0.3260 & -0.3260 & -0.1961 \end{bmatrix}^\top. \tag{2.83}$$

Both the expressions (2.82)-(2.83) exhibit component signs suggesting that the best partition for $\mathcal{V}$ is yielded by $\mathcal{V}_S = \{1, 2, 3\}$ and $\mathcal{V}_{\overline{S}} = \{4, 5, 6, 7\}$, as shown in Fig. 2.12. Approximating[12] $\mathfrak{h}_{\mathcal{G}}^{Ch} = 1/3$, $\mathfrak{h}_{\mathcal{G}}^{Con} = 1/6$ and $\mathfrak{h}_{\mathcal{G}} = 1/4$, one can be easily see that Cheeger's inequalities (B.26)-(B.27) are satisfied and, therefore, the chosen partition is a good candidate to be optimal according to the balance cut criterion.

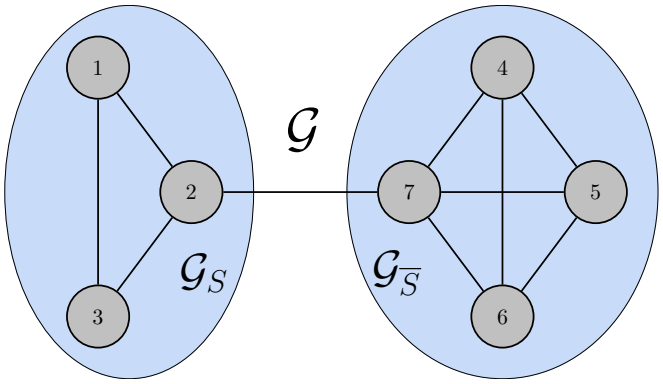


Figure 2.12. Graph partitioning of $\mathcal{G}$ into $\mathcal{G}_S$ and $\mathcal{G}_{\overline{S}}$, according to Alg. 1.

2.2.4 Iterative methods for Optimization

The search algorithms can be broadly classified into two types: (1) direct search algorithm and (2) indirect search algorithm. A direct search algorithm for numerical search optimization depends on the objective function only through ranking a countable set of function values. It does not involve the partial derivatives of the function and hence it is also called nongradient or zeroth order method. Whereas, indirect search algorithm, also called the descent method, depends on the first (first-order methods) and often second derivatives (second-order methods) of the objective function. In this paragraph, few basic indirect search algorithms, as the class of iterative descent methods for Optimization Kelley (1999), are introduced. Moreover, addressing the methods that can be exploited to solve distributed Convex Optimization problems, the Proximal Point algorithm Parikh and Boyd (2014) applied to separable cost functions, is discussed.

Classic descent algorithms

An iterative method is a mathematical procedure that uses an initial guess to generate a sequence of improving approximate solutions for a class of problems, in which the approximation at iteration t is derived from the previous ones. A specific implementation of an iterative method, including the termination criteria, is an algorithm of the iterative method. An iterative method is called convergent if the corresponding sequence converges for given initial approximations.

[12] Actually, this values are not only obtained by partitioning $\mathcal{G}$ as in Fig. 2.12, but also represent the effective quantities, in this example.

Iterative descent methods for Optimization are founded on first and second order necessary and sufficient conditions to minimize a given function $h : \mathbb{R}^N \to \mathbb{R} \cup \{+\infty\}$. The latter conditions are provided in the following theorems.

Theorem 2.2.7 (First and second order necessary conditions for optimality)**.** *Let h be twice continuously differentiable and let $\mathbf{x}^\star$ be a local minimizer of h. Then*

$$\nabla_{\mathbf{x}} h(\mathbf{x}^\star) = \mathbf{0}_N \tag{2.84}$$

Moreover, $\mathcal{H}_{\mathbf{xx}} h(\mathbf{x}^\star)$ is positive semidefinite.

Theorem 2.2.8 (First and second order sufficient conditions for optimality)**.** *Let h be twice continuously differentiable in a neighborhood of $\mathbf{x}^\star$. Assume that $\nabla h(\mathbf{x}^\star) = \mathbf{0}_N$ and that $\mathcal{H}_{\mathbf{xx}} h(\mathbf{x}^\star)$ is positive definite. Then $\mathbf{x}^\star$ is a local minimizer of h.*

In other words, Theorems 2.2.7 and 2.2.8 yield the key ideas to solve the unconstrained version of problem (2.1), i.e.

$$\min h(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^N. \tag{2.85}$$

Indeed, descent algorithms exploit the information on the derivatives of the cost h in order to converge, with an iterative procedure, to the solution. It is also important to observe that

1. a minimizer does not exist if h is unbounded from below;
2. the convergence towards a unique minimizer can be ensured if and only if h is a strictly convex function, since these kinds of functions are characterized by a positive definite Hessian matrix $\mathcal{H}_{\mathbf{xx}} h(\mathbf{x})$ for all $\mathbf{x} \in \mathbb{R}^N$.

In general, a descent algorithm can be described by the following *update equation*

$$\mathbf{x}(t+1) = \mathbf{x}(t) + \gamma_t \Delta\mathbf{x}(t), \quad t \in \mathbb{N} \tag{2.86}$$

where $\gamma_t > 0$, for all $t \in \mathbb{N}$, is the so-called *step size*, and $\Delta\mathbf{x}(t)$ is the *search direction*. The idea is to choose the search direction and the step size such that they satisfy the *descent condition*

$$h(\mathbf{x}(t+1)) < h(\mathbf{x}(t)), \tag{2.87}$$

until the optimum is reached and the algorithm terminates. The choice of step size at each iteration is called *line search*. Assuming h to be convex, condition (2.87) is ensured by a direction $\Delta\mathbf{x}(t)$ such that

$$\nabla_{\mathbf{x}} h(\mathbf{x}(t))^\top \Delta\mathbf{x}(t) < 0, \tag{2.88}$$

with such directions named *descent directions*.

Now, the problem of choosing a good step size is addressed. A possible choice for γ_t could be the value that minimizes h along the direction $\mathbf{x}(t) + \gamma_t \Delta\mathbf{x}(t)$, that is

$$\gamma_t = \arg\min_{\gamma \geq 0} h(\mathbf{x}(t) + \gamma \Delta\mathbf{x}(t)). \tag{2.89}$$

This method is called *exact line search*, but in practice it may be too computationally expensive. Therefore, one is often interested in methods that yield an approximate minimizer of h. The *backtracking line search* is one such method, since it chooses γ_t to satisfy the so-called Armijo's condition

$$h(\mathbf{x}(t) + \gamma_t \Delta\mathbf{x}(t)) \leq h(\mathbf{x}(t)) + \eta\gamma_t \nabla_{\mathbf{x}} h(\mathbf{x}(t))^\top \Delta\mathbf{x}(t) \tag{2.90}$$

where $\eta \in (0, 1)$, typically $\eta = 0.4$. Alg. 2 illustrates the method.

Algorithm 2 Backtracking line search
Input: h, $\nabla_{\mathbf{x}} h$, $\mathbf{x}(t)$, $\Delta\mathbf{x}(t)$, $\eta \in (0, 1)$
Output: γ_t

1: $\gamma_t \leftarrow 1$
2: **while** γ_t does not satisfy (2.90) **do**
3: $\quad \gamma_t \leftarrow \beta\gamma_t$, with $\beta \in (0, 1)$, typically $\beta = 0.7$
4: **end while**

Lastly, the few choices for the descent direction are briefly discussed. A very simple option for the descent is the negative gradient

$$\Delta\mathbf{x}(t) = -\nabla_{\mathbf{x}} h(\mathbf{x}(t)) \tag{2.91}$$

which gives the name to the *gradient method* or *steepest descent method*. It is worth to notice that this choice is indeed a descent direction since

$$\nabla_{\mathbf{x}} h(\mathbf{x}(t))^\top \Delta\mathbf{x}(t) = -\left\|\nabla_{\mathbf{x}} h(\mathbf{x}(t))\right\|_2^2 < 0. \tag{2.92}$$

Whereas, assuming that the Hessian $\mathcal{H}_{\mathbf{xx}} h$ of h is positive definite, another possible descent direction is given by

$$\Delta\mathbf{x}(t) = -\left(\mathcal{H}_{\mathbf{xx}} h(\mathbf{x}(t))\right)^{-1} \nabla_{\mathbf{x}} h(\mathbf{x}(t)), \tag{2.93}$$

which characterizes the so-called *Newton's method*. Clearly, this choice guarantees descent, indeed by the positive definiteness of $\mathcal{H}_{\mathbf{xx}} h$ it holds that

$$\nabla_{\mathbf{x}} h(\mathbf{x}(t))^\top \Delta\mathbf{x}(t) = -\nabla_{\mathbf{x}} h(\mathbf{x}(t))^\top \left(\mathcal{H}_{\mathbf{xx}} h(\mathbf{x}(t))\right)^{-1} \nabla_{\mathbf{x}} h(\mathbf{x}(t)) < 0. \tag{2.94}$$

Also, it is worth to mention other choices for the descent direction, as the *general descent method* in which

$$\Delta\mathbf{x}(t) = -\boldsymbol{\Omega}(\mathbf{x}(t), t) \nabla_{\mathbf{x}} h(\mathbf{x}(t)), \tag{2.95}$$

where $\boldsymbol{\Omega}(\mathbf{x}(t), t) \in \mathbb{R}^{N \times N}$ is a generic positive definite matrix that has to be assigned wisely in order to ensure convergence. For instance, this allows to implement a hybrid procedure that switches from the gradient method to the Newton's setting $\boldsymbol{\Omega}(\mathbf{x}(t), t) = \mathbf{I}_N$ or $\boldsymbol{\Omega}(\mathbf{x}(t), t) = \left(\mathcal{H}_{\mathbf{xx}} h(\mathbf{x}(t))\right)^{-1}$ for different t.

To conclude, an application to robotic coverage of descent algorithms is proposed in the following example.

Example 2.2.9 (Coverage algorithm based on gradient method)**.**
The coverage problem in (2.9) can be solved by a simple algorithm based on the gradient

method Mesbahi and Egerstedt (2010). The idea is to use the latter descent approach for moving agents, that is, to let

$$\dot{\mathbf{p}}_i(\tau) = -\frac{\partial h_{\hat{\Omega}}(\mathbf{p}_1, \ldots, \mathbf{p}_n)}{\partial \mathbf{p}_i} = -2 \int_{\hat{\Omega}_i} (\mathbf{p}_i(\tau) - \boldsymbol{\omega}) d\boldsymbol{\omega}. \tag{2.96}$$

This can be further improved upon allowing time-varying weights in the gradient descent algorithm, as for the general descent paradigm. In particular, by setting

$$\dot{\mathbf{p}}_i(\tau) = -\frac{1}{2\int_{\hat{\Omega}_i} d\boldsymbol{\omega}} \frac{\partial h_{\hat{\Omega}}(\mathbf{p}_1, \ldots, \mathbf{p}_n)}{\partial \mathbf{p}_i} \tag{2.97}$$

it is possible to obtain

$$\dot{\mathbf{p}}_i = f_{c_i}(\mathbf{p}_i(\tau)) - \mathbf{p}_i(\tau) \tag{2.98}$$

where $f_{c_i}(\mathbf{p}_i(\tau))$ is the center of mass of the Voronoi cell i at time τ.

Iterative approaches for Convex Optimization

In this final paragraph, an approach to tackle Convex Optimization is introduced. In particular, a class of algorithms, called proximal algorithms, for solving convex optimization problems is discussed.
Much like Newton's method is a standard tool for solving unconstrained smooth minimization problems of modest size, proximal algorithms can be viewed as an analogous tool for nonsmooth, constrained, large-scale, or distributed versions of these problems. They are very generally applicable, but they turn out to be especially well-suited to problems of recent and widespread interest involving large or high-dimensional datasets. Proximal methods sit at a higher level of abstraction than classical optimization algorithms like Newton's method. In the latter, the base operations are low-level, consisting of linear algebra operations and the computation of gradients and Hessians. In proximal algorithms, the base operation is evaluating the proximal operator of a function, which involves solving a small Convex Optimization problem. These sub-problems can be solved with standard methods, but they often admit closed form solutions or can be solved very quickly with simple specialized methods.

Let $h : \mathbb{R}^N \to \mathbb{R} \cup \{+\infty\}$ be a closed proper convex function, which means that its epigraph

$$\mathbf{epi}\ h = \{(\mathbf{x}, h_0) \in \mathbb{R}^N \times \mathbb{R} | h(\mathbf{x}) \leq h_0\} \tag{2.99}$$

is a nonempty closed convex set. The effective domain of h is

$$\mathbf{dom}\ h = \{\mathbf{x} \in \mathbb{R}^N | h(\mathbf{x}) < \infty\} \tag{2.100}$$

i.e., the set of points for which h takes on finite values. The proximal operator $\mathbf{prox}_h : \mathbb{R}^N \to \mathbb{R}^N$ of h is defined by

$$\mathbf{prox}_h(\boldsymbol{\omega}) = \arg\min_{\mathbf{x}} \left(h(\mathbf{x}) + \frac{1}{2} \|\mathbf{x} - \boldsymbol{\omega}\|_2^2 \right). \tag{2.101}$$

The function minimized on the r.h.s. of (2.101) is strongly convex and not everywhere infinite, so it has a unique minimizer for every $\boldsymbol{\omega} \in \mathbb{R}^N$ (even when $\mathbf{dom}\ h \subset \mathbb{R}^N$). Often

the proximal operator of the scaled function ρh, where $\rho > 0$, is encountered. This can be expressed as

$$\mathbf{prox}_{\rho h}(\boldsymbol{\omega}) = \arg\min_{\mathbf{x}} \left(h(\mathbf{x}) + \frac{\rho}{2} \|\mathbf{x} - \boldsymbol{\omega}\|_2^2 \right). \tag{2.102}$$

This is also called the proximal operator of h with parameter ρ, which plays the role of a regularizer. Also, it is worth to note that the point $\mathbf{x}^\star$ minimizes h if and only if

$$\mathbf{x}^\star = \mathbf{prox}_{\rho h}(\mathbf{x}^\star), \tag{2.103}$$

i.e. if $\mathbf{x}^\star$ is a fixed point of $\mathbf{prox}_h$ for all the values of $\rho > 0$. This fundamental property gives a link between proximal operator and Fixed Point Theory; e.g. many proximal algorithms for optimization can be interpreted as methods for finding fixed points of appropriate operators. This viewpoint is often useful in the analysis of these methods.

The proximal point algorithm (PP) is widely employed to deal with problems in which their structure is *separable*, in order to compute a distributed solution. To this purpose, two preliminary definitions are given in the following lines.

Definition 2.2.10 (Partition of a finite set)**.**
Let $[N] = \{1, \dots, N\}$. Given $\mathtt{c} \subseteq [N]$ let $\mathbf{x}_\mathtt{c} \in \mathbb{R}^\mathtt{c}$ denote the subvector of $\mathbf{x} \in \mathbb{R}^N$ referenced by the indices in $\mathtt{c}$. The collection $\mathtt{P} = \{\mathtt{c}, \dots, \mathtt{c}_N\}$, where $\mathtt{c}_i \subseteq [N]$, is a *partition* of $[N]$ if $\bigcup \mathtt{P} = [N]$ and $\mathtt{c}_i \cap \mathtt{c}_j = \varnothing$ for $i \neq j$.

Definition 2.2.11 (Separable function and scope)**.**
A function $h : \mathbb{R}^N \to \mathbb{R}$ is said to be $\mathtt{P}$-*separable* if

$$h(\mathbf{x}) = \sum_{i=1}^{N} h_i(\mathbf{x}_{\mathtt{c}_i}), \tag{2.104}$$

where $h_i : \mathbb{R}^{|\mathtt{c}_i|} \to \mathbb{R}$ and $\mathbf{x}_{\mathtt{c}_i}$ is the subvector of $\mathbf{x}$ with indices in $\mathtt{c}_i$. The set $\mathtt{c}_i$ is said *scope* of h_i. In other words, h is a sum of terms h_i, each of which depends only on part of $\mathbf{x}$; if each $\mathtt{c}_i = \{i\}$, then h is fully separable.

Separability is of interest because if h is $\mathtt{P}$-separable, then $[(\mathbf{prox}_h(\boldsymbol{\omega}))]_i = \mathbf{prox}_{h_i}(\boldsymbol{\omega}_i)$, where $\boldsymbol{\omega}_i \in \mathbb{R}^{|\mathtt{c}_i|}$, i.e., the proximal operator breaks into N smaller operations that can be carried out independently in parallel. Thus , evaluating the proximal operator of a separable function reduces to evaluating the proximal operators for each of the separable parts, which can be done independently.
The main problem belonging to this thesis in which the proximal operator will be used can be stated as follows:

$$\arg\min_{\mathbf{x} \in \mathbb{R}^N} h(\mathbf{x}), \qquad \text{with } h \text{ convex and separable} \tag{2.105}$$

Therefore, the PP solving (2.105) can be formulated as

$$\mathbf{x}(t+1) = \mathbf{prox}_{\rho h}(\mathbf{x}(t)) \tag{2.106}$$

where t is the iteration counter, and $\mathbf{x}(t)$ denotes the t-th iteration of the algorithm. If h has a minimum, then it is proven that $\mathbf{x}(t)$ converges to the set of minimizers of h and $h(\mathbf{x}(t))$ converges to its optimal value Bauschke and Combettes (2011).

2.3 Contributions and common thread

The major comprehensive contribution that has been devised in the entire dissertation rests upon the combination and exploitation of distinct disciplines with the purpose to consolidate both the theoretical and practical frameworks in which mobile multi-agent systems play a crucial role. As depicted in Fig. 2.13, it is worth to observe that various subjects are involved in this comprehensive methodology for tackling what can be delineated as Networked Optimization for Multi-Agent Systems (NO4MAS), such as Optimization Theory (OT), Dynamic Systems and Control (DS&C), Combinatorial Graph Theory (CGT), Optimal Control (OC), Trajectory Optimization (TO), Combinatorial Optimization (CO), Distributed Optimization (DO), Distributed Control Systems (DCS) and Distributed Estimation (DE).

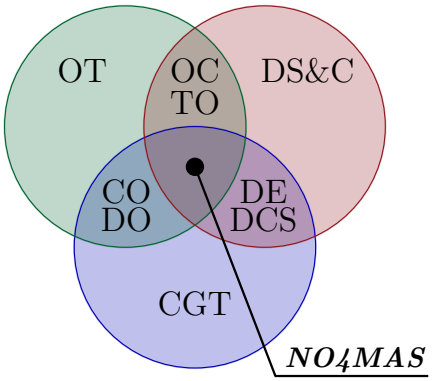


Figure 2.13. Numerous topics from several subjects sharing common interests are needed to find solutions to networked optimization problems for multi-agent systems.

The evidence that several abstract and tangible tools are required to concur in order to solve a networked optimization problem with a group of robots is, in fact, the fundamental aspect that broadens the scope of the existing methods already illustrated in this chapter. At the light of this panoramic sketch, the next paragraph is dedicated to the description of the research goals in this manuscript as well as the characterization of a more detailed outline for the forthcoming chapters.

2.3.1 Research goals

This work was performed at the research laboratories of the universities of Padova (Italy) and Colorado at Boulder (USA) for an overall period of three years. The most crucial aims that have been set from the beginning of the PhD activity are primarily represented by the following guidelines:

- the formalization of interesting engineering problems having practical consequences in the advancement of networked-based distributed architectures and multi-agent system research field;
- the development of analysis and design tools or mathematical methods for a group of cooperating mobile robots with assigned and specified common tasks;

- the creation, application and investigation of optimization-based strategies to achieve the most disparate required specifications by drawing inspiration from the existing methodologies and knowledge about agent technologies;
- the proof of theoretical statements settled in meaningful frameworks that are restricted, endowed with the least amount of hypotheses, to the problem of interest;
- the virtual implementation and numerical simulation of the devised techniques to assess numerical results of realistic case studies.

2.3.2 Detailed outline of the research activity

In this thesis, different types of (homogeneous) multi-agent systems are analyzed seeking efficient distributed solutions to face several typical and new issues within the common thread previously presented at the beginning of this section. For each of the next chapters containing research activity, namely Chapters 3, 4, 5, 6, a similar scheme as in Fig. 2.13 will be illustrated in order to highlight the most significant theoretical frameworks involved[13]. Such a scheme will be also associated to a box diagram showing a summarized version of the specific investigation objectives established for each particular activity. The single contributions given in the entire research complex are outlined as follows.

- Chap. 3. **Dynamic Coverage with Limited Sensing Capabilities**: a distributed algorithm is devised and tested through virtual simulations to accomplish robotic deployment and focus on event in an unknown scenario with no metric information available. In addition, further guarantees on the number of deployed robots covering the environment are provided.
- Chap. 4. **Optimal Time-Invariant Formation Tracking**: two approaches to govern a second order integrator system of multiple agents are derived, namely an inverse dynamics offline control and a distributed online controller. The first strategy leverages on a numerical tool for Trajectory Optimization, called Projection Operator based Newton's method, while the second one resorts on RT and CV. In this work, the main objective is to steer the system into a robotic formation that chases a desired trajectory while minimizing the energy spent.
- Chap. 5. **Distributed State Estimation from Relative Measurements**: a general regularized distributed linear model is analyzed and optimized to perform a state estimation from the noisy relative measurements provided by a network with a fixed topology and a bidirectional information exchange.
- Chap. 6. **On the Spectral Properties of κ-ring Graphs**: the theory of circulant matrices and graphs is the key for many distributed application, e.g. a camera network for surveillance purposes. This highly theoretical work explores a meaningful case study to ensure optimality conditions for activity 2.3.2, constituting a bridge between the spectrum of stochastic matrices induced by a peculiar class of circulant graphs – the κ-ring graphs – and the Dirichlet kernel.

[13] This will be done by emphasizing (in bold and italic) the acronyms in Fig. 2.13 corresponding to frameworks that are relevant for the study in question.

2.3.3 List of publications

The following list provides the full catalog of the author's publications. The items marked with a blue asterisk (*) are considered relevant for the Ph.D. activity and hence for this thesis. Other publications concerning these topics are envisaged.

*Fabris and Cenedese (2019). "*Distributed Strategies for Dynamic Coverage with Limited Sensing Capabilities*". In 27th Mediterranean Conference on Control and Automation 2019, Akko, Israel, pp. 203-208, Jul 1-4, 2019.

*Fabris, Cenedese, and Hauser (2019a). "*Optimal Time-Invariant Formation Tracking for a Second-Order Multi-Agent System*". In 18th European Control Conference 2019, Napoli, Italy, pp. 1556-1561, Jun 25-28, 2019.

*Fabris, Michieletto, and Cenedese (2019b). "*On the Distributed Estimation from Relative Measurements: a Graph-Based Convergence Analysis*". In 18th European Control Conference 2019, Napoli, Italy, pp. 1550-1555, Jun 25-28, 2019.

Varotto, Fabris, Michieletto, and Cenedese (2019). "*Distributed Dual Quaternion Based Localization of Visual Sensor Networks*". In 18th European Control Conference 2019, Napoli, Italy, pp. 1836-1841, Jun 25-28, 2019.

Frigo, Fabris, Galli, Gambarin, Marsili, Narduzzi, and Giorgi (2015). "*Efficient Tracking of Heart Rate under Physical Exercise from Photoplethysmographic Signals*". In IEEE 1st International Forum on Research and Technologies for Society and Industry, Torino, Italy, pp. 306-311, Sep 16-18, 2015.

3

Dynamic Coverage with Limited Sensing Capabilities

"Never send a human to do a machine's job."
Smith

Contents

The contents of this chapter are partly available in:
Fabris and Cenedese (2019). "*Distributed Strategies for Dynamic Coverage with Limited Sensing Capabilities*". In 27th Mediterranean Conference on Control and Automation 2019, Akko, Israel, pp. 203-208, Jul 1-4, 2019.

3.1 Overview

Sensor coverage of indoor environments using teams of mobile robots is a well-studied problem in robotics. This chapter is focused on unknown complex indoor scenarios that are typically hard to cover because there are no a priori information about the space

structure and there is not an optimal strategy to reach the complete coverage.
Generally, this task has to be performed using the lowest number of agents, the cheapest sensors and actuators to steer each mobile robot. In particular, the aim of this part of the thesis, is the development of an efficient algorithm to deploy and control a sensor network consisting of a collection of sensing devices that can coordinate their actions through wireless communication. Such a procedure has to guarantee the achievement of several tasks as event detection, reconnaissance, surveillance, target tracking, exploration of a specific region or rescue actions in environments that are hazardous to human operators. The advantages of this type of operations are the possibility of visiting and monitoring every point of the given space in an automatic and distributed fashion, the robustness to failures of single robots, the ability of the agents to adapt their arrangement and dynamics considering the obstacles in the environment, the reliability of a connected sensor network and the maximization of the sensing information utilizing the lowest amount of available local resources. In addition, it is worth to mention that swarms of robots employed for dynamic coverage are involved in many applications, such as active camera networks, mapping of unknown environments, transportation and delivery, fault detection, precision agriculture, cooperative robotics and wireless sensor networks.

To conclude this heading, Fig. 3.1 depicts the main features arising in this work. The principal methods employed resort on Optimization Theory, Combinatorial Graph Theory, Combinatorial Optimization and Distributed Optimization. Leveraging the latter tools, the main objectives for this study are represented by the design of a procedure for the automatic and dynamic deployment of robotic agents, event detection, clustering and dispatch. The developed algorithm is tested through virtual modeling and simulations of realistic scenarios.

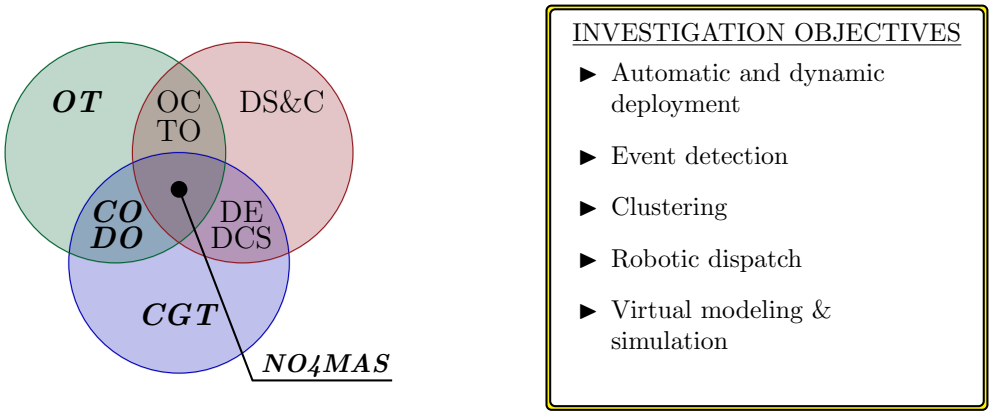


Figure 3.1. Theoretical fields and investigation objectives arising from this study.

3.1.1 Problem statement

In this chapter, the problem of an efficient indoor environment exploration by means of an expanding swarm of robots with limited and noisy local sensing is considered. By assumption, the indoor environment is unknown and two-dimensional and no global localization is available. Specifically, the only sensor capabilities assumed on each robot are those of an omni-directional camera with a limited radial range of vision and a touch

sensor to detect contact or collisions with obstacles and other robots. The disk around a robot representing a sensing radius of the omni-directional camera is called the robot's disk of visibility, within which the bearing to the neighboring robots and their identities can be detected. However, each camera does not provide a range measurement due to projection of the 3D world on to the camera plane. Thus, obstacles are detected through touch sensors near the base of the robots, and cannot be detected using the camera. The robots can also communicate with each other and can be driven in arbitrary direction. Three subsequent tasks are chosen for the swarm:

1. the maximization of local communication: algorithm in Ramaithitima et al. (2015) is used as starting point and reformulated in order to remove the assumption of a centralized server that collects local information from individual robots (see Fig. 3.2);

2. the possibility to perform event detection of the covered area, with specific sensor for it (see Fig. 3.3), e.g. sensing a increment of absolute temperature due to a fire;

3. the capacity of monitoring the event surface by sending a group of robots, as shown in the example in Fig. 3.4.

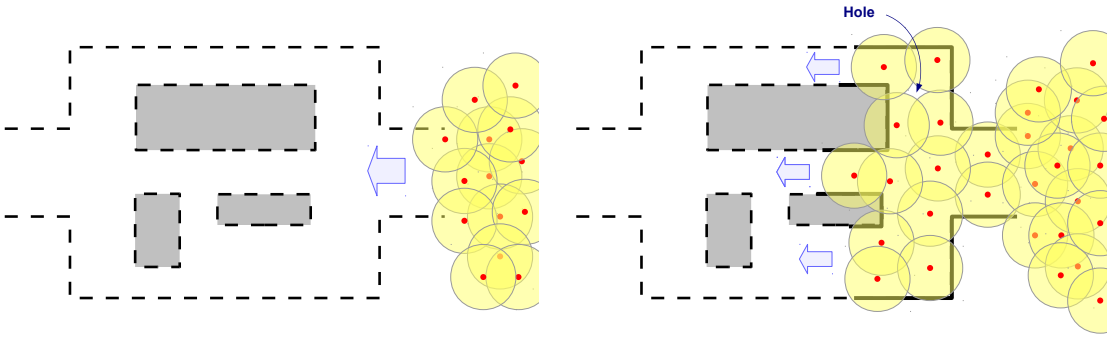


Figure 3.2. Illustration of a swarm of robots entering an environment and attaining coverage. [*Credits* : Ramaithitima et al. (2015)]

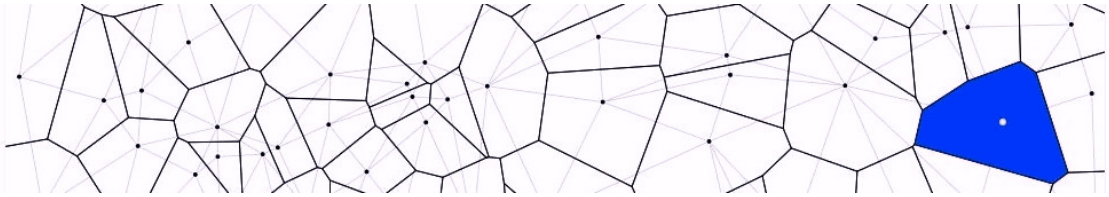


Figure 3.3. Voronoi partitions obtained after the deployments have divided the covered region. A partition, e.g. the blue one, is then monitored by a cluster of robots made by the agent in the centroid and its neighbors.

3.1.2 Related works

Classic coverage applications are based on centralized localization for each robot (e.g. GPS) in order to accomplish equitable partitions of the environment (as illustrated in Subsec. 2.1.3) by minimizing a coverage cost. However, such approaches need a priori

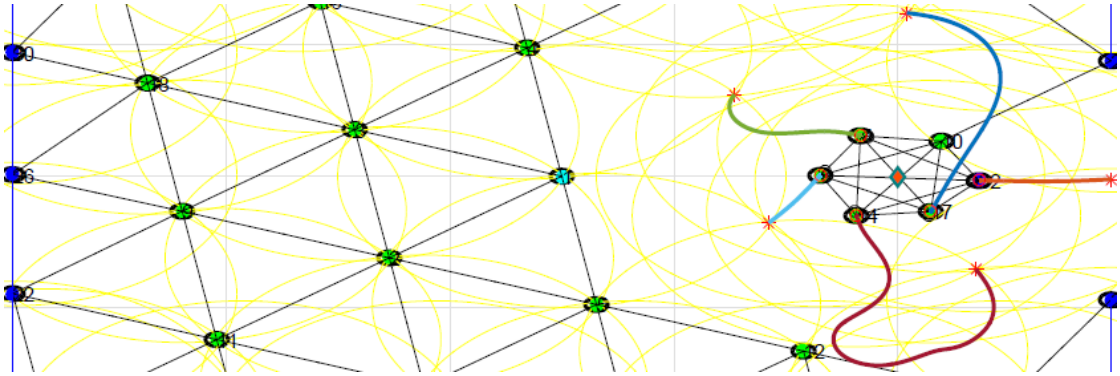


Figure 3.4. Six agents are driven from their initial position (red stars) towards a specific target while maintaining the connection with the rest of the network.

information about the environment where robots are deployed. Lately, complete sensor coverage of indoor environments using swarms of robots has been studied Dudek, Jenkin, Milios, and Wilkes (1991); Rutishauser, Correll, and Martinoli (2009) by modeling the known network as a graph and assuming robots to have global localization, with the possibility to navigate independently from one location to another in a global coordinate frame. In all of these lines of research, global localization of the robots, a priori knowledge of the environment (obstacle configuration), availability of metric information and ability to control the robots from one point in the environment to another have been presumed. In recent years, coverage by sensor network is studied more formally by using Simplicial Complex Theory Zomorodian (2010) and homological tools from algebraic topology, as in de Silva and Ghrist (2006, 2007); Ghrist, Lipsky, Derenick, and Speranzon (technical report, 2012), requiring little to no metric information. One of the latest research, from which this work draws inspiration, is Ramaithitima et al. (2015). Here, authors offer new perspectives improving the previous studies by using a different approach based on three aspects. Firstly, their method is robust to agent failure at any configuration. Secondly, the approach of pushing agents through the network links in order to expand the graph frontier, instead of navigating, does not require any restriction on the minimum distance between two neighbors. Lastly, their algorithm does not demand the workspace to be simply-connected, as demonstrated via experimental results.

3.1.3 Contribution and outline of the chapter

Starting from Ramaithitima et al. (2015), a more advanced and fully decentralized algorithm to fulfill coverage and focus on event is implemented, motivated by several works in the fields of event detection Wittenburg, Dziengel, Adler, Kasmi, Ziegert, and Schiller (2012), cluster selection Afsar and Tayarani-N (2014) and robotic dispatch Lukic and Stojmenovic (2013). A completely distributed approach, i.e. requiring local communication only, is presented: each agent needs to share with its neighbors just the information about measured bearing angles and neighborhoods. In particular, useless agents are sought by exploiting bearing angles, instead of computing graph cycles. Moreover, geometric models for the agents are adopted, e.g. circular shapes: this allow to take into account collisions and space occupation in order to provide a lower bound

for the number of agents deployed by the algorithm in question on virtual scenarios with rectangular features. Once coverage is completed, the focus on a preset event is then attained: in this final procedure, representing the new and original contribution of this chapter, each agent builds on a network deformation that needs to increase the event information while not losing connectivity and enhancing communication among activated companions. The remainder of this paper is organized as follows. In Sec. 3.2, some preliminary definitions and mathematical models are introduced, while Sec. 3.3 illustrates central ideas for the implementation of this algorithm. The discussion continues in Sec. 3.4 with the analysis of lower and upper bounds for the minimum number of deployed agents. In addition, numerical results reporting performances and limitations of this algorithm can be found in Sec. 3.5. Finally, conclusions are drawn in Sec. 3.6.

3.2 Assumptions and models

In this section, the fundamental operating hypotheses are presented. Geometrical assumptions are taken into consideration in order to define a proper environment for the agents; whereas, many mathematical models are adopted in order to describe scenario, event, agent and network features.

3.2.1 Geometrical assumptions

Hereafter, letters i, j, k and their variants are used as indexes; while letter L and its variants indicate finite lengths of vectors or number of elements in a sequence.
Let Ω be a set, such that $\mathrm{cl}(\Omega)$ be its closure and $\mathrm{int}(\Omega)$ be its interior. Whenever Ω is discrete and finite or empty its cardinality is denoted with $|\Omega| \in \mathbb{N}$, otherwise $|\Omega| = +\infty$. Let $\mathbf{p}_k = (x_k, y_k) \in \mathbb{R}^2$ be a point in the plain. Whenever Ω_k represents either a segment or a polygon, the surface of Ω_k is denoted with $\Omega_k^2 \subset \mathbb{R}^2$, setting $\Omega_k^2 = \Omega_k$ if and only if Ω_k is a segment, such that if $\Omega = \bigcup_{k=1}^{L} \Omega_k$ then $\Omega^2 = \bigcup_{k=1}^{L} \Omega_k^2$. Moreover, $\|\Omega_k\|_1 > 0$ and $\|\Omega_k\|_2 \geq 0$ indicate the perimeter of Ω_k and the area of Ω_k^2 respectively, setting $\|\Omega_k\|_2 = 0$ if and only if Ω_k is a segment. Lastly, if $\mathrm{int}(\Omega_i) \cap \mathrm{int}(\Omega_j) = \varnothing$ for all $i, j = 1, ..., L$, with $i \neq j$, then $\|\Omega\|_{\bar{k}} = \sum_{k=1}^{L} \|\Omega_k\|_{\bar{k}}$ for $\bar{k} = 1, 2$.
Further useful preliminaries are covered on geometrical and topological entities (see Subsec. B.1.1).

3.2.2 Mathematical models

Scenario

As Fig. 3.5 depicts, a scenario $SC = (EN, OB)$ is the planar space where actions take place, composed of an enclosure EN and, possibly, a set of obstacles $OB = \bigcup_{k=1}^{L_{ob}} ob_k$ with finite number L_{ob} of physical barriers ob_k. Whenever a scenario is obstacle-free, it is assumed that $SC = EN$. For the sake of simplicity, it is imposed that EN is a polygon and each obstacle ob_k is either a polygon or a segment. Moreover, to characterize SC as a realistic environment, the following properties on its potential obstacles are also

imposed:

- $\text{int}(ob_i) \cap \text{int}(ob_j) = \varnothing$ for all $i, j = 1, \ldots, L_{ob}$, $i \neq j$;
- $\text{cl}(OB^2) \subset \text{int}(EN^2)$;
- $\|OB\|_2 \leq k_{SC} \|EN\|_2$ for a given $k_{SC} \in (0, 1)$.

These three properties state that obstacles cannot overlap each other, they must be contained inside the enclosure, and their space occupation in the enclosure must be reasonably low with respect to the total available space. Finally, the surface to be covered and the scenario boundaries are defined as $CS^2 = \text{int}(EN^2) \setminus \text{cl}(OB^2)$ and $SB = EN \cup OB$, respectively.

Base station

A base station is the point $BS = \mathbf{p}_1 \in CS^2$ that generates all robots during coverage (see Fig. 3.5) and it also represents the position of the first agent a_1. Because of this choice, it is assumed that a_1 cannot be removed while seeking potential redundant agents in the network (see Alg. 4 in Subsec. 3.3.1).

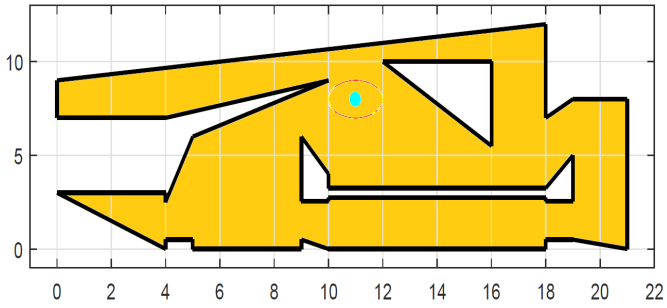


Figure 3.5. A scenario (yellow area delimited by the black contour) and a base station (dot in cyan).

Event

As shown in Fig. 3.6, an event is a point $EV \in EN^2$ that becomes significant inside the scenario after a complete coverage has been already attained and with a relevance that is radially decreasing. The latter is modeled by the real scalar function

$$f_{EV}(\mathbf{p}_p) = k_{EV} \exp\left(-\|\mathbf{p}_p - EV\|^2 / r_{EV}^2\right), \quad \forall \mathbf{p}_p \in EN^2 \tag{3.1}$$

where $k_{EV} > 0$ and $r_{EV} > 0$ represent the maximum intensity and the decay distance respectively.

Agents

Any agent $a_k = (v_k, \mathbf{p}_k, r_b, r_v)$ is represented by a vertex in a graph $\mathcal{G}$ and modeled by a circle with radius[1] $r_b > 0$ centered in $\mathbf{p}_k \in CS^2$. They are also provided with a

[1]Because of this fact, corridors in the scenario with a width less than the body diameter $2r_b$, can never be accessed by agents.

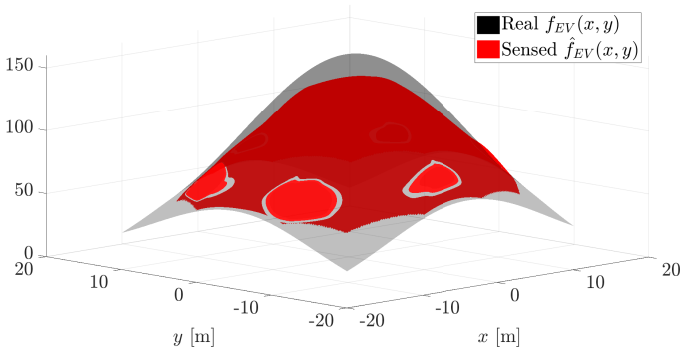
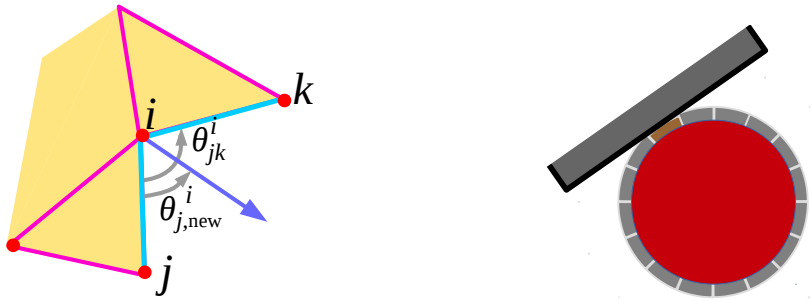


Figure 3.6. Intensity of an event located in the middle of a squared scenario, also adopted for simulation in Fig. 3.12. Black: real values of the event function. Red: cubic spline interpolation of the estimated values providing a reconstruction of the event function.



(a) Bearing-based sensing and control

(b) Touch sensing (gray)

Figure 3.7. Driving sensors and actuators mounted on each agent. [*Credits* : Ramaithitima et al. (2015)]

camera installed on the center with a visibility radius $r_v > 0$ in order to acquire bearing measurements (Ramaithitima et al. (2015), Sec. II-A). As depicted in Fig. 3.7(a), these are used to measure the angles agents create with one or more neighbors. A bearing measurement (in radians) of an agent a_k w.r.t. one of its neighbor a_i is denoted with $\theta^k_i \in [-\pi, \pi)$. If it measures the bearing to another agent a_j as θ^i_j, then the bearing to a_k relative to the bearing to a_j is defined as $\theta^k_{ij} = ((\theta^k_i - \theta^k_j) \mod 2\pi) - \pi$, such that $\theta^k_{ij} \in [-\pi, \pi)$. As illustrated in Fig. 3.8, two agents (a_i, a_j) are said neighbors if:

- $\|\mathbf{p}_i - \mathbf{p}_j\| \leq r_v$, i.e., their centers are not distant more than the visibility radius r_v;
- $(st_{ij} \cap CS^2) \subset CS^2$, i.e., there is no barrier between them;

Figure 3.8. Visibility radii determine whether two agents are neighbors (right) or not (left). Image taken from Ramaithitima et al. (2015).

It is assumed that a_i and a_j communicate even though between them another agent has already been placed. Furthermore, agents are provided with contact sensors (see Fig. 3.7(b)) that are triggered when touching any barrier in the scenario or another agent, able to roughly estimate the direction of an impact (Ramaithitima et al. (2015), Sec. II-C) within an error $\Delta e_\tau = \pi/N_T$, where N_T is the number of contact points in each touch sensor. Finally, each agent is provided by an event sensor in order to sense an estimate $\hat{f}_{EV}(\mathbf{p}_p)$ of the event intensity and it is controlled using the bearing-based visual homing controller (Ramaithitima et al. (2015), Sec. II-D). This means that the velocity $\dot{\mathbf{p}}_i$ of robot a_i is given by $\dot{\mathbf{p}}_i = k_p \sum_{v_j \in \mathcal{M}_i} (\theta^i_{j,des} - \theta^i_j)$, where $\mathcal{M}_i \subseteq \mathcal{N}_i$ is the list of robots that are neighbors of a_i and which can be used as landmarks, $\theta^i_{j,des}$ is the desired bearing with landmark a_j and k_p is a positive gain.

Multi-Agent network

The multi-agent network arising from the interconnections of the deployed agents can be modeled as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with all its structural properties and related concepts introduced in Sec. B.3. Each node $v_i \in \mathcal{V}$ represents a robot a_i and an edge $e_{ij} \in E$ between two robots (v_i, v_j) is created if only if (v_i, v_j) are neighbors.
In addition, the hexagonal-packing-based coverage policy discussed in Subsec. 2.2.2 is adopted, since it provides the best approach to save resources in an obstacle-free planar environments. Specifically, the deployment strategy employed in this work makes use of the following simplexes (see Subsec. B.1.2 for more details):

- type 0: points, which represent nodes (agents);
- type 1: segments, which represent edges (communication links);
- type 2: triangles, to guarantee hexagonal packing policy (optimality condition).

For this reason, tools as the Vietoris-Rips Complex are applied in order to achieve complete coverage.

3.3 Coverage and focus on event: algorithm design

A general overview of the main procedure is illustrated in Alg. 3 and described as follows.

- *Coverage*: Firstly, agents are deployed until full coverage is attained (line 1).
- *Clustering*: Subsequently, they sense an event EV in the scenario and a cluster $\mathcal{G}_{CL} \subseteq \mathcal{G}$ with preselected cardinality $n_{CL} = |\mathcal{G}_{CL}|$ is created (line 10). The cluster formation begins from a leader vertex $v^\star$ that measures the highest intensity: this node can be elected by means of max-consensus algorithms Oliva and Setola (2013) (line 8). Then, the cluster grows with a greedy approach, maintaining its connectivity.
- *Dispatch*: Finally, agents belonging to $\mathcal{G}_{CL}$ perform a dispatch (line 16) according to the minimization of an isoperimetric functional $h_{\mathcal{G}}(\mathcal{G}_{CL})$ with the purpose to drive the cluster's elements close to the event origin EV as far as possible, maintaining

the connections constituted right after the coverage. This minimization takes place iterating on cluster $\mathcal{G}_{CL}$, where a maximum number of iterations[2] $MaxIter$ is fixed and each session can disable the flag $\mathrm{f}_d^\star$, breaking the dispatch loop. Also, the leader node $v^\star$ (line 15) is chosen in an adaptive fashion, since, at each cycle, the event sensing for cluster nodes may vary while they move.

3.3.1 Coverage stage

As proposed in Ramaithitima et al. (2015), let us denote the Vietoris-Rips Complex (see B.1.2) of the set of the deployed agents[3] with $\mathcal{R}_{r_v}$, the frontier subcomplex with $\mathcal{F} \subseteq \mathcal{R}_{r_v}$, the obstacle subcomplex with $\mathcal{O} \subseteq \mathcal{R}_{r_v}$ and the fence subcomplex with $\mathcal{K} = \mathcal{F} \cup \mathcal{O}$. To accomplish this stage, the COVERAGE function is implemented at line 1 in Alg.3 adopting their hexagonal-packing-based coverage algorithm (see Fig. 3.9). However, since the relative homology $H_2(\mathcal{R}_{r_v}, \mathcal{K})$ (see B.1.3) is required to be stored in a centralized manner (Ramaithitima et al. (2015), Sec. II-B), that procedure is not completely distributed. To account for this aspect, an improvement to this scheme is introduced: conversely to the homology-based approach utilized in Ramaithitima et al. (2015), Sec. II-E, a redundant agent in the network is pinpointed exploiting reciprocal bearing measurement values, as shown in Fig. 3.10. This allows to rule out agents belonging to the interior of a 1-simplex (a segment) or 2-simplex (a triangle) taking advantage of simple angle properties[4]. In Alg. 4, these two different cases are examined considering each triplet of distinct agents $\boldsymbol{a}_k = (a_{k_1}, a_{k_2}, a_{k_3})$ that forms a 2-simplex and determines whether an agent is redundant to coverage purposes.

Algorithm 3 Outline of the main procedure

1: $\mathcal{G} \leftarrow$COVERAGE();
2: **for** each agent a_i, s.t. $i = 1, ..., n$ **do**
3: $\quad |v_i| \leftarrow \hat{f}_{EV}(\mathbf{p}_i)$;
4: **end for**
5: **for** all $e_{ij} \in \mathcal{E}$ **do**
6: $\quad |e_{ij}| \leftarrow (|v_i| + |v_j|)/2$;
7: **end for**
8: $v^\star \leftarrow$MAX-CONSENSUS($\mathcal{G}$,BS);
9: $\mathcal{G}_{CL} \leftarrow \{v^\star\}$
10: CLUSTERING($v^\star$,1);
11: **for** all nodes $v_i \in \mathcal{G}_{CL}$ **do**
12: $\quad [c_{di}, \mathrm{f}_{di}] \leftarrow [0, \textbf{false}]$;
13: **end for**
14: **while** $c_d^\star < MaxIter$ **and** $\mathrm{f}_d^\star = \textbf{false}$ **do**
15: $\quad v^\star \leftarrow$MAX-CONSENSUS($\mathcal{G}_{CL}, v^\star$);
16: $\quad [c_d^\star, \mathrm{f}_d^\star] \leftarrow$DISPATCH($v^\star$,$c_d^\star + 1$,**true**);
17: **end while**

[2] Iterations are counted by variables c_{di}, stored in each node $v_i \in \mathcal{V}_{CL}$.
[3] Abuse of notation: a simplicial complex depends on the algorithm status.
[4] The remarkable facts that the summation result of the three convex angles in any triangle is equivalent to a straight angle allows us to identify 1-simplex-redundant agents. Moreover, 2-simplex-redundant agents are spotted whenever three explementary angles with the same vertex exist.

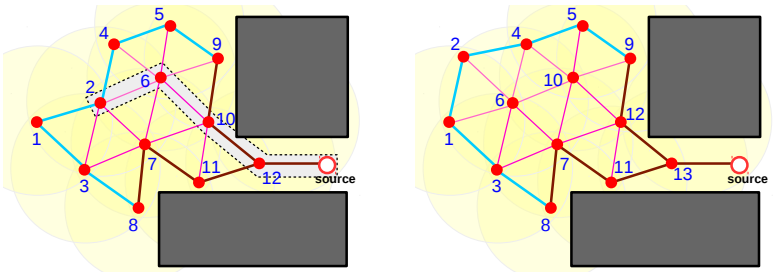


Figure 3.9. Coverage algorithm developed in Ramaithitima et al. (2015) leverages an innovative pushing technique and an expansion policy that allows to attain an hexagonal packing whenever border effect are not present. [*Credits* : Ramaithitima et al. (2015)]

Algorithm 4 Redundant agent search

for $k = 1, 2, ...$ s.t. $\boldsymbol{a}_k$ is a 2-simplex **do**
  $\boldsymbol{\theta}_k \leftarrow \begin{bmatrix} \theta^{k_1}_{k_2k_3} & \theta^{k_2}_{k_1k_3} & \theta^{k_3}_{k_1k_2} \end{bmatrix}$;
  if $\exists i \in \{1,2,3\}$ s.t. $|[\boldsymbol{\theta}_k]_i| = \pi$ **and** $k_i \neq 1$ **then**
    label a_{k_i} as 1-simplex-redundant;
  end if
  $\mathcal{N}_{k\cap} \leftarrow \mathcal{N}_{k_1} \cap \mathcal{N}_{k_2} \cap \mathcal{N}_{k_3}$;
  for $j = 1, ..., |\mathcal{N}_{k\cap}|$ s.t. $a_{k_j} \in \mathcal{N}_{k\cap}$ **do**
    if $\left|\theta^{k_j}_{k_1k_2} + \theta^{k_j}_{k_2k_3} + \theta^{k_j}_{k_3k_1}\right| = 2\pi$ **and** $k_j \neq 1$ **then**
      label a_{k_j} as 2-simplex-redundant;
    end if
  end for
end for

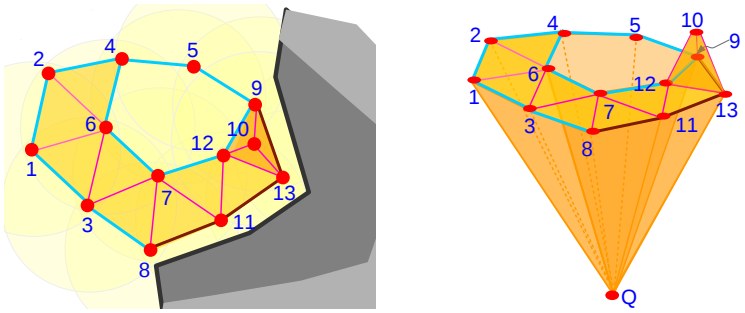


Figure 3.10. In Ramaithitima et al. (2015), redundant agents are identified through non-trivial 2-cycles belonging to the topological space H_2 where the union of the fence and obstacle nodes are collapsed into a a 0-simplex Q. [*Credits* : Ramaithitima et al. (2015)]

3.3.2 Clustering stage

In this intermediate stage, illustrated in Alg. 5, it is supposed that each node in the network has already sensed the event intensity, and hence each edge weight in the graph $\mathcal{G}$ has already been assigned by a specific function of the measurements (e.g. the function[5] at line 6 in Alg. 3): it follows that each neighborhood $\mathcal{N}_i$ in $\mathcal{G}$ can be sorted in a descending order according to each weight $|v_j|$, where $v_j = [\mathcal{N}_i]_k$, $k = 1, ..., |\mathcal{N}_i|$. The

[5]This weighting function has to be chosen according to condition in the r.h.s. at line 10 in Alg. 5

latter assumption contributes to implement a greedy approach: each agent v_i is labeled as a cluster node if it is chosen as the best option in the neighborhood $\mathcal{N}_i$ according to criteria that reflect local optimality. Alg. 5 should be actually seen as a distributed one-hop loop and not as a centralized recursive procedure, since it is not required to save the whole state of an agent in its stack while hopping. Keeping track of the last frame by including three counters and two additional flags – used to report the affiliations to cluster $\mathcal{G}_{CL}$ and the subgraph of completed vertices $\mathcal{G}_{CO} \subseteq \mathcal{G}$ – is enough to preserve information. Indeed, for every agent a_i, counters $\bar{k}_i$, k_i, $|\mathcal{G}_{CL}|$ are just counting increasing quantities and the two aforementioned flags, implementing lines 8 and 18, would be set only once. Moreover, when the CLUSTERING function is invoked to perform a communication hop from v_i to v_j in a real framework, only the information related to the cardinality $|\mathcal{G}_{CL}|$ has to be passed and updated.

Three steps can be identified in Alg. 5, which generally stops once $|\mathcal{G}_{CL}| = n_{CL}$ holds (line 13). Firstly, for $\bar{k}_i = 1$, all vertices in neighborhood $\mathcal{N}_i$ are inserted into cluster $\mathcal{G}_{CL}$ (line 8), as far as possible: if a new leader v_j is found (line 10) a hop is performed according to a greedy paradigm. Secondly, for $\bar{k}_i = 2$, nodes in $\mathcal{N}_i$ are forcefully visited in order to expand the cluster, in case condition $|\mathcal{G}_{CL}| = n_{CL}$ is yet to be attained. If the latter is not verified at the end of this loop then vertex v_i is labeled as complete (line 18); namely, all nodes in $\{v_i \cup \mathcal{N}_i\}$ already belong to cluster $\mathcal{G}_{CL}$. Lastly, for $\bar{k}_i = 3$, vertex v_i is definitively left and its non-complete neighboring nodes are selected (line 12) to be visited again.

Algorithm 5 Focus on event: cluster formation
procedure CLUSTERING($v_i, |\mathcal{G}_{CL}|$)

```
1:  for k̄_i = 1, 2, 3 do
2:      k_i ← 0;
3:      while k_i < deg(v_i) and |G_CL| < n_CL do
4:          k_i ← k_i + 1;
5:          v_j ← [N_i]_{k_i};
6:          if v_j ∉ G_CL or k̄_i = 3 then
7:              if k̄_i < 3 then
8:                  G_CL ← G_CL ∪ {v_j};
9:              end if
10:             k_i1 ← k̄_i = 1 and |v_j| > |e_ij|;
11:             k_i2 ← k̄_i = 2;
12:             k_i3 ← k̄_i = 3 and v_j ∉ G_CO;
13:             if |G_CL| < n_CL and (k_i1 or k_i2 or k_i3) then
14:                 CLUSTERING(v_i, |G_CL|);
15:             end if
16:         end if
17:         if k̄_i = 2 and k_i = deg(v_i) then
18:             G_CO ← G_CO ∪ {v_i};
19:         end if
20:     end while
21: end for
```

3.3.3 Dispatch stage

In this final stage representing the original contribution of this chapter, agents belonging to cluster $\mathcal{G}_{CL}$ are dispatched as shown in Alg. 6. Given a temporary leader v_i and

its restricted neighborhood $\mathcal{N}_i^\star$ (defined at line 5) sorted decreasingly w.r.t. the event intensity, each node $v_j \in \mathcal{N}_i^\star$ is sent[6] towards v_i itself from position $\mathbf{p}_j(t)$ to $\mathbf{p}_j(t+T) = \mathbf{p}_j(t) + K_{c_{di},j}(\mathbf{p}_i(t) - \mathbf{p}_j(t))/\left\|\mathbf{p}_i(t) - \mathbf{p}_j(t)\right\|$, $K_{c_{di},j} \geq 0$, as far as collisions or visibility issues do not arise (line 10), i.e. $K_{c_{di},j} = 0$. Moreover, a further stopping criterion (line 18) for agent a_j is given by the impossibility to locally minimize an isoperimetric functional, precisely the cut bipartition function (see Subsec. B.3.1)

$$h_{\mathcal{G}}(\mathcal{G}_{CL}) := \underbrace{\frac{|\partial\mathcal{G}_{CL}|}{\mathrm{vol}_v(\mathcal{G}_{CL})}}_{h_{CL}} + \underbrace{\frac{|\partial\mathcal{G}_{\overline{CL}}|}{\mathrm{vol}_v(\mathcal{G}_{\overline{CL}})}}_{h_{\overline{CL}}}. \tag{3.2}$$

Drawing inspiration from isoperimetric problems Chung (1997), functional in (3.2) is chosen to represent the "bottleneckedness" of the subgraph $\mathcal{G}_{CL}$ w.r.t. graph $\mathcal{G}$: the higher $h_{\mathcal{G}}(\mathcal{G}_{CL})$ the interconnected $\mathcal{G}_{CL}$ appears inside $\mathcal{G}$ (see also Subsec. 2.2.3, where edge expansion techniques for partitioning are addressed). To minimize $h_{\mathcal{G}}(\mathcal{G}_{CL})$, it has been chosen not to add new edges from nodes in $\mathcal{G}_{CL}$ to nodes in $\mathcal{G}_{\overline{CL}}$: keeping $\partial\mathcal{E}_{CL} = \partial\mathcal{E}_{\overline{CL}}$ constant during the dispatch procedure allows to consider term h_{CL} only, since $\mathrm{vol}_v(\mathcal{G}_{\overline{CL}})$ cannot vary and, thus, to simplify the minimization. Defining the positive quantities $\epsilon_S := \mathrm{vol}_v(\mathcal{G}_{CL})$, $\epsilon_{\overline{S}} := \mathrm{vol}_v(\mathcal{G}_{\overline{CL}})$ and $\epsilon_C := |\partial\mathcal{G}_{CL}|$, this heuristics is justified by the fact that the isoperimetric functional variation

$$\begin{aligned} \Delta h_{\mathcal{G}}(\mathcal{G}_{CL}) = -\epsilon_C(\epsilon_S^{-1} + \epsilon_{\overline{S}}^{-1}) + \\ (\epsilon_C + \Delta\epsilon_C)\left[(\epsilon_S + \Delta\epsilon_S)^{-1} + (\epsilon_{\overline{S}} + \Delta\epsilon_{\overline{S}})^{-1}\right] \end{aligned} \tag{3.3}$$

is expected not to become positive in all frameworks of interest. Condition $\Delta h_{\mathcal{G}}(\mathcal{G}_{CL}) < 0$ represents a decrease for functional $h_{\mathcal{G}}(\mathcal{G}_{CL})$, denoting that cluster agents are driven close to the event origin and among themselves at the highest possible level to maintain connectivity with the rest of the network. Indeed, for a large number of nodes in $\mathcal{G}_{CL}$, it is reasonable to assume[7] that $|\Delta\epsilon_S| \ll \epsilon_S$ and $|\Delta\epsilon_{\overline{S}}| \ll \epsilon_{\overline{S}}$. Therefore, trying to preserve the previously established topology without removing edges, it would hold that $\Delta h_{\mathcal{G}}(\mathcal{G}_{CL}) \simeq \Delta\epsilon_C(\epsilon_S^{-1} + \epsilon_{\overline{S}}^{-1}) > 0$. Whereas, in the setting where $h_{\mathcal{G}}(\mathcal{G}_{CL}) := h_{CL}$ and $\Delta\epsilon_C := 0$, it is possible – by showing that $\Delta\epsilon_S + \epsilon_S > 0$ – to conclude that $\Delta h_{\mathcal{G}}(\mathcal{G}_{CL}) = -\epsilon_C\Delta\epsilon_S/\left[\epsilon_S(\epsilon_S + \Delta\epsilon_S)\right] < 0$ if and only if $\Delta\epsilon_S > 0$ (line 12). Denoting with $\Delta\mathcal{N}_j(t+\Delta t) := \left\{v_{\bar{k}} \in \mathcal{G}_{CL} \mid e_{j\bar{k}}(t+\Delta t) \in \mathcal{E}\right\}$ the set of new vertices that node v_j can acquire as neighbors moving in $\mathbf{p}_j(t+\Delta t)$, s.t. $\Delta t \in [0, T]$, and with $\mathcal{N}_j(t+\Delta t) := \mathcal{N}_j(t) \cup \Delta\mathcal{N}_j(t+\Delta t)$ the new neighborhood for v_j, the j-th contribution to the vertex volume variation of cluster $\mathcal{G}_{CL}$ is yielded by

$$\begin{aligned} \Delta_j\mathrm{vol}_v(\mathcal{G}) = \sum_{\forall v_{\bar{k}} \in \Delta\mathcal{N}_j(t+\Delta t)} |v_{\bar{k}}| - |\mathcal{N}_j(t) \cap \mathcal{G}_{CL}|\,|v_j(t)| + \\ |\mathcal{N}_j(t+\Delta t) \cap \mathcal{G}_{CL}|\,\hat{f}_{EV}(\mathbf{p}_j(t+\Delta t)). \end{aligned} \tag{3.4}$$

[6] It is assumed that the bearing-based visual homing controller follows a suitable feedback control law to perform this navigation, e.g. steering the bearing measurement θ_j^i to zero while a_j is moving.

[7] Moving just one agent in cluster $\mathcal{G}_{CL}$ do not change graph volumes ϵ_S, $\epsilon_{\overline{S}}$ significantly, if a large number of agents has already been deployed.

Since each agent a_j moves while all the others do not change their position, i.e. each $\mathbf{p}_{\bar{k}}$ remains constant for all $v_{\bar{k}} \in \Delta\mathcal{N}_j(t+\Delta t)$, it holds that $\Delta_j \text{vol}_v(\mathcal{G}) = \Delta \text{vol}_v(\mathcal{G}) = \Delta\epsilon_S$: this relation proves that Alg. 6 is distributed. Moreover, since condition

$$\epsilon_S = \sum_{\forall v_{\bar{k}} \in \mathcal{G}_{CL}} |v_{\bar{k}}(t)|\,|\mathcal{N}_{\bar{k}}(t) \cap \mathcal{G}_{CL}| > |\mathcal{N}_j(t) \cap \mathcal{G}_{CL}|\,|v_j(t)| \tag{3.5}$$

is always verified, relation $\Delta\epsilon_S > -\epsilon_S$ holds true. In addition, to obtain an estimate of $\Delta_j \text{vol}_v(\mathcal{G})$ whenever the environment is noisy, it has been decided to take into account the signal-to-noise ratio SNR_{w} between the noise w and estimates $\hat{f}_{EV}$. Dividing term $-|\mathcal{N}_j(t) \cap \mathcal{G}_{CL}|\,|v_j(t)|$ in (3.4) by the quantity $(1 + \alpha_{\mathsf{w}} SNR_{\mathsf{w}}^{-1})$, where $\alpha_{\mathsf{w}} > 0$ is a tunable constant, facilitates the establishment of new communication links. Furthermore, whenever a dispatch session starts, node weights are updated (line 4), by means of a filtering procedure, e.g. by using a moving average FIR filter for each agent a_i, acting in the discrete time window where a_i does not change position.
Once again, Alg. 6 should be seen as a distributed one-hop loop[8], not as a centralized recursive procedure because, thanks to the properties of $\mathcal{N}_i^\star$, only counters c_{di} have to be stored and just incremented (line 2) for each node per each session. Lastly, variables $c_d^\star$, $\text{f}_d^\star$ are the only to be passed from node to node whenever a hop from v_i to v_j takes place.

Algorithm 6 Focus on event: agents' dispatch
function $[c_d^\star, \text{f}_d^\star]$ =DISPATCH(v_i,$c_d^\star$,$\text{f}_d^\star$)

1: **if** $c_{di} < c_d^\star$ **then**
2: $\quad c_{di} \leftarrow c_d^\star$;
3: **end if**
4: $|v_i| \leftarrow$FILTERING($|v_i|$,$\hat{f}_{EV}(\mathbf{p}_i)$);
5: $\mathcal{N}_i^\star \leftarrow (\mathcal{N}_i \cap \mathcal{G}_{CL}) \setminus \{v_k \in \mathcal{N}_i \mid c_{dk} = c_d^\star\}$;
6: **for** $k_i = 1, ..., |\mathcal{N}_i^\star|$ **do**
7: $\quad v_j \leftarrow [\mathcal{N}_i^\star]_{k_i}$;
8: $\quad$ **while** a_j is moving from $\mathbf{p}_j(t)$ to $\mathbf{p}_j(t+\Delta T)$ **do**
9: $\quad\quad$ **if** v_j cannot take a step forward **then**
10: $\quad\quad\quad$ **break while**
11: $\quad\quad$ **end if**
12: $\quad\quad$ **if** $\Delta_j \text{vol}_v(\mathcal{G}) > 0$ **then**
13: $\quad\quad\quad$ let a_j move from $\mathbf{p}_j(t)$ to $\mathbf{p}_j(t+\Delta t)$;
14: $\quad\quad\quad$ $|v_j| \leftarrow \hat{f}_{EV}(\mathbf{p}_j(t+\Delta t))$
15: $\quad\quad\quad$ $\mathcal{N}_j(t+\Delta t) \leftarrow \mathcal{N}_j(t) \cup \Delta\mathcal{N}_j(t+\Delta t)$;
16: $\quad\quad\quad$ $\text{f}_d^\star \leftarrow$ **false**;
17: $\quad\quad$ **else**
18: $\quad\quad\quad$ **break while**
19: $\quad\quad$ **end if**
20: $\quad$ **end while**
21: $\quad [c_d^\star, \text{f}_d^\star] \leftarrow$DISPATCH($v_j$,$c_d^\star$,$\text{f}_d^\star$);
22: **end for**

[8] Agents are labeled randomly at the coverage stage and move using local information only. Neither selection order nor navigation commands are imposed by a central hub at each dispatch session. The leader selected with max consensus just triggers the dispatch procedure.

3.4 Discussion on the number of deployed agents

Given the lack of metric information, knowing exactly beforehand how many agents are deployed in the coverage stage is generally arduous; thus, few bounds can be provided in a simplified framework. To begin, the case in which the scenario SC is an obstacle-free rectangle (A1) with dimensions $b_{SC} \times h_{SC}$ is considered; then, a further extension is proposed. In addition, it is also supposed that $r_v \geq 4r_b$ to allow a proper hexagonal packing policy implementation (A2), as far as it is possible. With these ideas in mind, the following proposition is stated.

Proposition 3.4.1. *Assume (A1) and (A2) hold. Let us define the dimensionless quantities* $\varrho_b := b_{SC}/r_v$, $\varrho_{b3} := \varrho_b/\sqrt{3}$, $\varrho_h := h_{SC}/r_v$, $\varrho_{h3} := \varrho_h/\sqrt{3}$ *and the real scalar functions in the positive variables* (ϱ_1, ϱ_2)

$$\overline{g_{SC}}(\varrho_1, \varrho_2) := 1 + \lfloor \varrho_1 \rfloor \lfloor \varrho_2 \rfloor + \lceil \varrho_1 \rceil \lceil \varrho_2 - 1/2 \rceil; \tag{3.6}$$

$$g_{SC}(\varrho_1, \varrho_2) := 1 + \lceil \varrho_1 - \sqrt{4 - \varrho_2^2} \rceil; \tag{3.7}$$

$$\begin{aligned} \underline{g_{SC}}(\varrho_1, \varrho_2) := & \lceil \varrho_1 \rceil (\lceil \varrho_2 + 1/2 \rceil + \lfloor \varrho_2 - 1 \rfloor) + \\ & + \lfloor \varrho_1 \rfloor \lceil \varrho_2 - 1 \rceil - \lfloor \varrho_1 + 1 \rfloor \lfloor \varrho_2 \rfloor. \end{aligned} \tag{3.8}$$

For $\varrho_{b3} > 1$ *and* $\varrho_{h3} > 1$*, the minimum number of deployed agents* n_a *to attain a complete coverage of the scenario can be upper bounded by* $\overline{n_a}$ *and lower bounded by* $\underline{n_a}$*, such that*

$$\overline{n_a} := \min(\overline{g_{SC}}(\varrho_b, \varrho_{h3}), \overline{g_{SC}}(\varrho_h, \varrho_{b3})); \tag{3.9}$$

$$\underline{n_a} := \min(\underline{g_{SC}}(\varrho_b, \varrho_{h3}), \underline{g_{SC}}(\varrho_h, \varrho_{b3})). \tag{3.10}$$

Moreover, it holds that

$$n_a = \begin{cases} g_{SC}(\varrho_b, \varrho_h), & \varrho_{h3} \leq 1 \text{ and } \varrho_b > 1; \\ g_{SC}(\varrho_h, \varrho_b), & \varrho_{b3} \leq 1 \text{ and } \varrho_h > 1; \\ 1, & \varrho_b \leq 1 \text{ and } \varrho_h \leq 1. \end{cases} \tag{3.11}$$

Proof. Whenever $\varrho_{b3} > 1$ and $\varrho_{h3} > 1$ holds, it is possible to cover a rectangular surface as shown in Fig. 3.11. The choice of an appropriate coverage policy allows to determine

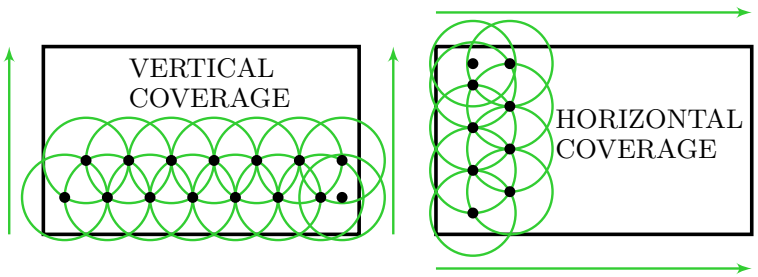


Figure 3.11. Vertical and horizontal coverage policies.

an upper and a lower bound for the number of deployed agents n_a, in this case scenario.

For this reason, in order to compute $\overline{n_a}$ and $\underline{n_a}$ in (3.9) and (3.10) respectively, it is necessary to minimize among $\overline{g_{SC}}(\varrho_1, \varrho_2)$ and $\underline{g_{SC}}(\varrho_1, \varrho_2)$, selecting the most convenient configuration among those shown in Fig. 3.11.
Let us define the fractional part of a real number $\omega \in \mathbb{R}$ as $\text{frac}(\omega) := \omega - \lfloor \omega \rfloor$, for $\omega \geq 0$, and the characteristic function $\chi_\Omega(\omega)$ on set Ω, 1-valued if $\omega \in \Omega$, 0-valued otherwise. W.l.o.g., let us adopt the vertical coverage policy in Fig. 3.11. Then, $\underline{g_{SC}}(\varrho_b, \varrho_{h3}) \leq n_a$ holds by assigning

$$\begin{aligned} \underline{g_{SC}}(\varrho_b, \varrho_{h3}) := (\lfloor \varrho_b \rfloor + \lceil \varrho_b \rceil) \lfloor \varrho_{h3} \rfloor + \\ \lceil \varrho_b \rceil \chi_{\{\omega | \omega > 1/2\}} (\text{frac}(\varrho_{h3})) + \\ - \lfloor \varrho_{h3} \rfloor \chi_{\{0\}} (\text{frac}(\varrho_b)) + \\ - \lfloor \varrho_b \rfloor \chi_{\{0\}} (\text{frac}(\varrho_{h3})) . \end{aligned} \tag{3.12}$$

In (3.12), first and second terms represent the space occupancy for agents at the first and second rows in Fig. 3.11, while third and fourth terms take into account redundant agents placed on the boundaries. Similarly, $\overline{g_{SC}}(\varrho_b, \varrho_{h3}) \geq n_a$ holds by assigning

$$\begin{aligned} \overline{g_{SC}}(\varrho_b, \varrho_{h3}) := 1 + (\lfloor \varrho_b \rfloor + \lceil \varrho_b \rceil) \lfloor \varrho_{h3} \rfloor + \\ \lceil \varrho_b \rceil \chi_{\{\omega | \omega > 1/2\}} (\text{frac}(\varrho_{h3})) . \end{aligned} \tag{3.13}$$

Expressions (3.6) and (3.8) are then obtained by leveraging floor and ceiling properties starting from (3.13) and (3.12), respectively. Now, w.l.o.g., let us assume that the width b_{SC} is larger than the height h_{SC}. If $\varrho_{h3} \leq 1$ and $\varrho_b > 1$ holds then the scenario surface can be exactly covered with the following number of agents:

$$n_a = g_{SC}(\varrho_b, \varrho_h) := 1 + \lceil \varrho_b - 2\sqrt{1 - (\varrho_h/2)^2} \rceil. \tag{3.14}$$

Equality in (3.14) holds true only if $\varrho_b - 2\sqrt{1 - (\varrho_h/2)^2} > -1$ is satisfied; however, the latter relation is already verified, since condition $\varrho_{h3} \leq 1$ and $\varrho_b > 1$ characterizes this case scenario. Finally, for $\varrho_b \leq 1$ and $\varrho_h \leq 1$, one agent is trivially sufficient to cover the entire surface. ■

This result, can be further extended by taking into account border effects while the algorithm compute deployment positions. For instance, the fact that a rectangular scenario SC has a well defined perimeter $p_{SC} = 2(b_{SC} + h_{SC})$ suggests that a good heuristic to approximatively improve lower bound in (3.10) can be adopted by adding to $\underline{n_a}$ the quantity $\underline{n_{ap}} = \lfloor p_{SC}/r_v \rfloor$. Finally, an additional step can be taken in order to provide a rough estimate of the lower bound $\underline{n_a}(SC)$ for a generic scenario SC. Since any connected space in $\mathbb{R}^2$ can be easily approximated by a segmentation into rectangles $\{re_k\}_{k=1}^{L_{re}}$, it holds that

$$\underline{n_a}(SC) \simeq \sigma_{p0} \underline{n_{ap}}(EN) + \sum_{k=1}^{L_{ob}} \sigma_{pk} \underline{n_{ap}}(ob_k) + \sum_{k=1}^{L_{re}} \sigma_{gk} \underline{n_a}(re_k^2) \tag{3.15}$$

where the dependency of each bound on a precise element of the scenario is indicated inside round brackets. In (3.15), coefficients σ_{gk}, for $k = 1, ..., L_{re}$, are either equal to 1,

if $re_k^2 \subseteq CS^2$ or 0, otherwise and coefficients $\sigma_{pk} \in [0,1]$, for $k = 0, \ldots, L_{ob}$, are selected to describe how much a delimitation conveys border effects.

3.5 Numerical simulations

The validity and the limitations of this work are discussed in this section providing numerical simulations in

- A) an obstacle-free scenario $SC = EN$;
- B) structured environment $SC = (EN, OB)$ with presence of obstacles.

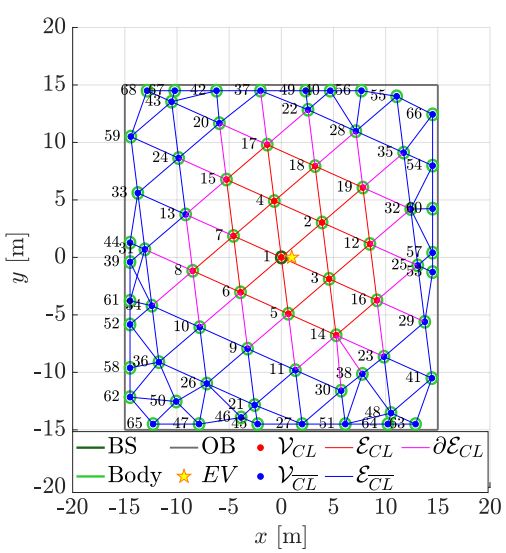
In this set-up, the following quantities are assigned: $r_v = 5$ m, $N_T = 1$, $k_{EV} = 160$, $r_{EV} = 15$ m. Impacts against barriers, collision between agents and general dynamics are virtually implemented in a realistic fashion for all instances of the algorithm. It is assumed that estimates $\hat{f}_{EV}(\mathbf{p}) = f_{EV}(\mathbf{p}) + \mathsf{w}(f_{EV}(\mathbf{p}))$ are affected by a uniformly distributed noise $\mathsf{w}(f_{EV}(\mathbf{p}))$ with zero mean and variance $f_{EV}(\mathbf{p})^2\sigma_{\mathsf{w}}^2/3$; thus, $\alpha_{\mathsf{w}} = 3$ is set.

3.5.1 Obstacle-free scenario

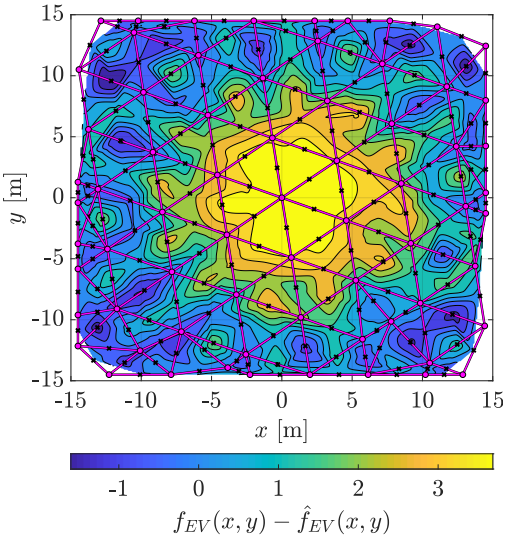
In this simulation, Alg. 3 is tested in a 15 m $\times$ 15 m squared scenario as Fig. 3.12(a) shows. Neither barriers nor noise are present. A coverage with $n_a = 68$ agents is attained. Cluster formation takes place thanks to event sensing, illustrated in Fig. 3.12(b), after a greedy selection of $n_{CL} = 15$ members. Once the dispatch is completed, the spacial distribution of the agents does not practically change (Fig. 3.12(c)), since the event $EV = (1\text{m}, 0)$ is situated in the middle of a quasi-perfect hexagonal packing. Finally, Fig. 3.12(d) exhibits a slight decrease of functional $h_{\mathcal{G}}(\mathcal{G}_{CL})$: this phenomenon is due to borders effects in the scenario that influence agents located in a marginal position w.r.t. the scenario centroid.

3.5.2 Structured and noisy environment

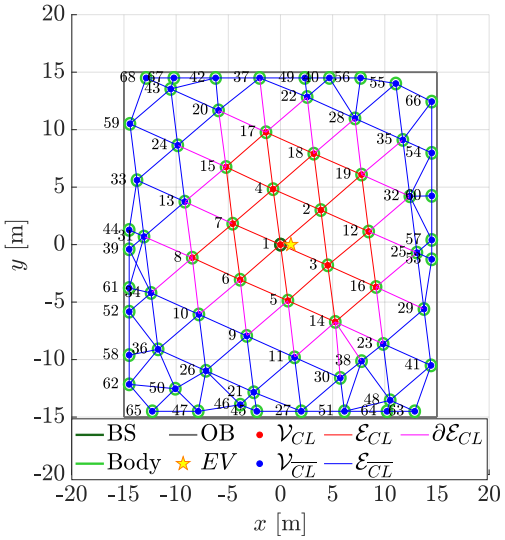
In Fig. 3.13(a), a simulation in a structured squared scenario is shown choosing the event in $EV = (1\text{m}, 0)$. It is remarkable to note that more agents w.r.t. the obstacle-free case have been deployed ($n_a = 98$). Cluster formation ($n_{CL} = 15$ is set) mainly arises where communication links are concentrated and around the event source after the sensing phase, illustrated in Fig. 3.13(b). Differently from the previous case, during the dispatch stage (Fig. 3.12(c) vs. Fig. 3.13(c)), agents focus on the event and, as a result, the graph topology shrinks around and towards the point EV. Remarkably, many additional edges are added in cluster $\mathcal{G}_{CL}$: this fact can be observed in Fig. 3.13(d), where, in correspondence to each sharp decrease of functional $h_{\mathcal{G}}(\mathcal{G}_{CL})$, a new link between two cluster nodes is created. Moreover, another peculiarity is highlighted: the minimization of $h_{\mathcal{G}}(\mathcal{G}_{CL})$ requires several sessions and much more iterations to be accomplished w.r.t. to the obstacle-free case. This fact is also due to the presence of noise ($\sigma_{\mathsf{w}} = 0.01$ is set): the



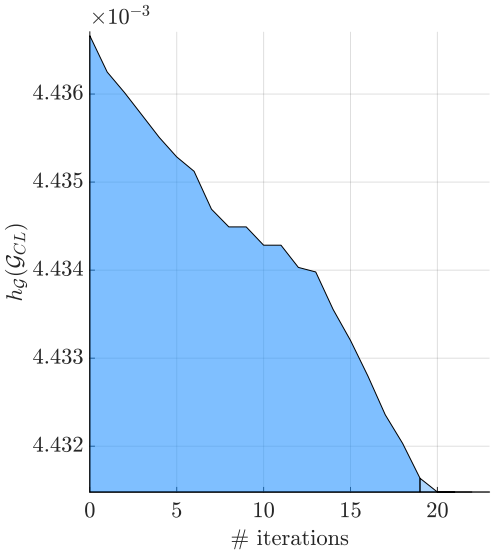
(a) Coverege and cluster selection



(b) Event intensity



(c) Dispatch of the selected cluster

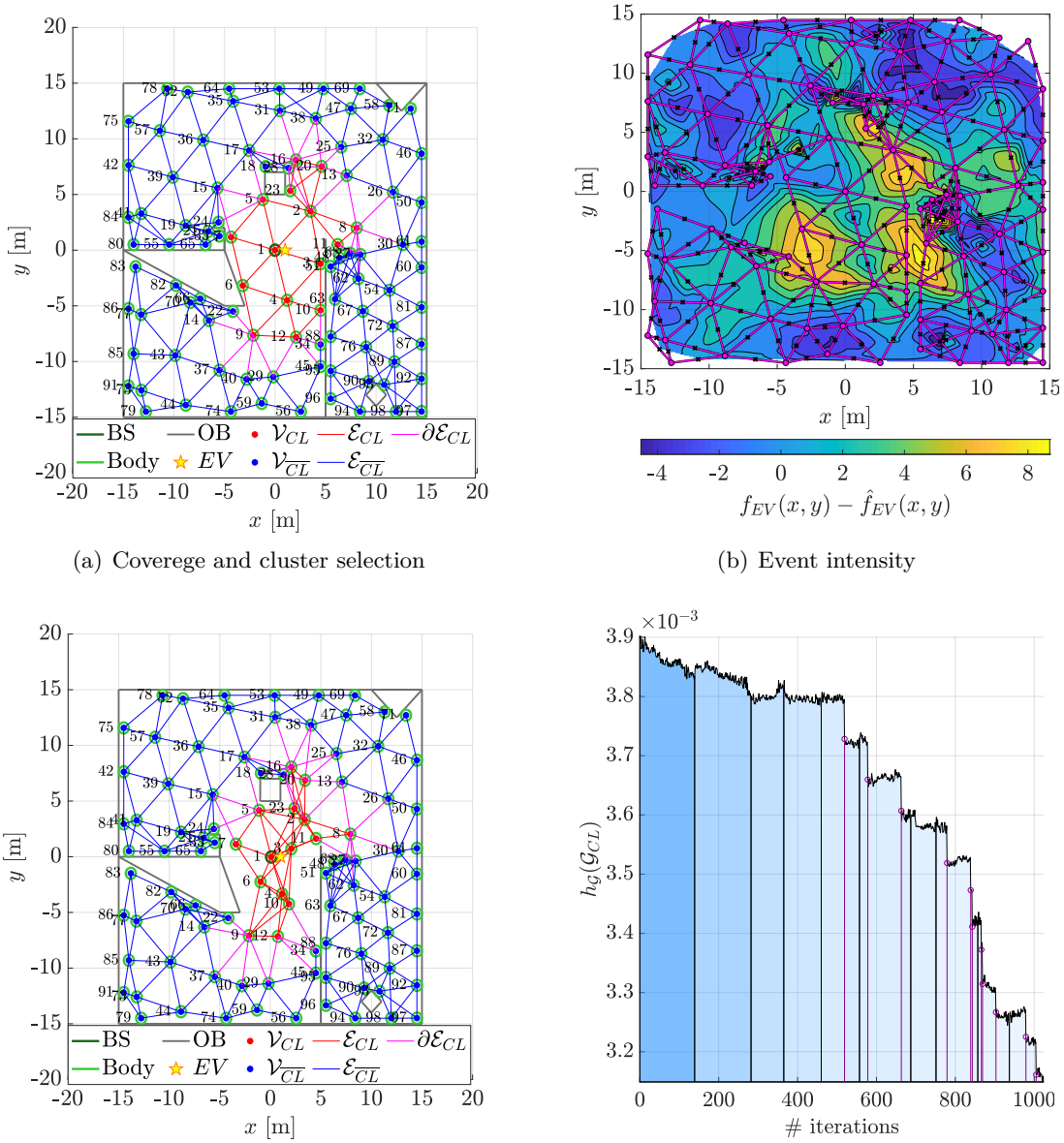


(d) Isoperimetric functional

Figure 3.12. Dynamic coverage in an obstacle-free scenario. Hexagonal packing is mostly achieved, up to border effects, and a cluster (red dots) is selected around the event source (yellow star). The agent dispatch has practically no effect as expected, since topology cannot shrink towards the event. After 20 iterations and 2 sessions, the execution is terminated.

computation of volume variation $\Delta\text{vol}_v(\mathcal{G})$ is affected by uncertainty; therefore, wrong descent directions for $h_{\mathcal{G}}(\mathcal{G}_{CL})$ are selected during dispatch. Finally, one can observe that agent number 28, as few other relevant agents close to the event, is not involved in the cluster because of the noisy environment, representing a potential limitation for this approach.

As a further assessment, the algorithm remains robust enough when events are selected in unreachable points of the scenario, e.g. $EV = (-15\text{m}, 0)$ on the enclosure (Fig. 3.14(a)) and $EV = (0, 6\text{m})$ inside an inaccessible obstacle (Fig. 3.14(c)), even though these two



(a) Coverege and cluster selection

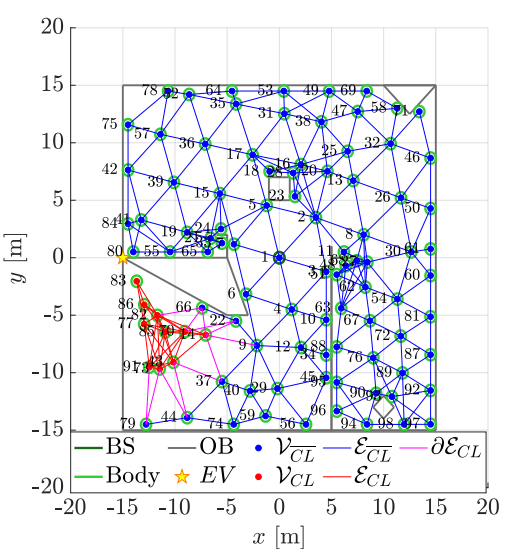
(b) Event intensity

(c) Dispatch of the selected cluster

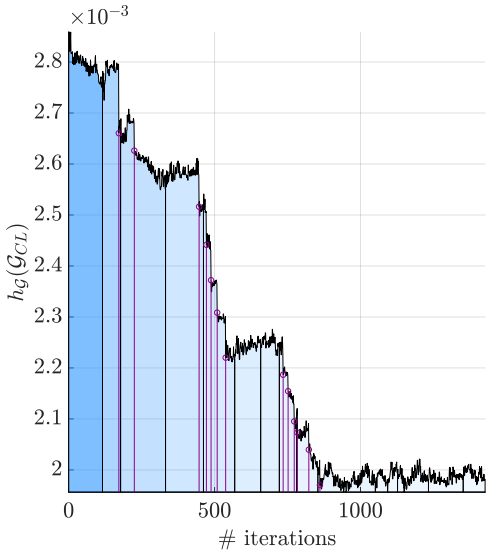
(d) Isoperimetric functional

Figure 3.13. Dynamic coverage in a noisy structured scenario. Hexagonal packing is achieved only in few areas, because of the large presence of obstacle borders. The agent dispatch leads to a graph topology narrowing by increasing the cluster volume: whenever an edge between two nodes in the cluster is added, the adopted isoperimetric functional decreases with a discontinuity (purple spikes). Execution stops after 10 sessions, with more than 1000 iterations.

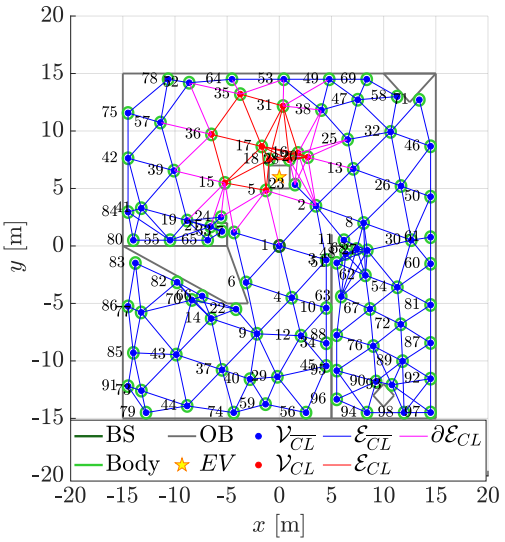
event sources are arduous or impossible to be covered. In both cases, especially when the event is located on the enclosure, functional $h_{\mathcal{G}}(\mathcal{G}_{CL})$ requires a large number of iterations and potentially infinite sessions to be minimized, as shown in Figs. 3.14(b) and 3.14(d), but a focus on the event is eventually achieved, to some extent. However, in these more critical cases, few limitations emerge: the considerable presence of noise ($\sigma_{\mathsf{w}} = 0.1$) and the lower desired number of agents selected ($n_{CL} = 10$) affect both clustering and dispatch performances negatively. For instance, cluster in Fig. 3.14(c) does not involve agent 23; moreover, in Fig. 3.14(a) the dispatch drives agents fairly far from the event.



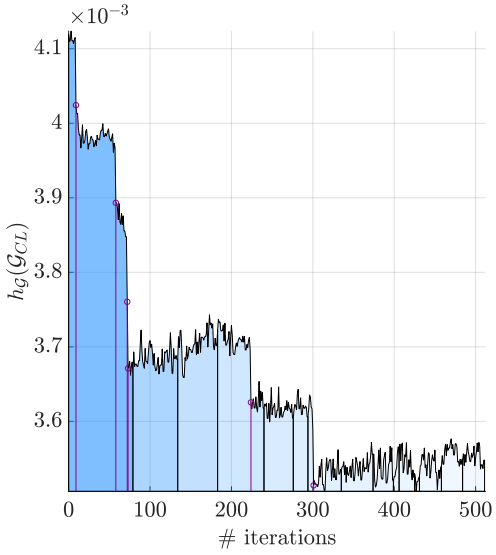
(a) Dispatch of the selected cluster



(b) Isoperimetric functional



(c) Dispatch of the selected cluster



(d) Isoperimetric functional

Figure 3.14. Dynamic coverage in a highly noisy structured scenario with unaccessible event sources. In (a) and (b) the dispatch is attained towards an event placed on the enclosure. In (c) and (d) the dispatch is attained around squared obstacle. The abundant presence of noise limits the cluster selection and the poor sensing affects the dispatch performances.

3.6 Chapter summary

In this chapter, an algorithm for dynamic coverage and focus on event has been designed. The agents employed for this task are provided by a bearing-based visual homing controller relying on limited sensing capabilities and local information. Geometrical models can simulate space occupation in an unknown scenario admitting the presence of obstacles.
The devised algorithm is divided into three subsequent phases. Firstly, in the deployment stage, a fully distributed implementation is proposed, starting from the work in Ramaithitima et al. (2015). Furthermore, bounds for the number of deployed agents are given. Secondly, in the clustering stage, the same paradigm of Oliva, Panzieri, Setola, and Gasparri (2019) has been exploited to carry out this task. Lastly, leveraging the minimization of an isoperimetric functional to increase the cluster volume and, consequently, maximizing the communication over the cluster, the dispatch of agents has been attained towards an event belonging to the environment.

4

Optimal Time-Invariant Formation Tracking

"There is a difference between knowing the path and walking the path."
Morpheus

Contents

The contents of this chapter are partly available in:
Fabris et al. (2019a). "*Optimal Time-Invariant Formation Tracking for a Second-Order Multi-Agent System*". In 18th European Control Conference 2019, Napoli, Italy, pp. 1556-1561, Jun 25-28, 2019.

4.1 Overview

In the framework of MAS applications, researchers are often interested in designing and controlling groups of agents in order to attain specific collective goals, such us, for example, rendezvous in mobile robotics Luo, Kim, Parasuraman, Bae, Matson, and Min (2019); Ozsoyeller, Beveridge, and Isler (2019), cooperative transportation Belbachir, Fallah-Seghrouchni, Casals, and Pasin (2019); Shirani, Najafi, and Izadi (2019), reliable communication over networks Bhanu, Reddy, and Hanumanthappa (2019); Redhu and Hegde (2019), exploration and mapping of unknown environments Geng, Zhou, Ding, Wang, and Zhang (2018), formation control for unmanned aerial vehicles He, Bai, Liang, Zhang, and Wang (2018); Shao, Yan, Zhou, and Zhu (2019); Yan, Shi, Lim, Wu, and Shi (2018); Zhou, Zhang, and Bi (2019), surveillance and monitoring Du, Sun, Cao, and Wang (2017); Oliva et al. (2019); Tsochev, Trifonov, Yoshinov, Manolov, Popov, and Pavlova (2018).

Recently, distributed formation tracking (FT) has emerged as one of the most appealing topics related to MAS Chu, Peng, Wen, and Rahmani (2018); Li, Xie, and Yan (2017); Peng, Sun, and Geng (2019); Wang, He, and Huang (2019), since many FT problems arise in trajectory optimization, path planning and maneuver regulation when a multitude of robotic agents is employed. Many recent works developed for autonomous vehicles exploit Trajectory Optimization to devise such control strategies, even in constrained environments. Practically, the employment of direct methods for the minimization of a cost functional represent a fundamental approach to provide solutions to this kind of problems.

To conclude this heading, Fig. 4.1 depicts the main features arising in this work. The methods employed in this chapter widely range over almost every discipline related to the manuscript, considering Optimization Theory, Dynamic Systems and Control, Combinatorial Graph Theory, Distributed Optimization and Distributed Control Systems. Nonetheless, the major focus in this work is directed toward the design of an optimal controller able to accomplish the given specification tasks in a distributed fashion, drawing inspiration from Optimal Control and exploiting a Trajectory-Optimization-based numerical tool in order to explore the formation of agents involved in the flocking.

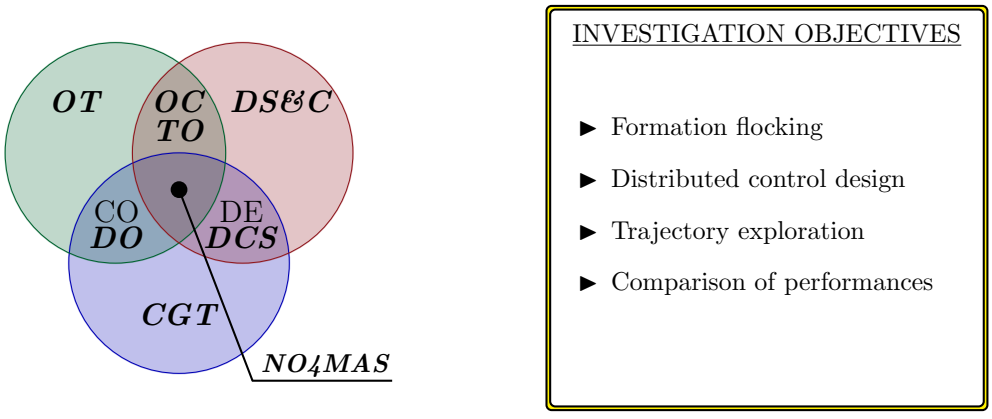


Figure 4.1. Theoretical fields and investigation objectives arising from this study.

4.1.1 Problem statement

The problem of FT Li and Duan (2015); Mesbahi and Egerstedt (2010) is usually formalized as determining a coordinated control law that keeps the MASs maintain a desired, not necessarily time-invariant, formation while following a reference orbit or tracking a target.
In the past years, this topic was initially tackled separately as Formation Control (FC) and Tracking Control (TC), when the subject of the study did not have a direct focus on a particular realistic application Xiao, Wang, Chen, and Gao (2009); Zhao, Duan, Wen, and Zhang (2013). Whereas, nowadays, a large part of the literature about this field addresses both of these two aspects simultaneously Liu, Zhao, and Chen (2016); Santiaguillo-Salinas and Aranda-Bricaire (2017); Yang, Cao, Garcia de Marina, Fang, and Chen (2018). However, in the majority of these studies, the energy consumption of the system under analysis is not taken into examination in details, even though it may represent a crucial constraint for the design.
Given the intrinsic complexity of the problem, which exhibits multiple facets, and considered the specific framework in which it is studied (along with the presence of novel contributions regarding the energy consumption aspect) its formalization will be better described in the dedicated Sec. 4.2.

4.1.2 Related works

Historically, TC, i.e. the field that studies how to generate proper input functions to track a desired reference trajectory, has been faced in several fashions; yet, controllers taking into account proportional actions to the position and velocity disagreement vectors have been largely devised, analyzed and successfully implemented in second order integrator MASs Zuo (2015). Furthermore, many open challenges can be classified as FC problems Cenedese, Favaretto, and Occioni (2016); Fathian, Doucette, Curtis, and Gans (2018); Schiano, Franchi, Zelazo, and Giordano (2016), in which the aim is represented by the accomplishment of prescribed constraints on their states (e.g. shapes or, more generally, geometrical patterns) for the multiple agents involved. Depending on the interaction and the sensing capabilities of the agents, a variety of FC problems can be encountered in the literature (see Anderson, Yu, Fidan, and Hendrickx (2008) for a comprehensive overview). For instance, in Oh, Park, and Ahn (2015), authors discern FC problems among distance-based, displacement-based and position-based according to the types of controlled and sensed variables. Many models in FC make use of potential based functions in order to implement one of the latter paradigm, since the maintenance of a formation depends on trade off between single dynamics behaviors, collision avoidance, and dispersion Chuang, Huang, D'Orsogna, and Bertozzi (2007); D'Orsogna, Chuang, Bertozzi, and Chayes (2006); Hauser and Cook (2016). These and similar laws often resort to a potential that determines the pairwise interactions among the agents, and along this line, many FC algorithms adopt a formulation that contains an attractive part to maintain the swarm cohesion and a repulsive one to avoid agent collision (see also Subsec. 2.1.3).

4.1.3 Contribution and outline of the chapter

In this work, it is proposed a novel approach to solve the formation problem for mobile robots MAS in which the FT task is formulated as a single time-invariant optimal control problem where the different subtasks are considered. Specifically, it is proposed a design strategy based on a cost functional that accommodates the tasks of 1) tracking a given a desired path (TC), 2) reach and maintain a formation (FC), specified by a feasible geometric shape, and 3) control input energy minimization. In particular, starting from the preliminary results in Fabris et al. (2019a), in which the optimization framework PRONTO (PRojection Operator based Newton's method for Trajectory Optimization, Aguiar, Bayer, Hauser, Häusler, Notarstefano, Pascoal, Rucco, and Saccon (2017)) is successfully adopted to obtain an inverse dynamics to steer the agents in formation along a chosen trajectory. In this work, it is provided a more thorough analysis of the approach and proposed a distributed control design that accounts for the presence of communication constraints in the system of mobile robots, within a scenario where information exchange between pairs of agents is not ensured. In order to provide a real time solution for such a formalization, an online networked strategy based on an optimal distributed proportional derivative (PD) controller is devised. No comparison between such a controller and other PD controllers in the literature is here presented: future developments on this side may be envisaged extending the work in this chapter. Instead, the original contribution of this PD controller rests upon the capability and versatility to accomplish all the aforementioned optimization tasks in a distributed fashion, according to a given settling time specification.

In the remainder of the chapter, Secs. 4.2 and 4.3 describe, in a general way, how the FT problem can be formulated and solved by using an optimization framework, and specifically PRONTO. Then, the main technical Sec. 4.3 presents the derivation of the distributed controller to govern the system's dynamics and a brief analysis of the system's equilibria. Finally, Secs. 4.4 and 4.5 illustrate some numerical results, providing a validation of the proposed approach and interesting final remarks for this research.

4.2 Problem setup

In this section, some assumptions related to the agents' dynamics are reported and a formalization of an optimal control problem is proposed, enriching the description of the approach developed in Fabris et al. (2019a). In doing so, this is still addressed as the Optimal Time-Invariant Formation Tracking (OIFT) problem, maintaining the aim of finding a potential-based solution by minimizing a global cost functional able to capture all the different assignments simultaneously. Nonetheless, communication topology constrains are now taken into consideration: along with the application of PRONTO as the chosen optimization framework for FT, the development of a distributed optimal control law to accomplish these tasks represents a novel and crucial goal for this work, that goes beyond the centralized formulation presented in Fabris et al. (2019a).

4.2.1 Dynamics of the agents

It is supposed that $n > 1$ robotic agents with a linear dynamics are already located in an M-dimensional space, $M \in \{1, 2, 3\}$. It is also assumed that each agent i, for $i = 1, ..., n$, is aware of its own absolute position $\mathbf{p}_i = \mathbf{p}_i(\tau) \in \mathbb{R}^M$, with τ indicating the continuous time, and velocity $\dot{\mathbf{p}}_i \in \mathbb{R}^M$, and can be governed by means of a control law acting on its absolute acceleration $\ddot{\mathbf{p}}_i \in \mathbb{R}^M$. With $N := nM$, the expressions of the state $\mathbf{x} \in \mathbb{R}^{2N}$ and the input $\mathbf{u} \in \mathbb{R}^N$ of this group of mobile agents are provided respectively by

$$\mathbf{x} = \begin{bmatrix} \mathbf{p}_1^\top & \dots & \mathbf{p}_n^\top & \dot{\mathbf{p}}_1^\top & \dots & \dot{\mathbf{p}}_n^\top \end{bmatrix}^\top = \begin{bmatrix} \mathbf{p}^\top & \dot{\mathbf{p}}^\top \end{bmatrix}^\top; \tag{4.1}$$

$$\mathbf{u} = \begin{bmatrix} \ddot{\mathbf{p}}_1^\top & \dots & \ddot{\mathbf{p}}_n^\top \end{bmatrix}^\top = \ddot{\mathbf{p}}. \tag{4.2}$$

Differently from Fabris et al. (2019a), it is assumed that the state information is not globally accessible for each agent, but just locally, so that an estimate of the centroid position $\mathbf{p}_c = n^{-1} \sum_{i=1}^{n} \mathbf{p}_i$ and velocity $\dot{\mathbf{p}}_c$ be available to each agent at each time instant via an average consensus procedure. For this reason, it is considered a connected undirected graph $\mathcal{G}$ to model the communication network among agents in the system. It is set $\mathbf{x}_c = \begin{bmatrix} \mathbf{p}_c^\top & \dot{\mathbf{p}}_c^\top \end{bmatrix}^\top$, with $\mathbf{x}_c \in \mathbb{R}^{2M}$, to be treated as an output for this linear dynamics.

Since the aim is to steer the agents such that their positions and velocities are controlled through their accelerations, a double integrator model is chosen: the following linear state space represents the adopted second-order dynamics

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \mathbf{x}_c = \mathbf{C}\mathbf{x} \end{cases}. \tag{4.3}$$

State matrix $\mathbf{A} \in \mathbb{R}^{2N \times 2N}$, input matrix $\mathbf{B} \in \mathbb{R}^{2N \times N}$ and output matrix $\mathbf{C} \in \mathbb{R}^{2M \times 2N}$ in (4.3) are given by

$$\mathbf{A} = \begin{bmatrix} \mathbf{Z}_N & \mathbf{I}_N \\ \mathbf{Z}_N & \mathbf{Z}_N \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} \mathbf{Z}_N \\ \mathbf{I}_N \end{bmatrix} \tag{4.4}$$

$$\mathbf{C} = \frac{1}{n} \begin{bmatrix} \mathbf{I}_M & \dots & \mathbf{I}_M & \mathbf{Z}_M & \dots & \mathbf{Z}_M \\ \mathbf{Z}_M & \dots & \mathbf{Z}_M & \mathbf{I}_M & \dots & \mathbf{I}_M \end{bmatrix}, \tag{4.5}$$

where $\mathbf{I}_\natural$ indicates the identity matrix of dimensions $\natural \times \natural$, $\mathbf{Z}_{\natural_1 \times \natural_2}$ denotes null matrices of dimensions $\natural_1 \times \natural_2$ and if $\natural_1 = \natural_2 = \natural$ the convention $\mathbf{Z}_{\natural \times \natural} = \mathbf{Z}_\natural$ is used.

4.2.2 Minimization problem formulation

In this context, it is required to perform tracking of a given path $\mathbf{x}_{c,des} = \begin{bmatrix} \mathbf{p}_{c,des}^\top & \dot{\mathbf{p}}_{c,des}^\top \end{bmatrix}^\top$ with the formation centroid $\mathbf{x}_c$ while minimizing the energy spent by the input $\mathbf{u}$ and attaining a desired formation, by stating an optimal control problem and designing a distributed proportional derivative (PD) control law. A more general OIFT problem can now be stated as:

Problem 4.2.1. *Given the second-order dynamics (4.3) and its trajectory manifold $\mathcal{T}$ over the time interval $[0, T]$, such that $\boldsymbol{\xi} = (\mathbf{x}(\cdot), \mathbf{u}(\cdot)) \in \mathcal{T}$, find a solution $\boldsymbol{\xi}^\star$ such that*

$$\min_{\boldsymbol{\xi} \in \mathcal{T}} h(\boldsymbol{\xi}) \tag{4.6}$$

is attained, where

$$h(\mathbf{x}(\cdot), \mathbf{u}(\cdot)) = m(\mathbf{x}(T)) + \int_0^T l\left(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau\right) d\tau \tag{4.7}$$

represents the cost functional to be minimized over $[0, T]$.

Generalizing the approach in Fabris et al. (2019a), two different terms explicitly appear in (5.1), to refer the three objectives defined before: the instantaneous cost

$$l\left(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau\right) := l^{tr}(\mathbf{x}_c(\tau)) + l^{in}(\mathbf{u}(\tau)) + l_d^{fo}(\mathbf{p}(\tau)) + l^{al}(\dot{\mathbf{p}}(\tau)) \tag{4.8}$$

and the final cost

$$m(\mathbf{x}(T)) := l^{tr}(\mathbf{x}_c(T)) + l_d^{fo}(\mathbf{p}(T)) + l^{al}(\dot{\mathbf{p}}(T)). \tag{4.9}$$

Each term in (4.8) is minimized with the purpose to obtain the fulfillment of a specific task. Indeed, setting $r_{ij}(\tau) := \|\mathbf{p}_i(\tau) - \mathbf{p}_j(\tau)\|_2$ as the inter-agent Euclidean distance, each contribution is defined as:

$$l^{tr}(\mathbf{x}_c(\tau)) = \frac{1}{2} \sum_{i=1}^{n} \|\mathbf{x}_c(\tau) - \mathbf{x}_{c,des}(\tau)\|^2_{\mathbf{Q}_{c,\dot{c},i}} \tag{4.10}$$

for the tracking task;

$$l^{in}(\mathbf{u}(\tau)) = \frac{1}{2} \sum_{i=1}^{n} \|\mathbf{u}_i(\tau)\|^2_{\mathbf{R}_i} \tag{4.11}$$

for the input energy task and, given a family of potential functions $\sigma_{d_{ij}} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$, s.t. $i \neq j$,

$$l_d^{fo}(\mathbf{p}(\tau)) = \frac{k_F}{4} \sum_{i=1}^{n} \sum_{\forall j \neq i} \sigma_{d_{ij}}\left(r_{ij}^2(\tau)\right) \tag{4.12}$$

for the formation task. Moreover, it is also accounted for term

$$l^{al}(\dot{\mathbf{p}}(\tau)) = \frac{k_A}{4} \sum_{i=1}^{n} \sum_{\forall j \neq i} \|\dot{\mathbf{p}}_i(\tau) - \dot{\mathbf{p}}_j(\tau)\|^2_{q_{A_{ij}}} \tag{4.13}$$

in order to implement a velocity alignment paradigm between agents. This allows the system to behave according to the well known Boids model, also based on cohesion and separation principles Reynolds (1987).

For all $i = 1 \dots n$, matrix $\mathbf{Q}_{c,\dot{c},i} = \text{Diag}\{\mathbf{Q}_{c,i}, \mathbf{Q}_{\dot{c},i}\} \in \mathbb{R}^{2M \times 2M}$ in (4.10), such that $\mathbf{Q}_{c,i} \in \mathbb{R}^{M \times M}$ and $\mathbf{Q}_{\dot{c},i} \in \mathbb{R}^{M \times M}$ weight the i-th centroid position and velocity respectively, is symmetric positive semidefinite and matrix $\mathbf{R}_i \in \mathbb{R}^{M \times M}$ in (4.11) is symmetric positive definite. In (4.12)-(4.13) coefficients $k_F > 0$ and $k_A > 0$ are constant weights, as well as $q_{A_{ij}} \geq 0$, that are null, if agents i and j do not exchange information, and larger

than zero, otherwise. All these parameters can be tuned according to given specifications in order to penalize the trajectory tracking, the energy spent by the inputs and the convergence to a desired shape respectively. Furthermore, considering the potential functions $\sigma_{d_{ij}}$ in the formation cost (4.12), each d_{ij} represents the desired inter-agent distance between a pair of agents (i, j) that is allowed to exchange information: an accurate selection of $\sigma_{d_{ij}}$ leads to the implementation of the cohesion and separation paradigms while using the optimization framework PRONTO, as explained next.

4.2.3 Potential-based formations

With regard to the formation objective, it is required the system of agents to achieve desired 1D, 2D or 3D geometric shapes induced by a set of constraints of the form

$$r_{ij} = d_{ij}, \quad \text{s.t. } i = 1, ..., n, \ \forall j \neq i; \tag{4.14}$$

by means of opportunely designed potential functions. Let $s_{ij} = r_{ij}^2$ be the squared inter-agent distance between agents i and j. Among several possible choices, the potential function

$$\sigma_{d_{ij}}(s_{ij}) := \begin{cases} k_{r_{ij}}(1 - s_{ij}/d_{ij}^2)^{\beta_{ij}} & \text{for } 0 \leq s_{ij} < d_{ij}^2 \\ k_{a_{ij}}\left((s_{ij}/d_{ij}^2)^{\alpha_{ij}} - 1\right)^{\beta_{ij}} & \text{for } s_{ij} \geq d_{ij}^2 \end{cases} \tag{4.15}$$

is selected since it is a power function in r_{ij} with degree $\beta_{ij} \in \mathbb{N}$, $\beta_{ij} > 2$, and differentiable with respect to $s_{ij} \geq 0$ until the second order. In addition, (4.15) is $\mathscr{C}^2$ and its magnitude can be adjusted by tuning constants $k_{r_{ij}} \geq 0$ and $k_{a_{ij}} \geq 0$ independently. These two parameters are on purpose designed to be directly proportional to the repulsion and attraction actions between agents respectively, playing a crucial role in the intensity regulation[1] of the potential itself (see Fig. 4.2). Moreover, exponent $\alpha_{a_{ij}} > 0$ allows to tune the magnitude of the attractive potential: specifically, if $s_{ij} > 2^{1/\alpha_{a_{ij}}} d_{ij}^2$, it is possible to say that the attraction force becomes significantly high ($\sigma_{d_{ij}} > k_{a_{ij}}$) for the system of agents; on the other hand, the maximum repulsion intensity is reached as s_{ij} approaches 0 to avoid collisions, with $\sigma_{d_{ij}} \to k_{r_{ij}}$, being $\sigma_{d_{ij}} \leq k_{r_{ij}}$ in this regime. A default choice can be preferred over others by setting $\alpha_{a_{ij}} = 1/2$, since this implies that the attractive potential starts to play an important role whenever an effective inter-agent distance s_{ij} exceeds twice its reference d_{ij}^2. In practical applications, though, $\alpha_{a_{ij}}$ should be chosen accordingly to the communication range.

As depicted in Fig. 4.2, the qualitative graphic of (4.15) and its derivatives emphasize the fact that

1. $\sigma'_{d_{ij}}(s_{ij}) \leq 0$ for $0 \leq s_{ij} < d_{ij}^2$;
2. $\sigma'_{d_{ij}}(s_{ij}) \geq 0$ for $s_{ij} \geq d_{ij}^2$;
3. $\sigma''_{d_{ij}}(s_{ij}) \geq 0$ for all s_{ij};

where the equalities in 1)-3) hold if and only if $s_{ij} = d_{ij}^2$.

[1] Gains $k_{r_{ij}}$ and $k_{a_{ij}}$ represent the nominal repulsive and attractive magnitudes respectively and are set to zero if and only if no communication occurs between i and j.

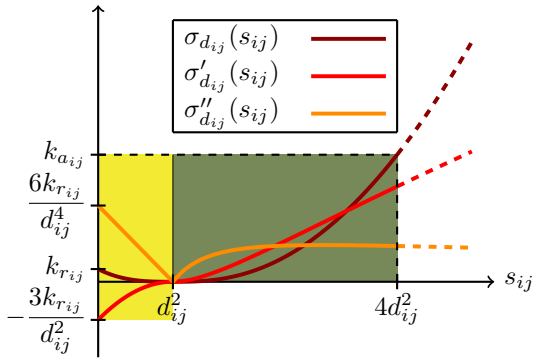


Figure 4.2. Potential function $\sigma_{d_{ij}}(s_{ij})$ with $\beta_{ij} = 3$, $\alpha_{a_{ij}} = 1/2$, $k_{a_{ij}} > k_{r_{ij}}$ and its derivatives w.r.t. s_{ij} up to the second order. Attractive and repulsive behaviors can be associated to the values in the dark green and yellow areas respectively.

The employment of potential functions such as (4.15) can be justified by the fact that they can lead to formations that verify the maximum number of feasible relations[2] in (4.14), since the dynamics is purposely driven to minimum-potential trajectories. Indeed, it can be easily proven that $\sigma_{d_{ij}}(s_{ij})$ is nonnegative for all s_{ij} and exhibits a unique global minimum point $(s_{ij}, \sigma_{d_{ij}}(s_{ij})) = (d_{ij}^2, 0)$, i.e. these potentials go to 0 when the desired distance d_{ij} within pair (i, j) is achieved.

4.3 Solutions for the OIFT problem

Here, it is provided a brief introduction to PRONTO[3], the numerical tool used for trajectory optimization, and clarify its adaptation to solve in a centralized fashion the OIFT problem introduced before. In the second part of this section, we devise an online distributed controller approximating the performance of inverse dynamics computed by PRONTO, which serves as a reference to compare the performances of the decentralized approach.

4.3.1 Basics of PRONTO

In its basic form, PRONTO is an iterative numerical algorithm for the minimization (see also Sec. A.1), through a Newton method, of a general cost functional (5.1). To work on the (infinite dimensional) trajectory Banach manifold $\mathcal{T}$ of a nonlinear control system $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$, $f \in \mathscr{C}^k$, $k \geq 1$, state-control curves $(\boldsymbol{\alpha}(\cdot), \boldsymbol{\mu}(\cdot))$ in the ambient Banach space are projected onto $\mathcal{T}$, $(\boldsymbol{x}(\cdot), \boldsymbol{u}(\cdot)) \in \mathcal{T}$, by using a linear time-varying trajectory tracking

[2] Meaning that they can be satisfied concurrently.

[3] For further details about properties of PRONTO, the interested reader is referred to the accurate explanations provided in chapter 4 of thesis Häusler (2015) and in Aguiar et al. (2017); Hauser and Saccon (2006).

controller. To this end, a generic nonlinear feedback system

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}), \quad \mathbf{x}(0) = \mathbf{x}_0; \tag{4.16}$$

$$\mathbf{u} = \boldsymbol{\mu} + \mathbf{K}(\boldsymbol{\alpha} - \mathbf{x}), \tag{4.17}$$

where matrix $\mathbf{K}$ acts as a (time-varying) controller, defines a nonlinear operator ($\mathscr{C}^k$ when $f \in \mathscr{C}^k$)

$$\mathcal{P}(\boldsymbol{\xi}) : \boldsymbol{\xi} = (\boldsymbol{\alpha}(\cdot), \boldsymbol{\mu}(\cdot)) \mapsto \boldsymbol{\eta} = (\mathbf{x}(\cdot), \mathbf{u}(\cdot)) \tag{4.18}$$

mapping bounded curves (in its domain) to trajectories.

Variables $\boldsymbol{\alpha}(\cdot)$ and $\boldsymbol{\mu}(\cdot)$ in (4.18) have to be intended as a curve $\boldsymbol{\xi} = (\boldsymbol{\alpha}(\cdot), \boldsymbol{\mu}(\cdot))$ that is not necessarily a trajectory and, initially, at iteration $t = 0$, represents the first attempt to compute input $\mathbf{u}(\cdot)$. Terms $\boldsymbol{\alpha}(\cdot)$ and $\boldsymbol{\mu}(\cdot)$ are then corrected and improved at each updating step of the PRONTO algorithm, minimizing cost $h(\boldsymbol{\xi})$, by means of the peculiar projection operator features. Among the many properties of the projection operator $\mathcal{P}$, the most relevant in terms of the practical use of PRONTO refers to the fact that it makes possible to easily switch between constrained and unconstrained minimization procedures to solve a given optimization problem, as

$$\min_{\boldsymbol{\xi} \in \mathcal{T}} h(\boldsymbol{\xi}) \quad \Leftrightarrow \quad \min_{\boldsymbol{\xi}} g(\boldsymbol{\xi}) \tag{4.19}$$

where $g(\boldsymbol{\xi}) = h(\mathcal{P}(\boldsymbol{\xi}))$.

In general, for any tangent curve $\boldsymbol{\zeta} = (\boldsymbol{\beta}(\cdot), \boldsymbol{\nu}(\cdot))$ at a certain $\boldsymbol{\xi} = (\boldsymbol{\alpha}(\cdot), \boldsymbol{\mu}(\cdot))$, the Fréchet derivative (see also Sec. B.5) of $\mathcal{P}(\cdot)$ is denoted with the continuous linear operator

$$D\mathcal{P}(\boldsymbol{\xi}) : \boldsymbol{\zeta} = (\boldsymbol{\beta}(\cdot), \boldsymbol{\nu}(\cdot)) \mapsto \boldsymbol{\gamma} = (\mathbf{z}(\cdot), \mathbf{v}(\cdot)) \tag{4.20}$$

such that the approximation

$$\mathcal{P}(\boldsymbol{\xi} + \boldsymbol{\zeta}) \approx \mathcal{P}(\boldsymbol{\xi}) + D\mathcal{P}(\boldsymbol{\xi}) \cdot \boldsymbol{\zeta} \tag{4.21}$$

holds for all $\boldsymbol{\xi} \in \mathcal{T}$. Then, the PRONTO algorithm can thus be formalized as

Algorithm 7 Projection operator Newton's method Hauser (2002)
Input: initial trajectory $\boldsymbol{\xi}_0 \in \mathcal{T}$
Output: final trajectory $\boldsymbol{\xi}_{t+1} \in \mathcal{T}$

1: **for** $t = 0, 1, 2, \ldots$ **do**
2: Redesign feedback $\mathbf{K}(\cdot)$ for $\mathcal{P}(\cdot)$, if desired/needed
3: $\boldsymbol{\zeta}_t = \underset{\boldsymbol{\zeta} \in T_{\boldsymbol{\xi}_t}\mathcal{T}}{\arg\min}\ Dh(\boldsymbol{\xi}_t) \cdot \boldsymbol{\zeta} + \frac{1}{2} D^2 g(\boldsymbol{\xi}_t) \cdot (\boldsymbol{\zeta}, \boldsymbol{\zeta})$
4: $\gamma_t = \underset{\gamma \in (0,1]}{\arg\min}\ g(\boldsymbol{\xi}_t + \gamma \boldsymbol{\zeta}_t)$
5: $\boldsymbol{\xi}_{t+1} = \mathcal{P}(\boldsymbol{\xi}_t + \gamma_t \boldsymbol{\zeta}_t)$
6: **end for**

In particular, Step 3 in Alg. 7 encodes the search direction problem for the Newton's method and can be solved by finding a solution to the time-varying linear quadratic (LQ)

optimal control problem (see Ex. 2.2.3 for its time-invariant version) of the form

$$\min \int_0^T \left(\mathbf{a}^\top \mathbf{z} + \mathbf{b}^\top \mathbf{v} + \frac{1}{2} \begin{bmatrix} \mathbf{z} \\ \mathbf{v} \end{bmatrix}^\top \begin{bmatrix} \mathbf{Q}_o & \mathbf{S}_o \\ \mathbf{S}_o^\top & \mathbf{R}_o \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{v} \end{bmatrix} \right) d\tau + \mathbf{r}_1^\top \mathbf{z}(T) + \frac{1}{2} \mathbf{z}(T)^\top \mathbf{P}_1 \mathbf{z}(T) \tag{4.22}$$

$$\text{subject to } \dot{\mathbf{z}} = \bar{\mathbf{A}}\mathbf{z} + \bar{\mathbf{B}}\mathbf{v}, \quad \mathbf{z}(0) = \mathbf{z}_0 \tag{4.23}$$

where vectors $\mathbf{a}$, $\mathbf{b}$ and matrices $\mathbf{Q}_o$, $\mathbf{S}_o$, $\mathbf{R}_o$, $\bar{\mathbf{A}}$, $\bar{\mathbf{B}}$ are time-varying parameters, obtained from the instantaneous cost l and the adopted dynamics, and, furthermore, vector $\mathbf{r}_1$ and matrix $\mathbf{P}_1$ are given constants that depend on the final cost. The relation between the optimal problem (4.22)-(4.23) and the formulation of the cost functional (4.8)-(4.9) will be made clearer in next subsection.

Interestingly, when problem (4.22)-(4.23) has a unique minimizing trajectory, it can be solved resorting to the following differential Riccati equation in the matrix variable $\mathbf{P}$:

$$\begin{cases} -\dot{\mathbf{P}} = \bar{\mathbf{A}}^\top \mathbf{P} + \mathbf{P}\bar{\mathbf{A}} - \mathbf{K}_o^\top \mathbf{R}_o \mathbf{K}_o + \mathbf{Q}_o, \quad \mathbf{P}(T) = \mathbf{P}_1 \\ \mathbf{K}_o = \mathbf{R}_o^{-1} \left(\mathbf{S}_o^\top + \bar{\mathbf{B}}^\top \mathbf{P} \right) \end{cases} \tag{4.24}$$

4.3.2 Application of PRONTO to solve the OIFT problem

The application of PRONTO to the OIFT problem allows to compute a complete trajectory $\boldsymbol{\xi}_{t+1} = (\mathbf{x}_{t+1}(\cdot), \mathbf{u}_{t+1}(\cdot))$ satisfying (4.6). Indeed, as mentioned before, the definitions in Sec. 4.2 are related to PRONTO variables and parameters and, by choosing in this framework $f(\mathbf{x}, \mathbf{u}) = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$, the following list of relations can be obtained:

$$\bar{\mathbf{A}}(\tau) = f_{\mathbf{x}} = \mathbf{A} \tag{4.25}$$

$$\bar{\mathbf{B}}(\tau) = f_{\mathbf{u}} = \mathbf{B} \tag{4.26}$$

$$\mathbf{a}(\tau) = l_{\mathbf{x}}^\top = \mathbf{C}^\top \mathbf{Q}_{c,\dot{c}} \left(\mathbf{x}_c - \mathbf{x}_{c,des} \right) + \nabla_{\mathbf{x}} l_d^{fo} + \nabla_{\mathbf{x}} l^{al} \tag{4.27}$$

$$\mathbf{b}(\tau) = l_{\mathbf{u}}^\top = \mathbf{R}\mathbf{u} \tag{4.28}$$

$$\mathbf{r}_1(T) = m_{\mathbf{x}}^\top = \mathbf{C}^\top \mathbf{Q}_{c,\dot{c}} \left(\mathbf{x}_c(T) - \mathbf{x}_{c,des}(T) \right) + \nabla_{\mathbf{x}} l_d^{fo}(T) + \nabla_{\mathbf{x}} l^{al}(T) \tag{4.29}$$

$$\mathbf{Q}_o(\tau) = l_{\mathbf{xx}} = \mathbf{C}^\top \mathbf{Q}_{c,\dot{c}} \mathbf{C} + \mathcal{H}_{\mathbf{xx}} l_d^{fo} + \mathcal{H}_{\mathbf{xx}} l^{al} \tag{4.30}$$

$$\mathbf{S}_o(\tau) = l_{\mathbf{xu}} = \mathbf{Z}_{2N \times N} \tag{4.31}$$

$$\mathbf{R}_o(\tau) = l_{\mathbf{uu}} = \mathbf{R} \tag{4.32}$$

$$\mathbf{P}_1(T) = m_{\mathbf{xx}} = \mathbf{C}^\top \mathbf{Q}_{c,\dot{c}} \mathbf{C} + \mathcal{H}_{\mathbf{xx}} l_d^{fo}(T) + \mathcal{H}_{\mathbf{xx}} l^{al}(T). \tag{4.33}$$

where $\mathbf{Q}_{c,\dot{c}} := \sum_{i=1}^n \mathbf{Q}_{c,\dot{c},i} \in \mathbb{R}^{2M \times 2M}$, $\mathbf{R} := \text{Diag}\{\mathbf{R}_1 \dots \mathbf{R}_n\} \in \mathbb{R}^{N \times N}$ and symbols ∇_* and $\mathcal{H}_{**}$ indicate standard gradient and Hessian operators w.r.t. $*$.

Therefore, a time-invariant PD controller as

$$\mathbf{K} = \begin{bmatrix} k_p \mathbf{I}_N & k_v \mathbf{I}_N \end{bmatrix}, \quad k_p, k_v > 0 \tag{4.34}$$

is an immediate and efficient option to choose, meaning that step 2 in Alg. 7 is not required to be processed. Moreover, let us analyze the gradient and Hessian matrix of l_d^{fo} w.r.t. the state $\mathbf{x}$ in (4.27) and (4.30) respectively. Indicating the zero vector of

dimension $\natural$ with $\mathbf{0}_\natural$ and the relative position displacements $\mathbf{e}_{ij} := \mathbf{p}_i - \mathbf{p}_j$ for all pairs (i, j), their formal expressions are provided by

$$\nabla_{\mathbf{x}} l_d^{fo} = \begin{bmatrix} \nabla_{\mathbf{p}}^\top l_d^{fo} & \nabla_{\dot{\mathbf{p}}}^\top l_d^{fo} \end{bmatrix}^\top = \begin{bmatrix} \nabla_{\mathbf{p}}^\top l_d^{fo} & \mathbf{0}_N^\top \end{bmatrix}^\top \tag{4.35}$$

$$\mathcal{H}_{\mathbf{xx}} l_d^{fo} = \begin{bmatrix} \mathcal{H}_{\mathbf{pp}} l_d^{fo} & \mathcal{H}_{\mathbf{p}\dot{\mathbf{p}}} l_d^{fo} \\ \mathcal{H}_{\dot{\mathbf{p}}\mathbf{p}} l_d^{fo} & \mathcal{H}_{\dot{\mathbf{p}}\dot{\mathbf{p}}} l_d^{fo} \end{bmatrix} = \begin{bmatrix} \mathcal{H}_{\mathbf{pp}} l_d^{fo} & \mathbf{Z}_N \\ \mathbf{Z}_N & \mathbf{Z}_N \end{bmatrix} \tag{4.36}$$

where, by assigning $\mathbf{\Pi}_{ij} := \mathbf{e}_{ij}\mathbf{e}_{ij}^\top \in \mathbb{R}^{M\times M}$, it holds that

$$\nabla_{\mathbf{p}_i} l_d^{fo}(\tau) = k_F \sum_{\forall j \neq i} \sigma'_{d_{ij}}\left(s_{ij}(\tau)\right) \mathbf{e}_{ij}(\tau) \tag{4.37}$$

and, for all $i \neq j$,

$$\mathcal{H}_{\mathbf{p}_i\mathbf{p}_j} l_d^{fo}(\tau) = -k_F \left[2\sigma''_{d_{ij}}\left(s_{ij}(\tau)\right) \mathbf{\Pi}_{ij}(\tau) + \sigma'_{d_{ij}}\left(s_{ij}(\tau)\right) \mathbf{I}_M\right]. \tag{4.38}$$

Remarkably, the choice of maintaining $\sigma_{d_{ij}} \in \mathscr{C}^2$ is required to use PRONTO. In addition, an expression for the diagonal blocks in the Hessian $\mathcal{H}_{\mathbf{pp}} l_d^{fo}$ can be provided by

$$\mathcal{H}_{\mathbf{p}_i\mathbf{p}_i} l_d^{fo}(\tau) = -\sum_{\forall j \neq i} \mathcal{H}_{\mathbf{p}_i\mathbf{p}_j} l_d^{fo}(\tau). \tag{4.39}$$

Similarly, the gradient and Hessian matrix of the velocity alignment term l^{al} are yielded by

$$\nabla_{\mathbf{x}} l^{al} = \begin{bmatrix} \nabla_{\mathbf{p}}^\top l^{al} & \nabla_{\dot{\mathbf{p}}}^\top l^{al} \end{bmatrix}^\top = \begin{bmatrix} \mathbf{0}_N^\top & \nabla_{\dot{\mathbf{p}}}^\top l^{al} \end{bmatrix}^\top \tag{4.40}$$

$$\mathcal{H}_{\mathbf{xx}} l^{al} = \begin{bmatrix} \mathcal{H}_{\mathbf{pp}} l^{al} & \mathcal{H}_{\mathbf{p}\dot{\mathbf{p}}} l^{al} \\ \mathcal{H}_{\dot{\mathbf{p}}\mathbf{p}} l^{al} & \mathcal{H}_{\dot{\mathbf{p}}\dot{\mathbf{p}}} l^{al} \end{bmatrix} = \begin{bmatrix} \mathbf{Z}_N & \mathbf{Z}_N \\ \mathbf{Z}_N & \mathcal{H}_{\dot{\mathbf{p}}\dot{\mathbf{p}}} l^{al} \end{bmatrix} \tag{4.41}$$

where

$$\nabla_{\mathbf{p}_i} l^{al}(\tau) = k_A \sum_{\forall j \neq i} q_{A_{ij}} \dot{\mathbf{e}}_{ij}(\tau); \tag{4.42}$$

$$\mathcal{H}_{\dot{\mathbf{p}}_i\dot{\mathbf{p}}_j} l^{al}(\tau) = \begin{cases} -k_A q_{A_{ij}}, & \text{if } i \neq j \\ k_A \sum\limits_{\forall j \neq i} q_{A_{ij}}, & \text{otherwise} \end{cases}. \tag{4.43}$$

As a matter of fact, the LQ problem in (4.24) needs for a positive semidefinite matrix $\mathbf{Q}_o$ to be solved; hence, care has to be taken while search directions in Alg. 7 are selected. In reality, it can be observed that expression (4.30) does not always verify this condition, given the general undetermined definiteness of the Hessian related to the formation $\mathcal{H}_{\mathbf{pp}} l_d^{fo}$. To this aim, a "safe version" of $\mathbf{Q}_o$, say $\mathbf{Q}_o^{safe}$, is implemented by exploiting a Gerschgorin-circle-theorem-based heuristic (see also Sec. B.2 for the statement of the Gerschgorin circle theorem) as in Alg. 8, i.e. by computing for $i \neq j$ the off-diagonal blocks[4]

$$\mathcal{H}^{safe}_{\mathbf{p}_i\mathbf{p}_j} l_d^{fo}(\tau) = -k_F \left[2\sigma''_{d_{ij}}\left(s_{ij}(\tau)\right) \mathbf{\Pi}_{ij}(\tau) + \chi_{>0}(\sigma'_{d_{ij}}\left(s_{ij}(\tau)\right))\sigma'_{d_{ij}}\left(s_{ij}(\tau)\right) \mathbf{I}_M\right], \tag{4.44}$$

[4]The diagonal blocks for the "safe version" are obtained exploiting (4.39).

where $\chi_{>0}$ denotes the classic unitary step function centered in 0.

Algorithm 8 Heuristic for the modification of $\mathcal{H}_{\mathbf{pp}} l_d^{fo}$ to search effectively a descent direction in PRONTO

for $i = 1, ..., n$ **do**
  $\mathcal{H}_{\mathbf{p}_i\mathbf{p}_i} l_d^{fo} \leftarrow \mathbf{Z}_M$
  for $j = 1, ..., n$ such that $j \neq i$ **do**
    $\mathcal{H}_{\mathbf{p}_i\mathbf{p}_j} l_d^{fo} \leftarrow -2k_F \sigma''_{d_{ij}}(s_{ij})\, \mathbf{\Pi}_{ij}$
    if $\mathbf{Q}_o^{safe}$ is not required **or** $\sigma'_{d_{ij}}(s_{ij}) > 0$ **then**
      $\mathcal{H}_{\mathbf{p}_i\mathbf{p}_j} l_d^{fo} \leftarrow \mathcal{H}_{\mathbf{p}_i\mathbf{p}_j} l_d^{fo} - k_F \sigma'_{d_{ij}}(s_{ij})\, \mathbf{I}_M$
    end if
    $\mathcal{H}_{\mathbf{p}_i\mathbf{p}_i} l_d^{fo} \leftarrow \mathcal{H}_{\mathbf{p}_i\mathbf{p}_i} l_d^{fo} - \mathcal{H}_{\mathbf{p}_i\mathbf{p}_j} l_d^{fo}$
  end for
end for

4.3.3 OIFT solution via distributed control law

The OIFT problem can be, in principle, solved even in a distributed way and online. In order to design a distributed control law, it is drawn inspiration from the Calculus of Variations (see Sec. B.5). Specifically, it is sought the expression of a feedback PD controller $\mathbf{u}(\tau) = \mathbf{u}(\mathbf{x}(\tau))$ in which each term represents an error to achieve tracking, formation or alignment.
Firstly, let $h_l := h - m$ be the objective functional to be minimized, such that

$$h_l(\mathbf{x}(\cdot), \mathbf{u}(\cdot)) = \int_0^T l\left(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau\right) d\tau. \tag{4.45}$$

It is assumed that the extrema for functions $\mathbf{x}(\cdot)$ and $\mathbf{u}(\cdot)$ are somehow fixed, since $\mathbf{x}(0)$ is specified, $\mathbf{x}(T)$ belongs to the manifold determined by final cost $m(\mathbf{x}(T))$, $\mathbf{u}(0)$ depends on $\mathbf{x}(0)$ and $\mathbf{u}(T) = \mathbf{0}_N$ is expected. It is worth to notice that instantaneous cost $l\left(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau\right)$ in (4.45) can be decomposed agent-wise as

$$l\left(\mathbf{x}(\tau), \mathbf{u}(\tau), \tau\right) = \sum_{i=1}^{n} l_i\left(\mathbf{x}(\tau), \mathbf{u}(\tau)\right), \tag{4.46}$$

and, according to decomposition (4.46), for the i-th agent, one has

$$l_i = \frac{1}{2}\left\|\mathbf{x}_c - \mathbf{x}_{c,des}\right\|^2_{\mathbf{Q}_{c,\dot{c},i}} + \frac{1}{2}\left\|\mathbf{u}_i\right\|^2_{\mathbf{R}_i} + \frac{k_F}{4}\sum_{\forall j \neq i} \sigma_{d_{ij}}(s_{ij}) + \frac{k_A}{4}\sum_{\forall j \neq i}\left\|\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_j\right\|^2_{q_{A_{ij}}}. \tag{4.47}$$

With this premise, the following Thm. 4.3.1 is stated in order to design a proper PD controller to drive the group of agents.

Theorem 4.3.1.
Let us define $\overline{\mathbf{Q}}_c := \sum_{j=1}^n \mathbf{Q}_{c,j}/n$, $\overline{\mathbf{Q}}_{\dot{c}} := \sum_{j=1}^n \mathbf{Q}_{\dot{c},j}/n$, *with* $\overline{\mathbf{Q}}_{\dot{c}}$ *non-singular. Assuming to adopt a distributed PD controller* $\mathbf{u} = \begin{bmatrix}\mathbf{u}_1^\top & \cdots & \mathbf{u}_n^\top\end{bmatrix}^\top$ *in order to govern the dynamics of system* (4.3)*, the objective in* (4.45) *is stationary with the following distributed control law:*

$$\mathbf{u}_i = -\mathbf{R}_i^{-1}\left[k_{P,i}^{tr}\overline{\mathbf{Q}}_c(\mathbf{p}_c - \mathbf{p}_{c,des}) + k_{D,i}^{tr}\overline{\mathbf{Q}}_{\dot{c}}(\dot{\mathbf{p}}_c - \dot{\mathbf{p}}_{c,des})\right] \tag{4.48}$$

$$-\mathbf{R}_i^{-1}\left[k_{P,i}^{fo}k_F\sum_{j\in\mathcal{N}_i}\sigma'_{d_{ij}}(s_{ij})\mathbf{e}_{ij} + k_{D,i}^{al}k_A\sum_{j\in\mathcal{N}_i}q_{Aij}\dot{\mathbf{e}}_{ij}\right] \tag{4.49}$$

$$-\mathbf{R}_i^{-1}k_{D,i}^{fo}k_F\sum_{j\in\mathcal{N}_i}\left[2\sigma''_{d_{ij}}(s_{ij})\mathbf{\Pi}_{ij} + \chi_{>0}(\sigma'_{d_{ij}}(s_{ij}))\sigma'_{d_{ij}}(s_{ij})\mathbf{I}_M\right]\dot{\mathbf{e}}_{ij}; \tag{4.50}$$

where $\mathcal{N}_i$ is the neighborhood of the i-th agent, i.e. the set of nodes that have established a communication with i, and $k_{P,i}^{tr}$, $k_{D,i}^{tr}$, $k_{P,i}^{fo}$, $k_{D,i}^{fo}$, $k_{D,i}^{al}$ are positive tunable gains.

Proof. Recalling that the system under analysis is controlled through the acceleration quantities, that is $\mathbf{u} = \ddot{\mathbf{p}}$, the i-th equation obtained by the Fundamental Lemma B.5.16 of CV applied to (4.45) is written in a distributed fashion:

$$\sum_{j=1}^{n}\left(-\frac{d^2}{dt^2}\frac{\partial l_j(\mathbf{x},\mathbf{u})}{\partial\ddot{\mathbf{p}}_i} + \frac{d}{dt}\frac{\partial l_j(\mathbf{x},\mathbf{u})}{\partial\dot{\mathbf{p}}_i} - \frac{\partial l_j(\mathbf{x},\mathbf{u})}{\partial\mathbf{p}_i}\right) = \mathbf{0}_M, \quad i = 1,\dots,n. \tag{4.51}$$

Let us observe that the i-th equation in (4.51) can be rewritten as

$$\sum_{j=1}^{n}\left[\frac{\mathbf{Q}_{\dot{c},j}}{n}(\ddot{\mathbf{p}}_c - \ddot{\mathbf{p}}_{c,des}) - \frac{\mathbf{Q}_{c,j}}{n}(\mathbf{p}_c - \mathbf{p}_{c,des})\right] + \sum_{\forall j\neq i}\left[k_A q_{A_{ij}}\ddot{\mathbf{e}}_{ij} - k_F\sigma'_{d_{ij}}(s_{ij})\mathbf{e}_{ij}\right] = \mathbf{R}_i\ddot{\mathbf{u}}_i. \tag{4.52}$$

Since the information on the centroid $\mathbf{x}_c$ is not directly available, its computation is allowed via an average consensus procedure with Metropolis-Hastings weights Garin and Schenato (2010). The same strategy can be employed to compute terms $\sum_{j=1}^{n}\mathbf{Q}_{c,j}/n$ and $\sum_{j=1}^{n}\mathbf{Q}_{\dot{c},j}/n$ in (4.52), since they represent two average weights. That said, let us denote the estimates of $\sum_{j=1}^{n}\mathbf{Q}_{c,j}/n$ and $\sum_{j=1}^{n}\mathbf{Q}_{\dot{c},j}/n$ obtained via the consensus with $\overline{\mathbf{Q}}_c$ and $\overline{\mathbf{Q}}_{\dot{c}}$ respectively. Now, in order to design a proper PD controller, it is needed an explicit linear expression of $\mathbf{u}$ depending on position and velocity errors only. Hence, to satisfy the first-order optimality condition in (4.52), it is reasonable, due to its structure, to assume the centroid error acceleration and the relative[5] accelerations to be governed by the following decoupled dynamics[6]

$$(\ddot{\mathbf{p}}_c - \ddot{\mathbf{p}}_{c,des}) = -k_{P1,i}^{tr}\overline{\mathbf{Q}}_{\dot{c}}^{-1}\overline{\mathbf{Q}}_c(\mathbf{p}_c - \mathbf{p}_{c,des}) - k_{D1,i}^{tr}(\dot{\mathbf{p}}_c - \dot{\mathbf{p}}_{c,des}); \tag{4.53}$$

$$\ddot{\mathbf{e}}_{ij} = -\frac{k_{P1,i}^{fo}k_F}{k_A q_{A_{ij}}}\sigma'_{d_{ij}}(s_{ij})\mathbf{e}_{ij} - k_{D1,i}^{al}\dot{\mathbf{e}}_{ij} \tag{4.54}$$

$$-\frac{k_{D1,i}^{fo}k_F}{k_A q_{A_{ij}}}\left[2\sigma''_{d_{ij}}(s_{ij})\mathbf{\Pi}_{ij} + \chi_{>0}(\sigma'_{d_{ij}}(s_{ij}))\sigma'_{d_{ij}}(s_{ij})\mathbf{I}_M\right]\dot{\mathbf{e}}_{ij}; \tag{4.55}$$

where $k_{P1,i}^{tr}$, $k_{D1,i}^{tr}$, $k_{P1,i}^{fo}$, $k_{D1,i}^{fo}$, $k_{D1,i}^{al}$ are positive constants for the PD controller under construction. It is worth to note that the last term of (4.55) accounts for the fact that also the second order time derivative of the formation term l^{fo} affects each mutual

[5] Whenever a couple of agents (i,j) exchanges information.

[6] This is an important assumption since it allows for a decoupling of the system's dynamics. Also it is worth to observe that, summing up all the relative accelerations in (4.54)-(4.55), one obtains the null vector.

acceleration $\ddot{\mathbf{e}}_{ij}$ between agents; thus, the same switching-based heuristics used in (4.44) is adopted even in this framework not to contrast the alignment task. The regulation in (4.54) and (4.55) allows us to state that the dynamics of input $\mathbf{u}_i$ is exponentially decaying to $\mathbf{0}_M$, as constants $k^{tr}_{D1,i}$ and $k^{al}_{D1,i}$ can be tuned large enough such that both the dynamics of $(\ddot{\mathbf{p}}_c - \ddot{\mathbf{p}}_{c,des})$ and $\ddot{\mathbf{e}}_{ij}$ be exponentially and asymptotically stable. Thus, it is also possible to assume that[7]

$$\ddot{\mathbf{u}}_i = \mathbf{K}_{u,i}\mathbf{u}_i, \quad i = 1, \dots, n \tag{4.56}$$

where constants $\mathbf{K}_{u,i} = k_{u,i}\mathbf{I}_M \in \mathbb{R}^{M\times M}$, $k_{u,i} > 0$, are adopted. As a remark, assumptions (4.54), (4.55) and (4.56) are suitable choices in order to obtain distributed a PD controller[8]; nonetheless, they have the drawback to add new constraints to which the control law has to abide. Finally, let us define the positive tunable gains for the PD controller as $k^{tr}_{P,i} := (1 + k^{tr}_{P1,i})/k_{u,i}$, $k^{tr}_{D,i} := k^{tr}_{D1,i}/k_{u,i}$, $k^{fo}_{P,i} := (1 + k^{fo}_{P1,i})/k_{u,i}$, $k^{fo}_{D,i} := k^{fo}_{D1,i}/k_{u,i}$, $k^{al}_{D,i} := k^{al}_{D1,i}/k_{u,i}$. Taking into consideration the proposed online average consensus approach to estimate global quantities, the distributed control law in (4.48) for the i-th agent can be designed exploiting (4.54)-(4.56) starting from relation (4.52). ■

Remark 4.3.2. Law (4.48) does not guarantee the minimization of (4.45). To prove this fact, the second order variation of (4.45) has to be analyzed and it has to be shown that this is a positive definite quadratic form for the input in (4.48).

In what follows, the derivation of the control law found in Thm. 4.3.1 is motivated.

4.3.4 Analysis of the equilibria

In this paragraph, the equilibria for the centralized and distributed solutions developed so far are investigated. To do so, it is accounted for a graph-based formalism that leverages concepts as incidence and rigidity matrices[9] (see also Sec. B.4).
Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of vertices ($n = |\mathcal{V}|$ in total) and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is the set of edges ($E = |\mathcal{E}|$ in total), the incidence matrix $\mathbf{E} \in \mathbb{R}^{n\times E}$ is defined as

$$[\mathbf{E}]_{ij} = \begin{cases} -1, & \text{the } j\text{-th edge sinks at node } i; \\ 1, & \text{the } j\text{-th edge leaves node } i; \\ 0, & \text{otherwise;} \end{cases} \tag{4.57}$$

and, consequently, the Laplacian matrix $\mathbf{L}$ associated to $\mathcal{G}$ can be defined as $\mathbf{L} = \mathbf{E}\mathbf{E}^\top$ (see also Subsec. B.3.2). With the same ordering for the edges of $\mathcal{E}$ given to the columns of $\mathbf{E}$, the rigidity functions $r_\mathcal{G} : \mathbb{R}^N \to \mathbb{R}^E$ associated to the frameworks $(\mathcal{G}, \mathbf{p})$ and $(\mathcal{G}, \dot{\mathbf{p}})$ are respectively given by

[7] This assumption is reasonable because (i) the second derivative of an exponential function is proportional (with a positive constant) to the function itself; (ii) the desired trajectory is expressed until its second derivative only; (iii) by imposing $\ddot{\mathbf{p}}_{des} = \mathbf{0}_N$, one seeks minimum energy solutions, since $\mathbf{u}_{des} = \ddot{\mathbf{p}}_{des}$. In principle, to be rigorous, (4.56) should be proven a posteriori.

[8] Choices justified by the fact that in the PRONTO optimization framework a PD controller can be employed successfully.

[9] For a detailed treatise on this, the reader is addressed to Oh et al. (2015) and Zhao and Zelazo (2016).

$$r_{\mathcal{G}}(\mathbf{p}) = \frac{1}{2}\left[\ldots, \mathbf{e}_{ij}^{\top}\mathbf{e}_{ij}, \ldots\right]^{\top}, \quad (i,j) \in \mathcal{E}; \tag{4.58}$$

$$r_{\mathcal{G}}(\dot{\mathbf{p}}) = \frac{1}{2}\left[\ldots, \dot{\mathbf{e}}_{ij}^{\top}\dot{\mathbf{e}}_{ij}, \ldots\right]^{\top}, \quad (i,j) \in \mathcal{E}. \tag{4.59}$$

One useful tool to characterize the rigidity property of a framework is the rigidity matrix $\boldsymbol{\mathcal{R}} \in \mathbb{R}^{E\times N}$, which is defined as the Jacobian of the rigidity function w.r.t. the spatial quantity (either $\mathbf{p}$ or $\dot{\mathbf{p}}$) considered in the relative framework. Let us also define $\mathrm{Diag}_{\mathcal{E}}$ as the operator that maps the set of displacement vectors $\mathbf{e}_{ij}^{\top}$ or $\dot{\mathbf{e}}_{ij}^{\top}$ ordered as in (B.34)-(4.59) into a diagonal matrix of dimensions $E \times N$, such that its k-th diagonal element of dimensions $1 \times M$ is exactly given by the k-th displacement vector, for $k = 1, \ldots, E$. Then, the rigidity matrices for frameworks $(\mathcal{G}, \mathbf{p})$ and $(\mathcal{G}, \dot{\mathbf{p}})$ are yielded by

$$\boldsymbol{\mathcal{R}}(\mathbf{p}) = \frac{\partial r_{\mathcal{G}}(\mathbf{p})}{\partial \mathbf{p}} = \mathrm{Diag}_{\mathcal{E}}\left\{\mathbf{e}_{ij}^{\top}\right\}\left(\mathbf{E}^{\top} \otimes \mathbf{I}_M\right), \quad \text{for } (\mathcal{G}, \mathbf{p}); \tag{4.60}$$

$$\boldsymbol{\mathcal{R}}(\dot{\mathbf{p}}) = \frac{\partial r_{\mathcal{G}}(\dot{\mathbf{p}})}{\partial \dot{\mathbf{p}}} = \mathrm{Diag}_{\mathcal{E}}\left\{\dot{\mathbf{e}}_{ij}^{\top}\right\}\left(\mathbf{E}^{\top} \otimes \mathbf{I}_M\right), \quad \text{for } (\mathcal{G}, \dot{\mathbf{p}}); \tag{4.61}$$

where symbol $\otimes$ indicates the Kronecker product operator[10].

Now, with these graph-based theoretical tools, it is possible to provide a different point of view for the expression characterizing gradients in (4.37), (4.42) and Hessian matrices in (4.39), (4.38), (4.43), offering a perspective from the networked control of the system. Indeed, fixing an order for the edges in $\mathcal{E}$, it is worth to note that one has

$$\nabla_{\mathbf{p}} l_d^{fo} = k_F \boldsymbol{\mathcal{R}}(\mathbf{p})^{\top} \sigma_d'(\mathbf{s}); \tag{4.62}$$

$$\mathcal{H}_{\mathbf{pp}} l_d^{fo} = k_F \left[\boldsymbol{\mathcal{R}}(\mathbf{p})^{\top} \mathrm{Diag}(2\sigma_d''(\mathbf{s}))\boldsymbol{\mathcal{R}}(\mathbf{p}) + \mathbf{E}\mathrm{Diag}(\sigma_d'(\mathbf{s}))\mathbf{E}^{\top} \otimes \mathbf{I}_M\right]; \tag{4.63}$$

$$\nabla_{\dot{\mathbf{p}}} l = k_A \boldsymbol{\mathcal{R}}(\dot{\mathbf{p}})^{\top} \mathbf{q}_A; \tag{4.64}$$

$$\mathcal{H}_{\dot{\mathbf{p}}\dot{\mathbf{p}}} l^{al} = k_A \mathbf{E}\mathrm{Diag}(\mathbf{q}_A)\mathbf{E}^{\top} \otimes \mathbf{I}_M; \tag{4.65}$$

where vector $\mathbf{s} \in \mathbb{R}^E$ contains all the terms s_{ij} in the order adopted for the edges whenever $(i,j) \in \mathcal{E}$, function σ_d and its derivatives are applied component-wise to $\mathbf{s}$, vector $\mathbf{q}_A \in \mathbb{R}^E$ contains all the terms $\mathbf{q}_{A_{ij}}$ in the order adopted for the edges whenever $(i,j) \in \mathcal{E}$ and $\mathrm{Diag}(\cdot)$ is the operator that maps a vector argument into its correspondent diagonal matrix.

The equilibria relative to the formation task are given by gradient in (4.62); in particular, three conditions can be distinguished:

(i) $\sigma_d'(\mathbf{s}) = \mathbf{0}_E$, i.e. $s_{ij} = d_{ij}^2$ for all $(i,j) \in \mathcal{E}$;

(ii) $\sigma_d'(\mathbf{s}) \in \ker\{\mathrm{Diag}_{\mathcal{E}}\{\mathbf{e}_{ij}\}\}$;

(iii) $\mathrm{Diag}_{\mathcal{E}}\{\mathbf{e}_{ij}\}\,\sigma_d'(\mathbf{s}) \in \ker\{\mathbf{E} \otimes \mathbf{I}_M\}$;

where $\ker\{\cdot\}$ denotes the kernel of a matrix and $\mathrm{Diag}_{\mathcal{E}}\{\mathbf{e}_{ij}\} = \mathrm{Diag}_{\mathcal{E}}\left\{\mathbf{e}_{ij}^{\top}\right\}^{\top}$. The first condition is exactly what the system should satisfy to accomplish the desired formation and represents a stable equilibrium, since this would make the corresponding instantaneous cost l_d^{fo} vanishing[11], being part of a minimum energy configuration. The

[10] It is assumed that the Kronecker product has a lower priority with respect to the matrix multiplication.
[11] Provided that the required geometric shape be feasible.

second condition arises whenever an undesired consensus takes place, e.g. a rendezvous, a collinear positioning or the achievement of an equivalent formation due to the lack of global rigidity. Finally, the third condition means that $\mathrm{Diag}_{\mathcal{E}}\{\mathbf{e}_{ij}\}\,\sigma'_d(\mathbf{s})$ is a signed path vector Mesbahi and Egerstedt (2010) associated to one of the cycles in $\mathcal{G}$. It is plausible that all the equilibria described by conditions (ii) and (iii) are unstable in case of global rigidity guarantees, due to the fact that $\sigma'_d(s_{ij})$ is monotonic and $\sigma'_d(s_{ij}) = 0$ holds only for $s_{ij} = d_{ij}^2$.
Differently, the equilibria relative to the alignment task are given by gradient in (4.64). Here, two conditions can be distinguished:

(i) $\mathbf{q}_A \in \ker\{\mathrm{Diag}_{\mathcal{E}}\{\dot{\mathbf{e}}_{ij}\}\}$, i.e. $\dot{\mathbf{e}}_{ij} = \mathbf{0}_M$ for all $(i,j) \in \mathcal{E}$;

(ii) $\mathrm{Diag}_{\mathcal{E}}\{\dot{\mathbf{e}}_{ij}\}\,\mathbf{q}_A \in \ker\{\mathbf{E} \otimes \mathbf{I}_M\}$;

The first condition leads to the fulfillment of the alignment, since it is verified if and only if the $\ker\{\mathrm{Diag}_{\mathcal{E}}\{\dot{\mathbf{e}}_{ij}\}\}$ reaches its maximum dimension. Whereas, the second condition is not only satisfied under the latter assumption, but even when $\mathrm{Diag}_{\mathcal{E}}\{\dot{\mathbf{e}}_{ij}\}\,\mathbf{q}_A$ is a signed path vector relative to one of the cycles in $\mathcal{G}$. Then, an alignment equilibrium is stable only if the first condition is verified, since this would make the relative instantaneous cost l^{al} vanishing, being part of a minimum energy configuration for the system too.

The tracking task also exhibits equilibria for $\mathbf{C}^\top \mathbf{Q}_{c,\dot{c}}\,(\mathbf{x}_c - \mathbf{x}_{c,des}) = \mathbf{0}_{2M}$, condition that can be easily separated into

$$\begin{cases} (\mathbf{p}_c - \mathbf{p}_{c,des}) \in \ker\left\{\overline{\mathbf{Q}}_c\right\}; \\ (\dot{\mathbf{p}}_c - \dot{\mathbf{p}}_{c,des}) \in \ker\left\{\overline{\mathbf{Q}}_{\dot{c}}\right\}; \end{cases} \tag{4.66}$$

and, trivially, (4.66) describes stable equilibria, since the corresponding tracking cost l^{tr}, which is a quadratic form weighted by positive semidefinite matrices $\overline{\mathbf{Q}}_c$ and $\overline{\mathbf{Q}}_{\dot{c}}$, vanishes when these conditions hold. In particular, one observes that equalities $\mathbf{p}_c = \mathbf{p}_{c,des}$ and $\dot{\mathbf{p}}_c = \dot{\mathbf{p}}_{c,des}$ must hold at the equilibrium, if these average weights are chosen to be positive definite.

Let us denote $\mathbb{1}_\natural \in \mathbb{R}^\natural$ as the vector in which all its $\natural$ components are equal to 1. In order to inspect the equilibria of the system governed by the distributed control law devised previously, its matrix representation is provided:

$$\begin{aligned} \mathbf{u} = &- \mathbf{R}^{-1}\left(\mathrm{Diag}_{\mathcal{V}}\left\{k_{P,i}^{tr}\right\} \otimes \mathbf{I}_M\right)\left(\mathbb{1}_n \otimes \overline{\mathbf{Q}}_c(\mathbf{p}_c - \mathbf{p}_{c,des})\right) + \\ &- \mathbf{R}^{-1}\left(\mathrm{Diag}_{\mathcal{V}}\left\{k_{D,i}^{tr}\right\} \otimes \mathbf{I}_M\right)\left(\mathbb{1}_n \otimes \overline{\mathbf{Q}}_{\dot{c}}(\dot{\mathbf{p}}_c - \dot{\mathbf{p}}_{c,des})\right) + \\ &- \mathbf{R}^{-1}\left[\left(\mathrm{Diag}_{\mathcal{V}}\left\{k_{P,i}^{fo}\right\} \otimes \mathbf{I}_M\right)\nabla_{\mathbf{p}} l_d^{fo} + \left(\mathrm{Diag}_{\mathcal{V}}\left\{k_{P,i}^{al}\right\} \otimes \mathbf{I}_M\right)\nabla_{\dot{\mathbf{p}}} l^{al}\right] + \\ &- \mathbf{R}^{-1}\left(\mathrm{Diag}_{\mathcal{V}}\left\{k_{D,i}^{fo}\right\} \otimes \mathbf{I}_M\right)(\mathbf{E} \otimes \mathbf{I}_M)\mathrm{Diag}_{\mathcal{E}}\left\{\mathcal{H}^{safe}_{\mathbf{p}_i\mathbf{p}_j}\dot{\mathbf{e}}_{ij}\right\}\mathbb{1}_E \end{aligned} \tag{4.67}$$

where $\mathrm{Diag}_{\mathcal{V}}$ is the operator that maps its arguments indexed by $i = 1, \ldots, n$ into a diagonal matrix of dimensions $n \times n$, such that the i-th diagonal element coincides with the i-th argument, and the expressions of gradients $\nabla_{\mathbf{p}} l_d^{fo}$, $\nabla_{\dot{\mathbf{p}}} l^{al}$ are provided in (4.62) and (4.64) respectively. It is worth to notice that the last term of (4.67) has a similar structure of a rigidity matrix in which the mutual velocities $\dot{\mathbf{e}}_{ij}$ are weighted by the "safe version" of the Hessian $\mathcal{H}_{\mathbf{pp}}$ computed in Alg. 8. This allows to take into account a

second order curvature effect to better fulfill both the formation and alignment tasks. Moreover, it is remarkable to observe that the equilibrium relations discussed so far represent a sufficient condition to obtain $\mathbf{u} = \mathbf{0}_N$ as $\tau \to T$ and thus the control aims at minimizing the global energy spent by the system of agents. Lastly, it is worth to mention that control laws similar to (4.67), leading a group of agents to be seen as dissipative second order Hamiltonian system, ensure local stability for the formations to be attained in an infinite time horizon Mesbahi and Egerstedt (2010); Oh and Ahn (2014). Moreover, closely related control laws working with finite-time specifications for second-order multi-agent systems have been already developed in Fu, Wang, and Wang (2019); Sun, Mou, Deghat, Anderson, and Morse (2014), supporting the robustness of the approach devised in this chapter.
To sum up, Thm. 4.3.1 allows to devise the online control in (4.67) without providing enough information about the stability properties concerning the formation aspect. Nonetheless, this derivation follows the principle of passivity while designing a coordination law for the group of agent in question Arcak (2007). Therefore, it can be concluded the generic convergence to the desired formation from all initial conditions except for those that lie on the stable manifolds of the unstable equilibria. This issue can be partially avoided assuming global rigidity (or, at least, infinitesimal rigidity, if equivalent – but not congruent – frameworks are accepted for the system of agents).

4.4 Numerical results

In this section, the focus is put on a singular case study[12] where desired distances d_{ij} are imposed such that the formation assumes a cubic shape in a space of dimension $M = 3$. Precisely, the desired cube has side equal to $d = 5$ m and each of the $n = 8$ agents have a neighborhood with degree $|\mathcal{N}_i| = 5$, $i = 1, \dots, n$. These choices allow to present an example where the formation to be achieved is globally rigid, thus ensuring the system to converge towards stable equilibria. Also, the group of agents have to track the straight line $\mathbf{\Gamma}(\tau) = \mathrm{v}(\tau, \tau, \tau)$ with a constant desired velocity $\mathrm{v} = 1$ ms^{-1} for each component. The initial state of the system is assigned as follows:

$$\mathbf{p}(0) = \left(\begin{bmatrix} -6 \\ 3 \\ 24 \end{bmatrix}, \begin{bmatrix} -9 \\ -3 \\ 3 \end{bmatrix}, \begin{bmatrix} 6 \\ -6 \\ 15 \end{bmatrix}, \begin{bmatrix} 15 \\ 3 \\ 15 \end{bmatrix}, \begin{bmatrix} -3 \\ -18 \\ 6 \end{bmatrix}, \begin{bmatrix} 6 \\ 6 \\ 6 \end{bmatrix}, \begin{bmatrix} -3 \\ -15 \\ -3 \end{bmatrix}, \begin{bmatrix} 15 \\ 6 \\ 9 \end{bmatrix} \right) \text{m};$$

$$\dot{\mathbf{p}}(0) = \left(\begin{bmatrix} 10 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 10 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 15 \\ -5 \\ 5 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 5 \end{bmatrix}, \begin{bmatrix} 5 \\ 5 \\ 5 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} -10 \\ -25 \\ 10 \end{bmatrix} \right) \text{ms}^{-1};$$

The input weighting matrix in (5.1) is selected as $\mathbf{R} = r_a \mathbf{I}_N$, with $r_a = 1$ m^{-2}s^4, in (4.11), and keep the integration time $T = 20$ s fixed for all simulations, as well as the structure of the output weighting matrices

[12] Further simulation examples involving unfeasible formation configurations can be found in Fabris et al. (2019a).

$$\mathbf{Q}_{c,c,i} = \begin{bmatrix} q_p \mathbf{I}_M & \mathbf{Z}_M \\ \mathbf{Z}_M & q_v \mathbf{I}_M \end{bmatrix}, \quad i = 1, \dots, n,$$

for (4.10), where $q_p = 1.25\ \mathrm{m}^{-2}$, $q_v = 0.125\ \mathrm{m}^{-2}\mathrm{s}^2$ are nonnegative constants. Moreover, it is set, for all edges $(i,j) \in \mathcal{E}$, the following parameters: $k_{r_{ij}} = 250$, $k_{a_{ij}} = 100$, $\beta_{ij} = 3$, $\alpha_{a_{ij}} = 1/2$, in (4.15), $k_F = 2$, in (4.12) and $k_A = 0.25$, $q_{A_{ij}} = 1\ \mathrm{m}^{-2}\mathrm{s}^2$ if $(i,j) \in \mathcal{E}$, $q_{A_{ij}} = 0$ $(i,j) \notin \mathcal{E}$, in (4.13). In each numerical simulation presented, it has been decided to stop the execution of Alg. 7 by taking two actions: the algorithm can stop at the iteration $t \leq MaxIter$ if, given a threshold $\varepsilon_g > 0$, the inequality $-Dg\left(\boldsymbol{\xi}_t\right) \cdot \boldsymbol{\zeta}_t < \varepsilon_g$ is verified. In particular, it is chosen $\varepsilon_g = 10^{-8}$. In addition, an interruption occurs when the preset maximum number of iterations $MaxIter = 80$ is exceeded.
Constant gains for the PD controller in (4.34) are selected leveraging the linear dynamics of (4.3), as

$$\begin{cases} \omega_n = 3\ \mathrm{rad\,s}^{-1} \\ \xi = 0.7 \end{cases} \Rightarrow \begin{cases} k_p = \omega_n^2 \\ k_v = 2\xi\omega_n \end{cases}.$$

Whereas, for the distributed PD controller in (4.48) it has been chosen gains $k_{P,i}^{tr} = 0.8\ \mathrm{s}^2$, $k_{D,i}^{tr} = 11.2\ \mathrm{s}^2$, $k_{P,i}^{fo} = 1.3\ \mathrm{s}^2$, $k_{D,i}^{fo} = 1\ \mathrm{s}^2$, $k_{D,i}^{al} = 0.3\ \mathrm{s}^2$ with a trial and error procedure.

4.4.1 Performances of the devised distributed PD controller

With the purpose of validating the distributed PD controller in (4.48), the two dynamics obtained by the latter control law and the offline inverse dynamics provided by PRONTO are compared. As a criterion, it is decided to look at the global energy spent by the input and the settling time of the system dynamics. Specifically, it is accounted for the global input energy $l^{in}(\cdot)$ and the average input energy. This quantity is defined as the weighted mean $\bar{l}^{in}(\cdot) := l^{in}(\cdot)/\left\|\mathbf{R}\right\|_F$, where $\left\|\mathbf{R}\right\|_F^2 = \sum_{i=1}^n \left\|\mathbf{R}_i\right\|_F^2$ and $\left\|\mathbf{R}_i\right\|_F = \sqrt{\mathrm{tr}(\mathbf{R}_i\mathbf{R}_i^\top)}$ has to be considered the dimensionless version of the Frobenius norm of $\mathbf{R}_i$ used to compute $l^{in}(\cdot)$, for $i = 1, \dots, n$. Furthermore, selecting $\eta \in [0,1]$, the settling time instant $\tau_\eta \in [0,T]$ for the system is established whenever condition $|l^{st}(\tau)| \leq \eta$ holds for all $\tau \geq \tau_\eta$, in which function $l^{st}(\tau) : \mathbb{R} \to [-1,1]$ is defined[13] as

$$l^{st} := \begin{cases} \dfrac{l^{fo} + l^{al} - l^{tr}}{l^{fo} + l^{al} + l^{tr}}, & \text{if } l^{fo} + l^{al} + l^{tr} \neq 0; \\ 0, & \text{otherwise.} \end{cases} \tag{4.68}$$

It is worth to note that function in (4.68) measures a weighted trade-off between relative-related and centroid-related control performances, as it is expected to go to 0 and remain null over the time if a control action is being applied.

In Fig. 4.3 the performances of these numerical simulations are illustrated, presenting the inverse dynamics obtained by PRONTO on the left column and the online dynamics governed by the distributed PD controller on the right column. In Figs. 4.3(a) and 4.3(b) one can observe that, in both cases, the positional trajectories described by the system in the three-dimensional space, noting that each agent contributes eventually to achieve a proper position as a vertex of the desired cubic shape. Moreover, the system's

[13] This definition must be revisited when constraints (4.14) describe a unfeasible shape.

centroid tracks the required straight line $\mathbf{\Gamma}(\cdot)$. However, it is worth to note that the trajectories of the system governed by the distributed control law are less smooth w.r.t. those of PRONTO. This indicates that more effort is needed to steer the agents using the distributed control (4.48); indeed, Fig. 4.3(d) depicts a higher energy consumption levels over the time w.r.t. to that in Fig. 4.3(c). It can be observed that the input energy spent with the distributed approach is approximately $n = 8$ times greater than that spent in the PRONTO framework: this fact might suggest that the suboptimality[14] degree of controller (4.48) could be quantified by the number of driven agents, provided that the performances related to time appear fairly similar. Figs. 4.3(e) and 4.3(f) show that this is the case for the numerical results here presented, since each settling time instant can be compared among the two approaches with a maximum deviation of about 2 s and occurs before final time T. Also, the latter fact proves that law in 4.48 works as a finite-time controller according to the choice of using function l^{st} as a reference for settling time, in this specific example.

Remarkably, three important evidences can be highlighted:

(i) the distributed control law in 4.48 allows to govern the state dynamics of the system similarly to what PRONTO algorithm produces as optimal inverse dynamics;

(ii) the distributed control law in 4.48 is suboptimal w.r.t. the inverse dynamics computed numerically, since it requires the system to spend more input energy, probably scaling proportionally with the number of agents, if time performances are comparable;

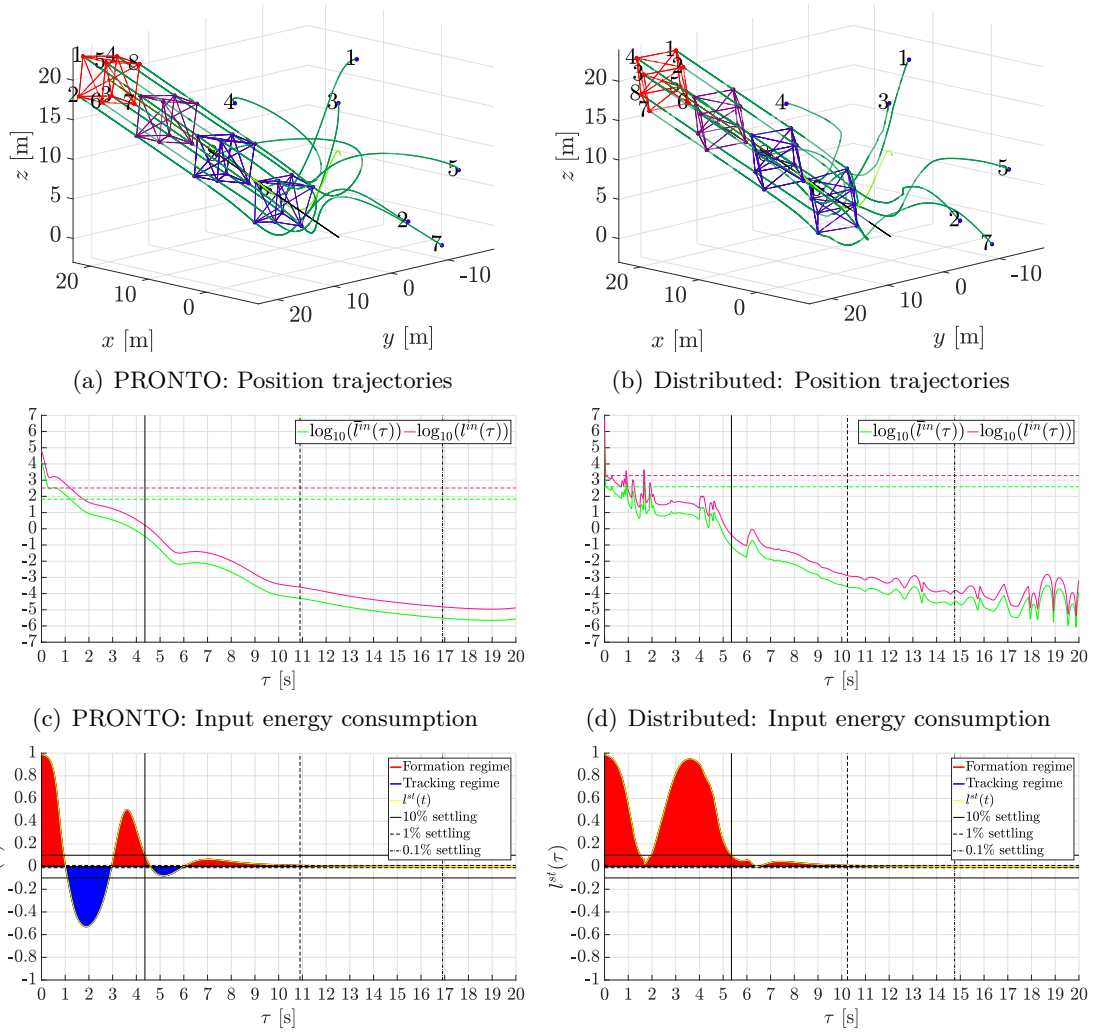
(iii) for global rigid geometrical shapes the convergence seems to be attained in a finite time, provided that gains of the distributed PD controller are properly tuned.

4.4.2 Distributed control law as an agreement protocol

As shown in (4.54) and (4.55), the two control actions – relative distance and controid regulations – can be analyzed separately. Moreover, control law in (4.67) points out that the relative distance regulation acts exploiting the potentials in (4.15) as distance-based error functions and the alignment term, indicating how these control actions can be seen as agreement protocols (see also Subsec. 2.1.3).

Fig. (4.4) provides an overview on the whole convergent dynamics of the system governed by the distributed PD controller. Firstly, Figs. (4.4(a)), (4.4(c)) and (4.4(e)) exhibit the convergence to zero of the distance errors depicted in Fig. 4.2 as $\tau \to T = 20$ s, aiming at the achievement of the formation. The same happens in Fig. 4.4(b) for the relative velocities responsible for the alignment task. Lastly, Figs. 4.4(d) and 4.4(f) depict the response of positions and velocities of the centroid, showing that the tracking task is fulfilled as required.
In each of these diagrams, it is worth to notice that the 1% settling time instant computed through function l^{st} gives significant information about the convergence of the state towards all the specifications. Indeed, reasonably, this allow us to state that the entire dynamics controlled by law in (4.67) practically attains a stable equilibrium after $T/2 = 10$ s, for $\tau > \tau_{1\%}$.

[14] w.r.t. to the inverse dynamics obtained with PRONTO



(a) PRONTO: Position trajectories

(b) Distributed: Position trajectories

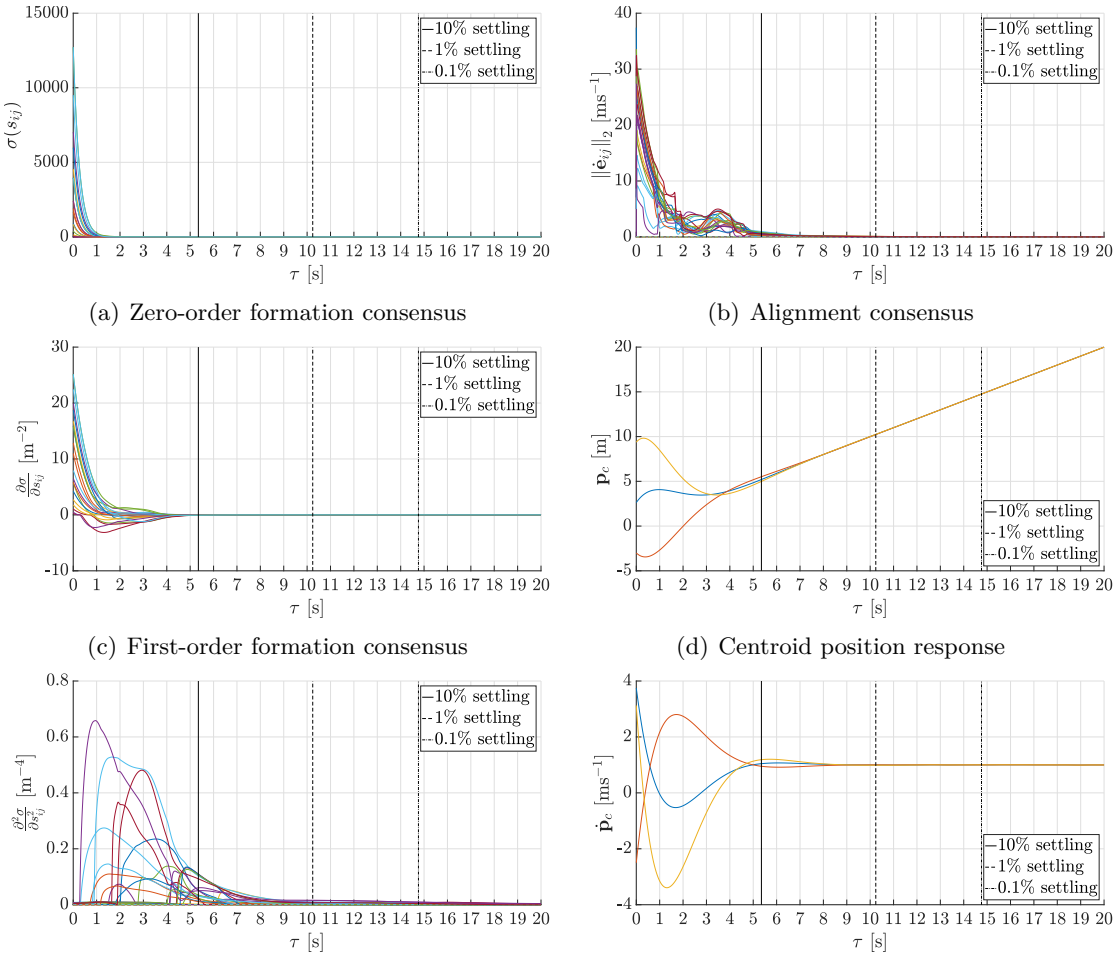
(c) PRONTO: Input energy consumption

(d) Distributed: Input energy consumption

(e) PRONTO: Settling time

(f) Distributed: Settling time

Figure 4.3. Comparison between the offline solution provided by PRONTO and the system dynamics governed by the distributed controller. (a)-(b): trajectories of the positions, in dark green; trajectory of the centroid position, in light green; desired straight path, in black; cubic shape formation, depicted progressively with ordered shades blue-indigo-purple-red, beginning with blue. (c)-(d): total input energy, in magenta, and average input energy, in green, spent by the system; dashed lines show their time-averaged values. (e)-(f): settling times at 10%, 1%, 0.1%, (i.e. $\eta = 0.1, 0.01, 0.001$ respectively) depicted by the vertical lines, and established evaluating a quantitative weighted trade-off between formation/alignment and tracking instantaneous costs.



(a) Zero-order formation consensus

(b) Alignment consensus

(c) First-order formation consensus

(d) Centroid position response

(e) Second-order formation consensus

(f) Centroid velocity response

Figure 4.4. Distributed control law seen as agreement protocol. (a)-(c)-(e): distance-based potential error functions and its derivatives converging to 0, guaranteeing agents to reach the required geometrical shape. (b): relative velicities converging to 0, ensuring the agents to follow the desired path with straight trajectories. (d)-(f): centroid positions and velocities (first, second and third components in blue, orange and yellow respectively) converging to the given specifications.

4.5 Chapter summary

In this chapter, it has been introduced a complete version of the Optimal Time-Invariant Formation Tracking problem whose minimum-energy solutions can be obtained by the numerical tool PRONTO (offline) and a distributed control law (online). In order to fulfill the required final configurations for the formations to be steered, generic potential functions have been proposed. Specifically, simulations show the validity of the online PD controller devised w.r.t. the trajectory and the energy profiles yielded by PRONTO, which have been considered as a reference for the distributed control law. A short yet detailed analysis of the equilibria has also been provided in order to give an overview on the working principles of developed input law. Numerical results also illustrate the versatility of this paradigm, since general communication constraints on the information sharing among agents have been adopted during the whole dissertation.

5

Distributed State Estimation from Relative Measurements

"Comprehension is not a requisite of cooperation."
Councillor Hamann

Contents

The contents of this chapter are partly available in:
Fabris et al. (2019b). "*On the Distributed Estimation from Relative Measurements: a Graph-Based Convergence Analysis*". In 18th European Control Conference 2019, Napoli, Italy, pp. 1550-1555, Jun 25-28, 2019.

5.1 Overview

In the latest years, MASs have received much attention as the number of distributed energy components and devices continues to increase globally Xie and Liu (2017) and, thanks to their versatile graph-based paradigm, their usage has considerably become interconnected with relevant leading areas in research nowadays, such as Artificial Intelligence Rocha and Kudenko (2019), Deep Learning Chen, Ding, Mao, Liu, and Hou (2019), Internet of Things Fouad and Moskowitz (2019) and Intelligent Sensing Putra, Trilaksono, Riyansyah, Laila, Harsoyo, and Kistijantoro (2019).
One of the main challenging aspects of the embedded multi-agent systems consists in the wise management and exploitation of the local information since they are typically involved in the solution of large-scale logically/spatially distributed optimization problems. Along this line, *system state estimation* constitutes a popular task in several control and learning smart networks applications, requiring to estimate the state of each agent through the adoption of a distributed paradigm exploiting local information and interactions. Remarkably, meaningful ongoing efforts have been directed towards the accomplishment of distributed state estimation Dehghanpour, Wang, Wang, Yuan, and Bu (2019) in relevant novel fields as networked microgrids Cintuglu and Ishchenko (2019) and global navigation satellite systems Khodabandeh and Teunissen (2019). Many algorithms for this peculiar purpose have been designed over the years for sundry kinds of sensor networks Dong, Wang, Alsaadi, and Ahmad (2015); Olfati-Saber (2007); Xu, Lu, Shi, Li, and Xie (2018); Yong, Liyi, Jianfeng, Zhe, and Yi (2019), exploiting methodologies that mainly focus on consensus-based agreement protocols Chen, Yin, Zhou, Wang, Wang, and Chen (2018); Jalalmaab, Pirani, Fidan, and Jeon (2019); Soatti, Nicoli, Savazzi, and Spagnolini (2017) and the Alternating Direction Method of Multipliers (ADMM) Du, Li, Li, Chen, Fei, and Wu (2019); Zhang, Li, Wu, and Zhou (2019a), especially in concurrence with the resolution of least squares (LS) problems Aster, Borchers, and Thurber (2019).

To conclude this heading, Fig. 5.1 depicts the main features arising in this work. The principal methods employed resort on Optimization Theory, Combinatorial Graph Theory, Dynamic Systems and Control, Distributed Optimization and Distributed Estimation. Leveraging the latter tools, the main objectives for this study are represented by the design of distributed algorithms for networked estimations from relative measurements and their performance analysis and comparison.

5.1.1 Problem statement

According to the model introduced in Subsec. 2.1.1, let us consider a multi-agent system composed of n devices whose interactions are represented by an undirected and connected graph $\mathcal{G}$. It is assumed that the i-th agent in the network is characterized by a multi-dimensional attribute $\mathbf{x}_i \in \mathbb{R}^M$ (the *i-th agent state*) corresponding to a physical quantity, and by a set of noisy relative measurements of the same quantity, namely $\{\tilde{\mathbf{x}}_{ij} \in \mathbb{R}^M,\ v_j \in \mathcal{N}_i\}$ where $\tilde{\mathbf{x}}_{ij}$ represents a noisy measurement of the difference between $\mathbf{x}_j$ and $\mathbf{x}_i$. The single components of $\mathbf{x}_i$ and $\tilde{\mathbf{x}}_{ij}$ are denoted by $x_{i,\mathrm{l}}$ and $\tilde{x}_{ij,\mathrm{l}}$ respectively, with $\mathrm{l} = 1 \dots M$.

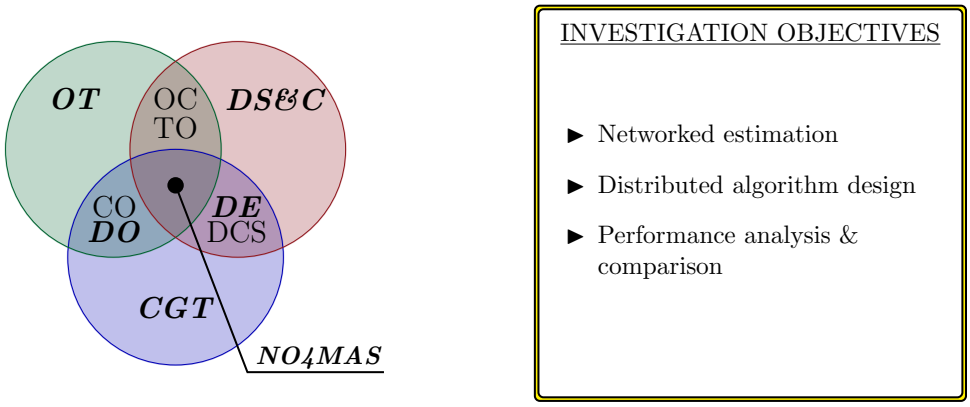


Figure 5.1. Theoretical fields and investigation objectives arising from this study.

In this chapter, the focus is directed on the problem of network state determination, i.e., the estimation of the set $\{\mathbf{x}_1^\star, \dots, \mathbf{x}_n^\star\}$ that allows to best approximate (and be consistent with) the set of the existing measurements. Formally, the major aim consists in solving the following convex minimization problem:

Problem 5.1.1. *Given the set of noisy relative measurements* $(\tilde{\mathbf{x}}_{ij}, \tilde{\mathbf{x}}_{ji})$ *for all* i, j*, where* $\tilde{\mathbf{x}}_{ij}$ *is not necessarily equal to* $\tilde{\mathbf{x}}_{ji}$*), find a solution to*

$$\underset{\{\mathbf{x}_1,\dots,\mathbf{x}_n\}}{\arg\min} \frac{1}{2} \sum_{v_i \in \mathcal{V}} \sum_{v_j \in \mathcal{N}_i} (\mathbf{x}_i - \mathbf{x}_j + \tilde{\mathbf{x}}_{ij})^\top (\mathbf{x}_i - \mathbf{x}_j + \tilde{\mathbf{x}}_{ij}). \tag{5.1}$$

Setting $N := nM$, the cost function in (5.1), hereafter denoted by $h := h(\mathbf{x})$, where $\mathbf{x} = \begin{bmatrix} \mathbf{x}_1^\top & \dots & \mathbf{x}_n^\top \end{bmatrix}^\top \in \mathbb{R}^N$.

5.1.2 Related works

Several existing works explore the properties of the distributed state estimation solution depending on the graph modeling the network in terms of available measurements. In Barooah and Hespanha (2005); Barooah and Hespanha (2007) the distributed estimation from relative measurements begun to be taken into account significantly. Whereas, in Rossi, Frasca, and Fagnani (2017), the estimation task is studied supposing the availability of both absolute and relative measurements and proving that the underlying system topology determines whether the error of the LS optimal estimator decreases to zero as the number of unknown variables grows to infinity. Similarly, the authors of Barooah and Hespanha (2009) investigates how the variance of the estimation error of a node variable grows with the distance of the node to an arbitrary reference one, providing then a graphs classification according to their sparsity. In Carron, Todescato, Carli, and Schenato (2014) the problem of optimal estimating the position of each agent in a network from relative noisy distances is solved through a consensus-based algorithm whose rate of convergence is determined for regular graphs such as Cayley, Ramanujan, and complete graphs. Also, in Zelazo, Franchi, Bülthoff, and Giordano (2015), the decentralized rigidity maintenance control with range measurements for multi-robot systems is tackled and, recently, authors of Ravazzi, Chan, and Frasca (2019) studied the distributed estimation from relative measurements of heterogeneous and uncertain quality.

5.1.3 Contribution and outline of the chapter

The aim of this chapter is the presentation and comparison of three iterative linear algorithms for the distributed state estimation from relative measurements (RM) in a MAS modeled as in Subsec. 2.1.1. In particular, the consensus-like iterative scheme Σ_η devised in Fabris et al. (2019b) and the new equivalent schemes (referred to Σ_ρ and Σ_ϵ in the sequel) are here proposed and formalized in their multivariate version. In addition, common features and notable distinctions are highlighted through a topological analysis. Finally, since robustness frequently represents an important issue for linear multi-agent protocols Xue, Wu, and Yuan (2019), a sensitivity analysis Hammond and Axelrod (2006) is proposed in order to discern which iterative procedure is more appropriate as their tunable regularizers η, ρ and ϵ change.

The remainder of the chapter is organized as follows. In Sec. 5.2, the distributed schemes Σ_η, Σ_ρ and Σ_ϵ that yield the resolution of Problem 5.1.1 are discussed in details, focusing on their convergence properties towards the centralized solution – as the network topology varies – in the subsequent Sec. 5.3. In Sec. 5.4, the distributed approaches for estimation are compared by means of a sensitivity analysis. In Sec. 5.5, numerical simulations to support the theoretical results are illustrated. Lastly, Sec. 5.6 concludes the dissertation summarizing this chapter.

5.2 Solutions for the state estimation from RM

To solve the state estimation problem defined in 5.1, in the following paragraphs different approaches, both centralized and decentralized, are presented. In particular, a discussion devoted to the different distributed iterative schemes is yielded as a starting point for a comparison between this kind of algorithms.

5.2.1 Centralized vs Distributed Solution

In order to derive a centralized solution to minimization problem in (5.1), let us account for the gradient of h whose i-th component is given by

$$\nabla_{\mathbf{x}_i} h = 2\deg(v_i)\mathbf{x}_i - 2\sum_{v_j \in \mathcal{N}_i} \mathbf{x}_j - \sum_{v_j \in \mathcal{N}_i} (\tilde{\mathbf{x}}_{ji} - \tilde{\mathbf{x}}_{ij}). \tag{5.2}$$

such that, indicating with $\mathbf{0}_N$ the null vector of dimension N, condition

$$\nabla_{\mathbf{x}} h = \begin{bmatrix} \nabla_{\mathbf{x}_1} h^\top & \cdots & \nabla_{\mathbf{x}_n} h^\top \end{bmatrix}^\top = \mathbf{0}_N \tag{5.3}$$

be satisfied. Indicating with $\mathbf{x}^{\mathrm{l}} = \begin{bmatrix} x_{1,\mathrm{l}} & \cdots & x_{n,\mathrm{l}} \end{bmatrix}^\top \in \mathbb{R}^n$, it is worth to note that equation (5.3) can be rewritten as

$$\nabla_{\mathbf{x}^{\mathrm{l}}} h = \mathbf{0}_n, \qquad \text{for } \mathrm{l} = 1, \dots, M \tag{5.4}$$

since components of (5.2) can be mutually decoupled. As a consequence, conditions (5.4) yields the following system of equations

$$2\mathbf{L}\mathbf{x}^{\mathrm{l}} = \tilde{\mathbf{x}}^{\mathrm{l}}, \qquad \text{for } \mathrm{l} = 1, \dots, M, \tag{5.5}$$

where $\tilde{\mathbf{x}}^{\mathrm{l}} = \left[\sum_{v_j \in \mathcal{N}_1} (\tilde{x}_{j1,\mathrm{l}} - \tilde{x}_{1j,\mathrm{l}}) \cdots \sum_{v_j \in \mathcal{N}_n} (\tilde{x}_{jn,\mathrm{l}} - \tilde{x}_{nj,\mathrm{l}})\right]^\top \in \mathbb{R}^n$. Hence, the next lemma can be straightforwardly obtained.

Lemma 5.2.1 (Centralized solution)**.**
Given a n-agents network associated to the graph $\mathcal{G}$, if the relative noisy measurements are such that $\tilde{\mathbf{x}}^{\mathrm{l}} \notin \ker(\mathbf{L}) \setminus \{\mathbf{0}_n\}$ for $\mathrm{l} = 1, \dots, M$, the minimum norm solution of (5.1) is given by

$$(\mathbf{x}^\star)^{\mathrm{l}} = \frac{1}{2}\mathbf{L}^\dagger \tilde{\mathbf{x}}^{\mathrm{l}}, \qquad \textit{for } \mathrm{l} = 1 \dots M, \tag{5.6}$$

where $\mathbf{L}^\dagger \in \mathbb{R}^{n \times n}$ is the pseudo-inverse of the Laplacian matrix related to the graph $\mathcal{G}$.

Relation (5.6) represents the *centralized* solution to the optimization problem (5.1) and it only requires to be aware of the network topology and the noisy relative measurements. On the other hand, each equation of system (5.5) refers to a single agent information, in terms of local topology and measurements, thus suggesting a *distributed* approach. Indeed, imposing $\nabla_{x_{i,\mathrm{l}}} h = 0$, from (5.2) one has

$$\sum_{v_j \in \mathcal{N}_i} (x_{j,\mathrm{l}} - x_{i,\mathrm{l}}) + \frac{1}{2} \sum_{v_j \in \mathcal{N}_i} (\tilde{x}_{ji,\mathrm{l}} - \tilde{x}_{ij,\mathrm{l}}) = 0, \qquad \text{for } \mathrm{l} = 1, \dots, M, \tag{5.7}$$

or, equivalently

$$x_{i,\mathrm{l}} = \frac{1}{\deg(v_i)} \left(\sum_{v_j \in \mathcal{N}_i} x_{j,\mathrm{l}} + \tfrac{1}{2} \sum_{v_j \in \mathcal{N}_i} (\tilde{x}_{ji,\mathrm{l}} - \tilde{x}_{ij,\mathrm{l}}) \right), \qquad \text{for } \mathrm{l} = 1, \dots, M, \tag{5.8}$$

i.e., the i-th node state depends exclusively on the states of its neighbors and on its set of relative measurements. In the following paragraphs, three consensus procedures, that can be classified as variants of the descent iterative method (see Subsec. 2.2.4), are characterized according to the different categories given in Olfati-Saber, Fax, and Murray (2007).

5.2.2 Development of distributed iterative schemes

In the next paragraphs, four distributed iterative schemes (Σ_η, Σ_ρ, Σ_ϵ and a particular case of them, Σ_0) deriving from the distributed solution (5.7) are introduced and briefly discussed.

Distributed iterative schemes Σ_0 and Σ_η

By considering the l-th state vector $\mathbf{x}^{\mathrm{l}}$, a discrete time system Σ_0^{l} of the form

$$\Sigma_0^{\mathrm{l}}: \quad \mathbf{x}^{\mathrm{l}}(t+1) = \mathbf{F}_0 \mathbf{x}^{\mathrm{l}}(t) + \mathbf{u}_0^{\mathrm{l}} \tag{5.9}$$

can be derived from (5.8) as an update rule driven by the input measurements. According to (5.9) the i-th state at the t-th step affects the neighbors estimates at the $(t+1)$-th step, but it is not considered for the recursive self-estimate. The state matrix $\mathbf{F}_0 \in \mathbb{R}^{n\times n}$ is indeed equal to the adjacency matrix $\mathbf{A}$ normalized by the node degrees and it is thus a row-stochastic matrix. Formally, it occurs that

$$[\mathbf{F}_0]_{ij} = \frac{1}{\deg(v_i)}\,[\mathbf{A}]_{ij} \quad \Rightarrow \quad \mathbf{F}_0 = \mathbf{D}^{-1}\mathbf{A}. \tag{5.10}$$

Similarly, the l-th input $\mathbf{u}_0^{\mathrm{l}} \in \mathbb{R}^n$ in (5.9) is given by the vector of the normalized relative measurements

$$\mathbf{u}_0^{\mathrm{l}} = \frac{1}{2}\mathbf{D}^{-1}\tilde{\mathbf{x}}^{\mathrm{l}}. \tag{5.11}$$

From Landau and Odlyzko (1981), it is possible to observe that if the graph $\mathcal{G}$ representing the given system is connected, then the state matrix $\mathbf{F}_0$ has n real eigenvalues $\lambda_0^{\mathbf{F}_0} \geq \cdots \geq \lambda_{n-1}^{\mathbf{F}_0}$ in the range $[-1,1]$ with $\lambda_0^{\mathbf{F}_0} = 1$ having single algebraic multiplicity. Because of this spectral distribution, (5.10) exhibits stability issues that can be overcome with a regularized distributed model that is constructed by introducing some memory in the system and adopting (5.9) to provide only a weighted correction to current estimate. This leads to

$$\begin{aligned}\Sigma_\eta^{\mathrm{l}}: \quad \mathbf{x}^{\mathrm{l}}(t+1) &= \eta\mathbf{x}^{\mathrm{l}}(t) + (1-\eta)\left(\mathbf{F}_0\mathbf{x}^{\mathrm{l}}(t) + \mathbf{u}_0^{\mathrm{l}}\right)\\ &= \left(\eta\mathbf{I}_n + (1-\eta)\mathbf{F}_0\right)\mathbf{x}^{\mathrm{l}}(t) + (1-\eta)\mathbf{u}_0^{\mathrm{l}}\\ &= \mathbf{F}_\eta\mathbf{x}^{\mathrm{l}}(t) + \mathbf{u}_\eta^{\mathrm{l}},\end{aligned} \tag{5.12}$$

where $\eta \in [0,1)$ and $\mathbf{I}_n$ is the identity matrix of dimension n, such that $\Sigma_\eta^{\mathrm{l}} = \Sigma_0^{\mathrm{l}}$ if and only if $\eta = 0$. It is immediate to see that this approach considers a convex combination of the result obtained in (5.8) between the i-th state and the rest of the states and RM, i.e.

$$x_{i,\mathrm{l}} = \eta x_{i,\mathrm{l}} + \frac{1-\eta}{\deg(v_i)}\left(\sum_{v_j\in\mathcal{N}_i} x_{j,\mathrm{l}} + \tfrac{1}{2}\sum_{v_j\in\mathcal{N}_i}(\tilde{x}_{ji,\mathrm{l}} - \tilde{x}_{ij,\mathrm{l}})\right), \qquad \text{for } \mathrm{l} = 1,\ldots,M. \tag{5.13}$$

The state matrix $\mathbf{F}_\eta \in \mathbb{R}^{n\times n}$ is still row-stochastic but with eigenvalues $\lambda_0^{\mathbf{F}_\eta} \geq \cdots \geq \lambda_{n-1}^{\mathbf{F}_\eta}$ in the range $(-1+2\eta, 1] \subseteq (-1,1]$, as $\eta \in (0,1)$. In particular, one can observe that, exploiting the linearity of the spectrum, eigenvalues of $\mathbf{F}_\eta$ can be characterized as $\lambda_i^{\mathbf{F}_\eta} = \eta + (1-\eta)\lambda_i^{\mathbf{F}_0}$, $i = 0,\ldots,n-1$, and hence they preserve the same order of the spectrum of $\mathbf{F}_0$.

Distributed iterative scheme Σ_ρ

Exploiting the contents of Subsec. 2.2.4, another distributed solution can be found resting upon an approach based on the so-called Proximal Point (PP) algorithm. Let us define the regularized PP objective function

$$h_\rho(\mathbf{x}^{\mathrm{l}}, \mathbf{x}^{\mathrm{l}}(t)) := h(\mathbf{x}^{\mathrm{l}}) + \frac{\rho}{2}\sum_{\mathrm{l}=1}^{M}\left\|\mathbf{x}^{\mathrm{l}} - \mathbf{x}^{\mathrm{l}}(t)\right\|_2^2, \qquad \text{for } \mathrm{l} = 1,\ldots,M \tag{5.14}$$

with $\rho \geq 0$, such that if $\rho = 0$ then $h_0 = h$. One observes that gradient and Hessian of (5.14) w.r.t. $\mathbf{x}^{\mathrm{l}}$ are respectively[1]

$$\nabla_{\mathbf{x}^{\mathrm{l}}} h_\rho(\mathbf{x}^{\mathrm{l}}, \mathbf{x}^{\mathrm{l}}(t)) = \nabla_{\mathbf{x}^{\mathrm{l}}} h + \rho(\mathbf{x}^{\mathrm{l}} - \mathbf{x}^{\mathrm{l}}(t)), \quad t = 1, 2, \ldots \tag{5.15}$$

$$\mathcal{H}_{\mathbf{x}^{\mathrm{l}}\mathbf{x}^{\mathrm{l}}} h_\rho(\mathbf{x}^{\mathrm{l}}, \mathbf{x}^{\mathrm{l}}(t)) = 2\mathbf{L} + \rho \boldsymbol{I}_n \succ 0, \quad t = 1, 2, \ldots \tag{5.16}$$

Since Hessian in (5.16) is positive definite for all $\rho > 0$ and all $t = 1, 2, \ldots$ then cost function in (5.14) is strictly convex for these values of ρ and its minimization admits one solution only. Indeed, the latter can be provided by the optimality condition

$$\nabla_{\mathbf{x}^{\mathrm{l}}} h_\rho(\mathbf{x}^{\mathrm{l}}(t+1), \mathbf{x}^{\mathrm{l}}(t)) = \mathbf{0}_n, \tag{5.17}$$

which, for all $i = 1, \ldots, n$, yields

$$x_{i,\mathrm{l}}(t+1) = \frac{\rho x_{i,\mathrm{l}}(t) + 2\sum_{v_j \in \mathcal{N}_i} x_{j,\mathrm{l}}(t) + \sum_{v_j \in \mathcal{N}_i} (\tilde{x}_{ji,\mathrm{l}} - \tilde{x}_{ij,\mathrm{l}})}{2\mathrm{deg}(v_i) + \rho}, \quad \text{for } \mathrm{l} = 1, \ldots, M. \tag{5.18}$$

In (5.18), the i-th node state depends, as in (5.8), on the states of its neighbors and on its set of RM. However, the use of the penalty ρ affects the i-th node state with the previous information about itself. Indeed, it is remarkable to notice that (5.18) can be also obtained as an alternative version of (5.7) in which, for $\mathrm{l} = 1, \ldots, M$, it holds that

$$x_{i,\mathrm{l}} = \left(\mathrm{deg}(v_i) + \frac{\rho}{2}\right)^{-1} \left(\sum_{v_j \in \mathcal{N}_i} (x_{j,\mathrm{l}}) + \frac{\rho}{2}\right) + \frac{1}{2}\left(\mathrm{deg}(v_i) + \frac{\rho}{2}\right)^{-1} \sum_{v_j \in \mathcal{N}_i} (\tilde{x}_{ji,\mathrm{l}} - \tilde{x}_{ij,\mathrm{l}}). \tag{5.19}$$

By considering the state vector $\mathbf{x}^{\mathrm{l}}$, another discrete time system, namely Σ_ρ^{l}, can be derived from (5.18) as an update rule driven by the input measurements, whenever $\rho > 0$, for all $\mathrm{l} = 1, \ldots, M$:

$$\Sigma_\rho^{\mathrm{l}}: \quad \mathbf{x}^{\mathrm{l}}(t+1) = \mathbf{F}_\rho \mathbf{x}^{\mathrm{l}}(t) + \mathbf{u}_\rho^{\mathrm{l}}. \tag{5.20}$$

On the other hand, if and only if $\rho = 0$, $\Sigma_\rho = \Sigma_0$ is obtained (the same happens for scheme in (5.10)), as for the previous case in which $\eta = 0$. In model (5.20) the i-th state at step t affects the neighbors estimate at iteration $t+1$ and it is also taken into account for a recursive self-estimate. As a consequence, the state matrix $\mathbf{F}_\rho$ is equal to a regularized adjacency matrix normalized by the regularized node degrees

$$[\mathbf{F}_\rho]_{ij} = \frac{[\mathbf{A}]_{ij} + \frac{\rho}{2}[\mathbf{I}_n]_{ij}}{\mathrm{deg}(v_i) + \frac{\rho}{2}} \;\Rightarrow\; \mathbf{F}_\rho = \left(\mathbf{D} + \frac{\rho}{2}\mathbf{I}_n\right)^{-1} \left(\mathbf{A} + \frac{\rho}{2}\mathbf{I}_n\right) \tag{5.21}$$

and it is still a row-stochastic matrix. The structure of (5.21) suggests that this approach can be reformulated by adding a self-loop at each node whose value corresponds to $\rho/2$, uniformly. Analogously, the input $\mathbf{u}_\rho$ in (5.20) is given by the vector of the regularized

[1] Let $\boldsymbol{\Omega} \in \mathbb{R}^{\natural \times \natural}$ be a matrix. The symbology $\boldsymbol{\Omega} \succeq 0$ and $\boldsymbol{\Omega} \succ 0$ is used to denote that $\boldsymbol{\Omega}$ is positive semidefinite and positive definite respectively. Moreover, given two $\natural \times \natural$ matrices $\boldsymbol{\Omega}_1$ and $\boldsymbol{\Omega}_2$, the notation $\boldsymbol{\Omega}_1 \succeq \boldsymbol{\Omega}_2$ stands for $\boldsymbol{\Omega}_1 - \boldsymbol{\Omega}_2 \succeq 0$. See also Sec. B.2.

and normalized RM

$$\mathbf{u}_\rho^{\mathrm{l}} = \frac{1}{2}\left(\mathbf{D} + \frac{\rho}{2}\mathbf{I}_n\right)^{-1}\tilde{\mathbf{x}}^{\mathrm{l}} = \left(\mathbf{D} + \frac{\rho}{2}\mathbf{I}_n\right)^{-1}\mathbf{D}\mathbf{u}_0^{\mathrm{l}}. \tag{5.22}$$

Let d_M be the maximum degree in $\mathcal{G}$. If graph $\mathcal{G}$ is connected, $\mathbf{F}_\rho$ has n real eigenvalues $\lambda_0^{\mathbf{F}_\rho} \geq \cdots \geq \lambda_{n-1}^{\mathbf{F}_\rho}$ and they range in $((\rho - 2d_M)/(\rho + 2d_M), 1] \subseteq (-1, 1]$, as $\rho > 0$, with $\lambda_0^{\mathbf{F}_\rho} = 1$ always present with single algebraic multiplicity.

Distributed iterative scheme Σ_ϵ

In this final paragraph, the most famous choice for the consensus matrix to reach an agreement regarding the consensus-based techniques is mentioned. Let us start again considering (5.7). Choosing $\epsilon \in (0, 2/\rho_S(\mathbf{L}))$, where $\rho_S(\mathbf{L}) = \lambda_{n-1}^{\mathbf{L}} \in (0, 2d_M)$ is the spectral radius of the Laplacian matrix (see Sec. B.2), it is easy to show that

$$x_{i,\mathrm{l}} = x_{i,\mathrm{l}} - \epsilon \sum_{v_j \in \mathcal{N}_i} (x_{i,\mathrm{l}} - x_{j,\mathrm{l}}) + \frac{\epsilon}{2} \sum_{v_j \in \mathcal{N}_i} (\tilde{x}_{ji,\mathrm{l}} - \tilde{x}_{ij,\mathrm{l}}) \tag{5.23}$$

provides another type of recursion that can be exploited to estimation purposes. Clearly, relation (5.23) suggests an adoption of the classical scheme

$$\Sigma_\epsilon^{\mathrm{l}}: \quad \mathbf{x}^{\mathrm{l}}(t+1) = \mathbf{F}_\epsilon \mathbf{x}^{\mathrm{l}}(t) + \mathbf{u}_\epsilon^{\mathrm{l}} \tag{5.24}$$

where

$$[\mathbf{F}_\epsilon]_{ij} = \begin{cases} -\epsilon[\mathbf{L}]_{ij}, & \text{for } i \neq j \\ 1 - \epsilon[\mathbf{L}]_{ii}, & \text{otherwise} \end{cases} \quad \Rightarrow \quad \mathbf{F}_\epsilon = \mathbf{I}_n - \epsilon\mathbf{L}; \tag{5.25}$$

and

$$\mathbf{u}_\epsilon^{\mathrm{l}} = \frac{\epsilon}{2}\tilde{\mathbf{x}}^{\mathrm{l}} = \epsilon\mathbf{D}\mathbf{u}_0^{\mathrm{l}}. \tag{5.26}$$

System (5.24) is nothing but the nonnormalized version[2] of scheme Σ_η^{l} devised in (5.12) and therefore this approach can be seen as the choice of weighting each edge in the graph $\mathcal{G}$ with ϵ and the addition of a unitary-weight self-loop for each vertex.
If graph $\mathcal{G}$ is connected, $\mathbf{F}_\epsilon$ has n real eigenvalues $\lambda_0^{\mathbf{F}_\epsilon} \geq \cdots \geq \lambda_{n-1}^{\mathbf{F}_\epsilon}$ and they ranges in $(1 - 2\epsilon/\rho_S(\mathbf{L}), 1] \subseteq (-1, 1]$, as $\epsilon \in (0, 2/\rho_S(\mathbf{L}))$, with $\lambda_0^{\mathbf{F}_\epsilon} = 1$ always present with single algebraic multiplicity. Moreover, exploiting the linearity of the spectrum, it is recalled that $\lambda_i^{\mathbf{F}_\epsilon} = 1 - \epsilon\lambda_i^{\mathbf{L}}$, for $i = 0, \ldots, n-1$. Also, it is important to note that ϵ must be strictly smaller than $2/\rho_S(\mathbf{L})$, otherwise, it would not only cause matrix $\mathbf{F}_\epsilon$ to lose its nonnegativity, but also it would drive $\Sigma_\epsilon^{\mathrm{l}}$ to become unstable.

To conclude, it is worth to note that scheme $\Sigma_\epsilon^{\mathrm{l}}$ cannot be compared with Σ_0^{l} for small values of ϵ. On the contrary, this can be done as ϵ approaches $2/\rho_S(\mathbf{L})$ from the left, when the topology is bipartite. However, a full equivalence would hold only if $\mathcal{G}$ is also regular, as discussed in the incoming Subsec. 5.2.3 and, lately, in Subsec. 5.3.2.

[2] The asymmetric normalized Laplacian can be defined as $\mathbf{L}' = \mathbf{D}' - \mathbf{A}'$, where $\mathbf{D}' = \mathbf{I}_n$ and $\mathbf{A}' = \mathbf{D}^{-1}\mathbf{A}$, so that $\mathbf{I}_n - \epsilon\rho_S(\mathbf{L})\mathbf{L}'/2 = \mathbf{I}_n - \epsilon\rho_S(\mathbf{L})\mathbf{D}'/2 + \epsilon\rho_S(\mathbf{L})\mathbf{A}'/2 = \eta\mathbf{D}' + (1-\eta)\mathbf{A}' = \eta\mathbf{I}_n + (1-\eta)\mathbf{F}_0$, imposing $\eta = 1 - \epsilon\rho_S(\mathbf{L})/2$.

5.2.3 Equivalence condition for the distributed solutions

Let us assume that, w.l.o.g., $M = 1$: as it will be shown in 5.3.1, this hypothesis can be made since the distributed schemes devised in (5.12), (5.20) and (5.24) can be decoupled component-wise in the index l. Thus, let us omit the presence of index l in this paragraph.

In order to discover an insightful equivalence between schemes Σ_η, Σ_ρ and Σ_ϵ, equality

$$(1-\eta)\mathbf{D}^{-1} = \left(\mathbf{D} + \frac{\rho}{2}\mathbf{I}_n\right)^{-1} = \epsilon\mathbf{I}_n \tag{5.27}$$

can be derived imposing $\mathbf{u}_\eta = \mathbf{u}_\rho = \mathbf{u}_\epsilon$. It is crucial to note that (5.27) holds true if and only if

$$\mathbf{D} = \frac{\rho}{2}\frac{1-\eta}{\eta}\mathbf{I}_n = \frac{1-\eta}{\epsilon}\mathbf{I}_n = \frac{2-\rho\epsilon}{2\epsilon}\mathbf{I}_n = \mathrm{d}\mathbf{I}_n, \quad \mathrm{d} \in \mathbb{N} \setminus \{0\}\,. \tag{5.28}$$

Hence, one can observe that if $\mathcal{G}$ is d-regular then, by (5.28), relation

$$\mathrm{d} = \frac{\rho}{2}\frac{1-\eta}{\eta} = \frac{1-\eta}{\epsilon} = \frac{2-\rho\epsilon}{2\epsilon} \tag{5.29}$$

can be then used as a change of variables, allowing to delineate three bijective relationships among the different distributed schemes. Thus, in a d-regular framework, η can be chosen in function of ρ and vice-versa, for instance. Finally, by (5.28), the full equivalence for schemes Σ_η, Σ_ρ, and Σ_ϵ holds assuming of d-regularity; indeed, one has

$$\mathbf{F}_\eta = \mathbf{F}_\rho, \qquad \text{if} \;\; \mathrm{d} = \frac{\rho}{2}\frac{1-\eta}{\eta}; \tag{5.30}$$

$$\mathbf{F}_\eta = \mathbf{F}_\epsilon, \qquad \text{if} \;\; \mathrm{d} = \frac{1-\eta}{\epsilon}; \tag{5.31}$$

$$\mathbf{F}_\rho = \mathbf{F}_\epsilon, \qquad \text{if} \;\; \mathrm{d} = \frac{2-\rho\epsilon}{2\epsilon}. \tag{5.32}$$

Therefore, schemes Σ_η, Σ_ρ, and Σ_ϵ are exactly equivalent, meaning that there exists a bijective map to rewrite one scheme into another, if and only if (i) they share the same topological network, (ii) the topology is regular with common degree d and (iii) parameters η, ρ and ϵ range in the intervals $(0, 1)$, $(0, +\infty)$ and $(0, 2/\rho_S(\mathbf{L}))$ respectively.

5.3 Convergence analysis of the distributed solutions

This section is crucial for the devised distributed solutions, since a large variety of properties are proven to give guarantees of convergence from different points of view. In the first paragraph the convergence of the distributed solutions toward the centralized solution is taken into consideration. Then, the central paragraphs show the convergence of the estimates for a large number of iterations. Finally, a comparison concerning convergence properties among the different schemes is proposed.

5.3.1 Convergence toward the centralized solution

In this paragraph, the convergence properties of all schemes Σ_η^{l}, Σ_ρ^{l}, $\Sigma_\epsilon^{\mathrm{l}}$ is discussed. Since, in these three cases, each adopted approach can be decoupled component-wise, let us address a general distributed estimation procedure with $\Sigma_\vartheta^{\mathrm{l}}$, such that parameter $\vartheta \in U_\vartheta \subseteq \mathbb{R}_{\geq 0}$ represents one among η, ρ and ϵ. Setting $\mathbf{x}' := \begin{bmatrix} (\mathbf{x}^1)^\top & \cdots & (\mathbf{x}^M)^\top \end{bmatrix}^\top \in \mathbb{R}^N$ and $\mathbf{u}'_\vartheta := \begin{bmatrix} (\mathbf{u}_\vartheta^1)^\top & \cdots & (\mathbf{u}_\vartheta^M)^\top \end{bmatrix}^\top \in \mathbb{R}^N$, it holds that

$$\mathbf{x}'(t+1) = \begin{bmatrix} \mathbf{F}_\vartheta & & 0 \\ & \ddots & \\ 0 & & \mathbf{F}_\vartheta \end{bmatrix} \mathbf{x}'(t) + \mathbf{u}'_\vartheta. \tag{5.33}$$

Hence, the convergence of the whole distributed estimation dynamics towards its centralized version, described by means of system (5.33), is governed only by the eigenvalues of stochastic matrix $\mathbf{F}_\vartheta$ belonging to its spectrum $\Lambda(\mathbf{F}_\vartheta)$. This fact implies that the convergence of the multivariate scheme $\Sigma_\vartheta := \left(\Sigma_\vartheta^1, \ldots, \Sigma_\vartheta^M\right)$ univocally depends on the topology of $\mathcal{G}$ and the choice of parameter ϑ. Moreover, given the spectral properties of $\Lambda(\mathbf{F}_\vartheta)$, a reasonable criterion to tune ϑ is given by

$$\vartheta^\star = \arg\min_{\mathrm{r}_\vartheta \in U_\vartheta} \{\mathrm{r}_\vartheta\} = \arg\min_{\vartheta \in U_\vartheta} \left\{ \max_{i=1,\ldots,n-1} \left| \lambda_i^{\mathbf{F}_\vartheta} \right| \right\}. \tag{5.34}$$

The criterion in (5.34) is introduced in order to optimize the converge rate $\mathrm{r}_\vartheta \in [0,1]$ of scheme Σ_ϑ: the smaller the convergence rate r_ϑ is the faster scheme Σ_ϑ provides accurate estimates as $t \to +\infty$. It is worth to notice that convergence analysis of estimation procedures related to the multivariate problem in (5.1) boils down to its scalar version. Therefore, index l is omitted in the continuation of this study, if not necessary, assuming $M = 1$ for the sake of simplicity.

In the sequel, it is shown that if the equilibrium points of systems (5.12), (5.20) and (5.24) exist, then these are those stated by Lemma 5.2.1. In other words, these equilibria represent the *distributed* solutions of problem (5.1), as proven by the following

Theorem 5.3.1 (Convergence of the distributed schemes)**.**
Estimations $\hat{\mathbf{x}} := \lim_{t \to +\infty} \mathbf{x}(t)$ *provided by schemes* Σ_η, Σ_ρ *and* Σ_ϵ *converge in terms of relative differences*[3] *to the centralized solution* $\mathbf{x}^\star$ *in* (5.6)*, for all choices of the parameters* $\eta \in (0,1)$, $\rho \in (0, +\infty)$ *and* $\epsilon \in (0, 2/\rho_S(\mathbf{L}))$ *respectively.*

Proof. By the spectral properties of $\mathbf{F}_\eta$, $\mathbf{F}_\rho$ and $\mathbf{F}_\epsilon$, marginal stability characterizes the free dynamics

$$\mathbf{x}(t+1) = \mathbf{F}_\vartheta \mathbf{x}(t), \quad \vartheta = \eta, \rho, \epsilon \tag{5.35}$$

of each schemes Σ_η, Σ_ρ, Σ_ϵ for all $\eta \in (0,1)$, $\rho > 0$ and $\epsilon \in (0, 2/\rho_S(\mathbf{L}))$. Moreover, identities $\lambda_{n-1}^{\mathbf{F}_\eta} \geq -1 + 2\eta > -1$, $\lambda_{n-1}^{\mathbf{F}_\rho} \geq (\rho - 2d_M)/(\rho + 2d_M) > -1$ and $\lambda_{n-1}^{\mathbf{F}_\epsilon} \geq 1 - \epsilon\rho_S(\mathbf{L}) > -1$ imply that the equilibrium point for (5.35) exists, it is unique, and it is given by $\hat{\mathbf{x}}_H = \beta_H \mathbb{1}_n$, with $\beta_H \in \mathbb{R}$. Similarly, for Σ_ϑ, with $\vartheta = \eta, \rho, \epsilon$, the equilibrium

[3] In the sense that $\hat{x}_i - \hat{x}_j = x_i^\star - x_j^\star$ for all $i, j = 1, \ldots, n$.

point can be expressed as $\hat{\mathbf{x}} = \hat{\mathbf{x}}_H + \hat{\mathbf{x}}_P(\mathbf{u}_\vartheta)$. At the equilibrium, the next chain of equivalences follows:

$$\hat{\mathbf{x}} = F_\vartheta \hat{\mathbf{x}} + \mathbf{u}_\vartheta, \quad \vartheta = \eta, \rho, \epsilon; \tag{5.36}$$

$$\begin{cases} 2\mathbf{D}\hat{\mathbf{x}} = 2\left[\vartheta\mathbf{D} + (1-\vartheta)\mathbf{A}\right]\hat{\mathbf{x}} + (1-\vartheta)\tilde{\mathbf{x}}, & \text{if } \vartheta = \eta; \\ (2\mathbf{D} + \vartheta\mathbf{I}_n)\,\hat{\mathbf{x}} = (2\mathbf{A} + \vartheta\mathbf{I}_n)\,\hat{\mathbf{x}} + \tilde{\mathbf{x}}, & \text{if } \vartheta = \rho; \\ 2\mathbf{I}_n\hat{\mathbf{x}} = 2(\mathbf{I}_n - \vartheta\mathbf{L})\hat{\mathbf{x}} + \vartheta\tilde{\mathbf{x}}, & \text{if } \vartheta = \epsilon; \end{cases} \tag{5.37}$$

$$2\mathbf{L}\hat{\mathbf{x}} = \tilde{\mathbf{x}}, \quad \vartheta = \eta, \rho, \epsilon. \tag{5.38}$$

Since $\hat{\mathbf{x}}_H \in \ker\{\mathbf{L}\}$, equality (5.38) is a particular instance of (5.5) solved by $\hat{\mathbf{x}}$; thus, $\hat{\mathbf{x}}$ is a solution to problem (5.1). ■

5.3.2 More details on the convergence properties of the distributed schemes

At first, it is necessary to distinguish between the convergence of the cost function h towards its minimum and the convergence of the states to the equilibrium values. In this kind of analysis, only the latter is taken into account. It is desirable that the update schemes (5.9) and (5.12) guarantee a monotonic decrease of the cost function, e.g., by adopting a gradient descent iterative procedure, nevertheless the convergence of the states towards the equilibrium configuration depends on the eigenvalues of the state matrices of the evaluated model ($\mathbf{F}_0$ and $\mathbf{F}_\eta$) and, in particular, the convergence rate depends on the second largest eigenvalue in modulus.

Convergence properties of Σ_0

Considering system (5.9), whose dynamics is governed by $\mathbf{F}_0$, the states convergence is ensured only if $\lambda = -1$ is not among the eigenvalues of the matrix. Indeed, if $\lambda = -1$ belongs to the spectrum of $\mathbf{F}_0$ its multiplicity would be unitary and this would imply (for large observation times $t \gg 1$) constant oscillations of the states around their equilibrium values due to the presence of a dominant oscillatory mode. From a graphical model point of view, this case can be interpreted referring to the network topology since $\lambda = -1$ occurs if and only if the associated graph is *bipartite*, i.e., all its cycles consist of an even number of nodes Landau and Odlyzko (1981) (see also Subsec. B.3.3). This statement can be proven accounting for the next fact Chung (1997)

Fact 5.3.2.
Denoting by $\boldsymbol{\varpi}_i \in \mathbb{R}^n$ the i-th eigenvector of $\mathbf{F}_0$, the following change of variables $\boldsymbol{\varpi}'_i = \mathbf{D}^{1/2}\boldsymbol{\varpi}_i$ is considered. Because of the spectrum linearity, for $i = 0, \ldots, n-1$, it holds that

$$\mathbf{F}_0\,\mathbf{D}^{-1/2}\boldsymbol{\varpi}'_i = \lambda_i^{\mathbf{F}_0}\mathbf{D}^{-1/2}\boldsymbol{\varpi}'_i \;\Rightarrow\; \boldsymbol{\mathcal{L}}\,\boldsymbol{\varpi}'_i = (1-\lambda_i^{\mathbf{F}_0})\boldsymbol{\varpi}'_i, \tag{5.39}$$

where $\boldsymbol{\mathcal{L}} = \mathbf{I}_n - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2} \in \mathbb{R}^{n\times n}$ is the *normalized Laplacian matrix* of $\mathcal{G}$. Hence, indicating with $\lambda_i^{\boldsymbol{\mathcal{L}}}$ the eigenvalues of $\boldsymbol{\mathcal{L}}$, such that $0 = \lambda_0^{\boldsymbol{\mathcal{L}}} < \lambda_1^{\boldsymbol{\mathcal{L}}} \leq \cdots \leq \lambda_{n-1}^{\boldsymbol{\mathcal{L}}} \leq 2$, one has that

$$\lambda_i^{\boldsymbol{\mathcal{L}}} = 1 - \lambda_i^{\mathbf{F}_0}, \quad \text{for } i = 0 \ldots n-1. \tag{5.40}$$

Moreover, $\lambda_{n-1}^{\boldsymbol{\mathcal{L}}} = 2$ if and only if $\mathcal{G}$ is bipartite, and consequently, $\lambda_{n-1}^{\mathbf{F}_0} = -1$ having unitary multiplicity.

When the graph describing the network is bipartite, convergence to the equilibrium can be reached only if it is possible to modify such topology so that the resulting graph presents at least one cycle made of an odd number of nodes. For example, in the multi-agent scenario, the link addition case (corresponding to a graph topology modification) translates into the possibility of finding an additional connection among devices that are neighbors in the sense of both communication and sensing: such an edge selection solution may not be feasible or convenient in real-world applications.

Conversely, in the non-bipartite case, convergence is assured and its rate is governed by the second largest (in modulus) eigenvalue of the state matrix $\mathbf{F}_0$.

Convergence properties of Σ_η

The presence of self-loops in model (5.12), controlled by parameter η, allows to modify the eigenvalues domain from the unit circle to the circle centered in $(\eta, 0)$ with radius $R_\eta = 1 - \eta,\ R_\eta \in (0, 1)$, ruling out the possible presence of the critic eigenvalue $\lambda = -1$. Therefore, not only the stability of the system is obtained but also the states convergence is always assured. In detail, the η parameter can be tuned in order to control the convergence speed, governed by the second largest eigenvalue (in modulus) of state matrix $\mathbf{F}_\eta$. If this is negative the estimated states converge toward the equilibrium values through an oscillatory transient period. On the contrary, if the second largest eigenvalue is positive then the estimation trend is asymptotically convergent, monotonic for large observation times. From an applicative perspective, the former oscillatory behavior is preferable since an averaging operation might provide an approximate solution to the convergence value, while in the latter case the iterations might consistently underestimate or overestimate the asymptotic values. Remarkably, these different behaviors can be seen as dependent on the control parameter η.

A good and viable strategy is to select the parameter η as

$$\eta^\star = \underset{\eta \in [0,1)}{\arg\min} \left\{ \max_{i=1,\dots,n-1} \left| \lambda_i^{\mathbf{F}_\eta} \right| \right\}, \tag{5.41}$$

to minimize the convergence rate of scheme Σ_η. The value of $\eta^\star$ can be analytically computed as shown in the next proposition. Note that in (5.41) it is assumed that η (or $\eta^\star$) might be zero: this case corresponds to consider model (5.9).

Proposition 5.3.3.
Given a multi-agent network represented by graph $\mathcal{G}$, the optimal value $\eta^\star$ in (5.41) is univocally determined as

$$\eta^\star = \begin{cases} 1 - \varsigma_{\boldsymbol{\mathcal{L}}}^{-1}, & \varsigma_{\boldsymbol{\mathcal{L}}} > 1 \\ 0, & \varsigma_{\boldsymbol{\mathcal{L}}} \leq 1 \end{cases} \tag{5.42}$$

where $\varsigma_{\boldsymbol{\mathcal{L}}} = \frac{1}{2}(\lambda_1^{\boldsymbol{\mathcal{L}}} + \lambda_{n-1}^{\boldsymbol{\mathcal{L}}}) \in \mathbb{R}$ and $\boldsymbol{\mathcal{L}}$ is the normalized Laplacian matrix related to $\mathcal{G}$.

Proof. Because $\lambda_1^{\mathbf{F}_\eta} \geq \dots \geq \lambda_{n-1}^{\mathbf{F}_\eta}$ for all $\eta \in [0, 1)$ and the spectrum linearity, problem

(5.41) can be rewritten as

$$\eta^\star = \underset{\eta\in[0,1)}{\arg\min} \left\{ \max \left\{ \left|\lambda_1^{\mathbf{F}_\eta}\right|, \left|\lambda_{n-1}^{\mathbf{F}_\eta}\right| \right\} \right\}, \tag{5.43}$$

where $\lambda_1^{\mathbf{F}_\eta} = \eta + (1-\eta)\lambda_1^{\mathbf{F}_0}$ and $\lambda_{n-1}^{\mathbf{F}_\eta} = \eta + (1-\eta)\lambda_{n-1}^{\mathbf{F}_0}$. Let then distinguish two cases: (a) $\varsigma_{\mathcal{L}} > 1$ and (b) $\varsigma_{\mathcal{L}} \leq 1$.
(a) The solution of (5.43) follows from the relation

$$-\left(\eta + (1-\eta)\lambda_{n-1}^{\mathbf{F}_0}\right) = \eta + (1-\eta)\lambda_1^{\mathbf{F}_0}, \tag{5.44}$$

under the constraint $\eta \in [0,1)$. Exploiting (5.40), it results that

$$\eta^\star = \frac{\lambda_1^{\mathbf{F}_0} + \lambda_{n-1}^{\mathbf{F}_0}}{\lambda_1^{\mathbf{F}_0} + \lambda_{n-1}^{\mathbf{F}_0} - 2} = \frac{2 - (\lambda_1^{\mathcal{L}} + \lambda_{n-1}^{\mathcal{L}})}{-(\lambda_1^{\mathcal{L}} + \lambda_{n-1}^{\mathcal{L}})} = 1 - \varsigma_{\mathcal{L}}^{-1}. \tag{5.45}$$

(b) If $\varsigma_{\mathcal{L}} \leq 1$ solution (5.45) is not valid because of the constraint on η. Furthermore, problem (5.43) boils down to

$$\eta^\star = \underset{\eta\in[0,1)}{\arg\min} \left\{ \eta + (1-\eta)\lambda_1^{\mathbf{F}_0} \right\}, \tag{5.46}$$

because $\lambda_1^{\mathbf{F}_0} = 1 - \lambda_1^{\mathcal{L}}$, $\lambda_1^{\mathbf{F}_0} \in [0,1)$, results to be the largest eigenvalue (in modulus) of $\mathbf{F}_0$ inside the unit circle. Since term $\eta + (1-\eta)\lambda_1^{\mathbf{F}_0}$ in (5.46) is strictly increasing in the variable η, the minimization is attained for $\eta^\star = 0$. ■

Remark 5.3.4.
Prop. 5.3.3 shows that, when $\varsigma_{\mathcal{L}} \leq 1$, the solution of (5.41) is $\eta^\star = 0$. Thus, in this case, either η should be selected in the neighborhood of zero or the scheme Σ_0 should be considered optimal.

Remark 5.3.5.
Note that, given a multi-agent network represented by a bipartite graph $\mathcal{G}$, the optimal value $\eta^\star$ in (5.41) is yielded by $\eta^\star = \lambda_1^{\mathcal{L}}(2 + \lambda_1^{\mathcal{L}})^{-1} \in (0,1)$.

The results in Prop. 5.3.3 allows to provide a lower bound for the convergence rate $\mathrm{r}_\eta = \max_{i=1\dots n-1}|\lambda_i^{\mathbf{F}_\eta}|$ of model Σ_η. If $\varsigma_{\mathcal{L}} \leq 1$ then it trivially holds $\mathrm{r}_\eta \geq \mathrm{r}_{\eta^\star} = 1 - \lambda_1^{\mathcal{L}}$. Nevertheless, when $\varsigma_{\mathcal{L}} > 1$, it is possible to prove that the convergence rate depends on the diameter of the considered network. To do so, it is worth to observe that, if $\varsigma_{\mathcal{L}} > 1$, by combining (5.44) and (5.45), it occurs

$$\mathrm{r}_\eta \geq \mathrm{r}_{\eta^\star} = 1 - \frac{\lambda_1^{\mathcal{L}}}{\varsigma_{\mathcal{L}}} = \frac{\lambda_{n-1}^{\mathcal{L}} - \lambda_1^{\mathcal{L}}}{\lambda_{n-1}^{\mathcal{L}} + \lambda_1^{\mathcal{L}}}. \tag{5.47}$$

Then, let us recall the following inequality valid for any non-complete graph $\mathcal{G}$ with n vertices Chung (1997)

$$\phi(\mathcal{G}) \leq \left\lceil \frac{\operatorname{arcosh}(n-1)}{\operatorname{arcosh}\left((\lambda_{n-1}^{\mathcal{L}} + \lambda_1^{\mathcal{L}})/(\lambda_{n-1}^{\mathcal{L}} - \lambda_1^{\mathcal{L}})\right)} \right\rceil. \tag{5.48}$$

where $\phi(\mathcal{G})$ is the diameter of the graph $\mathcal{G}$ (see Subsec. B.3.1). Combining (5.47) and (5.48), a lower bound for the convergence rate can be derived depending only on topological properties:

$$\mathrm{r}_\eta \geq \mathrm{r}_{\eta^\star} \geq \operatorname{sech}\left(\frac{\operatorname{arcosh}(n-1)}{\phi(\mathcal{G})-1}\right). \tag{5.49}$$

Note that for a complete graph with n vertices it holds that $\lambda_1^{\mathcal{L}} = \lambda_{n-1}^{\mathcal{L}} = n(n-1)^{-1}$, hence $\eta^\star = n^{-1}$ and $\mathrm{r}_{\eta^\star} = 0$. This implies that the estimate procedure converges toward the equilibrium in one single step.

On the convergence of Σ_ρ

Similarly to the previous subparagraph, the presence of self-loops in model (5.20) controlled by parameter ρ allows to modify the spectral domain of $\mathbf{F}_0$. However, in this framework, convergence guarantees and general insights related to this topic are more complicated to be provided, due to the intrinsic structure of the algebraic model for scheme Σ_ρ. In the following Lem. 5.3.6, some properties of the spectrum of $\mathbf{F}_\rho$ are shown to begin the discussion.

Lemma 5.3.6.
Considering two values ρ_1, ρ_2 of the parameter ρ, it holds that $\lambda_i^{\mathbf{F}_{\rho_1}} > \lambda_i^{\mathbf{F}_{\rho_2}}$ for $i \in \{1 \dots n-1\}$ if and only if $\rho_1 > \rho_2 \geq 0$.

Proof. Consider $\mathbf{F}_\rho : \mathbb{R}_{>0} \to \mathbb{R}^{n\times n}$ as a continuous function that maps $\rho \mapsto \mathbf{F}_\rho = (2\mathbf{D} + \rho\mathbf{I}_n)^{-1}(2\mathbf{A} + \rho\mathbf{I}_n)$. By the Gershgorin circle theorem, the derivative of $\mathbf{F}_\rho$ w.r.t. ρ is positive semidefinite, namely $\mathbf{F}'_\rho = d\mathbf{F}_\rho/d\rho = 2(2\mathbf{D} + \rho\mathbf{I}_n)^{-2}\mathbf{L} \succeq 0$. Hence, $\mathbf{F}_\rho$ is a nondecreasing function of the parameter ρ, namely $\rho_1 > \rho_2 \geq 0$ implies that $\mathbf{F}_{\rho_1} \succeq \mathbf{F}_{\rho_2}$ and viceversa. As a consequence, it follows that $\lambda_i^{\mathbf{F}_{\rho_1}} > \lambda_i^{\mathbf{F}_{\rho_2}}$ for $i \in \{1, \dots, n-1\}$ if and only if $\rho_1 > \rho_2 \geq 0$. This holds strictly for all $i \in \{1, \dots, n-1\}$ because the unique case of equality is provided by $\lambda_0^{\mathbf{F}_{\rho_1}} = \lambda_0^{\mathbf{F}_{\rho_2}} = 1$, $\forall \rho_1, \rho_2$, given the stochasticity of $\mathbf{F}_\rho$ and the fact that $\mathbf{L}$ has only a unique zero eigenvalue. ∎

Exploiting the results of Lem. 5.3.6, the following Prop. 5.3.7 states an upper and lower bound for each eigenvalue of $\mathbf{F}_\rho$ as a function of the eigenvalues of $\mathbf{F}_0$, the penalty value ρ and the extremal (maximum and minimum) degrees in graph $\mathcal{G}$.

Proposition 5.3.7.
For any $\rho > 0$, all the eigenvalues $\lambda_i^{\mathbf{F}_\rho}$, $i \in \{1, \dots, n-1\}$, of matrix $\mathbf{F}_\rho$ admit the following lower $\underline{\lambda}_i^{\mathbf{F}_\rho}$ and upper $\overline{\lambda}_i^{\mathbf{F}_\rho}$ bounds:

$$\underline{\lambda}_i^{\mathbf{F}_\rho} = \frac{\rho + 2\lambda_i^{\mathbf{F}_0} d_M}{\rho + 2d_M} \quad \text{and} \quad \overline{\lambda}_i^{\mathbf{F}_\rho} = \frac{\rho + 2\lambda_i^{\mathbf{F}_0} d_m}{\rho + 2d_m}. \tag{5.50}$$

Trivially, it holds that $\underline{\lambda}_i^{\mathbf{F}_\rho} = \lambda_i^{\mathbf{F}_\rho} = \overline{\lambda}_i^{\mathbf{F}_\rho}$ if and only if the graph $\mathcal{G}$ is regular, i.e., $\deg(v_i) = \mathrm{d}$ $\forall v_i \in \mathcal{V}$ with $\mathrm{d} > 0$.

Proof. Let $\tilde{\mathbf{F}}_\rho(\mathbf{\Omega}) : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}$ be a function that maps a diagonal matrix $\mathbf{\Omega} \succ 0$ to the matrix $\tilde{\mathbf{F}}_\rho(\mathbf{\Omega}) = (2\mathbf{\Omega} + \rho\mathbf{I}_n)^{-1}(2\mathbf{\Omega}\mathbf{F}_0 + \rho\mathbf{I}_n)$. Exploiting (5.40) and the Gershgorin circle

theorem, it is possible to prove that the condition $d_M\mathbf{I}_n \succeq \mathbf{\Omega} \succeq d_m\mathbf{I}_n$, i.e. $\mathbf{\Omega} - d_m\mathbf{I}_n \succeq 0$ and $d_M\mathbf{I}_n - \mathbf{\Omega} \succeq 0$, implies

$$\begin{cases} 2\rho(\mathbf{\Omega} - d_m\mathbf{I}_n)(\mathbf{I}_n - \mathbf{F}_0) \succeq 0 \\ 2\rho(d_M\mathbf{I}_n - \mathbf{\Omega})(\mathbf{I}_n - \mathbf{F}_0) \succeq 0 \end{cases} \tag{5.51}$$
$$\begin{cases} (2\mathbf{\Omega} + \rho\mathbf{I}_n)(2d_m\mathbf{F}_0 + \rho\mathbf{I}_n) \succeq (2d_m + \rho)(2\mathbf{\Omega}\mathbf{F}_0 + \rho\mathbf{I}_n) \\ (2d_M + \rho)(2\mathbf{\Omega}\mathbf{F}_0 + \rho\mathbf{I}_n) \succeq (2\mathbf{\Omega} + \rho\mathbf{I}_n)(2d_M\mathbf{F}_0 + \rho\mathbf{I}_n) \end{cases}$$

and, thus, $\tilde{\mathbf{F}}_\rho(d_m\mathbf{I}_n) \succeq \tilde{\mathbf{F}}_\rho(\mathbf{\Omega}) \succeq \tilde{\mathbf{F}}_\rho(d_M\mathbf{I}_n)$. The provided implications are valid for $\mathbf{\Omega} = \mathbf{D}$. In this case, $\tilde{\mathbf{F}}_\rho(\mathbf{D}) = \mathbf{F}_\rho$, hence the eigenvalue bounds in (5.50) can be derived for any $\rho > 0$ leveraging Lem. 5.3.6.
Finally, when $\mathcal{G}$ is a regular graph, it trivially holds that $d_m = d_M$ implying $\underline{\lambda}_i^{\mathbf{F}_\rho} = \lambda_i^{\mathbf{F}_\rho} = \overline{\lambda}_i^{\mathbf{F}_\rho}$, for $i \in \{1, \dots, n-1\}$. ■

Given this premise, in the following it is shown how the parameter ρ can be tuned in order to improve the convergence rate $\mathrm{r}_\rho \in [0, 1]$ of Σ_ρ, which is related to the second largest eigenvalue (in modulus) of $\mathbf{F}_\rho$. In particular, because of the monotonicity of $\Lambda(\mathbf{F}_\rho)$ for a given ρ, the convergence rate corresponds to

$$\mathrm{r}_\rho = \max_{i \in \{1,\dots,n-1\}} |\lambda_i^{\mathbf{F}_\rho}| = \max\left(|\lambda_1^{\mathbf{F}_\rho}|, |\lambda_{n-1}^{\mathbf{F}_\rho}|\right). \tag{5.52}$$

To minimize r_ρ in (5.52), a good and viable strategy is thus to select the parameter ρ as

$$\rho^\star = \arg\min_{\rho \geq 0} \left\{ \max\left(|\lambda_1^{\mathbf{F}_\rho}|, |\lambda_{n-1}^{\mathbf{F}_\rho}|\right) \right\} \tag{5.53}$$

The next proposition focuses on the solution of the optimization problem (5.53). Note that $\rho^\star$ might be zero: this case corresponds the adoption of scheme (5.9).

Proposition 5.3.8.
The solution of problem (5.53) exists and is unique. Defining the function $\varsigma_{\mathbf{\Omega}}$ that acts on matrix $\mathbf{\Omega}$ with ordered spectrum such that $\varsigma_{\mathbf{\Omega}} : \mathbb{R}^{n\times n} \to \mathbb{R}$, $\mathbf{\Omega} \mapsto (\lambda_1^{\mathbf{\Omega}} + \lambda_{n-1}^{\mathbf{\Omega}})/2$, it holds that

$$\rho^\star = \begin{cases} \rho^+, & \varsigma_{\mathcal{L}} > 1 \\ 0, & \varsigma_{\mathcal{L}} \leq 1 \end{cases} \tag{5.54}$$

with

$$2(\varsigma_{\mathcal{L}} - 1)d_m \leq \rho^+ \leq 2(\varsigma_{\mathcal{L}} - 1)d_M. \tag{5.55}$$

Furthermore, both bounds in (5.55) coincides with ρ^+ if and only if $\mathcal{G}$ is regular.

Proof. Similarly to relation (5.40), for $\rho \geq 0$ it holds that

$$\frac{1}{2}\left(\left(1 - \lambda_{n-1}^{\mathbf{F}_\rho}\right) + \left(1 - \lambda_1^{\mathbf{F}_\rho}\right)\right) = \varsigma_{\mathcal{L}_\rho} = 1 - \varsigma_{\mathbf{F}_\rho}. \tag{5.56}$$

Thanks to Lem. 5.3.6, $\varsigma_{\mathbf{F}_\rho}$ is proven to be a continuous strictly increasing function of the parameter ρ, thus ensuring the existence of a unique solution for (5.53). This also implies that $\varsigma_{\mathbf{F}_\rho} > \varsigma_{\mathbf{F}_0}$ for $\rho > 0$.

Then, if $\varsigma_{\mathbf{F}_0} \geq 0$, namely $\varsigma_{\mathcal{L}} \leq 1$, it follows that $\varsigma_{\mathbf{F}_\rho} > 0$. Therefore, due to (5.56), the convergence rate results to be $\mathrm{r}_\rho = |\lambda_1^{\mathbf{F}_\rho}|$ and it is trivially minimized for $\rho = 0$. Conversely, if $\varsigma_{\mathbf{F}_0} < 0$, namely $\varsigma_{\mathcal{L}} > 1$, the condition $\varsigma_{\mathbf{F}_\rho} = 0$, corresponding to the optimal scenario wherein $|\lambda_1^{\mathbf{F}_\rho}| = |\lambda_{n-1}^{\mathbf{F}_\rho}|$ may occur. In particular, for Bolzano's theorem, it follows that the condition $\varsigma_{\mathbf{F}_\rho} = 0$ is fulfilled only for a value of ρ, namely ρ^+.
Finally, resting upon Prop. 5.3.7, inequalities (5.55) can be derived by imposing

$$(\underline{\lambda}_1^{\mathbf{F}_{\rho^\star}} + \underline{\lambda}_{n-1}^{\mathbf{F}_{\rho^\star}})/2 \leq \varsigma_{\mathbf{F}_{\rho^\star}} \leq (\overline{\lambda}_1^{\mathbf{F}_{\rho^\star}} + \overline{\lambda}_{n-1}^{\mathbf{F}_{\rho^\star}})/2 \tag{5.57}$$

and exploiting the fact that $\varsigma_{\mathbf{F}_{\rho^\star}} = 0$ only when $\varsigma_{\mathbf{F}_0} < 0$. Trivially, the bounds (5.55) are met for regular graphs since $d_m = d_M$ for these topologies. ■

Remark 5.3.9.
Prop. 5.3.8 suggests when it is useful to adopt the PP-based solution (5.20) basing on the graph describing the given multi-agent system through $\varsigma_{\mathcal{L}}$. By definition, this quantity depends on the network connectivity and similarity to bipartite graph. Hence, when $\varsigma_{\mathcal{L}} > 1$, the corresponding topology tends to circulate the information between node groups. In this case, the choice $\rho^* = \rho^+$ allows to improve the information management. On the other hand, when $\varsigma_{\mathcal{L}} \leq 1$, either $\rho^\star$ is set as an arbitrary relatively small value or the scheme Σ_0 has to be preferred, since the addition of self loop in the graph does not improve (and often worsen) the estimation speed.

To conclude, the next proposition discusses the bounds of the convergence rate for the optimal selection of ρ in (5.54)

Proposition 5.3.10.
If $\varsigma_{\mathcal{L}} \leq 1$, the rate of convergence $\mathrm{r}_{\rho^\star}$ for scheme $\Sigma_{\rho^\star} = \Sigma_0$ is yielded by $\mathrm{r}_{\rho^\star} = 1 - \lambda_1^{\mathcal{L}} \in (0, 1)$. Otherwise, if $\varsigma_{\mathcal{L}} > 1$, it is bounded as follows

$$\mathrm{r}_{\rho^\star} \leq \max\left(-\underline{\lambda}_{n-1}^{\mathbf{F}_{\rho^+}}, \overline{\lambda}_1^{\mathbf{F}_{\rho^+}}\right) < 1, \tag{5.58}$$

$$\mathrm{r}_{\rho^\star} \geq \max\left(-\min\left(0, \overline{\lambda}_{n-1}^{\mathbf{F}_{\rho^+}}\right), \max\left(0, \underline{\lambda}_1^{\mathbf{F}_{\rho^+}}\right)\right) \geq 0. \tag{5.59}$$

In particular, $\mathrm{r}_{\rho^\star} = 0$ if and only if $\mathcal{G}$ is the complete graph.

Proof. The result follows immediately from Prop. 5.3.7, Prop. 5.3.1, Prop. 5.3.8, the fact that if $\lambda_{n-1}^{\mathbf{F}_0} = -1$ then $\lambda_{n-1}^{\mathbf{F}_\rho} \geq (\rho - 2d_M)/(\rho + 2d_M) > -1$ for $\rho > 0$, and the general fact that $0 < \lambda_1^{\mathcal{L}} \leq 1$ holds for a non-complete connected graph $\mathcal{G}$. Moreover, since nontrivial eigenvalues of $\boldsymbol{\mathcal{L}}$ are all equal to $n(n-1)^{-1}$ if and only if $\mathcal{G}$ is the complete graph, the assignment of $\rho^+ = 2$ to the parameter ρ allows to place all the eigenvalues of $\mathbf{F}_\rho$ different from 1 in 0, achieving superlinear convergence ($\mathrm{r}_\rho^\star = 0$). ■

On the convergence of Σ_ϵ

As already described in Subsec. 5.2.2, scheme Σ_ϵ presents a little dissimilarity from schemes Σ_η and Σ_ρ, since, as ϵ approaches 0, Σ_ϵ does not approach the behavior of Σ_0. Nevertheless, this scheme exhibits convergence properties that are deeply related to both Σ_η and Σ_ρ, especially to the former scheme. The forthcoming preposition 5.3.11 will clarify the previous statement.

Again, let us recall that a good and viable strategy is to select the parameter ϵ as

$$\epsilon^\star = \underset{\epsilon\in(0,2/\rho_S(\mathbf{L}))}{\arg\min} \left\{ \max_{i=1,\dots,n-1} \left|\lambda_i^{\mathbf{F}_\epsilon}\right| \right\}, \tag{5.60}$$

to minimize the convergence rate of scheme Σ_ϵ. The value of $\epsilon^\star$ can be analytically computed as shown in the next proposition. Note that in (5.60) it is assumed that ϵ (or $\epsilon^\star$) must not be equal to $2/\rho_S(\mathbf{L})$: this case corresponds to consider exactly a model with similar properties of Σ_0 when the topology is bipartite (or fully equivalent properties, if the topology is also regular). Nonetheless, the value $2/\rho_S(\mathbf{L})$ assigned to ϵ drives system Σ_ϵ to instability for any topology because it allows the presence of multiple eigenvalues λ, such that $|\lambda| = 1$, for matrix $\mathbf{F}_\epsilon$.

Proposition 5.3.11.
Given a multi-agent network represented by graph $\mathcal{G}$, the optimal value $\epsilon^\star$ in (5.60) is univocally determined as

$$\epsilon^* = \varsigma_{\mathbf{L}}^{-1} \tag{5.61}$$

where $\varsigma_{\mathbf{L}} = \frac{1}{2}(\lambda_1^{\mathbf{L}} + \lambda_{n-1}^{\mathbf{L}}) \in \mathbb{R}$ and $\mathbf{L}$ is the Laplacian matrix related to $\mathcal{G}$.

Proof. The proof exhibits analogies to the one for Prop. 5.3.3. Because $\lambda_1^{\mathbf{F}_\epsilon} \geq \ldots \geq \lambda_{n-1}^{\mathbf{F}_\epsilon}$ for all $\epsilon \in (0, 2/\rho_S(\mathbf{L}))$ and the spectrum linearity, problem (5.60) can be rewritten as

$$\epsilon^\star = \underset{\epsilon\in(0,2/\rho_S(\mathbf{L}))}{\arg\min} \left\{ \max\left\{ \left|\lambda_1^{\mathbf{F}_\epsilon}\right|, \left|\lambda_{n-1}^{\mathbf{F}_\epsilon}\right| \right\} \right\}, \tag{5.62}$$

where $\lambda_1^{\mathbf{F}_\epsilon} = 1-\epsilon\lambda_1^{\mathbf{L}}$ and $\lambda_{n-1}^{\mathbf{F}_\epsilon} = 1-\epsilon\lambda_{n-1}^{\mathbf{L}}$. Here, conversely to Prop. 5.3.3, no distinction for $\varsigma_{\mathbf{L}}$ is made. Indeed, the solution of (5.62) follows from the relation

$$-\left(1-\epsilon\lambda_{n-1}^{\mathbf{L}}\right) = 1-\epsilon\lambda_1^{\mathbf{L}}, \tag{5.63}$$

under the constraint $\epsilon \in (0, 2/\rho_S(\mathbf{L}))$. Then, leveraging the connectivity of the graph $\mathcal{G}$, $\lambda_1^{\mathbf{L}} > 0$ and thus one has that

$$\epsilon^\star = \frac{2}{\lambda_1^{\mathbf{L}} + \lambda_{n-1}^{\mathbf{L}}} = \varsigma_{\mathbf{L}}^{-1} \in (0, 2/\rho_S(\mathbf{L})). \tag{5.64}$$

■

The results in Prop. 5.3.11 allows to provide a lower bound for the convergence rate $\mathrm{r}_\epsilon = \max_{i=1\dots n-1}|\lambda_i^{\mathbf{F}_\epsilon}|$ of model Σ_ϵ. Indeed, it is possible to state that by combining (5.63) and (5.64), it occurs

$$\mathrm{r}_\epsilon \geq \mathrm{r}_{\epsilon^\star} = 1 - \frac{\lambda_1^{\mathbf{L}}}{\varsigma_{\mathbf{L}}} = \frac{\lambda_{n-1}^{\mathbf{L}} - \lambda_1^{\mathbf{L}}}{\lambda_{n-1}^{\mathbf{L}} + \lambda_1^{\mathbf{L}}}. \tag{5.65}$$

Remark 5.3.12.
Note that, given a multi-agent network represented by a bipartite graph $\mathcal{G}$, the optimal value $\epsilon^\star$ in (5.60) results to be $\epsilon^\star = 2(\lambda_1^{\mathbf{L}} + 2d_M)^{-1} \in (0, 1/d_M)$. Generally, $\epsilon \in (0, 1/d_M)$ is just a sufficient condition to provide stability to Σ_ϵ, as authors in Olfati-Saber et al. (2007) point out, since the bipartite case represents a worst case scenario for choosing ϵ.

Moreover, for a complete graph with n vertices it holds that $\lambda_1^{\mathbf{L}} = \lambda_{n-1}^{\mathbf{L}} = n$, hence $\epsilon^\star = n^{-1}$ and $\mathrm{r}_{\epsilon^\star} = 0$. This implies that the estimate procedure converges toward the equilibrium in one single step.

5.3.3 Convergence comparison between the developed distributed schemes

Generally, one might wonder which of the three distributed parametric schemes in Table 5.1 should be adopted to achieve the best performances in terms of convergence rate. It is, indeed, a matter of fact that the latter strongly depends on the spectral properties of the state matrix involved during the estimation procedure (see Table 5.2). In particular, the optimal convergence rate is a consequence of the optimal parameter selection implemented, as summarized in Table 5.3, which is basically computed leveraging the first nonzero eigenvalue and the spectral radius of the normalized Laplacian or the Laplacian matrices.

Examples in Fig. 5.2 illustrate that, for nonregular frameworks, none of the distributed schemes among Σ_η, Σ_ρ and Σ_ϵ outperforms another. Indeed, even for networks with the same number of nodes and diameter, the convergence performances depend on the specific case under analysis. On the other hand, as shown in Subsec. 5.2.3 an equivalence condition for all the iterative procedures is represented, to some extent, by the adoption of the same regular topology, i.e. the best convergence rate for the different schemes is identical for any d-regular set-up.

Given all the propositions proven in the previous paragraph, it is thus possible to state that, in general, the convergence properties for schemes Σ_η, Σ_ρ and Σ_ϵ are similar. Care has to be taken just for bipartite networks while selecting the parameter in its domain, otherwise the convergence does not occur, since Σ_0 exhibits oscillating dynamics in this framework.

Scheme	Parameter	State matrix	Input vector
Σ_0		$\mathbf{F}_0 = \mathbf{D}^{-1}\mathbf{A}$	$\mathbf{u}_0 = \frac{1}{2}\mathbf{D}^{-1}\tilde{\mathbf{x}}$
Σ_η	$\eta \in [0,1)$	$\mathbf{F}_\eta = (\eta\mathbf{I}_n + (1-\eta)\mathbf{F}_0)$	$\mathbf{u}_\eta = (1-\eta)\mathbf{u}_0$
Σ_ρ	$\rho \geq 0$	$\mathbf{F}_\rho = \left(\mathbf{D} + \frac{\rho}{2}\mathbf{I}_n\right)^{-1}\left(\mathbf{A} + \frac{\rho}{2}\mathbf{I}_n\right)$	$\mathbf{u}_\rho = \left(\mathbf{D} + \frac{\rho}{2}\mathbf{I}_n\right)^{-1}\mathbf{D}\mathbf{u}_0$
Σ_ϵ	$\epsilon \in \left(0, \frac{2}{\lambda_{n-1}^{\mathbf{L}}}\right)$	$\mathbf{F}_\epsilon = \mathbf{I}_n - \epsilon\mathbf{L}$	$\mathbf{u}_\epsilon = \epsilon\,\mathbf{D}\mathbf{u}_0$

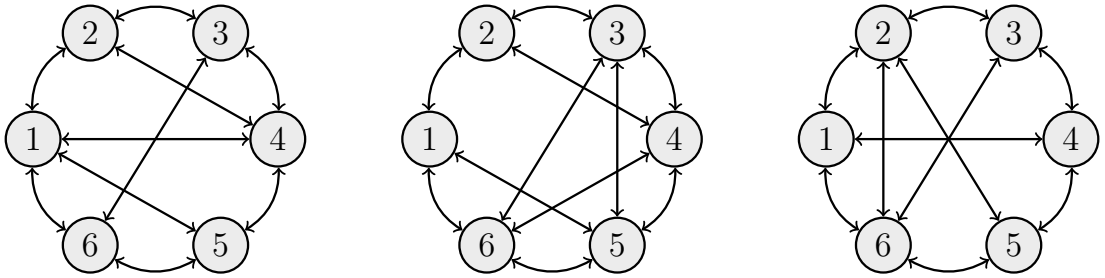
Table 5.1. Distributed iterative schemes examined in this chapter.

Scheme	Spectral properties of state state matrix
Σ_0	$\{1 = \lambda_0^{\mathbf{F}_0} > \cdots \geq \lambda_{n-1}^{\mathbf{F}_0}\} \subseteq [-1,1]$
Σ_η	$\{\lambda_0^{\mathbf{F}_\eta} \geq \cdots \geq \lambda_{n-1}^{\mathbf{F}_\eta}\} \subseteq (-1+2\eta, 1) \subseteq (-1,1]$
Σ_ρ	$\{1 = \lambda_0^{\mathbf{F}_\rho} > \cdots \geq \lambda_{n-1}^{\mathbf{F}_\rho}\} \subseteq \left(\left(\frac{\rho-2d_M}{\rho+2d_M}\right), 1\right) \subseteq (-1,1]$
Σ_ϵ	$\{\lambda_0^{\mathbf{F}_\epsilon} \geq \cdots \geq \lambda_{n-1}^{\mathbf{F}_\epsilon}\} \subseteq \left(1 - \frac{2\epsilon}{\lambda_{n-1}^{\mathbf{L}}}, 1\right) \subseteq (-1,1]$

Table 5.2. Spectral properties of the update matrices involved in the distributed iterative schemes examined, considering $\eta, \rho > 0$.

Scheme	Best convergence rate	Optimal parameter selection
Σ_0	$\mathrm{r}_0 = \begin{cases} \lambda^{\mathcal{L}}_{n-1} - 1, & \text{if } \varsigma_{\mathcal{L}} > 1 \\ 1 - \lambda^{\mathcal{L}}_1, & \text{if } \varsigma_{\mathcal{L}} \leq 1 \end{cases}$	no parameter available
Σ_η	$\mathrm{r}_{\eta^\star} = \begin{cases} 1 - \lambda^{\mathcal{L}}_1/\varsigma_{\mathcal{L}}, & \text{if } \varsigma_{\mathcal{L}} > 1 \\ 1 - \lambda^{\mathcal{L}}_1, & \text{if } \varsigma_{\mathcal{L}} \leq 1 \end{cases}$	$\eta^\star = \begin{cases} 1 - 1/\varsigma_{\mathcal{L}}, & \text{if } \varsigma_{\mathcal{L}} > 1 \\ 0, & \text{if } \varsigma_{\mathcal{L}} \leq 1 \end{cases}$
Σ_ρ	$\mathrm{r}_{\rho^\star} = \begin{cases} \mathrm{r}_{\rho^+}, & \text{if } \varsigma_{\mathcal{L}} > 1 \\ 1 - \lambda^{\mathcal{L}}_1, & \text{if } \varsigma_{\mathcal{L}} \leq 1 \end{cases}$	$\rho^\star = \begin{cases} \rho^+, & \text{if } \varsigma_{\mathcal{L}} > 1 \\ 0, & \text{if } \varsigma_{\mathcal{L}} \leq 1 \end{cases}$
Σ_ϵ	$\mathrm{r}_{\epsilon^\star} = 1 - \lambda^{\mathbf{L}}_1/\varsigma_{\mathbf{L}}$	$\epsilon^\star = 1/\varsigma_{\mathbf{L}}$

Table 5.3. Best convergence rate depending on the optimal parameter selection (ρ^+ can be found in the interval given by (5.55) via bisection method and r_{ρ^+} is bounded by 5.50).



(a) $\mathrm{r}_{\eta^\star} \simeq 0.3895$, $\eta^\star \simeq 0.1760$, $\mathrm{r}_{\rho^\star} \simeq 0.3932$, $\rho^\star \simeq 1.3607$, $\mathrm{r}_{\epsilon^\star} \simeq 0.4330$, $\epsilon^\star \simeq 0.2500$, $\mathrm{r}_0 \simeq 0.6863$.

(b) $\mathrm{r}_{\eta^\star} \simeq 0.3177$, $\eta^\star \simeq 0.2258$, $\mathrm{r}_{\rho^\star} \simeq 0.3162$, $\rho^\star \simeq 2.0000$, $\mathrm{r}_{\epsilon^\star} \simeq 0.3333$, $\epsilon^\star \simeq 0.2222$, $\mathrm{r}_0 \simeq 0.7020$.

(c) $\mathrm{r}_{\eta^\star} \simeq 0.3442$, $\eta^\star \simeq 0.2727$, $\mathrm{r}_{\rho^\star} \simeq 0.3395$, $\rho^\star \simeq 2.2422$, $\mathrm{r}_{\epsilon^\star} \simeq 0.3333$, $\epsilon^\star \simeq 0.2222$, $\mathrm{r}_0 \simeq 0.8482$.

Figure 5.2. Convergence rate comparison: simple examples of nonregular topologies with 6 nodes, minimum degree of 3, maximum degree of 4 and diameter of 2. Basically, all the algorithms offer similar performances but their rate of convergence is topology dependent, differing in absence of regularity. (a) Convergence of Σ_η scheme is faster in this example. (b) Convergence of Σ_ρ scheme is faster in this example. (c) Convergence of Σ_ϵ scheme is faster in this example.

Lastly, it is worth to recall that all the three algorithms based on a parameter converge in one step, if the topology is complete.

5.4 Sensitivity analysis to parametric variations

At the light of the premise given in Subsec. 2.2.1, an expression for the relative sensitivity function is provided for all the distributed iterative schemes, namely $\bar{S}_\eta(z)$ for Σ_η, $\bar{S}_\rho(z)$ for Σ_ρ and $\bar{S}_\epsilon(z)$ for Σ_ϵ. Moreover, in order to quantify precision of estimation performances and rejection to disturbances at the steady-state of these dynamic systems, the limits for $z \to 1$ for $\bar{S}_\eta(z)$, $\bar{S}_\rho(z)$ and $\bar{S}_\epsilon(z)$ are computed. Hereafter, the adjugate operator is indicated with $\mathrm{adj}\,[\cdot]$ and, with $\mathrm{Diag}\left[\boldsymbol{\varpi}^\top\right]$, it is denoted the diagonal matrix whose i-th diagonal element is given by the i-th component $[\boldsymbol{\omega}]_i$ of vector $\boldsymbol{\omega}$.

5.4.1 Computation of the sensitivity values at steady-state

The computation of $|\bar{S}_\vartheta(1)|$, with $\vartheta = \eta, \rho, \epsilon$, is carried out in the following paragraphs for all the distributed schemes developed.

Sensitivity of Σ_η

As regards scheme Σ_η, its discrete transfer function is given by

$$\mathbf{W}(z,\eta) = (z\mathbf{I}_n - \mathbf{F}_\eta)^{-1}\frac{1}{2}(1-\eta)\mathbf{D}^{-1} = \left(2\left(\frac{z-\eta}{1-\eta}\mathbf{D} - \mathbf{A}\right)\right)^{-1} \tag{5.66}$$

thus,

$$\frac{\partial \mathbf{W}(z,\eta)}{\partial \eta} = -\mathbf{W}(z,\eta)\frac{\partial \mathbf{W}^{-1}(z,\eta)}{\partial \eta}\mathbf{W}(z,\eta) = -2\frac{z-1}{(1-\eta)^2}\mathbf{W}(z,\eta)\mathbf{D}\mathbf{W}(z,\eta) \tag{5.67}$$

and, since $\mathbf{W}(z,\eta)$ is symmetric, it holds that

$$\bar{S}_\eta(z) = \operatorname{tr}\left[\mathbf{W}^{-\top}(z,\eta)\frac{\partial \mathbf{W}(z,\eta)}{\partial \eta}\right] = -2\frac{z-1}{(1-\eta)^2}\operatorname{tr}\left[\mathbf{D}\mathbf{W}(z,\eta)\right]. \tag{5.68}$$

Therefore, for Σ_η one has

$$\begin{aligned}
\bar{S}_\eta(1) &= \lim_{z\to 1} -\frac{z-1}{(1-\eta)^2}\operatorname{tr}\left[((z-\eta)/(1-\eta)\mathbf{I}_n - \mathbf{F}_0)^{-1}\right] && (5.69)\\
&= \lim_{z\to 1} -\frac{z-1}{(1-\eta)^2}\frac{\operatorname{tr}\left[\operatorname{adj}\left[(z-\eta)/(1-\eta)\mathbf{I}_n - \mathbf{F}_0\right]\right]}{(1-\eta)^{-n}\det\left[z\mathbf{I}_n - (\eta\mathbf{I}_n + (1-\eta)\mathbf{F}_0)\right]} && (5.70)\\
&= \lim_{z\to 1} -\frac{z-1}{(1-\eta)^{2-n}}\frac{\operatorname{tr}\left[\operatorname{adj}\left[\mathbf{I}_n - \mathbf{F}_0\right]\right]}{(z-1)\prod_{i=1}^{n-1}\left(z-\lambda_i^{\mathbf{F}_\eta}\right)} && (5.71)\\
&= -\frac{1}{(1-\eta)^{2-n}}\frac{\operatorname{tr}\left[\operatorname{adj}[\mathbf{T}_0]\operatorname{adj}[\mathbf{I}_n - \mathbf{\Lambda}_0]\operatorname{adj}[\mathbf{T}_0^{-1}]\right]}{(1-\eta)^{n-1}\prod_{i=1}^{n-1}\left(1-\lambda_i^{\mathbf{F}_0}\right)} && (5.72)
\end{aligned}$$

where $\mathbf{T}_0^{-1}(\mathbf{I}_n - \mathbf{\Lambda}_0)\mathbf{T}_0$ is the diagonalized Jordan form of $(\mathbf{I}_n - \mathbf{F}_0)$, such that $\mathbf{\Lambda}_0 = \operatorname{Jord}\left[\begin{bmatrix}\lambda_0^{\mathbf{F}_0} & \cdots & \lambda_{n-1}^{\mathbf{F}_0}\end{bmatrix}\right]$ is a block-diagonal Jordan matrix. Hence, by the trace properties, it holds that

$$\begin{aligned}
\bar{S}_\eta(1) &= -\frac{1}{(1-\eta)}\frac{\operatorname{tr}\left[\operatorname{adj}[\mathbf{T}_0^{-1}]\operatorname{adj}[\mathbf{T}_0]\operatorname{adj}[\mathbf{I}_n - \mathbf{\Lambda}_0]\right]}{\prod_{i=1}^{n-1}\left(1-\lambda_i^{\mathbf{F}_0}\right)} && (5.73)\\
&= -\frac{1}{(1-\eta)}\frac{\operatorname{tr}\left[\operatorname{Diag}\left[\begin{bmatrix}\prod_{i=1}^{n-1}\left(1-\lambda_i^{\mathbf{F}_0}\right) & 0 & \dots & 0\end{bmatrix}\right]\right]}{\prod_{i=1}^{n-1}\left(1-\lambda_i^{\mathbf{F}_0}\right)} && (5.74)\\
&= -(1-\eta)^{-1}\prod_{i=1}^{n-1}\left(1-\lambda_i^{\mathbf{F}_0}\right)/\prod_{i=1}^{n-1}\left(1-\lambda_i^{\mathbf{F}_0}\right) && (5.75)\\
&= -(1-\eta)^{-1} && (5.76)
\end{aligned}$$

Sensitivity of Σ_ρ

The discrete transfer function of scheme Σ_ρ is yielded by

$$\mathbf{W}(z,\rho) = (z\mathbf{I}_n - \mathbf{F}_\rho)^{-1}(2\mathbf{D} + \rho\mathbf{I}_n)^{-1} = (2(z\mathbf{D} - \mathbf{A}) + \rho(z-1)\mathbf{I}_n)^{-1} \tag{5.77}$$

thus,

$$\frac{\partial \mathbf{W}(z,\rho)}{\partial \rho} = -\mathbf{W}(z,\rho)\frac{\partial \mathbf{W}^{-1}(z,\rho)}{\partial \rho}\mathbf{W}(z,\rho) = -(z-1)\mathbf{W}^2(z,\rho) \tag{5.78}$$

and, since $\mathbf{W}(z,\rho)$ is symmetric, it holds that

$$\bar{S}_\rho(z) = \operatorname{tr}\left[\mathbf{W}^{-\top}(z,\rho)\frac{\partial \mathbf{W}(z,\rho)}{\partial \rho}\right] = -(z-1)\operatorname{tr}\left[\mathbf{W}(z,\rho)\right]. \tag{5.79}$$

Therefore, for Σ_ρ, one has

$$\begin{aligned}
\bar{S}_\rho(1) &= \lim_{z\to 1} -(z-1)\operatorname{tr}\left[(2(z\mathbf{D} - \mathbf{A}) + \rho(z-1)\mathbf{I}_n)^{-1}\right] && (5.80)\\
&= \lim_{z\to 1} -(z-1)\frac{\operatorname{tr}\left[\operatorname{adj}\left[2(z\mathbf{D} - \mathbf{A}) + \rho(z-1)\mathbf{I}_n\right]\right]}{\det\left[(2\mathbf{D} + \rho\mathbf{I}_n)(z\mathbf{I}_n - \mathbf{F}_\rho)\right]} && (5.81)\\
&= \lim_{z\to 1} -(z-1)\frac{2^{n-1}\operatorname{tr}\left[\operatorname{adj}\left[(z\mathbf{D} - \mathbf{A}) + \rho/2(z-1)\mathbf{I}_n\right]\right]}{(z-1)2^n\prod_{i=1}^n\left[\deg(v_i) + \rho/2\right]\prod_{i=1}^{n-1}\left(z - \lambda_i^{\mathbf{F}_\rho}\right)} && (5.82)\\
&= -\frac{\operatorname{tr}\left[\operatorname{adj}\left[\mathbf{D} - \mathbf{A}\right]\right]}{2\prod_{i=1}^n\left[\deg(v_i) + \rho/2\right]\prod_{i=1}^{n-1}\left(1 - \lambda_i^{\mathbf{F}_\rho}\right)} && (5.83)\\
&= -\frac{\operatorname{tr}\left[\operatorname{adj}\left[\mathbf{L}\right]\right]}{2\prod_{i=1}^n\left[\deg(v_i) + \rho/2\right]\prod_{i=1}^{n-1}\lambda_i^{\boldsymbol{\mathcal{L}}_\rho}} && (5.84)
\end{aligned}$$

where $\boldsymbol{\mathcal{L}}_\rho = \left(\mathbf{D} + \frac{\rho}{2}\mathbf{I}_n\right)^{-1/2}\mathbf{L}_\rho\left(\mathbf{D} + \frac{\rho}{2}\mathbf{I}_n\right)^{-1/2}$ is the normalized Laplacian induced by the graph $\mathcal{G}_\rho$ modified according to the PP-based approach. Remarkably, it also holds that $\mathbf{L}_\rho = \left(\mathbf{D} + \frac{\rho}{2}\mathbf{I}_n\right) - \left(\mathbf{A} + \frac{\rho}{2}\mathbf{I}_n\right) = \mathbf{L}$ and hence, similarly to the calculations shown for the numerator of (5.71)-(5.75), it is possible to obtain

$$n\mathfrak{t}(\mathcal{G}) = \prod_{i=1}^{n-1}\lambda_i^{\mathbf{L}} = \operatorname{tr}\left[\operatorname{adj}\left[\mathbf{L}\right]\right] = \operatorname{tr}\left[\operatorname{adj}\left[\mathbf{L}_\rho\right]\right] = \prod_{i=1}^{n-1}\lambda_i^{\mathbf{L}_\rho} = n\mathfrak{t}(\mathcal{G}_\rho) \tag{5.85}$$

resorting to Kirchhoff's matrix tree theorem (see Subsec. B.3.2) where $\mathfrak{t}(\mathcal{G})$ is the number of spanning trees in $\mathcal{G}$. Hence, exploiting (5.85), Theorem 2.1 of Huang and Li (2015) and the Handshaking lemma (see Subsec. B.3.1), relative sensitivity $\bar{S}_\rho(1)$ in (5.84) can be rewritten as

$$\begin{aligned}
\bar{S}_\rho(1) &= -\frac{n\mathfrak{t}(\mathcal{G}_\rho)}{2\cdot 2\left|\mathcal{E}_\rho\right|\mathfrak{t}(\mathcal{G}_\rho)} && (5.86)\\
&= -\frac{n}{2\sum_{i=1}^n\left[\deg(v_i) + \rho/2\right]} && (5.87)\\
&= -\frac{1}{2\mathrm{vol}(\mathcal{G})/n + \rho}. && (5.88)
\end{aligned}$$

Sensitivity of Σ_ϵ

For what concerns scheme Σ_ϵ, its discrete transfer function is given by

$$\mathbf{W}(z,\epsilon) = (z\mathbf{I}_n - \mathbf{F}_\epsilon)^{-1}\frac{1}{2}\epsilon = \left(2\left(\frac{z-1}{\epsilon}\mathbf{I}_n + \mathbf{L}\right)\right)^{-1} \tag{5.89}$$

thus,

$$\frac{\partial \mathbf{W}(z,\epsilon)}{\partial \epsilon} = -\mathbf{W}(z,\epsilon)\frac{\partial \mathbf{W}^{-1}(z,\epsilon)}{\partial \epsilon}\mathbf{W}(z,\epsilon) = \frac{2(z-1)}{\epsilon^2}\mathbf{W}(z,\epsilon)^2 \tag{5.90}$$

and, since $\mathbf{W}(z,\epsilon)$ is symmetric, it holds that

$$\bar{S}_\epsilon(z) = \operatorname{tr}\left[\mathbf{W}^{-\top}(z,\epsilon)\frac{\partial \mathbf{W}(z,\epsilon)}{\partial \epsilon}\right] = \frac{2(z-1)}{\epsilon^2}\operatorname{tr}\left[\mathbf{W}(z,\epsilon)\right]. \tag{5.91}$$

Therefore, for Σ_ϵ one has

$$\bar{S}_\epsilon(1) = \lim_{z\to 1}\frac{2(z-1)}{\epsilon^2}\operatorname{tr}\left[\frac{1}{2}\left(\frac{z-1}{\epsilon}\mathbf{I}_n + \mathbf{L}\right)^{-1}\right] \tag{5.92}$$

$$= \lim_{z\to 1}\frac{z-1}{\epsilon^2}\frac{\operatorname{tr}\left[\operatorname{adj}\left[\frac{z-1}{\epsilon}\mathbf{I}_n + \mathbf{L}\right]\right]}{\det\left[\frac{z-1}{\epsilon}\mathbf{I}_n + \mathbf{L}\right]} \tag{5.93}$$

$$= \lim_{z\to 1}\frac{z-1}{\epsilon^2}\frac{\operatorname{tr}\left[\operatorname{adj}\left[\mathbf{L}\right]\right]}{\prod_{i=0}^{n-1}\left(\frac{z-1}{\epsilon} + \lambda_i^{\mathbf{L}}\right)} \tag{5.94}$$

$$= \frac{1}{\epsilon^2}\frac{\operatorname{tr}\left[\operatorname{adj}\left[\mathbf{L}\right]\right]}{\frac{1}{\epsilon}\prod_{i=0}^{n-1}\lambda_i^{\mathbf{L}}} \tag{5.95}$$

Hence, exploiting again Kirchhoff's matrix tree theorem as in (5.85), it is then possible to obtain

$$\bar{S}_\epsilon(1) = \frac{1}{\epsilon}. \tag{5.96}$$

5.4.2 Final remarks about the sensitivity analysis

To conclude this discussion, the computation of the relative sensitivity at steady-state, i.e. as $z \to 1$, yields

- $|\bar{S}_\eta(1)| = (1-\eta)^{-1}$ for Σ_η;
- $|\bar{S}_\rho(1)| = (2\mathrm{vol}(\mathcal{G})/n + \rho)^{-1}$ for Σ_ρ;
- $|\bar{S}_\epsilon(1)| = \epsilon^{-1}$ for Σ_ϵ.

Now, let us consider inequality

$$1 + d_M \leq \rho_S(\mathbf{L}) \leq 2d_M \tag{5.97}$$

in Liu and Lu (2010) (with the equality on the r.h.s. holding for bipartite topologies) and denote with d_m the minimum degree in $\mathcal{G}$. It is worth to observe that, for any connected undirected network with $n \geq 2$ nodes, it holds that: $|\bar{S}_\eta(1)| \geq 1$ for all $\eta \in [0,1)$, $|\bar{S}_\rho(1)| \leq n/(2\text{vol}(\mathcal{G})) \leq 1/2$ for all $\rho \geq 0$ and $|\bar{S}_\epsilon(1)| > \rho_S(\mathbf{L})/2 \geq 1$ for all $\epsilon \in (0, 2/\rho_S(\mathbf{L})) \subseteq (0,1)$. This fact implies that scheme Σ_ρ is inherently more robust than the others in a steady state condition as their tunable parameters vary. On the contrary, scheme Σ_ϵ is the least reliable iterative procedure as ϵ is perturbed, since one has $|\bar{S}_\epsilon(1)| \geq 2$ for $\epsilon \in (0, 1/2]$ and $|\bar{S}_\eta(1)| \leq 2$ for $\eta \in (0, 1/2]$, being the interval $(0, 1/2]$ the most significant for selecting an optimal value of $\epsilon^\star$. Indeed, for connected undirected networks with $n > 2$ and $d_M > 2$, exploiting (5.97) and the Cheeger's bound for the algebraic connectivity given in Chung (1997), it is possible to show that the optimal values $\epsilon^\star$ belong to $(0, 1/2)$. Precisely, one has

$$\begin{cases} \lambda^{\mathbf{L}}_{n-1} = \rho_S(\mathbf{L}) \geq 1 + d_M \geq 4; \\ \lambda^{\mathbf{L}}_1 > \dfrac{2d_m}{n^4} \geq \dfrac{2}{n^4} \end{cases} \quad \Rightarrow \quad \epsilon^\star = \frac{2}{\lambda^{\mathbf{L}}_1 + \lambda^{\mathbf{L}}_{n-1}} < \frac{1}{2} \tag{5.98}$$

Additionally, exploiting Cor. 6.3.2 in the next chapter and relation (5.97) in the bipartite case, it is possible to prove without any difficulty that for cycle and path graphs (the only connected topologies with $d_M = 2$) it holds that $\epsilon^\star \in (0, 1/2]$. Therefore, the domain of all $\epsilon^\star$ can be restricted to $(0, 1/2]$, for any connected undirected graph with $n \geq 2$ nodes. Given the similar role of ϵ and η in schemes Σ_ϵ and Σ_η, the same restriction occurs for the optimal choice of latter parameter, since all $\eta^\star$ belong to $[0, 1/2]$ by Prop. 5.3.3 and Lem. B.3.66. These considerations can be also appreciated in Fig. 5.3.

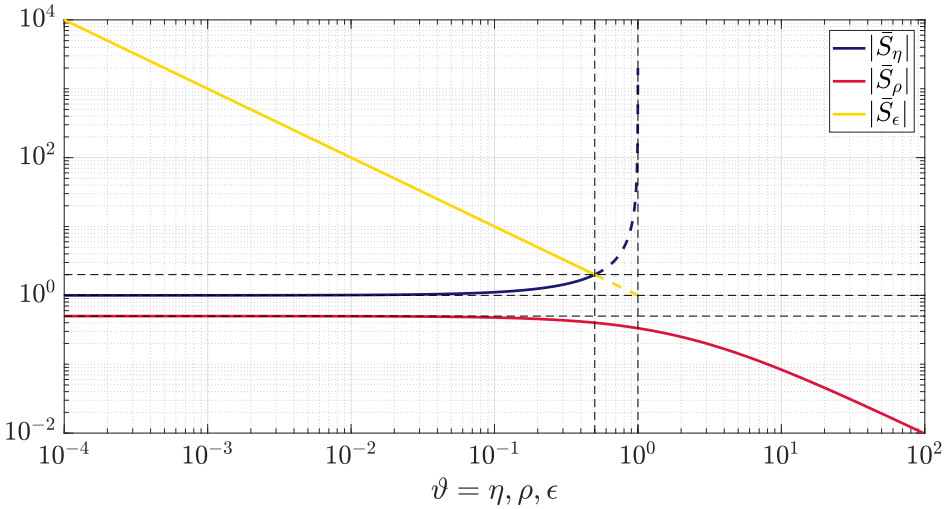


Figure 5.3. Sensitivity functions: qualitative comparison as the parameters $\vartheta = \eta, \rho, \epsilon$ vary, considering networks with $n \geq 2$ nodes. The red curve representing $|\bar{S}_\rho|$ assumes values that are smaller or equal to $1/2$, for all $\rho \geq 0$. The yellow curve representing $|\bar{S}_\epsilon|$ assumes values that are greater than 1 for all $\epsilon \in (0,1)$: its most important piece is, however, described by $\epsilon^\star \in (0, 1/2]$. The blue curve representing $|\bar{S}_\eta|$ assumes values that are greater or equal to 1 for all $\eta \in [0,1)$: its most important piece is, however, described by $\eta^\star \in [0, 1/2]$.

5.5 Numerical results

In this section, numerical results for schemes Σ_0, Σ_η and Σ_ρ, in (5.9), (5.12) and (5.20) respectively, are shown for different graph topologies, focusing on a particular setup that is suggestive for potential localization applications, such as the angle calibration for a planar camera network (see also Ex. 2.1.9). For the sake of clarity, scheme Σ_ϵ in (5.24) is not taken into account[4] in the following paragraph dedicated to simulation examples on convergence behaviors and sensitivity to parameter variations.
In this setup, $\mathrm{l} = 1$ has assigned[5] and a symmetric configuration has been chosen so that the estimations are expected to converge toward $x^\star_{i+1} - x^\star_i = 2\pi/n$, for $i = 1, \dots, n-1$. The relative measurements are assumed to be corrupted by an additive uniformly distributed noise in the range $[-\pi/36, \pi/36]$. In each simulation, the number of iterations is fixed to $\bar{t} = 20$ and the performance index

$$\mathrm{r}_e = \frac{1}{\bar{t} - 2t_{neg}} \sum_{t=t_{neg}}^{\bar{t}-t_{neg}-1} \frac{\|\mathbf{x}(t+1) - \mathbf{x}^\star\|_2}{\|\mathbf{x}(t) - \mathbf{x}^\star\|_2} \leq \mathrm{r} \tag{5.99}$$

is used to evaluate the effective convergence rate between estimations and centralized solutions, whenever convergence occurs. Index (5.99) neglects the first and last $t_{neg} = 2$ iterations and takes into account the average estimation error ratio over the procedure. Moreover, the final mean square error

$$\mathrm{MSE} = \frac{1}{n} \sum_{i=1}^{n} \left(x_i(\bar{t}) - x^\star_i\right)^2 \tag{5.100}$$

is also taken into account to assess the estimation performances[6] of the iterative schemes.

5.5.1 Case study: nonregular bipartite topology

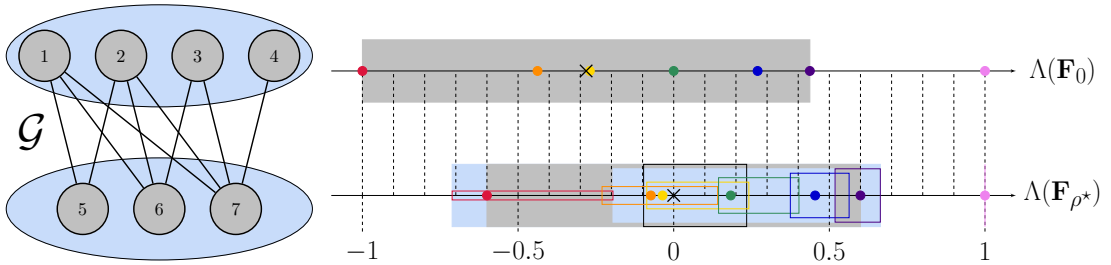
In this case study the nonregular bipartite network with $n = 7$ nodes in Fig. 5.4(a) is considered (see also Subsec. B.3.3). In the first paragraph, a simulation showing the convergence behavior of both the functional in (5.1) and the expected estimates is proposed via scheme Σ_ρ; whereas, in the second paragraph, a performance comparison between schemes Σ_η and Σ_ρ focused on this application is discussed. Further numerical results concerning scheme Σ_η are also available in Fabris et al. (2019b).

[4] This choice is motivated by various reasons; however, the most relevant are: (i) scheme Σ_η represents the normalized version of Σ_ϵ, thus they share very similar convergence properties; (ii) in Subsec. 5.4.2, it has been proven that the sensitivity to parameter variation of Σ_ϵ is the highest w.r.t. the other schemes.
[5] This choice is adopted w.l.o.g. and for the sake of clarity in the forthcoming illustrations.
[6] Each state component appearing in (5.99) and (5.100) has to be considered normalized in difference by, for instance, subtracting the first coordinate to all the others.

Distributed estimation through Σ_ρ as ρ varies in $[0, +\infty)$

Bipartite network $\mathcal{G}$ in Fig. 5.4(a) is the object of this case study. As it is illustrated in Fig. 5.4(b), eigenvalues belonging to the spectrum $\Lambda(\mathbf{F}_0)$ of state matrix $\mathbf{F}_0$ are mapped into spectrum $\Lambda(\mathbf{F}_{\rho^\star})$, when $\rho = \rho^\star$ is set. Moreover, recalling Lem. 5.3.6 and Prop. 5.3.7, all the the eigenvalues $\lambda_i^{\mathbf{F}_{\rho^\star}}$ are located within a certain neighborhood and maintain their order w.r.t. the elements in $\Lambda(\mathbf{F}_0)$. In addition to this fact, knowing the spectrum of the normalized Laplacian $\Lambda(\boldsymbol{\mathcal{L}})$, Prop. 5.3.8 entails that penalty parameter $\rho^\star$ belongs to the interval $[\underline{\rho}^\star, \overline{\rho}^\star] \simeq [0.5626, 2.2506]$; hence, the convergence rate $\mathrm{r}_{\rho^\star}$ can be bounded in the interval $[\underline{\mathrm{r}}_{\rho^\star}, \overline{\mathrm{r}}_{\rho^\star}] \simeq [0.5184, 0.7118]$.

Given the fact that $\mathcal{G}$ is bipartite, $\mathrm{r}_0 = 1$ trivially holds; then, the convergence of both cost and estimates by means of system Σ_0 does not occur, as depicted by the red lines in Fig. 5.5(a) and Fig. 5.5(b). Indeed, neither the cost function h approaches zero nor the estimates converge towards the centralized solution. On the other hand, this issue is easily overcome when the PP-based scheme is implemented: the choice of the penalty $\rho^\star \simeq 1.3469$ not only guarantees convergence but it also speeds up the estimation dynamics, which achieves the best possible convergence rate, as inequality $\mathrm{r}_0 \geq \mathrm{r}_\rho > \mathrm{r}_{\rho^\star} \simeq 0.5999$ indicates. Another evidence of this fact can be found in the values assumed by the final mean square errors. For the particular instance generated and depicted in Fig. 5.5(a) and Fig. 5.5(b), scheme Σ_0 yields MSE $\simeq 5.8291$; whereas, scheme $\Sigma_{\rho^\star}$ yields MSE $\simeq 2.7485 \cdot 10^{-4}$. To conclude this analysis, the values of the performance indexes $\mathrm{r}_e(\Sigma_0) \simeq 0.6034$ for Σ_0 and $\mathrm{r}_e(\Sigma_{\rho^\star}) \simeq 0.0471$ for $\Sigma_{\rho^\star}$ are also provided; however, the former value does not actually carry any useful information, since convergence is not attained in that specific case.

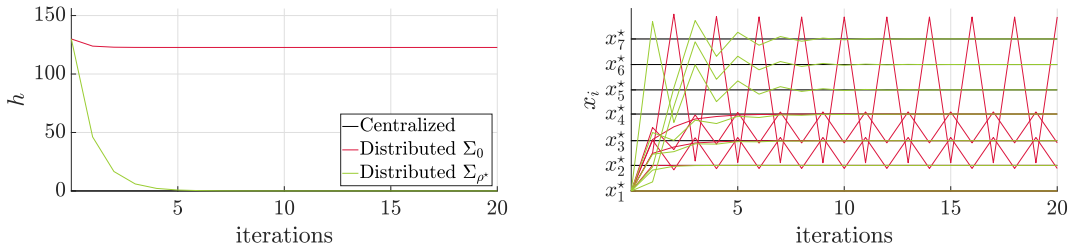


(a) Case study network (b) Spectral properties of algorithms

Figure 5.4. (a) A non-regular bipartite graph with 7 nodes is adopted in this case study. (b) Spectral mapping due to the tuning of penalty parameter $\rho^\star$. Eigenvalues (colored dots) belonging to spectrum $\Lambda(\mathbf{F}_{\rho^\star})$ are located inside their relative bounds (colored boxes). Penalty optimization allows the spectral centroid (black cross) to be placed in 0 and hence the essential spectral radius (see gray rectangle) is reduced.

Performance comparison between Σ_η and Σ_ρ

Again, let us consider bipartite network introduced in Fig. 5.4(a). Fig. 5.6 highlights the most important aspects regarding an overall comparison among the developed distributed schemes. Firstly, it is worth to note that, for each implementation of schemes Σ_η and Σ_ρ, regularizing parameters η and ρ play a key role for their convergence rate, as illustrated previously. These two quantities can be tuned accordingly to Fig. 5.6(a), by minimizing



(a) Cost function

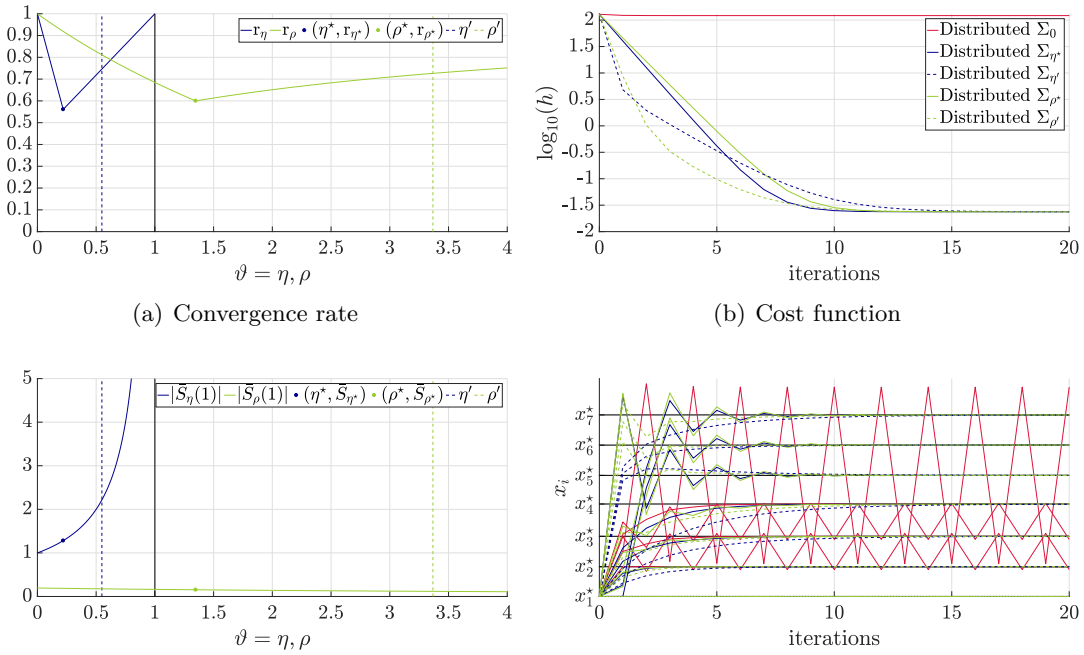
(b) Estimation dynamics

Figure 5.5. (a) Cost function decrease for systems Σ_0 and $\Sigma_{\rho^\star}$. The minimum for h is not attained by Σ_0 scheme, since graph $\mathcal{G}$ is bipartite. (b) Behavior of the iterative schemes. It is worth to notice that Σ_0 dynamics is oscillatory around the estimations while $\Sigma_{\rho^\star}$ dynamics is converging to the centralized solution.

Regularization	MSE(Σ_0)	MSE(Σ_η)	MSE(Σ_ρ)	$r_e(\Sigma_0)$	$r_e(\Sigma_\eta)$	$r_e(\Sigma_\rho)$
$\eta = \eta^\star$, $\rho = \rho^\star$	5.8157	$7.0147 \cdot 10^{-5}$	$2.7377 \cdot 10^{-4}$	–	0.0375	0.0471
$\eta = \eta'$, $\rho = \rho'$	5.8157	0.0082	0.0027	–	0.0494	0.0259

Table 5.4. Performance indexes for iterative schemes Σ_η and Σ_ρ acting on the bipartite graph reported in Fig. 5.4(a).

the curves relative to convergence rates r_η and r_ρ. In this particular settings, the optimal choice is given by $\eta^\star \simeq 0.2196$ and $\rho^\star \simeq 1.3469$, leading to $r_{\eta^\star} \simeq 0.5609 < r_{\rho^\star} \simeq 0.5999$. This fact can also be observed in the decrease of cost function h (see Fig. 5.6(b)) or in the estimation dynamics (provided in Fig. 5.6(d)), where schemes $\Sigma_{\eta'}$ and $\Sigma_{\rho'}$, with $\eta' := 1.5\eta^\star$ and $\rho' := 1.5\rho^\star$ are also taken into consideration to illustrates a potential deviation from the optimal $\eta^\star$ and $\rho^\star$.
Secondly, one observes that, in all cases, sensitivity to parameter variations in a steady-state condition is larger for procedure Σ_η. Indeed, this fact is the main reason for which, even though $r_{\eta^\star} < r_{\rho^\star}$ and parameter η' is set by means of the same perturbation applied to parameter ρ', convergence rate $r_{\eta'} \simeq 0.7462$ ends up being greater than $r_{\rho'} \simeq 0.7263$. In other words, when the relative variation from the optimal regularizing parameter exceeds a certain thresholds, it is guaranteed that scheme Σ_η exhibits deteriorated convergence performances w.r.t. those of scheme Σ_ρ. This holds in a general setting, since (i) if η goes to 1 then r_η tends to 1 because estimates are computed resorting only to their previous values and neglecting the rest of the network, meaning that the estimation result is provided with a large number of iterations; (ii) if η goes to 0 then r_η tends to r_0, as r_ρ does for ρ going to 0, slowing down the convergence of dynamics as a consequence. More generally, the latter statement explains the convergence performances of scheme Σ_0, which might be too slow or even presents oscillatory dynamics in the worst case scenario where bipartite topologies are employed, e.g. in this case study.
Another evidence supporting this fact can be found in Table 5.4 where mean square errors MSE and the effective estimation rate r_e are reported for each iterative scheme involved in this analysis.



(a) Convergence rate

(b) Cost function

(c) Sensitivity to parameter variations

(d) Estimation dynamics

Figure 5.6. Overall performance comparison relative to the developed distributed approaches, given the topology presented in Fig. 5.4(a). (a) Convergence rate minimization for both schemes Σ_η and Σ_ρ as η and ρ vary. (b) Decrease of the cost function for schemes Σ_0, Σ_η, $\Sigma_{\eta'}$, Σ_ρ and $\Sigma_{\rho'}$. (c) Sensitivity to parameter variations for both schemes Σ_η and Σ_ρ as η and ρ vary. (d) Estimation dynamics for schemes Σ_0, Σ_η, $\Sigma_{\eta'}$, Σ_ρ and $\Sigma_{\rho'}$.

5.5.2 Overall performance comparison for various topologies

In the following lines, several numerical results are also provided for a large variety of various topologies. In particular, this simulation section is concluded by validating the proposed algorithms for other network topologies (see also Subsec. B.3.3). In more details, Table 5.5 summarizes the parameter settings and all the main results related to the convergence behaviors. Furthermore, performances for tests accomplished on part of the graphs evaluated in Fabris et al. (2019b) are shown, e.g. the complete K_{36}, the circulant $C_{36}(1,2)$, a Ramanujan $\mathcal{R}_{36}$ and the Cayley $\Gamma_{36}(3)$, all with $n = 36$ nodes. In addition, other three topologies are also investigated:

- the star topology $\mathcal{S}_{36}$ with $n = 36$, i.e. a tree with 35 nodes characterized by a degree of 1 linked to a common root with 35 incidences;
- the small-world $SW_{9,27}$ with $n = 36$, constituted by two complete graphs K_9 and K_{27} linked together by one single edge;
- the complete binary trees $\mathcal{B}_4^+$, $\mathcal{B}_6^+$ (where $\mathcal{B}_{n_d}^+$ has 2^{n_d+1} vertices, hence respectively with $n = 32$ and $n = 128$ nodes) with an extra connection between its root and one of the leaves.

Lastly, some other graph-related quantities are also reported in Table 5.5, such as the

Topology	K_{36}	$C_{36}(1,2)$	$\mathcal{R}_{36}$	$\Gamma_{36}(3)$	$SW_{9,27}$	$\mathcal{S}_{36}$	$\mathcal{B}_4^+$	$\mathcal{B}_6^+$
Regular	yes	yes	yes	yes	no	no	no	no
Bipartite	no	no	no	yes	no	yes	no	no
Equivalence	yes	no	yes	yes	no	no	no	no
dens	1	0.1143	0.0857	0.0857	0.6159	0.0556	0.0667	0.0159
$\phi(\mathcal{G})$	1	9	6	6	3	2	8	12
$d_{av}(\mathcal{G})$	35	4	3	3	21.5556	1.9444	2	2
$d_m(\mathcal{G})$	35	4	3	3	8	1	1	1
$d_M(\mathcal{G})$	35	4	3	3	27	35	3	3
vol(()$\mathcal{G}$)	1260	144	108	108	776	70	62	254
$\lambda_1^{\mathbf{L}}$	36	0.1510	0.3408	0.3542	0.1304	1	0.0476	0.0096
$\lambda_1^{\boldsymbol{\mathcal{L}}}$	1.0286	0.0377	0.1136	0.1181	0.0133	1	0.0261	0.0050
$\lambda_{n-1}^{\mathbf{L}}$	36	6.2267	5.7326	6	28.0158	36	5.3787	5.5894
$\lambda_{n-1}^{\boldsymbol{\mathcal{L}}}$	1.0286	1.5567	1.9109	2	1.1456	2	1.9888	1.9980
$\varsigma_{\mathbf{L}}$	36	3.1888	3.0367	3.1771	14.0731	18.5	2.7131	2.7995
$\varsigma_{\boldsymbol{\mathcal{L}}}$	1.0286	0.7972	1.0122	1.0590	0.5795	1.5	1.0074	1.0015
$\eta^\star$	0.0278	0	0.0121	0.0557	0	0.3333	0.0074	0.0015
$\rho^\star$	2	0	0.0734	0.3542	0	1.8972	0.0283	0.0059
$\epsilon^\star$	0.0278	0.3136	0.3293	0.3148	0.0711	0.0541	0.3686	0.3572
r_0	0.0286	0.9623	0.9109	1	0.9867	1	0.9888	0.9980
$\mathrm{r}_{\eta^\star}$	$1.5543 \cdot 10^{-15}$	0.9623	0.8878	0.8885	0.9867	0.3333	0.9741	0.9950
$\mathrm{r}_{\rho^\star}$	0	0.9623	0.8878	0.8885	0.9867	0.4868	0.9741	0.9950
$\mathrm{r}_{\epsilon^\star}$	$1.3323 \cdot 10^{-15}$	0.9526	0.8878	0.8885	0.9907	0.9459	0.9825	0.9966
$\mathrm{r}_e(\Sigma_0)$	$5.2521 \cdot 10^{-5}$	0.2952	0.1842	–	0.7279	–	0.3298	0.4604
$\mathrm{r}_e(\Sigma_{\eta^\star})$	$7.7972 \cdot 10^{-16}$	0.2952	0.1478	0.0258	0.7279	0.0104	0.2871	0.4591
$\mathrm{r}_e(\Sigma_{\rho^\star})$	$8.0757 \cdot 10^{-16}$	0.2952	0.1478	0.0258	0.7279	0.0289	0.2922	0.4589
$\mathrm{r}_e(\Sigma_{\epsilon^\star})$	$7.7015 \cdot 10^{-16}$	0.3655	0.1478	0.0258	0.7029	0.6094	0.1957	0.5213
MSE(Σ_0)	$1.5916 \cdot 10^{-14}$	3.9867	0.4352	0.2509	14.4911	18.5854	2.1560	12.2234
MSE($\Sigma_{\eta^\star}$)	$1.6554 \cdot 10^{-14}$	3.9867	0.3474	0.0250	14.4911	$6.1042 \cdot 10^{-9}$	1.8040	12.1312
MSE($\Sigma_{\rho^\star}$)	$1.7156 \cdot 10^{-14}$	3.9867	0.3474	0.0250	14.4911	$1.1890 \cdot 10^{-5}$	1.7974	12.1263
MSE($\Sigma_{\epsilon^\star}$)	$1.6391 \cdot 10^{-14}$	3.5284	0.3474	0.0250	15.4689	7.0045	1.8190	12.1636

Table 5.5. Summary of the validation test for iterative schemes Σ_η and Σ_ρ in a heterogeneous topology framework. The 'Equivalence' tag indicates the existence of a full bijection between the different iterative schemes, according to Subsec. 5.2.3.

average degree $d_{av}(\mathcal{G}) := n^{-1}\sum_{i=1}^n \deg(v_i)$, the density $\mathrm{dens}(\mathcal{G}) = \mathrm{vol}(\mathcal{G})/(n(n-1))$ and the diameter $\phi(\mathcal{G})$ (see also Subsec. B.3.1).

From the data reported in Table 5.5, some considerations are in order:

- K_{36} and $\mathcal{R}_{36}$: the estimation performance is good for Σ_0, $\Sigma_{\eta^\star}$, $\Sigma_{\rho^\star}$ and $\Sigma_{\epsilon^\star}$, although the latter three methods allows for a faster convergence;

- $C_{36}(1,2)$: for these specific (regular) graphs the performances of Σ_0, $\Sigma_{\eta^\star}$, $\Sigma_{\rho^\star}$ and $\Sigma_{\epsilon^\star}$ are practically the same[7], for both indexes r_e and MSE (as a further remark, it is worth to note that for some circulant graphs as the ring graphs the performance results better by employing approaches Σ_η, Σ_ρ, Σ_ϵ);

- $\Gamma_{36}(3)$ and $\mathcal{S}_{36}$: for the bipartite nature of these graphs, only the PP-based estimation through Σ_ρ or the consensus schemes Σ_η, Σ_ϵ can be applied successfully ($\Gamma_{36}(3)$ is regular and $\mathcal{S}_{36}$ is not); however, scheme Σ_ϵ does not exhibit a good

[7]The lack of a full equivalence leads $\Sigma_{\epsilon^\star}$ to better performances.

convergence rate w.r.t. to the other iterative procedures in the star graph case (proven by the fact that its MSE is higher than the others);

- $SW_{9,27}$: the update rules Σ_0, Σ_η, Σ_ρ perform equally, with Σ_ϵ performing slightly worse from the others for this non-regular graph choice;
- $\mathcal{B}_4^+$ and $\mathcal{B}_6^+$: estimation performances are here similar between the four methods, nonetheless, the regularized procedures Σ_η, Σ_ρ, Σ_ϵ obtain better results both in estimation speed and estimation accuracy, with Σ_ϵ performing slightly worse.

By recalling Remark 5.3.9, the table highlights how bipartite and complete graphs improve information spreading through their characteristic connectivity patterns (with $\varsigma_{\mathcal{L}} > 1$ and adopted estimation scheme $\Sigma_{\rho^\star}$), while small world networks tend to isolate information in node clusters (with $\varsigma_{\mathcal{L}} < 1$ and adopted estimation scheme Σ_0).

5.6 Chapter summary

In this chapter, the optimal multivariate distributed state estimation problem from relative measurements, strongly motivated by the large use of multi-agent sensor networks, has been tackled and solved with different approaches. Specifically, a centralized solution has been provided, along with a distributed methodology that allows to formalize four iterative linear procedures. The first distributed scheme resulting from this approach, i.e. Σ_0, does not exhibit a convergent behavior for bipartite topologies; nonetheless, it represents the cornerstone for the application of three consensus-like schemes (Σ_η, Σ_ρ, Σ_ϵ), which yield general convergence properties to the system of sensors by tuning a regularization parameter opportunely. In particular, it is shown that the convergence performances of the three aforementioned distributed and regularized algorithms are topology-dependent and similar, excluding the general choice among one of the schemes. However, it is worth to underline that the PP-based scheme Σ_ρ is proven to be the most robust algorithm, among those presented, in terms of regularization parameter variations, for any given topology.

6

On the Spectral Properties of κ-ring Graphs

"Do not try and bend the spoon. That's impossible. Instead, only realize the truth... there is no spoon."
Child

Contents

6.1 Overview

Nowadays, Graph Spectral Theory is ubiquitous in a significant amount of diverse scientific fields Beineke and Wilson (1997), especially in numerous areas related to Engineering, as, for instance, Machine Learning Bekkerman, Bilenko, and Langford (2012), Data Mining

Cook and Holder (2007), Multi-agent Networks Mesbahi and Egerstedt (2010) and Social Networks Wasserman and Faust (1999). Many distributed algorithms for the estimation and control are implemented among the most heterogeneous kind of topologies, e.g. in Cui, Wang, Pei, and Zhu (2018); Grönkvist and Hansson (2001); Pourbabak, Luo, Chen, and Su (2018); Wang and Elia (2013).
Given the dramatic growth in the usage of these networked-based algorithms, the knowledge about eigenvalues of the Laplacian matrices Zhang (2011) and normalized Laplacian matrices Chung (1997) have gradually become fundamental to achieve a deep comprehension on the convergence performances and real effectiveness of such iterative schemes Makhdoumi and Ozdaglar (2017); Spielman (2007).
This research activity arose from the observations made in the Chap. 5 and Fabris et al. (2019b), in which the authors have shown how some algebraic entities connected to graphs are crucial to determine the rate of convergence for the adopted estimation approach. In particular, this chapter represents a marginal, yet significant, case study for that type of estimation from relative measurements applied to highly-symmetric distributed networks, e.g. a planar camera network for surveillance (see ex. 2.1.9) or scene reconstruction in which each node shares the same field-of-view and sensing capabilities. Nonetheless, this activity is undoubtedly the most prone to exhibit a theoretical approach to things, rather than showing MASs applications. For this reason, no problem statement is present.

To conclude this heading, Fig. 6.1 depicts the main aspects arising in this work – mainly related to Graph Theory and thus Combinatorial Graph Theory – to provide new results that might be exploited in Combinatorial Optimization and, more generally, Optimization Theory. Leveraging the graph-theoretic tools, the main objectives for this study are represented by the analysis of the spectral distribution of the eigenvalues of the Laplacian matrix induced by a specific class of graphs, named *κ-ring graphs*.

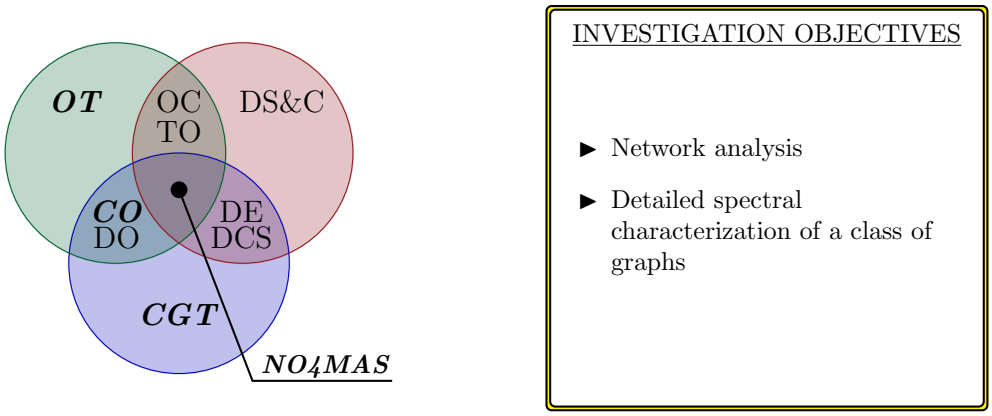


Figure 6.1. Theoretical fields and investigation objectives arising from this study.

6.1.1 Related works

Many spectral quantities related to graphs have been object of research over the years. The *Fiedler value* Fiedler (1973) has been widely investigated since 1969 with the work on the Cheeger's bound Cheeger (1969) and, in 1973, it was established that this quantity reflects how well-connected a graph is, delineating the gist of *algebraic connectivity* Li,

Guo, and Shiu (2013). Of the same relevance are the notions of *spectral radius* and *spread* Fan, Xu, Wang, and Liang (2008) for the Laplacian matrix. Afterwards, the graph spectral analysis was extensively exploited to compare performances for decentralized procedures as average consensus Lovisari and Zampieri (2012) and, more generally, is an efficient theoretical tool utilized in the vast field of Multi-Agent Systems Franceschelli, Gasparri, Giua, and Seatzu (2013).
Lastly, it is worth to recall that, in the last three decades, disparate researches on the eigenvalues of the Laplacian matrix have directed their focus on the bounds of the Fiedler value and other spectral quantities linked to the Randić matrix Banerjee and Mehatari (2016); Berman and Zhang (2000); Juvan and Mohar (1993); Kim and Mesbahi (2005); Rad, Jalili, and Hasler (2011); Rojo (2007); Shu, Hong, and Wen-Ren (2002); Sorgun (2013).

6.1.2 Contribution and outline of the chapter

As already disclosed, this chapter's contribution rests upon an exhaustive examination that aims at the spectral characterization of the graph Laplacian matrix induced by a certain class of topologies described by means of circulant adjacency matrices. Circulant matrices are very relevant in various research areas. Generally, they are employed to model networks of agents that share the same "local panorama" and are utilized – not only in the field of MASs – to model challenging problems and concepts: intelligent surveillance of public spaces Liu, Liu, Muhammad, Sangaiah, and Doctor (2018), tracking-by-detection Henriques, Caseiro, Martins, and Batista (2012), identification of sparse reciprocal graphical models Alpago, Zorzi, and Ferrante (2018), definition of *shift* in Graph Signal Processing Ortega, Frossard, Kovacević, Moura, and Vandergheynst (2018), modeling of quantum walks Seveso, Benedetti, and Paris (2019), video circulant sampling schemes Shu and Ahuja (2011), compressive 3D sensing techniques Antholzer, Wolf, Sandbichler, Dielacher, and Haltmeier (2019) and sensor network monitoring algorithms Gastpar and Vetterli (2005) represent few state-of-the-art topics that motivate the research in this work.
More specifically, several dependencies and relationships between the eigenvalues of this kind of Laplacian matrices under analysis and the Dirichlet kernel have been discovered: a special effort is directed toward the characterization of algebraic connectivity, spectral radius and essential spectral radius of the graph Laplacian. The implications of this fact are then leveraged to investigate the essential spectral radius (ESR) of the Laplacian matrix related to this class of graphs.

The remainder of the chapter is organized as follows: in Sec. 6.2, some mathematical preliminaries offer an overview on the circulant topologies and the definition of *κ-ring graph* is established. The main results of this work is then presented in Sec. 6.3, where spectral properties of κ-ring graphs are widely explored and characterized. The dissertation continues with the discussion in Sec. 6.4, in which a conjecture is proposed. Finally, conclusions in Sec. 6.5 summarize all the debated issues.

6.2 Mathematical preliminaries

This dissertation begins by illustrating some preliminaries about circulant matrices and circulant graphs, showing consolidated algebraic relations between them. Then, this section carries on with the introduction of a particular class of circulant graphs, object of this study.

6.2.1 Circulant matrices and circulant graphs

A circulant matrix is a specific Toeplitz matrix[1] where each row is shifted one entry to the right relative to the previous row vector. In formulas, given the n-dimensional vector $\boldsymbol{\omega} := \begin{bmatrix} \omega_0 & \cdots & \omega_{n-1} \end{bmatrix}^\top \in \mathbb{R}^n$, the definition of a circulant matrix $\mathbf{F} \in \mathbb{R}^{n \times n}$ rests upon the relation right below:

$$[\mathbf{F}]_{i+1,j+1} = \omega_{(j-i) \mod n} \quad \text{for } i, j = 0, \dots, n-1; \tag{6.1}$$

hence, indicating with $\mathrm{circ}(\cdot)$ the so-called *gallery* or *shift* operator, matrix $\mathbf{F}$ can be defined as follows:

$$\mathbf{F} = \mathrm{circ}(\boldsymbol{\omega}) := \begin{bmatrix} \omega_0 & \omega_1 & \cdots & \omega_{n-2} & \omega_{n-1} \\ \omega_{n-1} & \omega_0 & \cdots & \omega_{n-3} & \omega_{n-2} \\ \vdots & \ddots & \dots & \ddots & \vdots \\ \omega_2 & \omega_3 & \cdots & \omega_0 & \omega_1 \\ \omega_1 & \omega_2 & \cdots & \omega_{n-1} & \omega_0 \end{bmatrix}. \tag{6.2}$$

Let us denote with $\Lambda(\cdot)$ the spectral operator that maps a matrix into the correspondent set of eigenvalues. The spectrum $\Lambda(\mathbf{F})$ of a circulant matrix $\mathbf{F}$ is well-known in literature Gray (2005) and its (unsorted) elements $\lambda^{\mathbf{F}}(j) \in \Lambda(\mathbf{F})$ are provided by

$$\lambda^{\mathbf{F}}(j) = \sum_{k=0}^{n-1} \left[\omega_k \exp\left(-\frac{2k\pi \mathbf{i}}{n} j \right) \right] \quad \text{for } j = 0, \dots, n-1. \tag{6.3}$$

where $\exp(\cdot)$ indicates the exponential function.

Few basic notions of Graph Theory (see also Sec. B.3) are briefly recalled in the following lines. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph composed by the set of vertices $\mathcal{V} = \{v_1, \dots, v_n\}$ and the set of edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. An edge e_{ij} belongs to $\mathcal{V}$ if and only if the couple of vertices (v_i, v_j) is connected by a link. By assumption, $\mathcal{G}$ is undirected, meaning that an edge e_{ij} can be equivalently indicated by e_{ji}, and self-loops are not allowed, namely $e_{ii} \notin \mathcal{E}$ for any i. Furthermore, by hypothesis, $\mathcal{G}$ is simple, i.e. there exists at most one edge e_{ij} connecting a couple (v_i, v_j), and its adjacency matrix is denoted by $\mathbf{A} \in \{0,1\}^{n \times n}$, such that $[\mathbf{A}]_{ij} = [\mathbf{A}]_{ji} = 1$, if $e_{ij} \in \mathcal{E}$, and $[\mathbf{A}]_{ij} = [\mathbf{A}]_{ji} = 0$, otherwise. For each vertex $v_i \in \mathcal{V}$, the neighborhood $\mathcal{N}_i = \{v_j | e_{ij} \in \mathcal{E}\}$ is determined by the set of edges that possess v_i as end node and its cardinality contributes to the definition of the degree

[1] i.e. a matrix $\boldsymbol{\Omega} \in \mathbb{R}^{N \times N}$ with constant diagonal entries, such that $[\boldsymbol{\Omega}]_{ij} = [\boldsymbol{\Omega}]_{i-1,j-1} \mod N$, where mod indicates the modulus operation.

$\deg(v_i) = |\mathcal{N}_i| = \sum_{j=1}^{n}[\mathbf{A}]_{ij} \geq 0$ of the vertex v_i. The latter quantities account for the notion of degree matrix $\mathbf{D} \in \mathbb{R}_{\geq 0}^{n\times n}$, which presents the following characterization: $[\mathbf{D}]_{ii} = \deg(v_i)$; $[\mathbf{D}]_{ij} = 0$ holding for all $i \neq j$. Moreover, $\mathcal{G}$ is said *regular* if and only if $\deg(v_i) = \deg(v_j) =: \mathrm{d}$ for all $i \neq j$. All the previous definitions are fundamental to recall the concepts Chung (1997) of graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{A}$ and normalized graph Laplacian[2] $\boldsymbol{\mathcal{L}} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2} = \mathbf{I}_n - \mathscr{R}$, where $\mathbf{I}_n$ denotes the identity matrix of dimension n and $\mathscr{R} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ represents the canonical Randić matrix Gutman, Furtula, and Bozkurt (2014) associated to graph $\mathcal{G}$.

Now, chosen the vector $\overline{\boldsymbol{\omega}}$ in $\{0,1\}^n$, with its first component $[\overline{\omega}]_1 = \overline{\omega}_0 = 0$, a circulant graph of n nodes is a regular undirected simple graph $\mathcal{G} := C_n(\overline{\boldsymbol{\omega}})$ that can be identified with an adjacency matrix of the form $\mathbf{A} = \mathrm{circ}(\overline{\boldsymbol{\omega}})$. Denoting with d the common degree among the vertices of $\mathcal{G}$ and imposing $\boldsymbol{\omega} := \mathrm{d}^{-1}\overline{\boldsymbol{\omega}}$, the generic matrix $\mathbf{F}$ defined in (6.2) assumes the structure of a Randić matrix:

$$\mathbf{F} := \mathbf{D}^{-1}\mathbf{A} = \mathrm{d}^{-1}\mathbf{A} = \mathrm{d}^{-1/2}\mathbf{A}\mathrm{d}^{-1/2} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2} = \mathscr{R}. \tag{6.4}$$

Since in this setting it holds that

$$\mathbf{L} = \mathrm{d}\boldsymbol{\mathcal{L}} = \mathrm{d}(\mathbf{I}_n - \mathscr{R}), \tag{6.5}$$

equality (6.4) allows to establish a direct relationship between the spectral properties of matrix $\mathbf{F}$ to those of the graph Laplacian $\mathbf{L}$. Indeed, as already shown in Fact 5.3.2, it is possible to leverage the fact that relation

$$\lambda^{\mathbf{F}}(j) = \lambda^{\mathscr{R}}(j) = 1 - \lambda^{\boldsymbol{\mathcal{L}}}(j) \quad \text{for } j = 0, \ldots, n-1 \tag{6.6}$$

holds for a generic undirected graph $\mathcal{G}$ without isolated nodes to study the spectral properties of $\mathbf{F}$ via the normalized graph Laplacian $\boldsymbol{\mathcal{L}}$ and vice versa.

6.2.2 A class of circulant graphs: κ-ring graphs

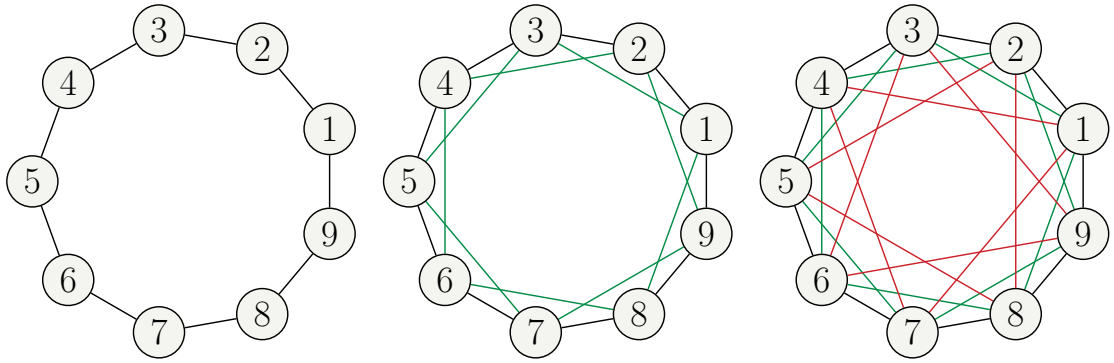
At the light of all the previous observations, a particular class of circulant graphs is taken into analysis hereafter, with the specific purpose of investigating its spectral properties. The elements belonging to the topology in question will be classified as *κ-ring graphs* $C_n(1,\kappa)$, where the constant $\kappa \in \mathbb{N}_{>0}$ represents the maximum length of the path that leads from one of the n nodes v_i to one of its neighbors v_j, neglecting the presence of all the other nodes that are not members of $\mathcal{N}_i$. In other words, for[3] $\kappa = 1, \ldots, \lfloor n/2 \rfloor - 1$, constant κ can be interpreted as the width of the identical panorama seen from each vertex, as depicted in Fig. 6.2. In addition, to provide further details in this description, it is worth to mention that κ-regular graphs are not only circulant but also connected, Hamiltonian and Eulerian, since this kind of topology is constructed by means of multiple edge layers beginning with the cycle of length n. Lastly, in Table 6.1, few standard quantities related to κ-ring graphs are summarized.

[2] Well-defined if and only if $\mathcal{G}$ has no isolated vertices.

[3] Values $\kappa = 0$ and $\kappa = \lfloor n/2 \rfloor$ are purposely excluded since they lead to degenerate well-known topologies, such as the void or complete graphs.

#Vertices	#Edges	Diameter	Radius	Girth	Regularity
$\lvert\mathcal{V}\rvert = n \geq 4$	$\lvert\mathcal{E}\rvert = n\kappa$	$\phi = \lceil n/2^{\kappa} \rceil$	$\mathfrak{r} = \phi$	$\mathrm{g} = \begin{cases} n, & \text{if } \kappa = 1 \\ 3, & \text{otherwise} \end{cases}$	$\mathrm{d} = 2\kappa$

Table 6.1. Basic topologic quantities of a κ-ring graph with n vertices.



(a) $C_9(1,1)$ (b) $C_9(1,2)$ (c) $C_9(1,3)$

Figure 6.2. All the three κ-ring graphs with $n = 9$ vertices. A layer of edges is added for each increasing value of κ: (a) first layer in black, (b) second layer in green, (c) third layer in red.

6.3 Spectral characterization of κ-ring graphs

In this section, the main results on the spectral properties of κ-ring graphs are given. Firstly, a spectral analysis of the graph Laplacian matrix $\mathbf{L}$ via the Dirichlet kernel is presented. The discussion yields a characterization of its spectrum $\Lambda(\mathbf{L})$, with particular attention directed towards the Fiedler value and its spectral radius. Then, the dissertation continues with a study on the so-called *essential spectral radius* of matrix $\mathbf{F}$.

6.3.1 Spectral analysis of the graph Laplacian matrix via the Dirichlet kernel

The analysis starts by showing the key insight to examine the spectral properties of κ-ring graphs through the theoretical support of the Dirichlet kernel $\mathcal{D}_{\natural} : \mathbb{R} \to \mathbb{R}$ of order $\natural \in \mathbb{N}$, which can be defined according to Brunckner, Brunckner, and Thomson (1997) as

$$\mathcal{D}_{\natural}(x) := \begin{cases} \dfrac{\sin((\natural + 1/2)x)}{2\sin(x/2)}, & \text{if } x \neq 2\pi\mathrm{l},\ \forall \mathrm{l} \in \mathbb{Z}; \\ \natural + 1/2, & \text{otherwise.} \end{cases} \tag{6.7}$$

A characterization for the eigenvalues of the graph Laplacian matrix $\mathbf{L}$ associated to the κ-ring graphs in terms of $\mathcal{D}_{\natural}$ is given by the following

Theorem 6.3.1 (Spectral characterization of κ-ring graphs)**.**
Let $\mathbf{L}$ *be the graph Laplacian matrix associated to a* κ*-ring graph* $C_n(1,\kappa)$*. Setting* $\theta := \pi/n$*, the eigenvalues* $\lambda^{\mathbf{L}}(j) \in \Lambda(\mathbf{L})$ *can be expressed in function of the Dirichlet*

kernel as

$$\lambda^{\mathbf{L}}(j) = 1 + 2\left(\kappa - \mathcal{D}_\kappa(2\theta j)\right), \qquad \text{for } j = 0, \dots, \lfloor n/2 \rfloor; \tag{6.8}$$

$$\lambda^{\mathbf{L}}(n-j) = \lambda^{\mathbf{L}}(j), \qquad \text{for } j = 1, \dots, \lfloor n/2 \rfloor. \tag{6.9}$$

Furthermore, each eigenvalue $\lambda^{\mathbf{L}}(j)$ belongs to $[0, 4\kappa]$ for all $j = 0, \dots, n-1$, eigenvalue $\lambda_0^{\mathbf{L}} := \lambda^{\mathbf{L}}(0) = 0$ is simple and, if there exists $j^\star \in \mathbb{N}$ such that $\lambda^{\mathbf{L}}(j^\star) = 4\kappa$ with $j^\star \in (0, n)$, then eigenvalue $\lambda^{\mathbf{L}}(j^\star)$ is simple.

Proof. Exploiting equality (6.3) and setting

$$[\boldsymbol{\omega}]_i := \begin{cases} \mathrm{d}^{-1}, & \text{if } e_{i1} \in \mathcal{E}; \\ 0, & \text{otherwise;} \end{cases} \tag{6.10}$$

the eigenvalues of the Randić matrix $\mathscr{R} = \mathbf{F}$ associated to graph $C_n(1, \kappa)$ can be rewritten as follows:

$$\lambda^{\mathscr{R}}(j) = \frac{1}{\mathrm{d}} \sum_{k=1}^{\mathrm{d}/2} [\exp(-\mathbf{i} 2k\theta j)] + \frac{1}{\mathrm{d}} \sum_{k=n-\mathrm{d}/2}^{n-1} [\exp(-\mathbf{i} 2k\theta j)] \tag{6.11}$$

$$= \frac{1}{\mathrm{d}} \sum_{k=1}^{\mathrm{d}/2} [\exp(-\mathbf{i} 2k\theta j)] + \frac{1}{\mathrm{d}} \sum_{k=1}^{\mathrm{d}/2} [\exp(\mathbf{i} 2k\theta j)] \tag{6.12}$$

$$= \frac{2}{\mathrm{d}} \left(\frac{1}{2} \sum_{|k| \le \mathrm{d}/2} [\exp(\mathbf{i} 2k\theta j)] - \frac{1}{2} \right). \tag{6.13}$$

Recalling that $\mathrm{d} = 2\kappa$ and applying Theorem 15.2 of Brunckner et al. (1997) to (6.13), it follows that the eigenvalues of the Randić matrix $\mathscr{R}$ are yielded by

$$\lambda^{\mathscr{R}}(j) = \kappa^{-1} \left(\mathcal{D}_\kappa(2\theta j) - 1/2 \right). \tag{6.14}$$

Therefore, leveraging (6.5) and (6.6), equality (6.8) is proven. Moreover, since the Dirichlet kernel is a 2π-periodic even function, equality (6.9) is also verified.

Lastly, regarding the rest of the statement, authors in Landau and Odlyzko (1981) have already shown that matrix $\mathscr{R}$ possesses eigenvalues belonging to the interval $[-1, 1]$, where $\lambda^{\mathscr{R}}(0) = 1$ and, possibly, $\lambda^{\mathscr{R}}(j^\star) = -1$ are both associated to a single eigenvector. Hence, the thesis follows resorting again to identity (6.6). ∎

The result provided by Theorem 6.3.1 contributes to equalities (6.8)-(6.9), yielding an interesting interconnection between the Dirichlet kernel and the eigenvalues of the graph Laplacian matrix $\mathbf{L}$. Denoting with $0 = \lambda_0^{\mathbf{L}} < \lambda_1^{\mathbf{L}} \le \cdots \le \lambda_{n-1}^{\mathbf{L}}$ the eigenvalues of $\Lambda(\mathbf{L})$, the analysis proceeds by focusing on the extremal (maximum and minimum) eigenvalues belonging to the restricted spectrum $\Lambda_0(\mathbf{L}) := \Lambda(\mathbf{L}) \setminus \{\lambda_0^{\mathbf{L}}\} \subseteq (0, 4\kappa]$ of $\mathbf{L}$. In the following lines, Corollary 6.3.2 and Corollary 6.3.3 investigate some properties related to the Fiedler value $\lambda_1^{\mathbf{L}}$ and the spectral radius $\lambda_{n-1}^{\mathbf{L}}$ of graph Laplacian $\mathbf{L}$ respectively. More details on the computation of $\lambda_{n-1}^{\mathbf{L}}$ are given in Sec. A.2.

Corollary 6.3.2 (Expression for the Fiedler value of κ-ring graphs)**.**
The smallest positive eigenvalue $\lambda_1^{\mathbf{L}}$ of the graph Laplacian $\mathbf{L}$ associated to the κ-ring

graph $C_n(1,\kappa)$ *is computed through indexes* $j=1$ *and* $j=n-1$, *i.e.*

$$\lambda_1^{\mathbf{L}} = \lambda^{\mathbf{L}}(1) = \lambda^{\mathbf{L}}(n-1) \in (0, 2\kappa). \tag{6.15}$$

Proof. Let us restrict w.l.o.g. the analysis to $j=1,\dots,\lfloor n/2\rfloor$ by exploiting the symmetry shown in (6.9). Firstly, it is worth to note that $\lambda^{\mathscr{R}}(1)$ is strictly positive, in general. Indeed, since $\theta=\pi/n\in(0,\pi/4]$, the relation

$$\kappa^{-1}(\mathcal{D}_\kappa(2\theta j)-1/2)>0 \Leftrightarrow \sin(\theta(\mathrm{d}+1))>\sin(\theta) \Leftrightarrow 0<\mathrm{d}<n-1 \tag{6.16}$$

holds true for any given (n,κ). Therefore, $\lambda^{\mathscr{R}}(1)$ actually belongs to the interval $(0,1)\subset[-1,1)$ and, consequently, $\lambda^{\mathbf{L}}(1)\in(0,2\kappa)\subseteq(0,4\kappa]$, by (6.6).

In order to prove that $\lambda^{\mathbf{L}}(1)<\lambda^{\mathbf{L}}(j)$ for $j=2,\dots,\lfloor n/2\rfloor$, that is $\lambda^{\mathscr{R}}(1)>\lambda^{\mathscr{R}}(j)$ for $j=2,\dots,\lfloor n/2\rfloor$, one has to show that inequalities

$$\mathcal{D}_\kappa(2\theta)>\mathcal{D}_\kappa(2\theta j) \Leftrightarrow \frac{\sin(\theta(\mathrm{d}+1))}{\sin(\theta)}>\frac{\sin(\theta(\mathrm{d}+1)j)}{\sin(\theta j)} \tag{6.17}$$

hold for all the values of j in question. Considering the second inequality in (6.17), formula 4.3.89 of Abramowitz and Stegun (1972) allows to expand the sine function as an infinite sequence of products. Thus, it is possible to obtain

$$\prod_{k=1}^{+\infty}\frac{n^2k^2-(\mathrm{d}+1)^2}{n^2k^2-1}>\prod_{k=1}^{+\infty}\frac{n^2k^2-(\mathrm{d}+1)^2j^2}{n^2k^2-j^2}. \tag{6.18}$$

Now, to satisfy relation in (6.18), it is sufficient to prove that:
(i) each term of the product on the l.h.s. is strictly positive for all $k\in\mathbb{N}$;
(ii) each term of the product on the l.h.s. is strictly greater than the correspondent term on the r.h.s. for all $k\in\mathbb{N}$.
Property (i) is verified, since this requirement boils down to the identity $\mathrm{d}<n\kappa-1$ for all $k\in\mathbb{N}$; while, property (ii) is also satisfied as this leads to the identity $\mathrm{d}>0$ for all $k\in\mathbb{N}$. Hence, relation (6.15) is proven. ∎

Corollary 6.3.3 (Spectral radius properties of κ-ring graphs)**.**
For the largest eigenvalue $\lambda_{n-1}^{\mathbf{L}}$ *of the graph Laplacian* $\mathbf{L}$ *associated to the* κ*-ring graph* $C_n(1,\kappa)$ *one has*

(i) $\lambda_{n-1}^{\mathbf{L}}\in[1+2\kappa,4\kappa]$, *with the equality for the upper bound holding if and only if* n *is even and* $\kappa=1$;

(ii) $\lambda_{n-1}^{\mathbf{L}}=\lambda^{\mathbf{L}}(j^\star)=\lambda^{\mathbf{L}}(n-j^\star)$, *where* $j^\star\in\mathbb{N}$ *belongs to* $[\underline{j},\overline{j}]\subset\mathbb{N}$ *with*

$$\underline{j}=1+\lfloor n/(2\kappa+1)\rfloor, \quad \overline{j}=\lceil(3n/(2\kappa+1)-1)/2\rceil; \tag{6.19}$$

(iii) $\lambda_{n-1}^{\mathbf{L}}=\lambda^{\mathbf{L}}(\lfloor n/2\rfloor)=\lambda^{\mathbf{L}}(\lceil n/2\rceil)$ *if and only if* $\kappa=1$;

(iv) $\lambda_{n-1}^{\mathbf{L}}=\lambda^{\mathbf{L}}(2)=\lambda^{\mathbf{L}}(n-2)$ *if* $\kappa\geq\kappa_n$ *with* $\kappa_n:=3n/10-1/2$.

Proof. Let us restrict w.l.o.g. the analysis to $j=1,\dots,\lfloor n/2\rfloor$ by exploiting the symmetry shown in (6.9).

(i) According to (6.5), let $\lambda^{\boldsymbol{\mathcal{L}}}_{n-1} := \lambda^{\mathbf{L}}_{n-1}/(2\kappa)$ be the spectral radius of the normalized graph Laplacian $\boldsymbol{\mathcal{L}}$. Since a connected graph is characterized by $\lambda^{\boldsymbol{\mathcal{L}}}_{n-1} = 2$ if and only if it is bipartite (Lem. 1.7 of Chung (1997)), then equality $\lambda^{\mathbf{L}}_{n-1} = 4\kappa$ holds if and only if the corresponding κ-ring graph $C_n(1,\kappa)$ is an even cycle. Then, thanks to relation (1) of Liu and Lu (2010), bounds for $\lambda^{\mathbf{L}}_{n-1}$ are proven.

(ii) By relation (6.6), the spectral radius $\lambda^{\mathbf{L}}_{n-1}$ can be found minimizing the j-th eigenvalue $\lambda^{\mathscr{R}}(j)$ of the Randić matrix $\mathscr{R}$. Firstly, exploiting the fact that relation $\lambda^{\mathscr{R}}_{n-1} = (1 - \lambda^{\boldsymbol{\mathcal{L}}}_{n-1}) \in [-1, -1/(2\kappa)]$ holds true by (6.6) applied on property (i), condition $-\kappa + 1/2 \leq \mathcal{D}_\kappa(2\theta j) \leq 0$ must be valid. So, $\mathcal{D}_\kappa(2\theta j) < 0$ can be imposed to compute lower bound $\underline{j}$ in (6.19) and checking afterwords that $\underline{j} \leq \lfloor n/2 \rfloor$ leads to identity $n \geq (4\kappa + 2)/(2\kappa - 1) \geq 2$. Moreover, it is ensured that

$$0 \leq j < \underline{j} \quad \Rightarrow \quad \mathcal{D}_\kappa(2\theta\underline{j}) < 0,\ \mathcal{D}_\kappa(2\theta\underline{j}) \leq \mathcal{D}_\kappa(2\theta j). \tag{6.20}$$

Secondly, upper bound $\overline{j}$ is determined considering that the denominator $2\sin(\theta j)$ of $\mathcal{D}_\kappa(2\theta j)$ is strictly positive for $j = 1, \ldots, \lfloor n/2 \rfloor$, and thus resorting to the following relaxed minimization problem:

$$\begin{aligned} &\arg\min_{j=2,\ldots,\lfloor n/2 \rfloor} \left|(2\kappa+1)\theta j - (3\pi/2 + 2\pi \mathrm{l})\right|, \quad \mathrm{l} \in \mathbb{Z} \\ &\text{s.t.: } j \in \mathbb{N}. \end{aligned} \tag{6.21}$$

Problem (6.21) is formulated observing that $(3\pi/2 + 2\pi\mathrm{l})$ is the argument that minimizes the numerator $\sin((2\kappa+1)\theta j)$ of $\mathcal{D}_\kappa(2\theta j)$ and is solved by imposing $\mathrm{l} = 0$, since term $2\sin(\theta j)$ is strictly increasing in j, for $j = 1, \ldots, \lfloor n/2 \rfloor$. From this approach, $\overline{j}$ in (6.19) is rounded so that both the increasing behavior of the denominator of $\mathcal{D}_\kappa(2\theta j)$ and constraint in (6.21) are taken into account. In addition, noting that $\overline{j} < 2n/(2\kappa+1)$ yields the identity $\mathrm{d} < n - 1$, the choice $\mathrm{l} = 0$ is justified and it is guaranteed that

$$\overline{j} < j \leq \lfloor n/2 \rfloor \quad \Rightarrow \quad \mathcal{D}_\kappa(2\theta\overline{j}) < 0,\ \mathcal{D}_\kappa(2\theta\overline{j}) \leq \mathcal{D}_\kappa(2\theta j). \tag{6.22}$$

Lastly, equivalence $\underline{j} \leq \overline{j}$ is satisfied, since $\underline{j} > 1$, $0 < (2\kappa+1)\theta \leq \pi$ and, by construction, it holds that $\pi < (2\kappa+1)\theta\underline{j} \leq (2\kappa+1)\theta\overline{j} < 2\pi$ because $\underline{j}$ is the first index $j \in [2, \lfloor n/2 \rfloor]$ that entails $\sin((2\kappa+1)\theta j) < 0$.

(iii) Setting $\kappa = 1$, equality $\lambda^{\mathbf{L}}(j) = 4\sin^2(\theta j)$ follows by resorting to the basic trigonometric formulas. Hence, for $\kappa = 1$, $\lambda^{\mathbf{L}}(j)$ is maximized selecting $j^\star = \lfloor n/2 \rfloor$. Furthermore, $\kappa = 1$ is the unique value that does not satisfy relation $\overline{j} < \lfloor n/2 \rfloor$ and, on the other hand, $\overline{j} < \lfloor n/2 \rfloor$ holds true for all $\kappa > 1$, since it leads to identities $n > (2\kappa+1)/(2(\kappa-1)) > 1$ and $n > (\kappa+1)/(\kappa-1) > 1$ for even and odd values of n respectively.

(iv) Equality $\lambda^{\mathbf{L}}_{n-1} = \lambda^{\mathbf{L}}(2)$ holds if $j^\star = \overline{j} = 2$, i.e. if $(3n/(2\kappa+1) - 1)/2 \leq 2$. This condition is, in fact, equivalent to require that $\kappa \geq 3n/10 - 1/2$.

■

6.3.2 Essential spectral radius analysis

Since matrix $\mathbf{F} := \mathbf{D}^{-1}\mathbf{A}$ is row-stochastic with eigenvalues[4] $\lambda_j^{\mathbf{F}} = \lambda_j^{\mathscr{R}} = (1 - \lambda_j^{\mathbf{L}}/(2\kappa))$, it is possible to define its *essential spectral radius* (see Sec. B.2) as

$$\mathrm{esr}(\mathbf{F}) = \lambda^{\mathbf{F}} := \max_{\lambda \in \Lambda(\mathbf{F}) \setminus \{\lambda_0^{\mathbf{F}}\}} |\lambda| \,. \tag{6.23}$$

This quantity is crucial to establish convergence performances in several frameworks, e.g. for graph-based distributed iterative algorithms (as illustrated in Chap. 5) or random walk processes Andrade, de Freitas, Robbiano, and Rodríguez (2018); Chung and Zhao (2010). Generally, relation (6.6) implies that $\lambda^{\mathbf{F}} = \lambda^{\mathscr{R}}$; hence, a further study on (6.23) can be provided starting from the preliminary

Lemma 6.3.4.
Let $\mathscr{R}$ be the Randić matrix of a κ-ring graph $C_n(1,\kappa)$ and $\theta := \pi/n \in (0, \pi/4]$. There exists a real number $\kappa_\theta \in (0, n/2)$ such that if $\kappa \geq \kappa_\theta$ then $\lambda^{\mathscr{R}}(1) + \lambda^{\mathscr{R}}(2) \leq 0$, with the equality holding if and only if $\kappa = \kappa_\theta$. Moreover, letting $\mathrm{c}_{2\theta} := \cos(2\theta)$, the value of κ_θ is yielded by

$$\kappa_\theta = \theta^{-1} \arcsin\left(\sqrt{x_\theta}\right), \tag{6.24}$$

where x_θ is the unique solution belonging to $(0,1)$ of the polynomial equation

$$\mathrm{p}_\theta(x) := x^3 + a_{\theta,2}x^2 + a_{\theta,1}x + a_{\theta,0} = 0, \tag{6.25}$$

with $a_{\theta,2} = -(\mathrm{c}_{2\theta} + 5)/2$, $a_{\theta,1} = (4\mathrm{c}_{2\theta}^2 + 7\mathrm{c}_{2\theta} + 13)/8$, $a_{\theta,0} = -(3\mathrm{c}_{2\theta} + 1)^2/16$.

Proof. From (6.14), the eigenvalues of the Randić matrix $\mathscr{R}$ can be rewritten using the prosthaphaeresis formula for the difference of two sines as

$$\lambda^{\mathscr{R}}(j) = \frac{\sin(\kappa\theta j)\cos((\kappa+1)\theta j)}{\kappa \sin(\theta j)}. \tag{6.26}$$

Thus, indicating with s_x and c_x the sine and cosine functions respectively, inequality $\lambda^{\mathscr{R}}(1) + \lambda^{\mathscr{R}}(2) \leq 0$ can be rephrased as follows by means of the triple angle identities $\mathrm{c}_{3x} = 4\mathrm{c}_x^3 - 3\mathrm{c}_x$, $\mathrm{s}_{3x} = 3\mathrm{s}_x - 4\mathrm{s}_x^3$, the Werner's formula for the product of two cosines and the basic trigonometric rules:

$$(1 - \mathrm{c}_{2\theta}^2)(5 - 4\mathrm{s}_{\kappa\theta}^2)^2\mathrm{s}_{\kappa\theta}^2 \geq (1 - \mathrm{s}_{\kappa\theta}^2)(4\mathrm{c}_{2\theta}(1 - \mathrm{s}_{\kappa\theta}^2) + 1 - \mathrm{c}_{2\theta})^2. \tag{6.27}$$

Now, assigning $x_\theta := \mathrm{s}_{\kappa_\theta\theta}^2 \in (0,1)$, inequality (6.27) can be solved in κ resorting to equation (6.25) and determining the solutions of $\mathrm{p}_\theta(x) \geq 0$. The application of the Routh-Hurwitz criterion to $\mathrm{p}_\theta(x)$, as illustrated in Table 6.2, ensures that x_θ has a strictly positive real part for any value of θ, since each pair of subsequent terms in the second column exhibits an alternating sign. Analogously, in order to show that x_θ has real part smaller than 1 for all θ, the Routh-Hurwitz criterion can be also applied to $-\mathrm{p}_\theta(y)$, obtained imposing $y := 1 - x$. This leads to the analysis of Table 6.3: the fact

[4] Sorted in a descending order as $j = 0, \cdots, n-1$, i.e. $\lambda_{n-1}^{\mathbf{F}} \leq, \ldots, \leq \lambda_1^{\mathbf{F}} < \lambda_0^{\mathbf{F}} = 1$.

x^3	1	$(4c_{2\theta}^2 + 7c_{2\theta} + 13)/8$
x^2	$-(c_{2\theta} + 5)/2$	$-(3c_{2\theta} + 1)^2/16$
x^1	$(2c_{2\theta}^3 + 9c_{2\theta}^2 + 21c_{2\theta} + 32)/(4(c_{2\theta} + 5))$	0
x^0	$-(3c_{2\theta} + 1)^2/16$	0

Table 6.2. Routh array for polynomial $p_\theta(x)$.

that each pair of subsequent terms in the second column exhibits an alternating sign finally ensures that $x_\theta \in (0, 1)$, provided that $x_\theta \in \mathbb{R}$.

y^3	1	$(4c_{2\theta}^2 - c_{2\theta} - 3)/8$
y^2	$-(1 - c_{2\theta})/2$	$-(1 - c_{2\theta}^2)/16$
y^1	$(2c_{2\theta}^2 - c_{2\theta} - 2)/4$	0
y^0	$-(1 - c_{2\theta}^2)/16$	0

Table 6.3. Routh array for polynomial $-p_\theta(y)$.

According to method 3.8.2 of Abramowitz and Stegun (1972), equation (6.25) can be solved by setting

$$q_\theta := a_{\theta,1}/3 - a_{\theta,2}^2/9, \quad r_\theta := (a_{\theta,1}a_{\theta,2} - 3a_{\theta,0})/6 - a_{\theta,2}^3/27, \tag{6.28}$$

through the computation and observation of the discriminant

$$\Delta_\theta := q_\theta^3 + r_\theta^2 = \frac{7\,(1 - c_{2\theta})\,(1 - c_{2\theta}^2)\,(c_{2\theta} + 13/14)}{1728}\left(c_{2\theta} - \frac{1}{2}\right)^2 \geq 0. \tag{6.29}$$

Expression in (6.29) is strictly positive if and only if the second quadratic factor is grater than zero: this occurs for values of $c_{2\theta} \neq 1/2$, i.e. for $n \neq 6$. In this case, the presence of only one real solution is guaranteed and it is yielded by

$$x_\theta = -\frac{a_{\theta,2}}{3} + \sqrt[3]{r_\theta + \sqrt{\Delta_\theta}} + \sqrt[3]{r_\theta - \sqrt{\Delta_\theta}}. \tag{6.30}$$

Otherwise, for $n = 6$, the discriminant Δ_θ vanishes and the solutions for (6.25) are given by $\{1/4, 5/4, 5/4\}$, meaning that $x_{\pi/6} = 1/4 \in (0, 1)$. ■

Lastly, the result on the ESR of $\mathbf{F}$ for κ-ring graphs is stated as follows:

Theorem 6.3.5 (ESR of κ-ring graphs: equality properties)**.**
For the ESR $\lambda^{\mathscr{R}}$ of the Randić matrix $\mathscr{R}$ associated to the κ-ring graph $C_n(1, \kappa)$ one has

(i) $\lambda^{\mathscr{R}} = 1$ *if and only if* n *is even and* $\kappa = 1$*;*

(ii) $\lambda^{\mathscr{R}} = \left|\lambda^{\mathscr{R}}(j')\right| = \left|\lambda^{\mathscr{R}}(n - j')\right|$*, where* $j' \in \mathbb{N}$ *belongs to* $\{1\} \cup [\underline{j}, \overline{j}] \subset \mathbb{N}$ *with* $(\underline{j}, \overline{j})$ *yielded by (6.19);*

(iii) $\lambda^{\mathscr{R}} = -\lambda^{\mathscr{R}}(\lfloor n/2 \rfloor) = -\lambda^{\mathscr{R}}(\lceil n/2 \rceil)$ *if and only if* $\kappa = 1$*;*

(iv) $\lambda^{\mathscr{R}} = -\lambda^{\mathscr{R}}(2) = -\lambda^{\mathscr{R}}(n - 2)$ *if* $\kappa \geq \max\{\kappa_n, \kappa_\theta\}$.

Proof. Let us restrict w.l.o.g. the analysis to $j = 1, \ldots, \lfloor n/2 \rfloor$ by exploiting the symmetry shown in (6.9).

(i) Cor. 6.3.2 and relations (6.5)-(6.6) imply that $\lambda^{\mathscr{R}} \in (0,1)$; thus, $\lambda^{\mathscr{R}} = 1$ if and only if $\lambda^{\mathscr{R}}_{n-1} = -1$. Exploiting Cor. 6.3.3, the equality is proven.

(ii) The interval in which j' lays is directly provided by Corollaries 6.3.2-6.3.3.

(iii) The fact that $\lambda^{\mathscr{R}} = -\lambda^{\mathscr{R}}(\lfloor n/2 \rfloor)$ if and only if $\kappa = 1$ derives from Cor. 6.3.3 and equality (6.26), which leads to $-\lambda^{\mathscr{R}}(\lfloor n/2 \rfloor) > \lambda^{\mathscr{R}}(1)$ for $\kappa = 1$.

(iv) The fact that $\lambda^{\mathscr{R}} = -\lambda^{\mathscr{R}}(2) = -\lambda^{\mathscr{R}}(n-2)$ if $\kappa \geq \max\{\kappa_n, \kappa_\theta\}$ is a consequence of Lem. 6.3.4 and Cor. 6.3.3.

■

6.4 Discussion

In this section, the results on the spectral properties of κ-ring graphs presented so far are depicted through numerical examples that illustrate spectral distributions in function of the number of vertices n, parameter κ and index j. In addition, a final conjecture on the ESR of the Randić matrix is proposed to conclude the discussion.

6.4.1 Spectral properties of κ-ring graphs

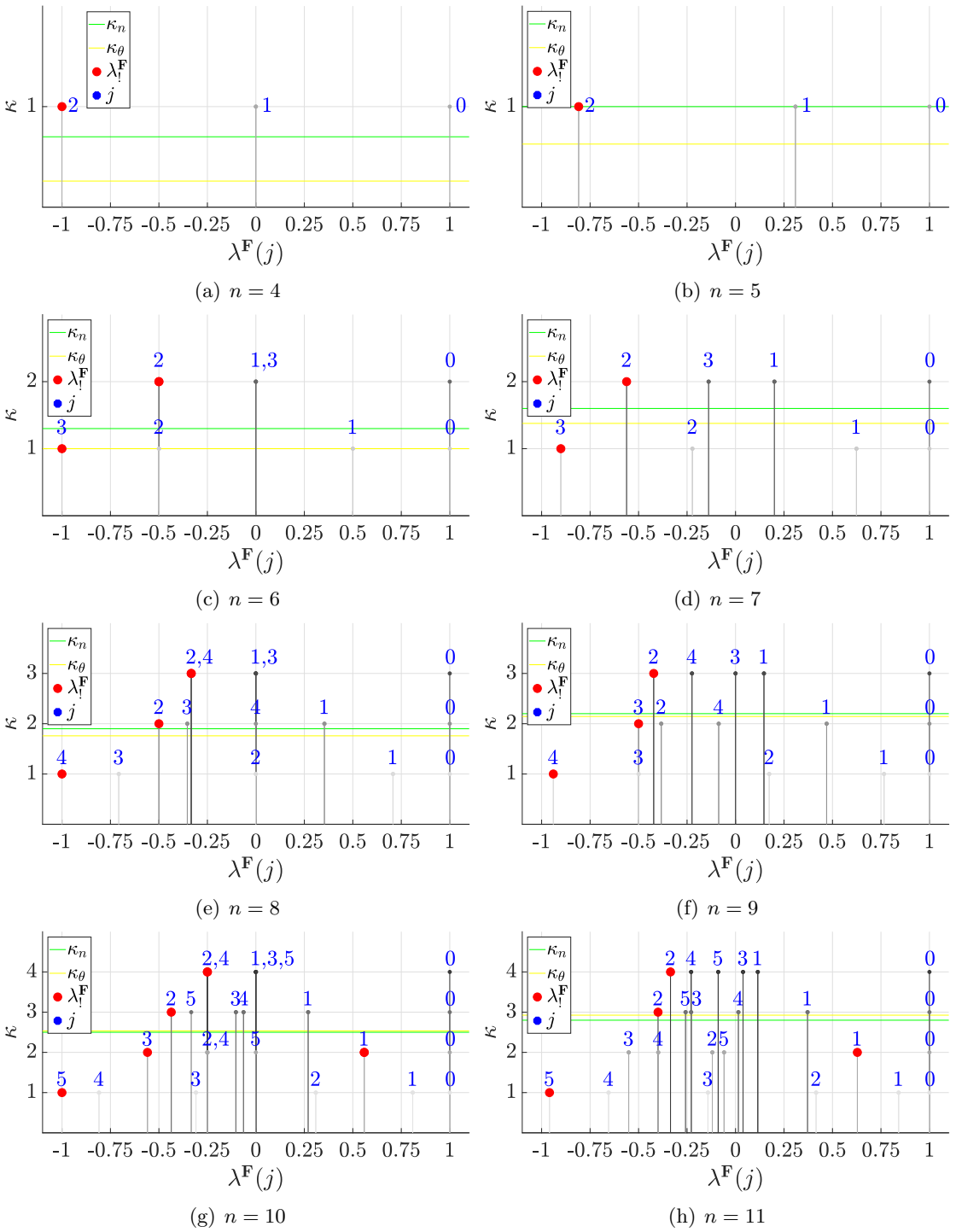
Examples in Fig. 6.3 grant to cover some of the most important aspects of this research, depicting a representation of the spectrum of the Randić matrix $\Lambda(\mathscr{R}) = \Lambda(\mathbf{F})$. Specifically, each diagram in Fig. 6.3 shows how the eigenvalues $\lambda^{\mathbf{F}}(j) = \lambda^{\mathscr{R}}(j)$ of $\mathbf{F}$ are located within interval $[-1,1]$, as parameter κ changes for a fixed n. Plots 6.3(a)-6.3(h) also illustrate indexes $j = 0, \ldots, \lfloor n/2 \rfloor$ of relation (6.14) in blue, thresholds $(\kappa_n, \kappa_\theta)$ with a green and a yellow line respectively, and the eigenvalue $\lambda^{\mathbf{F}}_! := \lambda^{\mathbf{F}}(j')$ that maximizes relation (6.23) with a red dot. One can observe the following global facts descending from all the previous statements presented in Sec. 6.3:

- $\lambda^{\mathbf{F}}(j) \in [-1,1]$ for all $j = 0, \ldots, \lfloor n/2 \rfloor$, with -1 and 1 simple eigenvalues;
- $\lambda^{\mathbf{F}}(1) > \lambda^{\mathbf{F}}(j)$ for all $j = 2, \ldots, \lfloor n/2 \rfloor$;
- for $\kappa = 1$, $\lambda^{\mathbf{F}}_! = \lambda^{\mathbf{F}}_{n-1} = \lambda^{\mathbf{F}}(\lfloor n/2 \rfloor)$ and $\lambda^{\mathbf{F}}_{n-1} = -1$ if n is even;
- $\lambda^{\mathbf{F}}_{n-1} = \lambda^{\mathbf{F}}(2)$ if $\kappa \geq \kappa_n$ and $\lambda^{\mathbf{F}}_! = \lambda^{\mathbf{F}}(2)$ if $\kappa \geq \max\{\kappa_n, \kappa_\theta\}$.

Some peculiarities and patterns can be found for:

- $n = 5$, where $\kappa_n = 1$ tightly verifies property (iv) in Corollary 6.3.3;
- $n = 6$, where $\kappa_\theta = 1$, so that the information about κ_n becomes necessary to satisfy property (iv) in Theorem 6.3.5;
- $n = 10$, where it holds that $\lambda^{\mathbf{F}} = \sqrt{5}/4 = -\lambda^{\mathbf{F}}(3) = \lambda^{\mathbf{F}}(1)$ for $\kappa = 2$ and $\kappa_\theta \simeq 2.5330$ is not smaller than $\kappa_n = 2.5$ as, conversely, it does in all the previous cases.

To sum up, each debated example gravitates, to some extent, around the key relation in (6.14), describing the spectrum $\Lambda(\mathscr{R})$ of the Randić matrix: this investigation completely leverages the central idea of studying the spectral properties of κ-regular graphs via the Dirichlet kernel definition given in (6.7).



(a) $n = 4$ (b) $n = 5$

(c) $n = 6$ (d) $n = 7$

(e) $n = 8$ (f) $n = 9$

(g) $n = 10$ (h) $n = 11$

Figure 6.3. General eigenvalue distribution of the Randić matrix spectrum $\Lambda(\mathscr{R}) = \Lambda(\mathbf{F})$ for the κ-ring graphs $C_n(1, \kappa)$ with $n = 4, \ldots, 11$, $\kappa = 1, \ldots, \lfloor n/2 \rfloor - 1$.

6.4.2 Conjecture on the Randić matrix spectrum for κ-ring graphs

All numerical results provided to support the previous discussion suggest few clues about one relevant issue for this research. A richer description for the behavior of index j' allows to achieve a deeper understanding of the ESR $\lambda^{\mathbf{F}} = \lambda^{\mathscr{R}}$. An insight on this quantity might be useful to prove novel results of a wider scope in graph theory and its applications, e.g. convergence rate of distributed algorithms and random walk processes. So, observing the fact that graph $C_9(1,2)$ in Fig. 6.3(f) is the unique example leading to $\lambda^{\mathbf{F}} = 1/2 = -\lambda^{\mathbf{F}}(3) > \lambda^{\mathbf{F}}(1) \simeq 0.4698$, graph $C_{10}(1,2)$ in Fig. 6.3(g) is the unique example leading to $\lambda^{\mathbf{F}} = \sqrt{5}/4 = -\lambda^{\mathbf{F}}(3) = \lambda^{\mathbf{F}}(1)$, in each diagrams of Fig. 6.3 it holds that $\lambda^{\mathbf{F}} = -\lambda^{\mathbf{F}}(\lfloor n/2 \rfloor)$ if $\kappa = 1$ and $\lambda^{\mathbf{F}} = -\lambda^{\mathbf{F}}(2)$ if $\kappa > \max\{\kappa_n, \kappa_\theta\}$, the following conjecture is drawn after having run many other analogous simulations[5]:

Conjecture 6.4.1 (Characterization of the essential spectral radius index)**.**
The ESR $\lambda^{\mathbf{F}}$ for a κ-ring graph $C_n(1,\kappa)$ is equal to $\left|\lambda^{\mathbf{F}}(j')\right|$ where

$$j' = \begin{cases} \lfloor n/2 \rfloor, & \text{if } \kappa = 1; \\ 3, & \text{if } n = 9 \text{ and } \kappa = 2; \\ 2, & \text{if } \kappa > \kappa_\theta; \\ 1, & \text{otherwise.} \end{cases} \tag{6.31}$$

6.5 Chapter summary

In this chapter, a class of circulant graphs, defined as κ-ring graphs, is presented highlighting the relationship between the spectrum of their characteristic matrices and the Dirichlet kernel. Several properties related to the eigenvalues are described extensively, with a particular focus on the Fiedler value, the spectral radius of the Laplacian and the essential spectral radius of the Randić matrix associated to these graphs. Part of the proven results is also discussed in details with auxiliary diagrams depicting the related spectral distributions. Along the dissertation, a conjecture on the computation of the debated essential spectral radius has emerged and its analysis is envisaged.

[5] Performed as n goes to infinity, until $n = 5000$, for every value of κ.

7

Conclusions and future works

"Everything that has a beginning has an end."
Oracle

Contents

7.1 Concluding remarks

This thesis introduces the role of mobile multi-agent systems and networked control trends belonging to this age in Chap. 1. It continues proposing a wide-ranging overview in Chap. 2, discussing general background, theoretical methodologies and common thread for the topics of interest in this research field. Lastly, the single contributions of the manuscript are represented by the core of the PhD activity, which can be summarized into four research tasks and a comprehensive final discussion. These conclusive considerations are reported in the following paragraphs.

7.1.1 First research task

Ideas in Chap. 3. dealing with *Dynamic Coverage with Limited Sensing Capabilities* compose the first research task. In this work, it is presented the development of a novel distributed algorithm performing robotic coverage, clustering and dispatch around an event in static-obstacle structured environments without relying on metric information. Specifically, the aim is to account for the trade-off between local communication given by bearing visibility sensors installed on each agent involved, optimal deployment in closed unknown scenarios and focus of a group of agents on one point of interest. The particular targets of this study can be summarized as 1. the computation, under certain topological

assumptions, of a lower bound for the number of required agents, which are provided by a realistic geometric model (e.g. a round shape) to emphasize physical limitations; 2. the minimization of the number of nodes and links maintaining a distributed approach over a connected communication graph; 3. the identification of an activation cluster around an event with a radial decreasing intensity, sensed by each agent; 4. the attempt to send the agents belonging to the cluster towards the most intense point in the scenario by minimizing a weighted isoperimetric functional.

7.1.2 Second research task

Studies in Chap. 4. coping with *Optimal Time-Invariant Formation Tracking* comprise the second research task. This work addresses the Optimal Time-invariant Formation Tracking (OIFT) control problem for second-order multi-agent systems. Most of the results related to distributed Formation Tracking do not take into account energy consumption aspects while designing control laws. In order to consider this important criterion, a new contribution is provided by formalizing and proposing a solution of an optimization problem that encapsulates trajectory tracking, distance-based formation control and input energy minimization. To this end, it is shown how to compute the inverse dynamics obtained with the Projector Operator based Newton's method for Trajectory Optimization (PRONTO), exploiting that as a general reference to devise a novel online distributed control law to drive the agents. To stabilize the formation, a specific choice of potential functions has been made. This work also gives a general overview about the equilibria that the system can achieve with such approaches. Finally, a numerical example involving a cubic formation following a straight path is simulated to validate the proposed control strategies.

7.1.3 Third research task

Analyses in Chap. 5. regarding the *Distributed State Estimation from Relative Measurements* constitute the third research task. This activity studies the multivariate least-squares problem of estimation from relative measurements in an undirected connected graph within the context of multi-agent systems, providing various distributed iterative schemes based on consensus. These allow to establish a connection between the convergence behavior of consensus algorithms toward the optimal estimate and the theory of the stochastic matrices that describe the network system dynamics. The most challenging aspect of this research resorts on the fact that agents are, by assumption, only able to access and exchange local information among a given topology.
One contribution is represented by the effort spent to prove that estimates converge to the correspondent centralized solution. Furthermore, to investigate the robustness and the differences between the different approaches introduced, a topological study and a sensitivity analysis are presented and followed by numerical simulations supporting the theoretical results proven along the dissertation.

7.1.4 Fourth research task

The fourth and last research task consists of the investigations *On the Spectral Properties of κ-ring Graphs* carried out in Chap. 6.
The use of Laplacian and Randić matrices induced by circulant graphs is extensively adopted in several fields, e.g. distributed scalable algorithms, and their eigenvalues play a central role on the convergence analysis of the latter. In this work, a specific class of circulant matrices, inducing the so-called κ-ring graphs, is considered and its spectral properties are studied, providing a deep insight based on the Dirichlet kernel. Further associated well-known fundamental quantities in Spectral Graph Theory as the Fiedler value of the Laplacian and the essential spectral radius of the Randić matrix are widely discussed along this research, with particular attention on their contextualization. Numerical examples, restricted to a relevant subset of the topology under analysis, illustrate the exactness of the proposed results, representing a remarkable starting point for novel conjectures in this topic.

7.1.5 Discussion

In the recent years, the field of automation has witnessed a tendency to employ networked control for groups of agents to fulfill complex tasks arising from robotic surveillance to smart-grid production, from autonomous vehicle servicing to sensor network monitoring and a variety of space-based applications. Such mobile systems are referred to as multi-agent systems (MASs). In this thesis, the main research interest concerns how to design the effective decentralized coordination among autonomous agents, by means of a limited local sensing, to perform common tasks, aiming at reaching a high-quality overall performances. This approach has had a transformative impact in several application domains and, indeed, MASs possess well-known capabilities for autonomous and intelligent operation based on planning, reactivity, learning, proactivity, mobility, adaptivity and reasoning. Furthermore, such elements collaborate to seek a solution to problems that are out of reach for the single entity.
What has led to this rapid progress in the last two decades is a combination of technological advances in price, scale of the platforms themselves, computational performance and a novel breakthrough of how the mobile robots should be arranged algorithmically. At the light of these increasing trends of investigation, it is crucial to reinforce the latest knowledge to keep up with the research demand. Pursuing this aim, the research activities proposed in this manuscript broaden the existing boundaries in the following fields: Optimization Theory (OT), Dynamic Systems and Control (DS&C), Combinatorial Graph Theory (CGT), Optimal Control (OC), Trajectory Optimization (TO), Combinatorial Optimization (CO), Distributed Optimization (DO), Distributed Control Systems (DCS) and Distributed Estimation (DE). A summary of the contributions to each field of the Ph.D. activity is contained in Table 7.1.

Addressing the main activities in Chapters 3, 4 and 5, a general control design guideline has been followed, as illustrated in the conceptual scheme of Fig. 7.1.

Research task	OT	DS&C	CGT	OC	TO	CO	DO	DCS	DE
First, Chap. 3	✓	✓	✓			✓	✓		
Second, Chap. 4	✓	✓	✓	✓	✓		✓	✓	
Third, Chap. 5	✓	✓	✓				✓		✓
Fourth, Chap. 6	✓		✓			✓			

Table 7.1. Disciplines covered (marked with ✓) for each research task in the manuscript.

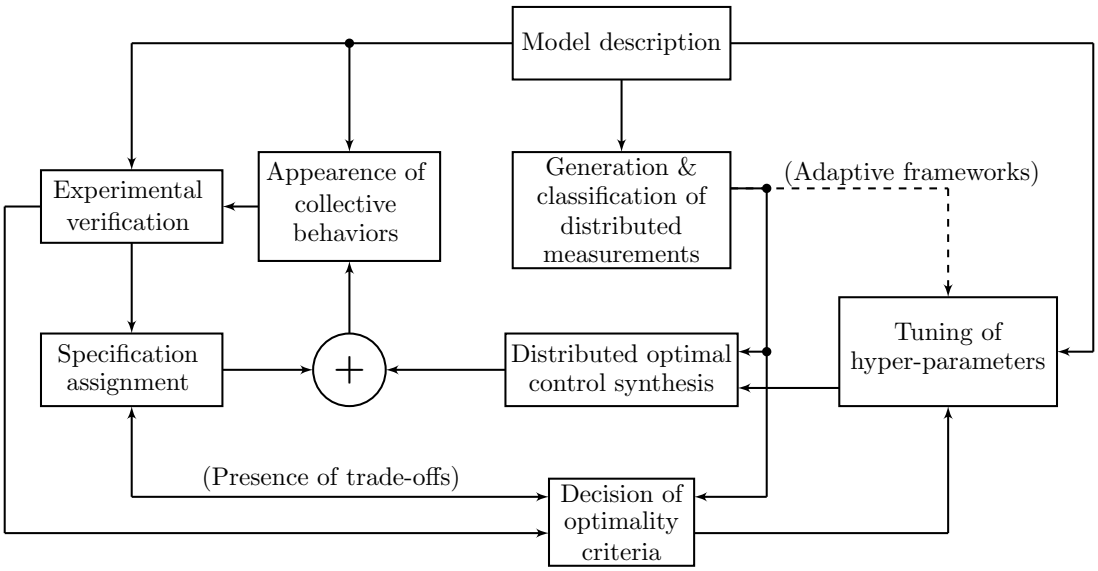


Figure 7.1. Conceptual scheme for MAS design followed in the major Ph.D. activities. Arrows indicate dependencies (head points depend on tail ones).

Specifically, given a multi-agent plant and certain specifications, the design starts providing a model description. Then, by means of the available measurements and information exchange modality one should develop various kinds of distributed control strategies (from the synthesis of control laws to the fine tuning of hyper-parameters), according to the optimality criteria decided. The latter might be modified after the outcome of experimental verification, e.g. numerical simulations or proofs, showing the collective behaviors emerging from the MAS under analysis, since either too many solutions or the lack of them may arise. The same procedure should be followed while assigning specifications: this aspect, the model describing the network and the establishment of optimality criteria, require, in general, delicate compromises in order to find a real and suitable implementation. In particular, the lesson that can be learned from this manuscript is that the design of multi-agent networks leads, inevitably, to the presence of potentially multiple trade-offs. Few examples of this statement are given in the sequel.

- In Chap. 3, the number of available mobile agents is fundamental to comply the task of robotic deployment and focus on an event: this aspect becomes problematic whenever a large area has to be covered (especially when a dynamic coverage is demanded). Moreover, the sensing capabilities of each device is crucial for the coordination of such agents and the possibility for them to arrange easily into optimal spacial configurations.

- Chap. 4 represents the clearest example of trade-off, in this thesis. Indeed, the aim of Optimal Time-Invariant Formation Tracking is that of performing both (position

& velocity) tracking and formation control while minimizing the energy spent by the system of agents. In a generic scenario, these requirements exhibit the tendency to play one against the other. Furthermore, the topology arising from the local information exchange modality and the number of agents is, again, very relevant in the accomplishment of this task, for any particular case scenario.

- In Chap. 5, many iterative procedures to attain the state estimation from a collection of relative measurements are developed. It is worth to observe that the performances of such schemes are inherently different from each other but also highly related to the kind of network topology involved. This is, for instance, the key to characterize the convergence rate with which the iterative schemes provide their estimates. This feature has been deeply investigated in Chap. 6, where the topology of κ-ring graphs[1] is adopted.

7.2 Future developments

For the first research task, future works on complete dynamic coverage in a noisy framework are envisaged, since uncovered regions could arise while agents are steered.
In the second research task, possible future directions might be represented by the application of PRONTO to a real OIFT problem with a nonlinear dynamics or the extension to a time-varying formation framework. Concerning the third research task, future directions for this work essentially consist in extending the relative state estimation capabilities over sensor networks in which information is not mutually exchanged among nodes, i.e. developing a novel version for the algorithms in question leveraging digraphs.
Regarding the fourth research task, future works might be represented by the study of isoperimetric problems related to κ-ring graphs, e.g. the Cheeger's inequality, as well as on the possibility to discover further algebraic interconnections between other topologies and the Dirichlet kernel.
To conclude, the comprehensive future directions can be summarized as the development of novel theoretical methods and applications aiming at

- the extension of the proposed activities to frameworks where the information exchange may be unidirectional or where the network structure may change over the time;
- the research of new strategies to reduce as far as possible the energy spent by the systems in the control process to lower production costs, maintenance, environmental impact and increase the life of the driven plant;
- the study of algebraic conditions that guarantee a distributed optimization over multi-agent networks consisting of mobile robots from the performance viewpoint;
- the robustness analysis of such methods and the possibility to ensure correct working conditions by the autonomous system involved, even in situations where uncertainty becomes extremely relevant.

[1] Which are described by two quantities only: the number of nudes in the network and the number of links to exchange information for each connected element.

A

Supplementary material

"'Why' is the only real source of power, without it you are powerless."
Merovingian

A.1 A short overview on PRONTO

This section provides a short overview on the various efforts in vehicle trajectory exploration and motion planning based on PRONTO and reviews few of its basics. Further details can be also found in Aguiar et al. (2017).

A.1.1 Introduction: PRONTO-based optimal motion planning

The development of efficient tools to compute optimal trajectories for autonomous vehicles has been receiving growing attention for robotic applications. These tools allow the designer to study and analyze – that is, *explore* – the dynamic performance of autonomous vehicles and, at the same time, give useful insight on how to improve vehicle performance. They are also vital in developing motion planning algorithms for single and multiple vehicles acting in cooperation, specifically when both detailed vehicle models and complex geometric constraints have to be taken into consideration. Motion Planning is a core robotics technology that, together with other actuation, control and perception technologies enables robot autonomy.

This section provides a quick overview about motion planning and vehicle trajectory exploration based on PRONTO, an indirect numerical method for solving optimal control problems. The name PRONTO stands for PRojection Operator based Newton's method for Trajectory Optimization. The method is also known, for short, as the projection operator approach. Interesting applications that have been tackled with this method include, e.g., cooperative motion planning of autonomous underwater vehicles (AUVs) for environmental surveying and computing minimum-time trajectories for a race car, exploiting the energy from the surrounding environment for long endurance missions of UAVs.

In its basic formulation, a motion planner is a theoretical method to generate a trajectory to connect a given initial state of a robotic system to a desired final state. Robot state means the position of the robotic system and, possibly, the orientation (of each vehicle) relative to a reference coordinate system. Depending on the specific application and problem at hand and the level of complexity of the mathematical model employed to describe the robotic system, other quantities such as angular and linear velocities can belong to the system state. Other factors that play a role and increase

complexity are the presence of obstacles in the environment and the typical need to avoid contact with these obstacles as well as inter-vehicle collisions in a multi-vehicle situation.

Every constrained optimal motion problem starts from a careful selection of the vehicle model(s), cost criteria (e.g., energy consumption and maneuvering time) to arrive at a well formulated optimal control problem and constraints (e.g., actuator saturation and collision avoidance). The problem formulation requires some experience in order to avoid defining a problem with too many solutions or with no solution so as to allow numerical resolution using the selected optimization method.

PRONTO has the unusual peculiarity of working directly in continuous-time, constructing a sequence of trajectories with descending cost. More commonly, numerical optimal control is to discretize the system dynamics, constraints, and cost functional giving a constrained nonlinear optimization problem that is subsequently solved by using off-the-shelf constrained nonlinear solvers, specialized to manage the sparsity of the constraints resulting in the discretization of the continuous-time dynamics. PRONTO skips this transcription phase, employing instead an infinite-dimensional Newton method (see also Subsec. 2.2.4) fulfilling second-order convergence.

A.1.2 More basics and geometric interpretation on PRONTO

Some basics and geometric interpretation of PRONTO have already been introduced in Subsec. 4.3.1. In this paragraph, further details about this numerical tool are provided.

Starting from (4.16)-(4.17), it is easy to see that $\boldsymbol{\xi}$ is a trajectory, $\boldsymbol{\xi} \in \mathcal{T}$, if and only if $\boldsymbol{\xi}$ is a fixed point of $\mathcal{P}$, $\boldsymbol{\xi} = \mathcal{P}(\boldsymbol{\xi})$. Since $\mathcal{P}(\boldsymbol{\xi}) \in \mathcal{T}$ for all $\boldsymbol{\xi}$ in the domain of $\mathcal{P}$, it holds that $\mathcal{P} = \mathcal{P}(\mathcal{P}(\boldsymbol{\xi}))$, briefly $\mathcal{P} = \mathcal{P}^2$, so that $\mathcal{P}$ is a (nonlinear) *projection operator*. Considering the Fréchet derivative of $\mathcal{P}$ and the approximation in (4.21), one observes that $D\mathcal{P}$ is the (continuous) linear projector operator (4.20) in which the dynamics of $\mathbf{z}$ and $\mathbf{v}$ is regulated by the following equations

$$\dot{\mathbf{z}} = \bar{\mathbf{A}}(\boldsymbol{\eta}(\tau))\mathbf{z} + \bar{\mathbf{B}}(\boldsymbol{\eta}(\tau))\mathbf{v}, \quad \mathbf{z}(0) = \mathbf{0}_N; \tag{A.1}$$

$$\mathbf{v} = \boldsymbol{\nu}(\tau) + \mathbf{K}(\tau)[\boldsymbol{\beta}(\tau) - \mathbf{z}]; \tag{A.2}$$

representing the linearization for $(\mathbf{x}, \mathbf{u})$ at the point $\boldsymbol{\xi}$. The system matrices $\bar{\mathbf{A}}(\boldsymbol{\eta}(\tau)) = \mathbf{f}_{\mathbf{x}}(\mathbf{x}(\tau), \mathbf{u}(\tau))$ and $\bar{\mathbf{B}}(\boldsymbol{\eta}(\tau)) = \mathbf{f}_{\mathbf{u}}(\mathbf{x}(\tau), \mathbf{u}(\tau))$ are evaluated on the trajectory $\boldsymbol{\eta} = \mathcal{P}(\boldsymbol{\xi})$. The set of bounded linearized trajectories at a given (nonlinear) trajectory $\boldsymbol{\xi} \in \mathcal{T}$ forms the tangent space to the manifold, denoted $T_{\boldsymbol{\xi}}\mathcal{T}$. It is remarkable that, as in the nonlinear case, tangent trajectories at $\boldsymbol{\xi} \in \mathcal{P}$ are fixed points: $\boldsymbol{\zeta} \in T_{\boldsymbol{\xi}}\mathcal{T} \Leftrightarrow \boldsymbol{\zeta} = D\mathcal{P}(\boldsymbol{\xi}) \cdot \boldsymbol{\zeta}$. The projection operator $\mathcal{P}$ provides a local representation of the trajectory manifold: given a trajectory $\boldsymbol{\xi} \in \mathcal{T}$, every nearby trajectory $\boldsymbol{\eta} \in \mathcal{T}$ is of the form $\boldsymbol{\eta} = \mathcal{P}(\boldsymbol{\xi} + \boldsymbol{\zeta})$ for a unique tangent trajectory $\boldsymbol{\zeta} \in T_{\boldsymbol{\xi}}\mathcal{T}$. That is, using $\mathcal{P}$, tangent trajectories can be used as local coordinates for the trajectory manifold. It is worth to note that the projection operator depends on the choice of the time-dependent feedback gain $\mathbf{K}(\cdot)$, which is typically chosen to provide local stability of the closed-loop system about a given trajectory. For instance, if $\boldsymbol{\xi} \in \mathcal{T}$ and $\dot{\mathbf{z}} = [\bar{\mathbf{A}}(\boldsymbol{\xi}(\tau)) - \bar{\mathbf{B}}(\boldsymbol{\xi}(\tau))\mathbf{K}(\tau)]\mathbf{z}$ is exponentially stable then the domain of the operator $\mathcal{P}$ will include a nice L_∞ neighborhood of $\boldsymbol{\xi}$.

At the light of these observations, Alg. 7 can be illustrated as in Fig. A.1.

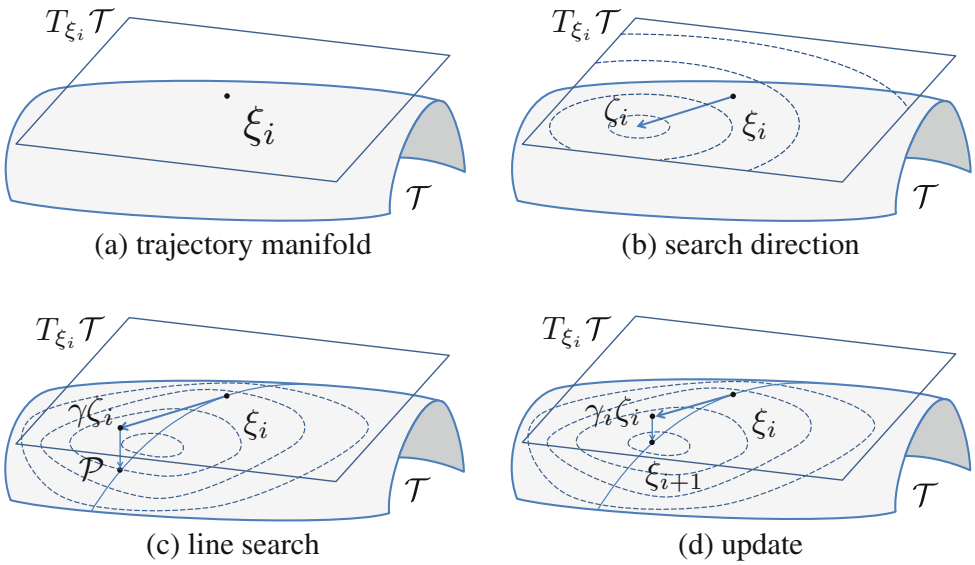


Figure A.1. The Projection Operator approach: at each iteration, (a) the linearization of the control system about the trajectory $\boldsymbol{\xi}_i$ defines the tangent space to the trajectory manifold $\mathcal{T}$ at $\boldsymbol{\xi}_i$; (b) minimization over the tangent space of a second-order approximation of the extended cost functional $g = h \circ \mathcal{P}$ yields a search direction $\boldsymbol{\zeta}_i$; (c) a step size is computed via a line search along $\boldsymbol{\zeta}_i$ with $\mathcal{P}$ bending that line onto the trajectory manifold $\mathcal{T}$; (d) the search direction $\boldsymbol{\zeta}_i$ and step size γ_i are combined and projected to obtain the updated trajectory $\boldsymbol{\xi}_{i+1}$. [*Credits* : Aguiar et al. (2017)]

Working with the cost functional $g(\boldsymbol{\xi}) := h(\mathcal{P}(\boldsymbol{\xi}))$ in an essentially unconstrained manner (see (4.19)), one may develop effective descent methods for trajectory optimization. For instance, a Newton descent step at $\boldsymbol{\xi} \in \mathcal{T}$ may be obtained by minimizing the quadratic functional $Dg(\boldsymbol{\xi}) \cdot \boldsymbol{\zeta} + \frac{1}{2} D^2 g(\boldsymbol{\xi}) \cdot (\boldsymbol{\zeta}, \boldsymbol{\zeta}) \simeq g(\boldsymbol{\xi} + \boldsymbol{\zeta}) - g(\boldsymbol{\xi})$ over the linear subspace of tangent trajectories $\boldsymbol{\zeta} \in T_{\boldsymbol{\xi}}\mathcal{T}$. In fact, one may derive many properties, including first and second-order optimality conditions, of the constrained optimal control problem though an unconstrained analysis of the functional g. Moreover, it is remarkable to observe that

- $Dg(\boldsymbol{\xi}) \cdot \boldsymbol{\zeta} = Dh(\boldsymbol{\xi}) \cdot \boldsymbol{\zeta}$
- $D^2 g(\boldsymbol{\xi}) \cdot (\boldsymbol{\zeta}, \boldsymbol{\zeta}) = D^2 h(\boldsymbol{\xi}) \cdot (\boldsymbol{\zeta}, \boldsymbol{\zeta}) + Dh(\boldsymbol{\xi}) D^2 \mathcal{P}(\boldsymbol{\xi}) \cdot (\boldsymbol{\zeta}, \boldsymbol{\zeta})$, where $D^2\mathcal{P}(\boldsymbol{\xi})$ is null whenever the constraint of the problem is represented by a linear dynamics.

To conclude, once one has an acceptable descent direction $\boldsymbol{\zeta} \in T_{\boldsymbol{\xi}}\mathcal{T}$ (see Step 3 of Alg. 7), a line search using the actual cost functional $g = h \circ \mathcal{P}$ is required and it is obtained solving the problem in Ex. 2.2.3 for the linearized trajectory. Step 4 in Alg. 7 suggests an exact line search, but it is more common and effective to use an Armijo-backtracking line search. Remarkably, in a sufficiently small neighborhood of a second-order sufficient condition minimizer, the Newton step will be accepted (with unit step size) leading to quadratic convergence. Lastly, it is worth to notice that the projection operator is used to bend the line of test points $\boldsymbol{\xi}_i + \gamma \boldsymbol{\zeta}_i$ along the trajectory manifold (see Fig. A.1), leading to the natural manifold update law in Step 5 of Alg. 7: $\boldsymbol{\xi}_{i+1} = \mathcal{P}(\boldsymbol{\xi}_i + \gamma_i \boldsymbol{\zeta}_i)$.

A.2 Computation of $\lambda^{\mathbf{L}}_{n-1}$ for κ-ring graphs

In Cor. 6.3.3, upper and lower bounds for the index j referring to the spectral radius $\lambda^{\mathbf{L}}_{n-1}$ of the Laplacian are provided. Let us recall that $\lambda^{\mathbf{L}}_{n-1} = 2\kappa(1 - \lambda^{\mathscr{R}}_{n-1})$ and denote with $\mathcal{D}'_\kappa(x)$ the first derivative of $\mathcal{D}_\kappa(x)$ w.r.t. x. The calculus of the quantity in question can be conveniently implemented whenever $\kappa \in (1, \kappa_n)$, by means of the procedure illustrated in Alg. 9, with a computational complexity[1] of $O\left(\log_2(n/(2\kappa+1))\right)$ as the ratio n/κ goes to infinity.

Algorithm 9 Computation of eigenvalue $\lambda^{\mathscr{R}}_{n-1}$ and its relative index $j^\star$

set $(\underline{j}, \overline{j})$ as the extrema given by relation (6.19);
if $\mathcal{D}'_\kappa(2\theta\overline{j}) \leq 0$ **then** $\left(j^\star, \lambda^{\mathscr{R}}_{n-1}\right) \leftarrow \left(\overline{j}, \lambda^{\mathscr{R}}(\overline{j})\right)$;
else if $\mathcal{D}'_\kappa(2\theta\underline{j}) \geq 0$ **then** $\left(j^\star, \lambda^{\mathscr{R}}_{n-1}\right) \leftarrow \left(\underline{j}, \lambda^{\mathscr{R}}(\underline{j})\right)$;
else
    $found \leftarrow$ **false**;
    while $\overline{j} - \underline{j} > 1$ **and not** $found$ **do**
        $j^\star \leftarrow \lfloor(\underline{j} + \overline{j})/2 + 1/2\rfloor$;
        if $\mathcal{D}'_\kappa(2\theta j^\star) < 0$ **then** $\underline{j} \leftarrow j^\star$;
        else if $\mathcal{D}'_\kappa(2\theta j^\star) > 0$ **then** $\overline{j} \leftarrow j^\star$;
        else $found \leftarrow$ **true**; $\lambda^{\mathscr{R}}_{n-1} \leftarrow \lambda^{\mathscr{R}}(j^\star)$;
        end if
    end while
    if not $found$ **then** $\left(j^\star, \lambda^{\mathscr{R}}_{n-1}\right) \leftarrow \min_{j \in \{\underline{j}, \overline{j}\}} \left\{\lambda^{\mathscr{R}}(j)\right\}$;
    end if
end if

The computational complexity of Alg. 9 is justified by the fact that it encodes an implementation of a binary search procedure Knuth (1998) that operates on a discrete set with cardinality $\overline{j} - \underline{j} + 1 < n/(4\kappa+2) + 3/2$. Whereas, its correctness can be proven as follows. Since $l = 0$ is set to choose $\overline{j}$ in (6.21), it is guaranteed that $\mathcal{D}_\kappa(2\theta j) < 0$ for all $j \in [\underline{j}, \overline{j}]$. Hence, exploiting (6.20) and (6.22), interval $[\underline{j}, \overline{j}]$ must lay within two consecutive roots of $\mathcal{D}_\kappa(x)$. The correctness of the binary search is then verified, as there exists only one local extremum for $\mathcal{D}_\kappa(x)$ between a pair of consecutive zeros Wiggins (2007) and the first derivative of the Dirichlet kernel $\mathcal{D}'_\kappa(x)$ is employed as *value* to select intervals which *key* $j^\star \in [\underline{j}, \overline{j}]$ belongs to.

As an example, Fig. A.2 represents eigenvalues $\lambda^{\mathbf{F}}(j) = \lambda^{\mathscr{R}}_{n-1}$ in function of their index j for a κ-ring graph with $n = 46$ vertices. Here, interval $[\underline{j}, \overline{j}]$ examined in Cor. 6.3.3 is highlighted and represented as the union of two separated areas $\mathcal{B}_L\left(\lambda^{\mathbf{F}}_{n-1}\right)$, in orange, and $\mathcal{B}_R\left(\lambda^{\mathbf{F}}_{n-1}\right)$, in light-blue, such that $\mathcal{B}_L\left(\lambda^{\mathbf{F}}_{n-1}\right) \cup \mathcal{B}_R\left(\lambda^{\mathbf{F}}_{n-1}\right) = [\underline{j}, \overline{j}]$ and intersection $\mathcal{B}_L\left(\lambda^{\mathbf{F}}_{n-1}\right) \cap \mathcal{B}_R\left(\lambda^{\mathbf{F}}_{n-1}\right) = \{j_\oplus\}$ is composed of the single splitting element

$$j_\oplus := \max\left\{\underline{j}, \min\left\{\overline{j} - 1, \lfloor u_5 n/(\pi(2\kappa+1)) + 1/2\rfloor\right\}\right\} \tag{A.3}$$

[1] In terms of number of operations directly proportional to the number of comparisons between the current estimate of $\lambda^{\mathscr{R}}_{n-1}$ and the next potential incumbent.

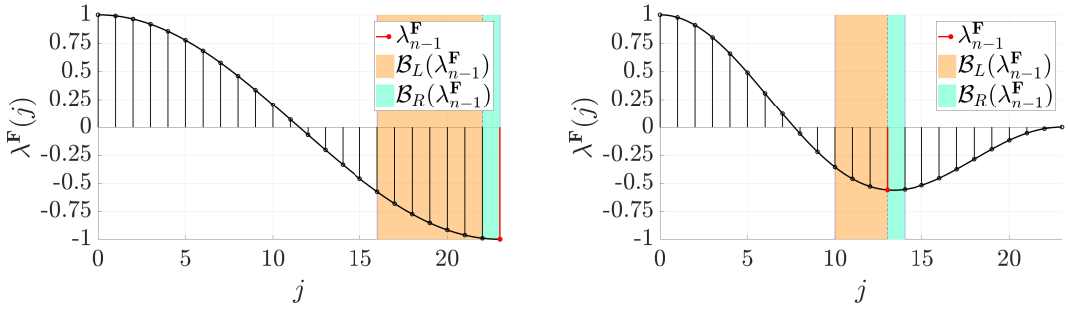
where constant $u_5 := 4.493409457909064175307880936048$ is generated and declared in Wiggins (2007). In addition, it is worth to notice that eigenvalue $\lambda^{\mathbf{F}}_{n-1}$, depicted with a red stem, belongs to set $\mathcal{B}_R(\lambda^{\mathbf{F}}_{n-1}) \cup \{j_\oplus\}$ for all κ adopted in this example and is equal to -1 for $\kappa = 1$, being, in such a case, indexed by $j^\star = \lfloor n/2 \rfloor = 23$.
Furthermore, one can observe that:

- index $j^\star$ (addressing $\lambda^{\mathbf{F}}_{n-1}$) is inversely proportional to κ;
- set $\mathcal{B}_L\left(\lambda^{\mathbf{F}}_{n-1}\right) \setminus \{j_\oplus\}$ degenerates quickly as κ increases and its cardinality reduces to 1 starting from $\kappa = 5$;
- index $j^\star$ may coincide with $j_\oplus$ also when sets $\mathcal{B}_L\left(\lambda^{\mathbf{F}}_{n-1}\right)$, $\mathcal{B}_R\left(\lambda^{\mathbf{F}}_{n-1}\right)$ possess cardinality greater or equal to 1 (see $\kappa = 2$ and $\kappa = 4$);
- for $\kappa > \kappa_n = 13.3$ set $\mathcal{B}_R\left(\lambda^{\mathbf{F}}_{n-1}\right)$ degenerates and $j^\star = 2$ begins to hold;
- for $\kappa > \kappa_\theta = 17.5506$ relation $\left|\lambda^{\mathbf{F}}_1\right| < \left|\lambda^{\mathbf{F}}_{n-1}\right|$ is satisfied.

Lastly, another important consideration is proposed in order to improve lower bound $\underline{j}$ in (6.19). Indeed, the example of binary search initialization illustrated in Fig. A.2, as many other similar preformed simulations omitted in this work, lead to the following

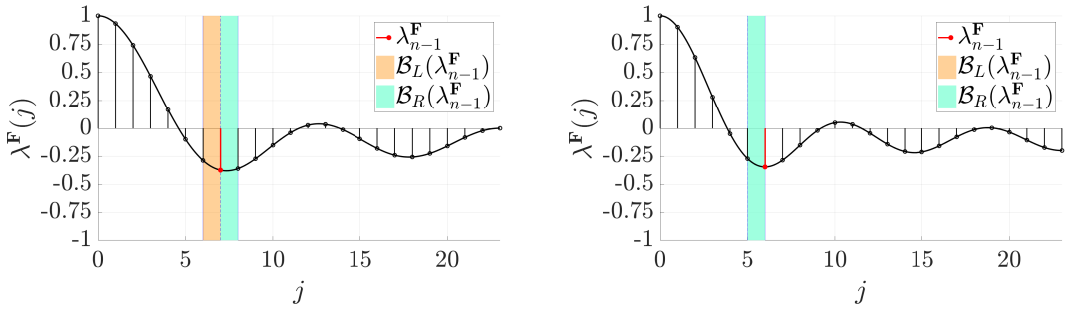
Conjecture A.2.1 (Bounds for the interval of the spectral radius index)**.**
Relation $\lambda^{\mathbf{L}}_{n-1} = \lambda^{\mathbf{L}}(j^\star) = \lambda^{\mathbf{L}}(n - j^\star)$ *holds with* $j^\star \in \mathbb{N}$ *belonging to* $[j_\oplus, \overline{j}] \subset \mathbb{N}$*, where* $j_\oplus$ *and* $\overline{j}$ *are defined as in* (A.3) *and* (6.19) *respectively.*

If statement of Conjecture A.2.1 held true, the computation of $\lambda^{\mathbf{L}}_{n-1}$ would be facilitated and the computational complexity of Alg. 9 would be substantially lowered from $O(\log_2(n/(2\kappa+1)))$ to $O(\log_2((3\pi - 2u_5)n/(4\pi\kappa))) \simeq O(\log_2(0.0348n/\kappa))$ as the ratio n/κ goes to infinity.



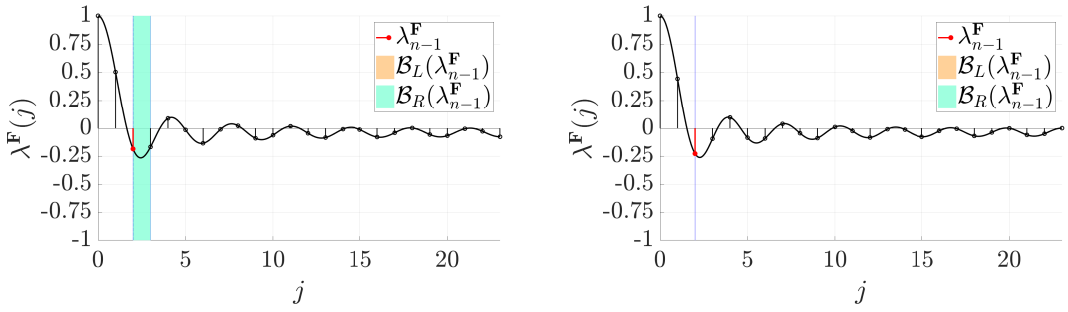
(a) $\kappa = 1$, $\lambda^{\mathbf{F}}_1 \simeq 0.9907$, $\lambda^{\mathbf{F}}_{n-1} = -1$

(b) $\kappa = 2$, $\lambda^{\mathbf{F}}_1 \simeq 0.9768$, $\lambda^{\mathbf{F}}_{n-1} \simeq -0.5603$



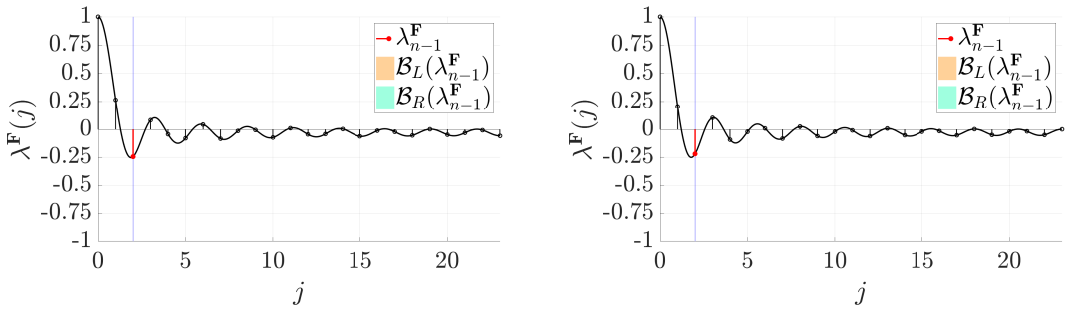
(c) $\kappa = 4$, $\lambda^{\mathbf{F}}_1 \simeq 0.9313$, $\lambda^{\mathbf{F}}_{n-1} \simeq -0.3742$

(d) $\kappa = 5$, $\lambda^{\mathbf{F}}_1 \simeq 0.9002$, $\lambda^{\mathbf{F}}_{n-1} \simeq -0.3458$



(e) $\kappa = 13$, $\lambda^{\mathbf{F}}_1 \simeq 0.5042$, $\lambda^{\mathbf{F}}_{n-1} \simeq -0.1852$

(f) $\kappa = 14$, $\lambda^{\mathbf{F}}_1 \simeq 0.4443$, $\lambda^{\mathbf{F}}_{n-1} \simeq -0.2274$



(g) $\kappa = 17$, $\lambda^{\mathbf{F}}_1 \simeq 0.2648$, $\lambda^{\mathbf{F}}_{n-1} \simeq -0.2449$

(h) $\kappa = 18$, $\lambda^{\mathbf{F}}_1 \simeq 0.2070$, $\lambda^{\mathbf{F}}_{n-1} \simeq -0.2200$

Figure A.2. Binary search initialization (not needed for $\kappa = 1$) to compute eigenvalue $\lambda^{\mathbf{F}}_{n-1} = \lambda^{\mathscr{R}}_{n-1}$ for κ-ring graphs with $n = 46$ vertices, $\kappa_n = 13.3$, $\kappa_\theta \simeq 17.5506$. Orange area $\mathcal{B}_L(\lambda^{\mathbf{F}}_{n-1}) \setminus \{j_\oplus\}$ does not contain estimates for $\lambda^{\mathbf{F}}_{n-1}$ from the beginning.

B

Appendix

"We need guns. Lots of guns."
Neo

Contents

B.1 Elements of Algebraic Topology and Geometry

This first section of the appendix is highly related to the concepts arising in Chap. 3, since many topological and geometrical objects are used while modeling. Specifically, all the following topics can be found in Coxeter (1947); Hatcher (2002); Nakahara (2003). The remainder of the section is organized as follows. Subsec. B.1.1 illustrates basic notions of Euclidean Geometry and Topology. Subsec. B.1.2 provides an explanation of all the complex topological tools (e.g. the simplicial complexes) characterizing procedure at line 1 in Alg. 3. Subsec. B.1.3 introduces the bases of Group Theory that are necessary to the gist of *homology* and the understanding of contribution given by Alg. 4.

B.1.1 Geometrical and topological entities

Definition B.1.1 (Point)**.**
A *point* $\mathbf{p}_i$ is an element belonging to the Euclidean space $\mathbb{R}^M$ of dimension $M \geq 1$.

Definition B.1.2 (Segment)**.**
A *segment* st_{ij} delimited by the pair of points $(\mathbf{p}_i, \mathbf{p}_j) \in \mathbb{R}^M \times \mathbb{R}^M$ is defined as the collection of points described by the set $st_{ij} = \{\eta\mathbf{p}_i + (1-\eta)\mathbf{p}_j \mid \mathbf{p}_i \neq \mathbf{p}_j,\ \forall\eta \in [0,1]\}$.

Definition B.1.3 (Polygonal, closed polygonal, planar polygonal)**.**
A *polygonal* $pl_{[0,L]}$ is defined as the finite collection of segments $pl_{[0,L]} = \{st_1, st_2, \ldots, st_L\} = \left\{st_{k_0k_1}, st_{k_1k_2}, \ldots, st_{k_{l-1}k_l}\right\}$, assuming $L \geq 2$, and such that a polygonal is *closed* whenever $\mathbf{p}_{k_0} = \mathbf{p}_{k_l}$, using the notation $pl_{[1,L]}$ in this case. A polygonal is *planar* if it spans a two-dimensional subspace of $\mathbb{R}^M$.

Definition B.1.4 (Polygon)**.**
For $M \geq 2$, a *polygon* $pn_{[1,L]}$ is defined as the closed set of points delimited and formed by the interior of the union $pn_{[1,L]} = \bigcup_{k=1}^{L} st_k$, such that each segment st_k belongs to a closed planar polygonal $pl_{[1,L]}$ that satisfies the following two properties:

- $\mathbf{p}_{k_i} \neq \mathbf{p}_{k_j}$ for all $i, j = 1, \ldots, L$, with $i \neq j$;
- $\sum_{i<j;\ i,j=1,\ldots,L} |st_i \cap st_j| = L$.

Definition B.1.5 (Regular polygon)**.**
A *regular polygon* is a polygon that is equiangular (all angles are equal in measure) and equilateral (all sides have the same length).

Definition B.1.6 (Polytope, edge, vertex)**.**
A *polytope* is an abstract extension[1] of a polygon spanning a subspace Ω_κ of $\mathbb{R}^M$ with dimension $\kappa \geq 2$, e.g. a polygon is a polytope spanning a two-dimensional subspace. Moreover, a polytope that spans a subspace of dimension $\kappa = 1$ is called *edge* (e.g. a segment), and a polytope that spans a subspace of dimension $\kappa = 0$ is called *vertex* (e.g. a point).

Definition B.1.7 (Convex set)**.**
A *convex set* Ω is a subset of a Euclidean space $\mathbb{R}^M$ such that every segment st_{ij} with vertices $(\mathbf{p}_i, \mathbf{p}_j) \in \Omega \times \Omega$ is a subset of Ω.

Definition B.1.8 (Convex hull)**.**
The *convex hull* of a set Ω of points in a Euclidean space $\mathbb{R}^M$ is the smallest convex set that contains Ω.

Definition B.1.9 (Topological space)**.**
Let $\mathcal{X}$ be any set and $\mathbb{T} = \{\Omega_i | i \in \Omega_I\}$ denote a certain collection of subsets of $\mathcal{X}$. The pair $(\mathcal{X}, \mathbb{T})$ is a *topological space* if $\mathbb{T}$ satisfies the following requirements:

(i) $\varnothing, \mathcal{X} \in \mathbb{T}$.

(ii) If $\mathbb{T}$ is any (maybe finite) subcollection of Ω_I, the family $\{\Omega_j | j \in \Omega_J\}$ satisfies $\bigcup_{j\in\Omega_J} \Omega_j \in \mathbb{T}$.

(iii) If Ω_K is any finite subcollection of Ω_I, the family $\{\Omega_k | k \in \Omega_K\}$ satisfies $\bigcap_{k\in\Omega_K} \Omega_k \in \mathbb{T}$.

[1] The details of this definition fall outside the purpose of the appendix.

$\mathcal{X}$ alone is sometimes called *topological space*. The Ω_i are called the *open sets* and $\mathbb{T}$ is said to give a *topology* to $\mathcal{X}$.

Definition B.1.10 (Metric and distance)**.**
A *metric* on a topological space $\mathcal{X}$ is the law chosen for a function $\mathcal{M}_{\mathcal{X}} : \mathcal{X} \times \mathcal{X} \to [0, +\infty)$ that maps two arbitrary elements $\omega_1, \omega_2 \in \mathcal{X}$ into their *distance* $\mathrm{dist}(\omega_1, \omega_2)$, satisfying the following conditions:

1. $\mathrm{dist}(\omega_1, \omega_2) \geq 0$ (separation axiom);
2. $\mathrm{dist}(\omega_1, \omega_2) = 0$ if and only if $\omega_1 = \omega_2$ (identity of indiscernible elements);
3. $\mathrm{dist}(\omega_1, \omega_2) = \mathrm{dist}(\omega_2, \omega_1)$ (symmetry);
4. for all $\omega_0 \in \mathcal{X}$, $\mathrm{dist}(\omega_0, \omega_2) \leq \mathrm{dist}(\omega_0, \omega_1) + \mathrm{dist}(\omega_1, \omega_2)$ (subadditivity).

Definition B.1.11 (Metric space)**.**
A *metric space* $(\mathcal{X}, \mathcal{M}_{\mathcal{X}})$ is a topological space $\mathcal{X}$ endowed with its metric $\mathcal{M}_{\mathcal{X}}$.

Definition B.1.12 (Diameter of a subset)**.**
Given a metric space $(\mathcal{X}, \mathcal{M}_{\mathcal{X}})$, let $\mathcal{X}_0$ be a subset of $\mathcal{X}$. The *diameter* of $\mathcal{X}_0$ is defined as $\phi(\mathcal{X}_0) = \sup\{\mathrm{dist}(\omega_1, \omega_2) | \omega_1, \omega_2 \in \mathcal{X}_0\}$.

Definition B.1.13 (Interior point and interior of a set)**.**
If Ω is a subset of a topological space $\mathcal{X}$, then ω is an interior point of Ω if there exists a neighborhood centered at ω which is completely contained in Ω. The interior $\mathrm{int}(\mathcal{X})$ of $\mathcal{X}$ is the set of all interior points of $\mathcal{X}$.

Definition B.1.14 (Limit point)**.**
Let Ω be a subset of a topological space $\mathcal{X}$. A point ω in $\mathcal{X}$ is a *limit point* of Ω if every neighborhood of ω contains at least one point of Ω different from ω itself.

Definition B.1.15 (Closure of a set)**.**
The *closure* of a subset Ω of points in a topological space $\mathcal{X}$ consists of all points in Ω together with all limit points of Ω. The closure of Ω may equivalently be defined as the union of Ω and its boundary, and also as the intersection of all closed sets containing Ω.

B.1.2 Simplices and simplicial complexes

Definition B.1.16 (Simplex)**.**
A *κ-simplex* is a κ-dimensional polytope which is the convex hull of its $\kappa + 1$ vertices. More formally, suppose the $\kappa+1$ points $\mathbf{p}_0, \ldots, \mathbf{p}_\kappa \in \mathbb{R}^\kappa$ are affinely independent, meaning that vectors $\mathbf{p}_1 - \mathbf{p}_0, \ldots, \mathbf{p}_k - \mathbf{p}_0$ are linearly independent. Then, the *κ-simplex* $S^{(\kappa)}$ determined by them is the set of points

$$S^{(\kappa)} = \left\{ \omega_0 \mathbf{p}_0 + \cdots + \omega_\kappa \mathbf{p}_\kappa \mid \omega_i \geq 0, \quad 0 \leq i \leq \kappa, \quad \sum_{i=0}^{\kappa} \omega_i = 1 \right\} \tag{B.1}$$

Conventionally, $S^{(-1)} = \varnothing$ is settled.

Example B.1.17 (Simplices of dimensions $\kappa = 0, \ldots, 3$)**.**
A simplex is a generalization of the notion of a triangle or tetrahedron to arbitrary dimensions. As illustrated in Fig. 2.10, a single point may be considered a 0-simplex, and a segment may be considered a 1-simplex. A 2-simplex is a triangle, a 3-simplex is a tetrahedron. Any simplex can be defined as the smallest convex set containing the given vertices.

Definition B.1.18 (Faces of a simplex)**.**
The convex hull of any nonempty subset of the $\kappa + 1$ points that define a κ-simplex is called a *face* of the simplex.

Remark B.1.19.
Faces are simplices themselves. In particular, if $0 \leq \kappa' \leq \kappa$, the convex hull of a subset of size $\kappa' + 1$ (of the $\kappa + 1$ defining points) is an κ'-simplex, called an κ'-face of the κ-simplex. The 0-faces (i.e., the defining points themselves as sets of size 1) are called the *vertices*, the 1-faces are called the *edges*, the $\kappa - 1$-faces are called the *facets*, and the sole κ-face is the whole κ-simplex itself.

Definition B.1.20 (Cofaces of a simplex)**.**
A simplex $S_j^{(\kappa_j)}$ is a *coface* of a simplex $S_i^{(\kappa_i)}$ if $S_i^{(\kappa_i)}$ is a face of $S_j^{(\kappa_j)}$.

Definition B.1.21 (Regular simplex)**.**
A *regular simplex* is a simplex that is also a regular polytope. A regular κ-simplex may be constructed from a regular $(\kappa - 1)$-simplex by connecting a new vertex to all original vertices by the common edge length.

Definition B.1.22 (Simplicial complex)**.**
As depicted in Fig. B.1, a *κ-simplicial complex* of dimension κ is a set $\mathcal{S}^{(\kappa)}$ of simplices in $\mathbb{R}^\kappa$ such that:

- each face $S_i^{(\kappa_i')}$ of a κ_i-simplex $S_i^{(\kappa_i)}$ belonging to $\mathcal{S}^{(\kappa)}$ is an element of $\mathcal{S}^{(\kappa)}$;
- the intersection of two simplices $S_j^{(\kappa_j)} \in \mathcal{S}^{(\kappa)}$ and $S_j^{(\kappa_j)} \in \mathcal{S}^{(\kappa)}$ is the empty set or a face that is common to both $S_j^{(\kappa_j)} \in \mathcal{S}^{(\kappa)}$ and $S_j^{(\kappa_j)} \in \mathcal{S}^{(\kappa)}$;
- the set $\mathcal{S}^{(\kappa)}$ is locally finite, i.e. for all bounded subsets $\Omega \subset \mathbb{R}^\kappa$ the intersection between Ω and $\mathcal{S}^{(\kappa)}$ is finite.

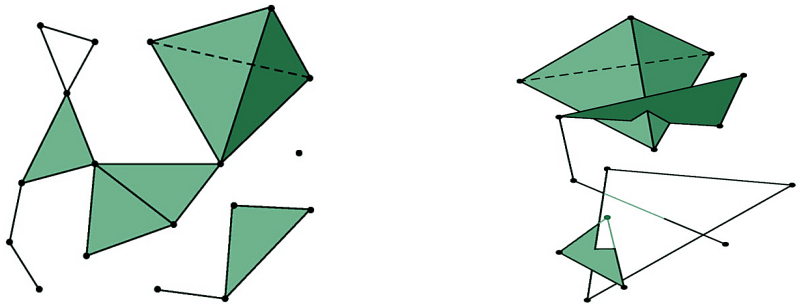
Remark B.1.23.
A κ-simplicial complex $\mathcal{S}^{(\kappa)}$ does not necessarily have a finite amount of elements. Its dimension κ is just an index of the maximum dimension among all the simplices contained in $\mathcal{S}^{(\kappa)}$.

Definition B.1.24 (Simplicial subcomplex)**.**
If $\mathcal{S}_i^{(\kappa_i)}$ is a subcollection of a simplicial complex $\mathcal{S}^{(\kappa)}$ that contains all faces of its elements, where $0 \leq \kappa \leq \kappa_i$, then $\mathcal{S}_i^{(\kappa_i)}$ is another simplicial complex called a *simplicial subcomplex*.

Definition B.1.25 (κ-skeleton)**.**
The *κ-skeleton* of a simplicial complex $\mathcal{S}^{(\kappa')}$ is defined as the simplicial subcomplex $\mathcal{S}^{(\kappa)}$ that is the union of all the simplices in $\mathcal{S}^{(\kappa')}$ of dimensions $\kappa \leq \kappa'$.

Definition B.1.26 (Abstract simplicial complex)**.**
A family $\mathcal{AS}^{(\kappa)}$ of non-empty finite subsets of a set Ω is an *abstract simplicial complex* if, for every set $S_j^{(\kappa_j)}$ (abstract coface) in $\mathcal{AS}^{(\kappa)}$, and every non-empty subset $S_i^{(\kappa_i)} \subseteq S_j^{(\kappa_j)}$, then $S_i^{(\kappa_i)}$ (abstract face) also belongs to $\mathcal{AS}^{(\kappa)}$. Concepts as the *abstract simplicial subcomplex* or the *abstract skeleton* are defined in a similar way to those in definitions B.1.24 and B.1.25.

Remark B.1.27.
An abstract simplicial complex is a purely combinatorial extension of the geometric notion of a simplicial complex (see Fig. B.1), consisting of a family of non-empty finite sets closed under the operation of taking non-empty subsets.



(a) A simplicial complex in $\mathbb{R}^3$

(b) An abstract simplicial complex in $\Omega = \mathbb{R}^3$ (not a simplicial complex)

Figure B.1. (a) A simplicial complex of dimension 3, since it is made of vertices, edges, triangles and tetrahedra. (b) An abstract simplicial complex that is not a simplicial complex, since it contains simplices whose intersections are neither the empty set nor a common face.

Definition B.1.28 (Vietoris-Rips complex)**.**
A *Vietoris-Rips complex* is an abstract simplicial complex that can be defined from any metric space $(\mathcal{X}, \mathcal{M}_\mathcal{X})$ and a constant maximum distance $\ell_\mathcal{X}$ by forming a simplex for every finite set of points $\mathcal{X}_0$ in $\mathcal{X}$ that has a diameter $\phi(\mathcal{X}_0) \leq \ell_\mathcal{X}$.

Example B.1.29 (Vietoris-Rips complex)**.**
Fig. B.2 depicts a Vietoris–Rips complex of a set of 42 points in the Euclidean plane. This complex has simplices of up to four points: the points themselves (black circles), pairs of points (black edges), triples of points (blue violet triangles), and quadruples of points (purple tetrahedrons). Circles in aqua marine represent the maximum diameter $\ell_\mathcal{X}$ attainable by the overall set of points.

B.1.3 Notions of Homology Groups

Definition B.1.30 (Group, identity element, inverse element)**.**
A *group* $(\mathbb{G}, \circledast)$ is a set $\mathbb{G}$ equipped with a binary operation $\circledast$ that combines any two elements $\omega_1, \omega_2 \in \mathbb{G}$ to form a third element $\omega_1 \circledast \omega_2$ in such a way that four conditions called *group axioms* are satisfied:

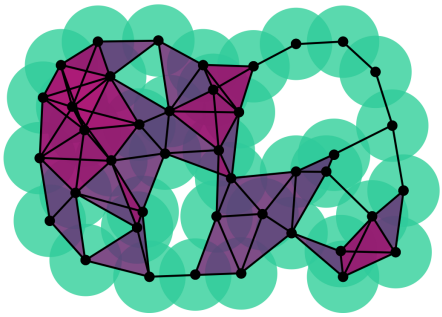


Figure B.2. A Vietoris–Rips complex of a set of 42 points in the Euclidean plane.

1. For all $\omega_1, \omega_2 \in \mathbb{G}$, the result of the operation $\omega_1 \circledast \omega_2$ also belongs to $\mathbb{G}$ (clusure);
2. For all $\omega_0, \omega_1, \omega_2 \in \mathbb{G}$ it holds that $(\omega_0 \circledast \omega_1) \circledast \omega_2 = \omega_0 \circledast (\omega_1 \circledast \omega_2)$ (associativity);
3. There exists a unique element $\mathscr{e} \in \mathbb{G}$ such that, for every element $\omega_0 \in \mathbb{G}$, the equation $\mathscr{e} \circledast \omega_0 = \omega_0 \circledast \mathscr{e}$ holds (identity element);
4. For each $\omega_1 \in \mathbb{G}$, there exists an element $\omega_2 \in \mathbb{G}$, commonly denoted ω_1^{-1} (or $-\omega_1$, if the operation is denoted with $+$), such that $\omega_1 \circledast \omega_2 = \omega_2 \circledast \omega_1 = \mathscr{e}$, where $\mathscr{e}$ is the identity element (inverse element).

Definition B.1.31 (Subgroup)**.**
A *subgroup* $(\mathbb{H}, \circledast)$ of a group $(\mathbb{G}, \circledast)$ is if $\mathbb{H}$ also forms a group under the operation $\circledast$. This is usually denoted with $\mathbb{H} \leq \mathbb{G}$.

Definition B.1.32 (Left and right cosets of a subgroup in a group)**.**
Let $(\mathbb{G}, \circledast)$ and $(\mathbb{H}, \circledast)$ be two groups such that $\mathbb{H} \leq \mathbb{G}$. Let also $\omega_{\mathbb{G}}$ be an element of $\mathbb{G}$. Then $\omega_{\mathbb{G}}\mathbb{H} = \{\omega_{\mathbb{G}} \circledast \omega_{\mathbb{H}} | \forall \omega_{\mathbb{H}} \in \mathbb{H}\}$ is the *left coset* of $\mathbb{H}$ in $\mathbb{G}$ w.r.t. $\omega_{\mathbb{G}}$, and $\mathbb{H}\omega_{\mathbb{G}} = \{\omega_{\mathbb{H}} \circledast \omega_{\mathbb{G}} | \forall \omega_{\mathbb{H}} \in \mathbb{H}\}$ is the *right coset* of $\mathbb{H}$ in $\mathbb{G}$ w.r.t. $\omega_{\mathbb{G}}$.

Definition B.1.33 (Normal subgroup)**.**
Let $\mathbb{H}$ be a subgroup of the group $\mathbb{G}$. The subgroup $\mathbb{H}$ is said to be *normal* if, for all $\omega_{\mathbb{G}} \in \mathbb{G}$ the left and right cosets of $\mathbb{H}$ in $\mathbb{G}$ are equivalent, namely $\omega_{\mathbb{G}}\mathbb{H} = \mathbb{H}\omega_{\mathbb{G}}$.

Definition B.1.34 (Abelian group)**.**
An *Abelian group* $(\mathbb{GA}, \circledast)$ is a group that satisfies the commutativity property, i.e. for all $\omega_1, \omega_2 \in \mathbb{GA}$, it holds that $\omega_1 \circledast \omega_2 = \omega_2 \circledast \omega_1$.

Proposition B.1.35 (Abelian subgroups are normal)**.**
Let $\mathbb{GA}$ be an Abelian group and $\omega_{\mathbb{G}}$ an element of $\mathbb{GA}$. Then the left and right cosets of $\mathbb{GA}$ coincide, namely $\omega_{\mathbb{G}}\mathbb{HA} = \mathbb{HA}\omega_{\mathbb{G}}$ for every $\omega_{\mathbb{G}} \in \mathbb{GA}$ and every Abelian subgroup such that $\mathbb{HA} \leq \mathbb{GA}$. This implies that every Abelian subgroup $(\mathbb{HA}, \circledast)$ is normal.

Definition B.1.36 (Quotient group)**.**
Let $(\mathbb{H}, \circledast)$ be a normal subgroup of a group $(\mathbb{G}, \circledast)$. Consider the set $\mathbb{G}/\mathbb{H}$ of all the left cosets of $\mathbb{H}$ in $\mathbb{G}$, that is $\mathbb{G}/\mathbb{H} = \{\omega_{\mathbb{G}}\mathbb{H} | \forall \omega_{\mathbb{G}} \in \mathbb{G}\}$, and an operation $\oslash$ satisfying $(\omega_{\mathbb{G},1}\mathbb{H}) \circledast (\omega_{\mathbb{G},2}\mathbb{H}) = (\omega_{\mathbb{G},1} \circledast \omega_{\mathbb{G},2})\mathbb{H}$ for all $\omega_{\mathbb{G},1}, \omega_{\mathbb{G},2} \in \mathbb{G}$. Then, $(\mathbb{G}/\mathbb{H}, \oslash)$ is said to be the *quotient group* of $(\mathbb{G}, \circledast)$ by $(\mathbb{H}, \circledast)$.

Example B.1.37 (Quotient group of $(\mathbb{Z},+)$ by $(3\mathbb{Z},+)$)**.**
Let $\mathbb{Z}$ be the set of integer numbers and $3\mathbb{Z}$ be the set in which only the multiples of the integer numbers by 3 are present, that is $3\mathbb{Z} = \{\ldots,-9,-6,-3,0,3,6,9,\ldots\}$. To compute the quotient group of $(\mathbb{Z},+)$ by $(3\mathbb{Z},+)$ it is possible to proceed as follows. Let us define all the possible cosets of $(3\mathbb{Z},+)$ in $(\mathbb{Z},+)$: $\bar{0} = 0\mathbb{Z} = \mathbb{Z}$, $\bar{1} = 1\mathbb{Z} = \{1+\omega|\omega \in 3\mathbb{Z}\}$ and $\bar{2} = 2\mathbb{Z} = \{2+\omega|\omega \in 3\mathbb{Z}\}$. Since the $+$ operation satisfies $\omega_1\mathbb{Z} + \omega_2\mathbb{Z} = (\omega_1+\omega_2)\mathbb{Z}$ for all $\omega_1,\omega_2 = 0,1,2$ and since $(3\mathbb{Z},+)$ is normal, the quotient group of $(\mathbb{Z},+)$ by $(3\mathbb{Z},+)$ is given by $(\{\bar{0},\bar{1},\bar{2}\},+)$.

Definition B.1.38 (Homomorphism)**.**
Let $\mathbb{G}$ and $\mathbb{H}$ two algebraic structures of the same type. Assume there exists an operation f defined on both $\mathbb{G}$ as $f_{\mathbb{G}}$ and on $\mathbb{H}$ as $f_{\mathbb{H}}$. A function $\Xi : \mathbb{G} \to \mathbb{H}$ is a *homomorphism* if, for every κ-tuple $\omega_1,\ldots,\omega_\kappa$ of $\mathbb{G}$ one has $\Xi(f_{\mathbb{G}}(\omega_1,\ldots,\omega_\kappa)) = f_{\mathbb{H}}(\Xi(\omega_1),\ldots,\Xi(\omega_\kappa))$.

Remark B.1.39. A homomorphism can be seen as a structure-preserving map between two algebraic structures of the same type, e.g. two groups endowed with an operation $\circledast$.

Definition B.1.40 (Trivial homomorphism)**.**
Let us assume the same hypotheses in Def. B.1.38. The *trivial homomorphism* Ξ_0 is the map that sends every element of $\mathbb{G}$ into the identity element $e_{\mathbb{H}}$ of $\mathbb{H}$. In formulas: $\Xi_0(f_{\mathbb{G}}(\omega_1,\ldots,\omega_\kappa)) = e_{\mathbb{H}}$.

Definition B.1.41 (Chain and boundary operator)**.**
A *chain* is a sequence of Abelian groups $(\mathbb{GA}_i,\circledast)$, indexed by $i = 1,2,\ldots$, and homomorphisms $\partial_i : \mathbb{GA}_i \to \mathbb{GA}_{i-1}$, called *boundary operators*, such that the composition of two subsequent homomorphisms has the trivial homomorphism as a result. In other words, it holds that $\partial_{i-1} \circ \partial_i = e_{\mathbb{GA}_{i-1}}$, where $e_{\mathbb{GA}_{i-1}}$ is the identity element of $\mathbb{GA}_{i-1}$.

Remark B.1.42.
A chain can be represented with the symbolic map defined as the composition of all the boundary operators:

$$\ldots \xrightarrow{\partial_{i+1}} \mathbb{GA}_i \xrightarrow{\partial_i} \mathbb{GA}_{i-1} \ldots \xrightarrow{\partial_2} \mathbb{GA}_1 \xrightarrow{\partial_1} \mathbb{GA}_0 \xrightarrow{\partial_0} e_{\mathbb{GA}_0}. \tag{B.2}$$

Proposition B.1.43.
Property $\partial_{i-1} \circ \partial_i = e_{\mathbb{GA}_{i-1}}$ *in Def. B.1.41 is equivalent to the requirement that*

$$\operatorname{im}\{\partial_{i+1}\} \subseteq \ker\{\partial_i\} \quad \text{for } i = 0,1,2,\ldots. \tag{B.3}$$

Definition B.1.44 (i-th homology group)**.**
Given a topological space $\mathcal{X}$, the *i-th homology group* $H_i(\mathcal{X})$ on $\mathcal{X}$ is defined as quotient group $H_i(\mathcal{X}) = \ker\{\partial_i\}/\operatorname{im}\{\partial_{i+1}\}$ in which the chain related to all boundary operators ∂_i is established from $\mathcal{X}$.

Remark B.1.45 (Relation between homology groups, boundaries and cycles)**.**
In Def. B.1.44, if $\operatorname{im}\{\partial_{i+1}\}$ and $\ker\{\partial_i\}$ were equal for all i then the chain is said to be *exact*. However, this does not generally occur; an homology group measures how much the chain is far to be exact. Furthemore, the i-th homology group $H_i(\mathcal{X})$ of a topological space $\mathcal{X}$ gives important information about the i-th dimensional holes in $\mathcal{X}$. For instance, $H_0(\mathcal{X})$ describes the path-connected components of $\mathcal{X}$.

Example B.1.46 (Homology groups of a torus)**.**
The topological groups of a torus $\mathcal{X}_T$ (see Fig. B.3) are given by $H_0(\mathcal{X}_T) = \mathbb{Z}$, $H_1(\mathcal{X}_T) = \mathbb{Z} \times \mathbb{Z}$, $H_2(\mathcal{X}_T) = \mathbb{Z}$. This tells us information about its topological structure. Indeed, $H_0(\mathcal{X}_T) = \mathbb{Z}$ corresponds to a single path-connected component, $H_1(\mathcal{X}_T)$ corresponds to two independent one-dimensional components (yellow and dark cyan circles) and $H_2(\mathcal{X}_T)$ corresponds to a one two-dimensional hole as the interior of the torus.

Definition B.1.47 (Simplicial homologies)**.**
A *simplicial i-homology* $H_i(\mathcal{AS}^{(\kappa)})$ is a homology group defined on a topological space $\mathcal{AS}^{(\kappa)}$ formed by simplices. In particular, for every simplex $S_j^{(\kappa_j)}$, $j = 1, 2, \ldots$, in an abstract simplicial complex $\mathcal{AS}^{(\kappa)}$, a specific ordering $(v_0^{(j)}, \ldots, \hat{v}_k^{(j)}, \ldots, v_{\kappa_j}^{(j)})$ can be assigned to the vertices of $S_j^{(\kappa_j)}$ with the aim to characterize its $(\kappa_j - 1)$-faces correspondent to the exclusion of vertices $\hat{v}_k^{(j)}$. Thanks to this possibility, the boundary operator $\partial_{\kappa_j}^{(j)}$ acting on the j-th simplex $S_j^{(\kappa_j)}$ can be defined as

$$\partial_{\kappa_j}^{(j)} = \sum_{k=0}^{\kappa_j} (-1)^k (v_0^{(j)}, \ldots, \hat{v}_k^{(j)}, \ldots, v_{\kappa_j}^{(j)}) \tag{B.4}$$

The choice in (B.4) allows to give proper meaning to the κ-chain complex $C^{(\kappa)}$ associated to $\mathcal{AS}^{(\kappa)}$, since $C^{(\kappa)}$ is established exploiting the Abelian group generated by means of the (formal) sum $\sum_j \omega_j S_j^{(\kappa_j)}$, with $\omega_j \in \mathbb{Z}$, for all $j = 1, 2, \ldots$.

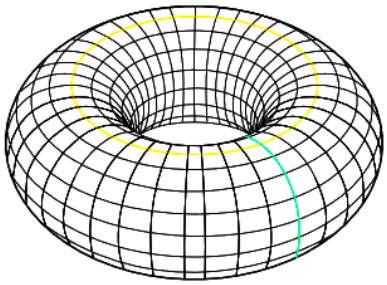


Figure B.3. The torus $\mathcal{X}_T$.

B.2 Elements of Linear Spectral Theory

In this second section of the appendix, many basic definitions, propositions, lemmas and fundamental theorems related to Spectral Theory, yet restricted to the framework of Linear Algebra, are stated. All the concept here illustrated can be found in Berman and Plemmons (1994); Fornasini (2013); Meyer (2000) and are widely employed in Chap. 5 and Chap. 6.

Definition B.2.1 (Hermitian matrix)**.**
Let $\mathbf{\Omega} \in \mathbb{C}^{N \times N}$ be a squared matrix and $\Re$, $\Im$ denote the real and imaginary part of a complex number respectively. Matrix $\mathbf{\Omega}$ is said *Hermitian* if $\mathbf{\Omega} = \mathbf{\Omega}^*$, where $*$ represents

the *conjugate transpose operator*, namely $\Re([\mathbf{\Omega}]_{ji}) = \Re([\mathbf{\Omega}]_{ij})$ and $\Im([\mathbf{\Omega}]_{ji}) = -\Im([\mathbf{\Omega}]_{ij})$, for $i = 1, \dots, N$, $j = 1, \dots, N$.

Definition B.2.2 (Symmetric matrix)**.**
Let $\mathbf{\Omega} \in \mathbb{R}^{N \times N}$ be a squared matrix. Matrix $\mathbf{\Omega}$ is said *symmetric* if $\mathbf{\Omega} = \mathbf{\Omega}^\top$, where $\top$ represents the *transpose operator*, namely $[\mathbf{\Omega}]_{ji} = [\mathbf{\Omega}]_{ij}$, for $i = 1, \dots, N$, $j = 1, \dots, N$.

Proposition B.2.3 (Relation between Hermitian and symmetric matrices)**.**
Let $\mathbf{\Omega} \in \mathbb{R}^{N \times N}$ be a squared matrix. Then $\mathbf{\Omega}$ is also Hermitian.

Definition B.2.4 (Characteristic polynomial, eigenvalue and algebraic multiplicity)**.**
Let $V_{\mathbf{\Omega}} \subseteq \mathbb{C}^N$ a vector space of dimension N and $f_{\mathbf{\Omega}} : V_{\mathbf{\Omega}} \to V_{\mathbf{\Omega}} : \boldsymbol{\omega} \mapsto \mathbf{\Omega}\boldsymbol{\omega}$ a *linear endomorphism* such that $\mathbf{\Omega} \in \mathbb{C}^{N \times N}$ is a squared matrix. It is said that $\bar{\lambda} \in \mathbb{C}$ is an *eigenvalue* of $\mathbf{\Omega}$ if $\bar{\lambda}$ is a root of the complex polynomial

$$\Delta(\lambda) = \det(\mathbf{\Omega} - \lambda \mathbf{I}_N), \qquad \lambda \in \mathbb{C}, \tag{B.5}$$

called *characteristic polynomial* associated to $f_{\mathbf{\Omega}}$. In other words, an eigenvalue $\bar{\lambda}$ satisfies the so-called *characteristic equation* $\Delta(\lambda) = 0$. The *algebraic multiplicity* $\text{mult}_A(\bar{\lambda}) \in \mathbb{N} \setminus \{0\}$ of $\bar{\lambda}$ is the number of times it is repeated as a root of the characteristic polynomial.

Definition B.2.5 (Spectrum of a matrix)**.**
Let $\mathbf{\Omega} \in \mathbb{C}^{N \times N}$ be a squared matrix. The set $\Lambda(\mathbf{\Omega}) = \left\{\lambda_0^{\mathbf{\Omega}}, \dots, \lambda_{N-1}^{\mathbf{\Omega}}\right\}$ containing all the eigenvalues[2] $\lambda_i^{\mathbf{\Omega}}$ of $\mathbf{\Omega}$, for $i = 0, \dots, N-1$, is called *spectrum* of $\mathbf{\Omega}$.

Definition B.2.6 (Spectral radius)**.**
Let $\lambda_0^{\mathbf{\Omega}}, \dots, \lambda_{N-1}^{\mathbf{\Omega}}$ the eigenvalues of a squared matrix $\mathbf{\Omega} \in \mathbb{C}^{N \times N}$. The operator $\rho_S : \mathbb{C}^{N \times N} \to \mathbb{R}$ that maps $\mathbf{\Omega}$ into the real positive value

$$\rho_S(\mathbf{\Omega}) = \max\left\{|\lambda_0^{\mathbf{\Omega}}|, \dots, |\lambda_{N-1}^{\mathbf{\Omega}}|\right\} \tag{B.6}$$

is called *spectral radius*.

Definition B.2.7 (Left and right eigenvector, eigenpair)**.**
Under the same assumptions of Def. B.2.4, let $\lambda_i^{\mathbf{\Omega}}$ be the i-th eigenvalue in $\Lambda(\mathbf{\Omega})$ and consider the following equations in the unknown $\boldsymbol{\omega} \in \mathbb{C}^N$

$$\mathbf{\Omega}\boldsymbol{\omega} = \lambda_i^{\mathbf{\Omega}}\boldsymbol{\omega}; \tag{B.7}$$

$$\boldsymbol{\omega}^\top\mathbf{\Omega} = \lambda_i^{\mathbf{\Omega}}\boldsymbol{\omega}^\top. \tag{B.8}$$

If $\boldsymbol{\varpi}_i^{\mathbf{\Omega}} \in V_{\mathbf{\Omega}}$ satisfies (B.7), then it is said to be a *right eigenvector* (or simply, an *eigenvector*) associated to the eigenvalue $\lambda_i^{\mathbf{\Omega}}$, forming the i-th *eigenpair* $(\boldsymbol{\varpi}_i^{\mathbf{\Omega}}, \lambda_i^{\mathbf{\Omega}})$. Furthermore, if $\boldsymbol{\upsilon}_i^{\mathbf{\Omega}} \in V_{\mathbf{\Omega}}$ satisfies (B.8), then it is said to be a *left eigenvector* associated to the eigenvalue $\lambda_i^{\mathbf{\Omega}}$.

Definition B.2.8 (Eigenspace and geometric multiplicity)**.**
Under the same assumptions of Def. B.2.4, the *eigenspace* associated to the i-th eigenvalue

[2] Sorted with an arbitrary criterion.

$\lambda_i^{\boldsymbol{\Omega}}$ is the vector subspace spanned by $W_i^{\boldsymbol{\Omega}} = \{\boldsymbol{\omega} \in V_{\boldsymbol{\Omega}} \setminus \{\mathbf{0}_N\} | \boldsymbol{\omega} \in \ker(\boldsymbol{\Omega} - \lambda_i^{\boldsymbol{\Omega}} \mathbf{I}_N)\}$, i.e. the subspace spanned by the eigenvectors associated to $\lambda_i^{\boldsymbol{\Omega}}$. The integer $\text{mult}_G(\lambda_i^{\boldsymbol{\Omega}}) = \dim(W_i^{\boldsymbol{\Omega}}) \in \mathbb{N} \setminus \{0\}$ is called the *geometric multiplicity* of $\lambda_i^{\boldsymbol{\Omega}}$.

Proposition B.2.9 (Algebraic and geometric multiplicity relationship)**.**
Under the same assumptions of Def. B.2.4, let $\lambda_i^{\boldsymbol{\Omega}}$ be the i-th eigenvalue of $\boldsymbol{\Omega}$. One has

$$\text{mult}_G(\lambda_i^{\boldsymbol{\Omega}}) \leq \text{mult}_A(\lambda_i^{\boldsymbol{\Omega}}), \quad \textit{for } i = 0, \dots, N-1. \tag{B.9}$$

Definition B.2.10 (Simple and semi-simple eigenvalue)**.**
An eigenvalue is said to be *semi-simple* if its algebraic multiplicity and geometric multiplicity are equal. Moreover, it is said *simple* if its algebraic multiplicity is unitary.

Definition B.2.11 (Definiteness of a matrix)**.**
Let $\boldsymbol{\Omega}$ a squared matrix in $\mathbb{R}^{N \times N}$ (resp. a squared Hermitian matrix in $\mathbb{C}^{N \times N}$). Consider the *quadratic form* $\langle \boldsymbol{\omega}, \boldsymbol{\omega} \rangle_{\boldsymbol{\Omega}} = \boldsymbol{\omega}^\top \boldsymbol{\Omega} \boldsymbol{\omega} \in \mathbb{R}$. Let also $\mathbb{R}_0^N$ (resp. $\mathbb{C}_0^N$) be equal to $\mathbb{R}^N \setminus \{\mathbf{0}_N\}$ (resp. $\mathbb{C}^N \setminus \{\mathbf{0}_N\}$). Matrix $\boldsymbol{\Omega}$ is said to be

(i) *positive definite* if $\langle \boldsymbol{\omega}, \boldsymbol{\omega} \rangle_{\boldsymbol{\Omega}} > 0$, i.e. $\boldsymbol{\Omega} \succ 0$, for all vectors $\boldsymbol{\omega} \in \mathbb{R}_0^N$ (resp. $\boldsymbol{\omega} \in \mathbb{C}_0^N$);

(ii) *positive semi-definite* if $\langle \boldsymbol{\omega}, \boldsymbol{\omega} \rangle_{\boldsymbol{\Omega}} \geq 0$, i.e. $\boldsymbol{\Omega} \succeq 0$, for all vectors $\boldsymbol{\omega} \in \mathbb{R}_0^N$ (resp. $\boldsymbol{\omega} \in \mathbb{C}_0^N$) and, for some $\boldsymbol{\omega} \in \mathbb{R}_0^N$ (resp. $\boldsymbol{\omega} \in \mathbb{C}_0^N$), it holds that $\langle \boldsymbol{\omega}, \boldsymbol{\omega} \rangle_{\boldsymbol{\Omega}} = 0$;

(iii) *negative definite* if $\langle \boldsymbol{\omega}, \boldsymbol{\omega} \rangle_{\boldsymbol{\Omega}} < 0$, i.e. $\boldsymbol{\Omega} \prec 0$, for all vectors $\boldsymbol{\omega} \in \mathbb{R}_0^N$ (resp. $\boldsymbol{\omega} \in \mathbb{C}_0^N$);

(iv) *negative semi-definite* if $\langle \boldsymbol{\omega}, \boldsymbol{\omega} \rangle_{\boldsymbol{\Omega}} \leq 0$, i.e. $\boldsymbol{\Omega} \preceq 0$, for all vectors $\boldsymbol{\omega} \in \mathbb{R}_0^N$ (resp. $\boldsymbol{\omega} \in \mathbb{C}_0^N$) and, for some $\boldsymbol{\omega} \in \mathbb{R}_0^N$ (resp. $\boldsymbol{\omega} \in \mathbb{C}_0^N$), it holds that $\langle \boldsymbol{\omega}, \boldsymbol{\omega} \rangle_{\boldsymbol{\Omega}} = 0$;

(v) *indefinite* if none of the above options holds.

Proposition B.2.12 (Spectral definiteness of Hermitian matrices)**.**
Let $\boldsymbol{\Omega} \in \mathbb{C}^{N \times N}$ be a Hermitian matrix. Then matrix $\boldsymbol{\Omega}$ is

(i) positive definite if and only if all its eigenvalues are strictly positive;

(ii) positive semi-definite if and only if all its eigenvalues are nonnegative and there exists a null eigenvalue;

(iii) negative definite if and only if all its eigenvalues are strictly negative;

(iv) negative semi-definite if and only if all its eigenvalues are nonpositive and there exists a null eigenvalue;

(v) indefinite if and only if none of the above options holds.

Corollary B.2.13 (Spectral Theorem applied to Hermitian matrices)**.**
If a squared matrix $\boldsymbol{\Omega} \in \mathbb{C}^{N \times N}$ is Hermitian, there exists an orthonormal basis for $V_{\boldsymbol{\Omega}}$, namely the space that defines the endomorphism generated by $\boldsymbol{\Omega}$, consisting of eigenvectors of $\boldsymbol{\Omega}$. Moreover, each eigenvalue of $\boldsymbol{\Omega}$ is real.

Definition B.2.14 (Gerschgorin disc)**.**
Let $\boldsymbol{\Omega} \in \mathbb{C}^{N \times N}$ be a squared matrix with entries $[\boldsymbol{\Omega}]_{ij} = \omega_{ij}$. The i-th *Gerschgorin disc* $D_{G,i}(\omega_{ii}, R_{ii})$ is defined as the circle centered at ω_{ii} with radius $R_{ii} = \sum_{j \neq i} |\omega_{ij}|$.

Theorem B.2.15 (Gerschgorin Circle Theorem)**.**
Let $\boldsymbol{\Omega} \in \mathbb{C}^{N\times N}$ be a squared matrix with entries $[\boldsymbol{\Omega}]_{ij} = \omega_{ij}$. Let us also define the union of Gerschgorin discs $D_{G,R} = \cup_{i=1}^{N} D_{G,i}$ for the rows of $\boldsymbol{\Omega}$ and the union of Gerschgorin discs $D_{G,C} = \cup_{j=1}^{N} D_{G,j}$ for the rows of $\boldsymbol{\Omega}^\top$, where $[\boldsymbol{\Omega}^\top]_{ji} = \omega_{ij}$. Then the spectrum of $\boldsymbol{\Omega}$ is contained in the intersection $D_{G,R} \cap D_{G,C}$.

Definition B.2.16 (Rayleigh quotient)**.**
For a given complex Hermitian matrix $\boldsymbol{\Omega}$ and nonzero vector $\boldsymbol{\omega}$, the *Rayleigh quotient* is defined as the map $\mathscr{R}(\boldsymbol{\Omega}, \boldsymbol{\omega}) : \mathbb{C}^{N\times N} \times \mathbb{C}^N \to \mathbb{R}$ such that

$$\mathscr{R}(\boldsymbol{\Omega}, \boldsymbol{\omega}) = \frac{\langle \boldsymbol{\omega}, \boldsymbol{\omega} \rangle_{\boldsymbol{\Omega}}}{\boldsymbol{\omega}^\top \boldsymbol{\omega}} = \frac{\boldsymbol{\omega}^\top \boldsymbol{\Omega} \boldsymbol{\omega}}{\boldsymbol{\omega}^\top \boldsymbol{\omega}}. \tag{B.10}$$

Proposition B.2.17 (Fundamental property of the Rayleigh quotient)**.**
Under the same assumptions of Def. B.2.16, let us consider the eigenpair $(\lambda_i^{\boldsymbol{\Omega}}, \boldsymbol{\varpi}_i^{\boldsymbol{\Omega}})$. Then

$$\mathscr{R}(\boldsymbol{\Omega}, \boldsymbol{\varpi}_i^{\boldsymbol{\Omega}}) = \lambda_i^{\boldsymbol{\Omega}}. \tag{B.11}$$

Theorem B.2.18 (Min-max Theorem)**.**
Let $\boldsymbol{\Omega} \in \mathbb{C}^{N\times N}$ be an Hermitian matrix with eigenvalues $\lambda_0^{\boldsymbol{\Omega}} \leq \cdots \leq \lambda_{N-1}^{\boldsymbol{\Omega}}$ then it holds that

$$\lambda_i^{\boldsymbol{\Omega}} = \min_{X_i} \max_{\boldsymbol{\omega} \in X_i} \mathscr{R}(\boldsymbol{\Omega}, \boldsymbol{\omega}), \quad \textit{such that } \dim(X_i) = i,\ \mathbf{0}_i \notin X_i \tag{B.12}$$

and

$$\lambda_i^{\boldsymbol{\Omega}} = \max_{X_i} \min_{\boldsymbol{\omega} \in X_i} \mathscr{R}(\boldsymbol{\Omega}, \boldsymbol{\omega}), \quad \textit{such that } \dim(X_i) = N - i + 1,\ \mathbf{0}_{N-i+1} \notin X_i. \tag{B.13}$$

In particular, it holds that

$$\lambda_0^{\boldsymbol{\Omega}} \leq \mathscr{R}(\boldsymbol{\Omega}, \boldsymbol{\omega}) \leq \lambda_{N-1}^{\boldsymbol{\Omega}}, \quad \forall \boldsymbol{\omega} \in \mathbb{C}^N \setminus \{\mathbf{0}_N\}, \tag{B.14}$$

where bounds of (B.14) are attained when $\boldsymbol{\omega}$ is an eigenvector of the correspondent marginal eigenvalues $\lambda_0^{\boldsymbol{\Omega}}$ and $\lambda_{N-1}^{\boldsymbol{\Omega}}$.

Definition B.2.19 (Nonnegative vector)**.**
A vector $\boldsymbol{\omega} \in \mathbb{R}^N$ is *nonnegative* if all its components $[\boldsymbol{\omega}]_i$ are nonnegative. This property is denoted with $\boldsymbol{\omega} \geq 0$.

Definition B.2.20 (Positive vector)**.**
A vector $\boldsymbol{\omega} \in \mathbb{R}^N$ is *positive* if all its components $[\boldsymbol{\omega}]_i$ are strictly positive. This property is denoted with $\boldsymbol{\omega} > 0$.

Definition B.2.21 (Nonnegative matrix)**.**
A matrix $\boldsymbol{\Omega} \in \mathbb{R}^{N\times N}$ is *nonnegative* if all its entries $[\boldsymbol{\Omega}]_{ij}$ are nonnegative. This property is denoted with $\boldsymbol{\Omega} \geq 0$.

Definition B.2.22 (Positive matrix)**.**
A matrix $\boldsymbol{\Omega} \in \mathbb{R}^{N\times N}$ is *positive* if all its entries $[\boldsymbol{\Omega}]_{ij}$ are strictly positive. This property is denoted with $\boldsymbol{\Omega} > 0$.

Definition B.2.23 (Primitive matrix)**.**
A matrix $\boldsymbol{\Omega} \in \mathbb{R}^{N\times N}$ is *primitive* if there exists a positive integer κ such that $\boldsymbol{\Omega}^\kappa > 0$.

Definition B.2.24 (Irreducible matrix)**.**
A matrix $\mathbf{\Omega} \in \mathbb{R}^{N\times N}$ is *irreducible* if in correspondence to each couple of indexes (i,j) there exists a positive integer $\kappa(i,j)$ such that $[\mathbf{\Omega}^{\kappa(i,j)}]_{ij} > 0$.

Proposition B.2.25 (Characterization of irreducible matrices)**.**
A matrix $\mathbf{\Omega} \in \mathbb{R}^{N\times N}$ is irreducible if $\sum_{i=1}^{N} \mathbf{\Omega}^i > 0$.

Definition B.2.26 (Reducible matrix)**.**
A matrix $\mathbf{\Omega} \in \mathbb{R}^{N\times N}$ is *reducible* if there exists a couple of indexes (i,j) such that, for all $\kappa \in \mathbb{N} \setminus \{0\}$, it holds that $[\mathbf{\Omega}^{\kappa}]_{ij} = 0$.

Lemma B.2.27 (Classification of nonnegative matrices)**.**
Let $\mathbf{\Omega} \in \mathbb{R}^{N\times N}$ be a squared matrix. It holds that,

(i) if $\mathbf{\Omega}$ is positive then it is also primitive;

(ii) if $\mathbf{\Omega}$ is primitive then it is also irreducible;

(iii) if $\mathbf{\Omega}$ is irreducible then it is also nonnegative.

Moreover, the set of reducible matrices in union with the set of irreducible matrices complete the set of nonnegative matrices.

Theorem B.2.28 (Perron-Frobenius Theorem)**.**
Let $\mathbf{\Omega} \in \mathbb{R}^{N\times N}$ be a squared matrix with dimension $N \geq 2$. If $\mathbf{\Omega}$ is nonnegative then

(i) there exists a real eigenvalue $\lambda_0^{\mathbf{\Omega}}$ of $\mathbf{\Omega}$ such that $\lambda_0^{\mathbf{\Omega}} \geq |\lambda_i^{\mathbf{\Omega}}| \geq 0$ holds, for all the other eigenvalues $\lambda_i^{\mathbf{\Omega}}$ of $\mathbf{\Omega}$ indexed by $i = 1, \ldots, N-1$;

(ii) the right and left eigenvectors associated to $\lambda_0^{\mathbf{\Omega}}$ can be selected nonnegative.

If additionally $\mathbf{\Omega}$ is irreducible, then

(iii) the eigenvalue $\lambda_0^{\mathbf{\Omega}}$ is strictly positive and simple[3], and the other eigenvalues with modulus equal to $\lambda_0^{\mathbf{\Omega}}$ are the complex numbers given by

$$\lambda_0^{\mathbf{\Omega}} \exp(\mathbf{i}2\pi k/\mu_I), \quad k = 0, \ldots, \mu_I - 1, \tag{B.15}$$

where the integer $\mu_I > 1$ is called index of imprimitivity*;*

(iv) the right and left eigenvectors associated to $\lambda_0^{\mathbf{\Omega}}$ are unique and positive, up to rescaling.

If additionally $\mathbf{\Omega}$ is primitive, then

(v) the eigenvalue $\lambda_0^{\mathbf{\Omega}}$ satisfies $\lambda_0^{\mathbf{\Omega}} > |\lambda_i^{\mathbf{\Omega}}| \geq 0$, for $i = 1, \ldots, N-1$, and the index of imprimitivity μ_I is unitary.

[3]For this reason, $\lambda_0^{\mathbf{\Omega}}$ is also called *leading eigenvalue*, in the Perron-Frobenius Theory. The leading eigenvalue trivially coincides with the value assumed by spectral radius $\rho_S(\mathbf{\Omega})$.

Definition B.2.29 (Essential spectral radius)**.**
Let $\mathbf{\Omega} \in \mathbb{R}^{N\times N}$ be a nonnegative irreducible matrix and $\lambda_0^{\mathbf{\Omega}} = \rho_S(\mathbf{\Omega})$ be the corresponding leading eigenvalue. In addition, let us define $\Lambda_0(\mathbf{\Omega}) = \Lambda(\mathbf{\Omega}) \setminus \{\lambda_i^{\mathbf{\Omega}} : |\lambda_i^{\mathbf{\Omega}}| = \lambda_0^{\mathbf{\Omega}}\}$. The *essential spectral radius* $\lambda^{\mathbf{\Omega}}$ of $\mathbf{\Omega}$ is defined as the map $\mathrm{esr}(\mathbf{\Omega}) : \mathbb{R}^{N\times N} \to \mathbb{R} : \mathbf{\Omega} \mapsto \lambda^{\mathbf{\Omega}}$ such that

$$\lambda^{\mathbf{\Omega}} = \max_{\lambda\in\Lambda_0(\mathbf{\Omega})} |\lambda|. \tag{B.16}$$

The eigenvalue that maximizes relation (B.16) is denoted with $\lambda_!^{\mathbf{\Omega}} \in \Lambda_0(\mathbf{\Omega})$.

Definition B.2.30 (Stochastic matrix)**.**
Let $\mathbf{\Omega} \in \mathbb{R}^N$ a squared matrix and denote it entries as $[\mathbf{\Omega}]_{ij} = \omega_{ij}$. If the relation

$$\omega_{ij} \geq 0, \qquad \sum_{j=1}^{N} \omega_{ij} = 1, \qquad \text{for } i = 1, \dots, N \tag{B.17}$$

is satisfied then $\mathbf{\Omega}$ is said to be a *(row)-stochastic* matrix. This property is indicated with $\mathbf{\Omega} \in \mathrm{stoc}(\mathbb{R}^{N\times N})$.

Definition B.2.31 (Doubly stochastic matrix)**.**
Let $\mathbf{\Omega} \in \mathbb{R}^{N\times N}$ a squared matrix and denote it entries as $[\mathbf{\Omega}]_{ij} = \omega_{ij}$. If relation (B.17) is satisfied by both $\mathbf{\Omega}$ and $\mathbf{\Omega}^\top$, namely $\mathbf{\Omega}$ is both row- and column-stochastic, then $\mathbf{\Omega}$ is said to be *doubly stochastic*. This property is indicated with $\mathbf{\Omega} \in \mathrm{stoc}^2(\mathbb{R}^{N\times N})$.

Theorem B.2.32 (Characterization of stochastic matrices)**.**
If $\mathbf{\Omega}$ is a stochastic matrix then the spectral radius of $\mathbf{\Omega}$ is equal to the unity, namely $\rho_S(\mathbf{\Omega}) = 1$, and its spectrum $\Lambda(\mathbf{\Omega})$ is a subset of the unit disc $\{\omega \in \mathbb{C} : |\omega| \leq 1\}$. On the other hand, a nonnegative matrix $\mathbf{\Omega} \in \mathbb{R}^{N\times N}$ is stochastic if and only if $\mathbb{1}_N$ is an eigenvector of $\mathbf{\Omega}$ corresponding to the eigenvalue 1, i.e.

$$\mathbf{\Omega}\mathbb{1}_N = \mathbb{1}_N. \tag{B.18}$$

Furthermore, a nonnegative matrix $\mathbf{\Omega} \in \mathbb{R}^{N\times N}$ is doubly stochastic if and only if (B.18) *holds together with*

$$\mathbb{1}_N^\top \mathbf{\Omega} = \mathbb{1}_N^\top, \tag{B.19}$$

namely $\mathbb{1}_N$ is also a left eigenvector.

Definition B.2.33 (Convergent and semi-convergent matrix)**.**
Let $\mathbf{\Omega} \in \mathbb{C}^{N\times N}$ be a squared matrix and consider the limit

$$\mathbf{\Omega}_\infty = \lim_{t\to+\infty} \mathbf{\Omega}^t. \tag{B.20}$$

Then matrix $\mathbf{\Omega}$ is said *semi-convergent* if $\mathbf{\Omega}_\infty$ exists finite. Moreover, denoting with $\mathbf{Z}_N$ the N-dimensional squared matrix whose entries are all zeros, i.e. $[\mathbf{Z}_N]_{ij} = 0$ for all $i, j = 1, \dots, N-1$, if $\mathbf{\Omega}_\infty = \mathbf{Z}_N$, then matrix $\mathbf{\Omega}$ is said *convergent*.

Lemma B.2.34 (Relation between spectral radius and convergence)**.**
For a squared matrix $\mathbf{\Omega} \in \mathbb{C}^{N\times N}$, the following statements hold:

(i) $\mathbf{\Omega}$ is convergent if and only if $\rho_S(\mathbf{\Omega}) < 1$;

(ii) $\boldsymbol{\Omega}$ *is semi-convergent if and only if* $\rho_S(\boldsymbol{\Omega}) \leq 1$*: either no eigenvalue has unit modulus or if* $\lambda_0^{\boldsymbol{\Omega}}$ *is an eigenvalue of* $\boldsymbol{\Omega}$ *such that* $|\lambda_0^{\boldsymbol{\Omega}}| = 1$*, then it is semi-simple.*

B.3 Elements of Graph Theory

In this third section of the appendix, several elements of Graph Theory, along with their notation, are introduced in order to provide support to the most important aspect characterizing the multi-agent system modeling throughout the overall chapters. All the concepts can be found in Chung (1997); Goodrich and Tamassia (2011); Lee et al. (2014); Marsden (2013); Mesbahi and Egerstedt (2010). A list of the most fundamental definitions, assumptions and facts for Graph Theory concepts involved in this thesis follows.

B.3.1 Basic notions of Graph Theory and hypotheses

Definition B.3.1 (Graph)**.**
A *graph* is a way of representing relationships that exist between pairs of objects. That is, a graph is a set of objects, called vertices, together with a collection of pairwise connections between them. Abstractly, a *graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is simply a set $\mathcal{V} = \{v_1, \dots, v_n\}$ of n vertices (or nodes) and a collection $\mathcal{E} = \{e_1, \dots, e_E\} \subseteq \mathcal{V} \times \mathcal{V}$ of E pairs of vertices from $\mathcal{V}$, called edges (or arcs).

Definition B.3.2 (Simple graph)**.**
A *graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is said to be *simple* if each edge $e_k \in \mathcal{E}$, $k = 1, \dots, E$, is associated to one and only one pair of vertices $e_{ij} = (v_i, v_j)$. In other words, if $e_k = e_{ij}$ belongs to $\mathcal{E}$ then it is unique.

Assumption B.3.3.
Henceforward, every graph is assumed to be a simple graph with a finite number n of vertices.

Definition B.3.4 (Edge orientation)**.**
Whenever a pair of vertices $e_{ij} = (v_i, v_j) \in \mathcal{V} \times \mathcal{V}$ is ordered, with v_i preceding v_j, each edge e_{ij} is oriented such that vertex v_i is called *origin* and vertex v_j is called *destination*.

Definition B.3.5 (Undirected graph)**.**
A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is said *undirected*[4] if each edge $e_k \in \mathcal{E}$, $k = 1, \dots, E$, does not possess an orientation, namely $e_{ij} = e_{ji}$, for all $i, j = 1, \dots, n$.

Definition B.3.6 (Directed graph)**.**
A graph $\mathcal{G}^d = (\mathcal{V}, \mathcal{E}^d)$ is said *directed* (or a *digraph*) if each edge $e_k \in \mathcal{E}^d$, $k = 1, \dots, E$, possesses an orientation.

Definition B.3.7 (Oriented graph)**.**
Any directed graph $\mathcal{G}^o = (\mathcal{V}, \mathcal{E}^o)$ obtained by assigning an arbitrary orientation to each edge belonging to an undirected graph $\mathcal{G}$ is said to be its *oriented* version (see Fig. B.4).

[4]With abuse of notation, both generic and undirected graphs are indicated by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$.

Fact B.3.8 (Graph representation)**.**
A graph $\mathcal{G}$ can be represented on a two-dimensional or three-dimensional space by means of all its vertices and edges. As shown in Fig. B.4, a vertex can be depicted as a circle, a dot or a sphere; an oriented edge can be depicted with an arrow going from its origin to its destination; a non-oriented edge as well as a doubly oriented edge can be depicted as a line or a double arrow.

(a) Undirected graph

(b) Oriented digraph

Figure B.4. The digraph $\mathcal{G}^o$ is obtained orienting the undirected graph $\mathcal{G}$.

Definition B.3.9 (Self loop on a vertex)**.**
For any graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, if edge e_{ii} belongs to $\mathcal{E}$ then e_{ii} is defined as the *self loop* related to vertex v_i.

Definition B.3.10 (Neighborhood of a vertex)**.**
Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be any graph. Given a vertex $v_i \in \mathcal{V}$, its *neighborhood* is defined as the set $\mathcal{N}_i = \{v_j | e_{ij} \in \mathcal{E}\}$. Each vertex $v_j \in \mathcal{N}_i$ is said to be a *neighbor* of v_i.

Definition B.3.11 (Adjacent vertices)**.**
Two vertices v_i and v_j are said to *adjacent* if there exist an edge e_{ij} connecting them, i.e. if v_i and v_j are neighbors.

Definition B.3.12 (Degree of a vertex)**.**
For an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, each vertex $v_i \in \mathcal{V}$ is characterized by the natural quantity $\deg(v_i)$ called *degree* and given by the cardinality of the corresponding neighborhood $\mathcal{N}_i$, such that $\deg(v_i) = |\mathcal{N}_i|$. In other words, the degree of the i-th vertex $\deg(v_i)$ is equivalent to the number of neighbors of vertex v_i. Furthermore, the minimum and maximum degrees in $\mathcal{G}$ are denoted by $d_m = d_m(\mathcal{G})$ and $d_M = d_M(\mathcal{G})$, respectively.

Definition B.3.13 (Volume of an undirected graph)**.**
The *volume* of an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is defined as $\mathrm{vol}(\mathcal{G}) = \sum_{i=1}^{n} \deg(v_i)$.

Lemma B.3.14 (Handshaking lemma)**.**
If $\mathcal{G}$ is an undirected graph with E edges and without self loops, then it holds that

$$\mathrm{vol}(\mathcal{G}) = 2E \tag{B.21}$$

Definition B.3.15 (Weighted graph)**.**
Let $\mathcal{W} \in \{\mathcal{W}_v, \mathcal{W}_e\}$ be the set of weights, consisting of n vertex weights $\mathcal{W}_v \subset \mathbb{C}$ and E edge weights $\mathcal{W}_e \subset \mathbb{C}$. A *weighted graph* $\mathcal{G}^w = (\mathcal{V}, \mathcal{E}, \mathcal{W})$ is, generally, a graph in

which each vertex $v_i \in \mathcal{V}$ is weighted by $w_{v_i} \in \mathcal{W}_v$ (*vertex-weighted* graph $\mathcal{G}^w = \mathcal{G}^v$), $i = 1, \ldots, n$ or each edge $e_k \in \mathcal{E}$ is weighted by $w_{e_k} \in \mathcal{W}_e$, $k = 1, \ldots, E$ (*edge-weighted* graph $\mathcal{G}^w = \mathcal{G}^e$).

Assumption B.3.16.
Henceforward, weights of weighted graphs are assumed to be real and non-negative. Therefore, by convention, $|v_i| = w_{v_i}$ and $|e_k| = w_{e_k}$ are set for all $i = 1, \ldots, n$, for all $w_{e_k} \in \mathcal{W}_e$. Moreover, whenever sets $\mathcal{W}_v$ and $\mathcal{W}_e$ are not specified, their weights are equal to the unit.

Definition B.3.17 (Weighted quantities for undirected graphs)**.**
For an undirected weighted graph $\mathcal{G}^w = (\mathcal{V}, \mathcal{E}, \mathcal{W})$, the following definitions are considered:

- vertex-degree $\deg_v(v_i) = |v_i|\deg(v_i)$;
- edge-degree $\deg_e(v_i) = \sum_{j=1}^{\deg(v_i)} |e_{ij}|$;
- weighted-degree $\deg_w(v_i) = \deg_v(v_i)$ or $\deg_w(v_i) = \deg_e(v_i)$, specified by the context;
- weighted volume $\mathrm{vol}_w(\mathcal{G}^w) = \sum_{i=1}^{n} \deg_w(v_i)$.

Definition B.3.18 (Subgraph)**.**
A *subgraph* $\mathcal{G}_S = (\mathcal{V}_S, \mathcal{E}_S) \subseteq (\mathcal{V}, \mathcal{E})$ is another graph of $n_S = |\mathcal{V}_S|$ vertices formed by a subset of the vertices and edges of a given graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$.

Definition B.3.19 (Subgraph complement)**.**
The *complement* $\mathcal{G}_{\overline{S}} = (\mathcal{V}_{\overline{S}}, \mathcal{E}_{\overline{S}})$ of a given subgraph $\mathcal{G}_S \subseteq \mathcal{G}$ is defined such that $\mathcal{G}_S \cup \mathcal{G}_{\overline{S}} = \mathcal{G}$ and $\mathcal{G}_S \cap \mathcal{G}_{\overline{S}} = \varnothing$ hold. As a consequence, the number of vertices $n_{\overline{S}} = |\mathcal{V}_{\overline{S}}|$ in $\mathcal{G}_{\overline{S}}$ is such that $n_S + n_{\overline{S}} = n$.

Definition B.3.20 (Cut of a graph)**.**
Given a subgraph $\mathcal{G}_S$ and its complement $\mathcal{G}_{\overline{S}}$, the cut of graph $\mathcal{G}$ is defined as the set $\partial\mathcal{E}_S = \partial\mathcal{E}_{\overline{S}} = \{(v_i, v_j) \mid v_i \in \mathcal{V}_S \ \ \& \ \ v_j \in \mathcal{V}_{\overline{S}}\} \subseteq \mathcal{E}$.

Definition B.3.21 (Cut ratio)**.**
The *cut ratio* of a vertex-weighted graph $\mathcal{G}^v$ partitioned into two subgraphs $\mathcal{G}_S$ and $\mathcal{G}_{\overline{S}}$ is defined as $h_{\mathcal{G}}^{Ch}(\mathcal{G}_S) = |\partial\mathcal{E}_S| / \min(\sum_{v_i \in \mathcal{V}_S} |v_i|, \sum_{v_i \in \mathcal{V}_{\overline{S}}} |v_i|)$.

Definition B.3.22 (Conductance of a cut)**.**
The *conductance* of a vertex-weighted graph $\mathcal{G}^v$ partitioned into two subgraphs $\mathcal{G}_S$ and $\mathcal{G}_{\overline{S}}$ is defined as $h_{\mathcal{G}}^{Con}(\mathcal{G}_S) = |\partial\mathcal{E}_S| / \min(\mathrm{vol}_v(\mathcal{G}_S), \mathrm{vol}_v(\mathcal{G}_{\overline{S}}))$.

Definition B.3.23 (Bipartition of a cut)**.**
The *bipartition* for a vertex-weighted graph $\mathcal{G}^v$ partitioned into two subgraphs $\mathcal{G}_S$ and $\mathcal{G}_{\overline{S}}$ is defined as $h_{\mathcal{G}}(\mathcal{G}_S) = |\partial\mathcal{E}_S| / \mathrm{vol}_v(\mathcal{G}_S) + |\partial\mathcal{E}_{\overline{S}}| / \mathrm{vol}_v(\mathcal{G}_{\overline{S}})$.

Definition B.3.24 (Cheeger isoperimetric constant of a graph)**.**
The *Cheeger isoperimetric constant* of a graph $\mathcal{G}^v$ is defined as $\mathfrak{h}_{\mathcal{G}}^{Ch} = \min_{\mathcal{V}_S \subseteq \mathcal{V}} h_{\mathcal{G}}^{Ch}(\mathcal{G}_S)$.

Definition B.3.25 (Conductance of a graph)**.**
The *conductance* of a graph $\mathcal{G}^v$ is defined as $\mathfrak{h}_{\mathcal{G}}^{Con} = \min_{\mathcal{V}_S \subseteq \mathcal{V}} h_{\mathcal{G}}^{Con}(\mathcal{G}_S)$.

Definition B.3.26 (Bipartition constant of a graph)**.**
The *bipartition constant* of a graph $\mathcal{G}^v$ is defined as $\mathfrak{h}_{\mathcal{G}} = \min_{\mathcal{V}_S \subseteq \mathcal{V}} h_{\mathcal{G}}(\mathcal{G}_S)$.

Definition B.3.27 (Adjacent edges)**.**
Two edges (e_{ik}, e_{kj}) are said to be *adjacent* if they share a common vertex v_k, i.e. v_k is adjacent to both v_i and v_j.

Definition B.3.28 (Walk, vertex sequence of a walk)**.**
Given an integer[5] L, a *walk* $\bar{\pi}_{k_0 k_L}$ from vertex v_{k_0} to vertex v_{k_L} is defined as the ordered sequence $\bar{\pi}_{k_0 k_L} = \{e_{k_0 k_1}, e_{k_1 k_2}, \ldots, e_{k_{L-2} k_{L-1}}, e_{k_{L-1} k_L}\}$, in which two consecutive elements are adjacent edges. The ordered sequence $\{v_{k_i}\}_{i=0}^{L}$ is called *vertex sequence of the walk*. For $L = 0$, the walk $\bar{\pi}_{k_0 k_0} = \varnothing$ is degenerate and v_0 is the only vertex belonging to the associated vertex sequence.

Definition B.3.29 (Path)**.**
A *path* π_{ij} from vertex v_i to vertex $v_j \neq v_i$ is defined as walk from v_i to v_j in which all the vertices belonging to the associated vertex sequence are distinct.

Definition B.3.30 (Length (of a walk or a path))**.**
The *length* $\ell(\cdot)$ of a walk or a path is a function that maps a walk or a path into the number of edges L belonging to the edge sequence that define them. The weighted version $\ell_W(\cdot)$ of the length is defined as the summation of the edge weights for each element in a walk or a path.

Definition B.3.31 (Shortest path)**.**
A *shortest path* between vertices v_i and v_j is a path $\pi_{ij}^{\star}$ such that its length $\ell(\pi_{ij}^{\star})$ is minimum among all the possible paths between the ending nodes v_i and v_j.

Definition B.3.32 (Distance)**.**
The *distance* $\mathrm{dist}(v_i, v_j)$ between any two vertices (v_i, v_j) in a graph is the length $\ell(\pi_{ij}^{\star})$ of any shortest path $\pi_{ij}^{\star}$ having the two vertices (v_i, v_j) as its endpoints.

Definition B.3.33 (Diameter)**.**
The *diameter* of a connected graph $\mathcal{G}$ is the maximum distance of a shortest path and is given by

$$\phi(\mathcal{G}) = \max_{v_j \in \mathcal{V}} \max_{v_i \in \mathcal{V}} \ell(\pi_{ij}^{\star}). \tag{B.22}$$

Definition B.3.34 (Radius)**.**
The *radius* $\mathrm{r}(\mathcal{G})$ of a connected graph $\mathcal{G}$ is the minimum among all the maximum distances between a vertex v_i to all other vertices, that is

$$\mathfrak{r}(\mathcal{G}) = \min_{v_j \in \mathcal{V}} \max_{v_i \in \mathcal{V}} \ell(\pi_{ij}^{\star}). \tag{B.23}$$

Definition B.3.35 (Connected and strongly connected graph)**.**
A *connected* graph $\mathcal{G}$ is one in which each pair of distinct vertices forms the endpoints of a path. Higher forms of connectivity include *strong connectivity* in directed graphs: for each two vertices there are paths from one to the other in both directions.

[5] L may be infinite.

Definition B.3.36 (Connected component of a graph)**.**
A *connected component* of a graph $\mathcal{G}$ is any maximal connected subgraph belonging to $\mathcal{G}$.

Definition B.3.37 (Circuit)**.**
A *circuit* is a walk in which its first and last vertices belonging to the associated vertex sequence coincide.

Definition B.3.38 (Cycle)**.**
A *cycle* is a circuit in which the only repeated vertices in the associated vertex sequence are the first and the last vertices.

Definition B.3.39 (Girth)**.**
The *girth* of a graph $\mathcal{G}$ is the length g of a shortest cycle contained in $\mathcal{G}$.

Definition B.3.40 (Acyclic graph)**.**
A graph $\mathcal{G}$ is said *acyclic* if it does not have any cycle.

Definition B.3.41 (Tree and root)**.**
A *tree* is an undirected graph that is both connected and acyclic, or a directed graph in which there exists a unique path from one vertex (the *root* of the tree) to all remaining vertices.

Definition B.3.42 (Spanning subgraph)**.**
A subgraph $\mathcal{G}_S \subseteq \mathcal{G}$ is spanning when it includes all of the vertices of the given graph $\mathcal{G}$.

Definition B.3.43 (Spanning tree)**.**
A *spanning tree* is a spanning subgraph that is also a tree.

B.3.2 Algebraic entities and properties related to graphs

Definition B.3.44 (Adjacency matrix)**.**
The *adjacency matrix* $\mathbf{A} \in \mathbb{R}^{n\times n}$ of a graph $\mathcal{G}$ with n vertices is defined as $[\mathbf{A}]_{ij} = 1$ if $e_{ij} \in \mathcal{E}$ (i.e., v_i and v_j are neighbors); $[\mathbf{A}]_{ij} = 0$ otherwise.

Definition B.3.45 (Degree matrix)**.**
The *degree matrix* $\mathbf{D} \in \mathbb{R}^{n\times n}$ of a graph $\mathcal{G}$ with n vertices is defined as $[\mathbf{D}]_{ii} = \deg(v_i)$; $[\mathbf{D}]_{ij} = 0$ for all $i \neq j$.

Lemma B.3.46 (Number of paths between two vertices)**.**
Let $\mathcal{G}$ be any graph with adjacency matrix $\mathbf{A} = (\mathcal{V}, \mathcal{E})$. The number of paths from vertex $v_i \in \mathcal{V}$ to vertex $v_j \in \mathcal{V}$ of length $\kappa \in \mathbb{N}$ is $[\mathbf{A}^\kappa]_{ij}$, if $i \neq j$. Otherwise, $[\mathbf{A}^\kappa]_{ii}$ provides the number of fundamental cycles of length κ passing through v_i.

Definition B.3.47 (Laplacian matrix)**.**
The *adjacency matrix* $\mathbf{L} \in \mathbb{R}^{n\times n}$ of a graph $\mathcal{G}$ with n vertices is defined as $\mathbf{L} = \mathbf{D} - \mathbf{A}$.

Definition B.3.48 (Incidence matrix)**.**
The *incidence matrix* $\mathbf{E} \in \mathbb{R}^{n\times E}$ of a graph $\mathcal{G}$ with n vertices and E edges is defined as $[\mathbf{E}]_{ij} = -1$ if the j-th edge sinks at node i; $[\mathbf{E}]_{ij} = 1$ if the j-th edge leaves node i; $[\mathbf{E}]_{ij} = 0$ otherwise.

Proposition B.3.49 (Relation between Laplacian and incidence matrices)**.**
Let $\mathbf{L}$ *be the Laplacian matrix associated to an undirected graph* $\mathcal{G}$*. Then* $\mathbf{L}$ *is symmetric, positive semi-definite and it can equivalently defined as* $\mathbf{L} = \mathbf{E}\mathbf{E}^\top$*, considering the incidence matrix* $\mathbf{E}$ *of the oriented graph* $\mathcal{G}^o$*, in case* $\mathcal{G}$ *be undirected.*

Theorem B.3.50 (First eigenvalue and eigenvector of the Laplacian Matrix)**.**
Let $\mathbb{1}_n \in \mathbb{R}^n$ *be the vector whose component are all equal to* 1*. A Laplacian* $\mathbf{L} \in \mathbb{R}^{n\times n}$ *associated to any graph* $\mathcal{G}$ *satisfies the following equality*

$$\mathbf{L}\mathbb{1}_n = 0\mathbb{1}_n. \tag{B.24}$$

In other words, the $\lambda_0^{\mathbf{L}} = 0$ *is the first eigenvalue of* $\mathbf{L}$ *and* $\boldsymbol{\varpi}_0^{\mathbf{L}} = \mathbb{1}_n$ *is its correspondent eigenvector.*

Theorem B.3.51 (Number of connected components of a graph)**.**
Let $0 = \lambda_0^{\mathbf{L}} \leq \ldots \leq \lambda_{n-1}^{\mathbf{L}}$ *be the eigenvalues of a Laplacian matrix* $\mathbf{L}$ *associated to an undirected graph* $\mathcal{G}$*. Then* 0 *is an eigenvalue of* $\mathbf{L}$ *and its multiplicity* $c_{\mathcal{G}} > 0$ *corresponds with the number of connected components in* $\mathcal{G}$*, namely*

$$\lambda_i^{\mathbf{L}} = 0, \quad \textit{for } i = 1, \ldots, c_{\mathcal{G}}. \tag{B.25}$$

Definition B.3.52 (Algebraic connectivity)**.**
For undirected graph $\mathcal{G}$, the *algebraic connectivity* is defined as the second eigenvalue $\lambda_1^{\mathbf{L}}$ of the Laplacian matrix $\mathbf{L}$ associated to $\mathcal{G}$.

Theorem B.3.53 (Connectedness of a graph)**.**
Let $\mathbf{L}$ *be the Laplacian matrix of an undirected graph* $\mathcal{G}$*. Then* $\mathcal{G}$ *is connected if and only if* $\lambda_1^{\mathbf{L}} > 0$*. This quantity provides a measure of the connectedness of* $\mathcal{G}$*, being* 0 *if and only if* $\mathcal{G}$ *is not connected.*

Definition B.3.54 (Normalized Laplacian matrix)**.**
The *normalized Laplacian matrix* $\boldsymbol{\mathcal{L}} \in \mathbb{R}^{n\times n}$ of a graph $\mathcal{G}$ is defined as $\boldsymbol{\mathcal{L}} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}$.

Definition B.3.55 (Randić matrix)**.**
The *Randić matrix* $\mathscr{R} \in \mathbb{R}^{n\times n}$ of a graph $\mathcal{G}$ is defined as $\mathscr{R} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$.

Theorem B.3.56 (Cheeger's inequalities)**.**
Let $\mathcal{G}$ *be an undirected graph. Two Cheeger's inequalities can be derived using the concepts provided so far:*

$$(\mathfrak{h}_{\mathcal{G}}^{Ch})^2/(2d_M(\mathcal{G})) \leq \lambda_1^{\mathbf{L}} \leq 2\mathfrak{h}_{\mathcal{G}}^{Ch} \tag{B.26}$$

$$\mathfrak{h}_{\mathcal{G}}^2/8 \leq (\mathfrak{h}_{\mathcal{G}}^{Con})^2/2 \leq \lambda_1^{\boldsymbol{\mathcal{L}}} \leq 2\mathfrak{h}_{\mathcal{G}}^{Con} \leq 2\mathfrak{h}_{\mathcal{G}} \tag{B.27}$$

Similar inequalities hold whenever $\mathcal{G}$ *is vertex-weighted just considering a weighted version of the Laplacian and normalized Laplacian matrices (i.e., by defining degree matrices in which the diagonal entries are the weights for vertices).*

Theorem B.3.57 (Kirchhoff's matrix tree theorem)**.**
Let $\mathfrak{t}(\mathcal{G})$ *be the number of spanning tree in the connected and undirected graph* $\mathcal{G}$ *with* n

vertices and let $\mathbf{L}$ *be the corresponding Laplacian matrix of* $\mathcal{G}$*. Then one has*

$$\mathfrak{t}(\mathcal{G}) = \frac{1}{n}\prod_{i=1}^{n-1}\lambda_i^{\mathbf{L}}. \tag{B.28}$$

B.3.3 Fundamental classes of undirected graphs and their characterization

Definition B.3.58 (Path graph)**.**
A *path graph* is a graph $P_n = (\mathcal{V}, \mathcal{E})$ in which, up to a permutation of the indexes, the set of edges $\mathcal{E}$ coincides with path π_{ij} from vertex v_i to vertex v_j and the set of vertices $\mathcal{V}$ coincides with the vertex sequence associated to π_{ij}. Fig. B.5(a) shows a cycle graph.

Definition B.3.59 (Cycle graph)**.**
A *cycle graph* or circular graph is a graph that consists of a single cycle, or in other words, some number of vertices (at least 3) arranged in a path graph in which one additional edge is inserted to connect the two nodes with degree 1. The *cycle graph* with n vertices is called $C_n(1,1)$. Fig. B.5(b) shows a cycle graph.

Definition B.3.60 (Complete graph)**.**
A *complete graph* is a graph in which every pair of distinct vertices is connected by an edge. The *complete graph* with n vertices is called K_n. Fig. B.5(c) shows a complete graph.

Proposition B.3.61 (Maximum number of edges in an undirected graph)**.**
The complete graph K_n *has the maximum number* E *of edges among all the undirected graphs with* n *and it holds that* $E = n(n-1)/2$*.*

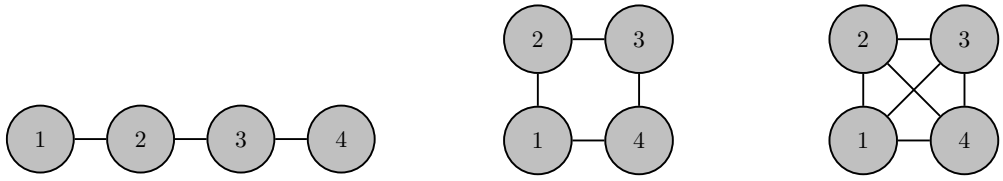
Definition B.3.62 (Density of an undirected graph)**.**
The *density* of an undirected graph $\mathcal{G}$ with n vertices is defined as the real quantity $\mathrm{dens}(\mathcal{G}) = \mathrm{vol}(\mathcal{G})/(n(n-1)) \in [0,1]$.

Definition B.3.63 (Dense and sparse undirected graphs)**.**
Let ε_D be a real number in the interval $[0,1)$. An undirected graph $\mathcal{G}$ is said to be *dense* if $\mathrm{dens}(\mathcal{G}) > \varepsilon_D$ or *sparse* if $\mathrm{dens}(\mathcal{G}) \leq \varepsilon_D$. If not specified by the context, it is generally assumed $\varepsilon_D = 0.5$.

Definition B.3.64 (Bipartite graph)**.**
A *bipartite graph* is a graph whose vertices can be divided into two disjoint and independent sets $\mathcal{V}_1$ and $\mathcal{V}_2$ such that every edge connects a vertex in $\mathcal{V}_1$ to one in $\mathcal{V}_2$. Fig. B.4(a) and Fig. B.5(b) show two bipartite graphs.

Definition B.3.65 (Regular graph)**.**
A d-*regular graph* is a graph in which all the vertices have the same common degree d, i.e. $\deg(v_i) = \mathrm{d} = \deg(v_j)$, for all $v_i, v_j \in \mathcal{V}$.

Lemma B.3.66 (Spectral properties of the normalized Laplacian matrix)**.**
For an undirected graph $\mathcal{G}$ *on* n *vertices, it holds that*



(a) Path graph P_4 (b) Cycle graph $C_n(1,1)$ (c) Complete graph K_4

Figure B.5. Different classes of graphs with $n = 4$ vertices. Setting $\varepsilon_D = 0.5$, (a) can be considered sparse and, on the contrary, (b) and (c) dense. (b) is 2-regular and bipartite. (c) is 3-regular and possesses the maximum number of edges $\max_{|\mathcal{V}|=4} E = 6$ for an undirected graph with 4 vertices.

(i)

$$\sum_{i=1}^{n} \lambda_i^{\mathcal{L}} \leq n \tag{B.29}$$

with equality holding if and only if $\mathcal{G}$ has no isolated vertices.

(ii) For $n \geq 2$,

$$\lambda_1^{\mathcal{L}} \leq \frac{n}{n-1} \tag{B.30}$$

with equality holding if and only if $\mathcal{G}$ is the complete graph on n vertices. Also, for a graph $\mathcal{G}$ without isolated vertices, one has

$$\lambda_{n-1}^{\mathcal{L}} \geq \frac{n}{n-1}. \tag{B.31}$$

(iii) For a graph which is not a complete graph, one has $\lambda_1^{\mathcal{L}} \leq 1$.

(iv) If $\mathcal{G}$ is connected, then $\lambda_1^{\mathcal{L}} > 0$. If $\lambda_i^{\mathcal{L}} = 0$ and $\lambda_{i+1}^{\mathcal{L}} \neq 0$, then $\mathcal{G}$ has exactly $i+1$ connected components.

(v) For all $i \leq n-1$, one has

$$\lambda_i^{\mathcal{L}} \leq 2 \tag{B.32}$$

with $\lambda_{n-1}^{\mathcal{L}} = 2$ if and only if a connected component of $\mathcal{G}$ is bipartite and non-trivial.

(vi) The spectrum of a graph is the union of the spectra of its connected components.

B.4 Elements of Rigidity Theory

In this fourth section of the appendix, some basic notions about Graph Rigidity are introduced to provide support to the online distributed control law devised for the formation tracking in Chap. 4. *Graph Rigidity* is the study of formation graphs for which the only permissible motions, while maintaining proper edge distances, are rigid motions. Graphs that satisfy these properties are promising candidates for specifying formations in that they describe the formation shape using only inter-agent distance specifications.

For a detailed treatise on this, the reader is addressed to Mesbahi and Egerstedt (2010); Oh et al. (2015); Zhao and Zelazo (2016).

Definition B.4.1 (Framework)**.**
A *framework* is classically defined as a couple $(\mathcal{G}, \mathbf{p})$ where

- $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is the formation graph[6] with n vertices and E edges;
- $\mathbf{p}$ is a collection of absolute positions, such that $\mathbf{p} = \begin{bmatrix} \mathbf{p}_1^\top & \cdots & \mathbf{p}_n^\top \end{bmatrix}^\top$, where the i-th position $\mathbf{p}_i$ belongs to the M-dimensional Euclidean space $\mathbb{R}^M$.

More generally, a framework can be composed of other spacial quantities, e.g. framework $(\mathcal{G}, \dot{\mathbf{p}})$ considers absolute velocities $\dot{\mathbf{p}} = \begin{bmatrix} \dot{\mathbf{p}}_1^\top & \cdots & \dot{\mathbf{p}}_n^\top \end{bmatrix}^\top$ instead of absolute positions.

Definition B.4.2 (Equivalent frameworks)**.**
Two frameworks $(\mathcal{G}, \mathbf{p})$ and $(\mathcal{G}, \mathbf{p}')$ are *equivalent* if $\|\mathbf{p}_i - \mathbf{p}_j\|_2 = \|\mathbf{p}'_i - \mathbf{p}'_j\|_2$ for all the edges $e_{ij} \in \mathcal{E}$.

Definition B.4.3 (Congruent frameworks)**.**
Two frameworks $(\mathcal{G}, \mathbf{p})$ and $(\mathcal{G}, \mathbf{p}')$ are *congruent* if $\|\mathbf{p}_i - \mathbf{p}_j\|_2 = \|\mathbf{p}'_i - \mathbf{p}'_j\|_2$ for all the couple of nodes $(v_i, v_j) \in \mathcal{V} \times \mathcal{V}$.

Definition B.4.4 (Globally rigid framework)**.**
A framework $(\mathcal{G}, \mathbf{p})$ is *globally rigid* if every framework that is equivalent to $(\mathcal{G}, \mathbf{p})$ is also congruent to $(\mathcal{G}, \mathbf{p})$.

Definition B.4.5 (Rigid framework)**.**
A framework $(\mathcal{G}, \mathbf{p})$ is *rigid* if there exists $\epsilon > 0$ such that every framework $(\mathcal{G}, \mathbf{p}')$ that is equivalent to $(\mathcal{G}, \mathbf{p})$ and satisfies $\|\mathbf{p}_i - \mathbf{p}'_i\|_2 < \epsilon$ for all $v_i \in \mathcal{V}$, is congruent to $(\mathcal{G}, \mathbf{p})$.

Proposition B.4.6 (Global rigidity and rigidity)**.**
Global rigidity implies rigidity, but the converse is not true.

Definition B.4.7 (Displacement vector, global displacement vector)**.**
The *displacement vector* between the i-th and j-th agent positions is defined as $\mathbf{e}_{ij} = \mathbf{p}_i - \mathbf{p}_j \in \mathbb{R}^M$. The *global displacement vector* $\mathbf{e}$ is defined as the collection of all displacement vectors, namely $\mathbf{e} = \begin{bmatrix} \mathbf{e}_1^\top & \cdots & \mathbf{e}_E^\top \end{bmatrix}^\top$.

Proposition B.4.8 (Relation between displacement and absolute measurements)**.**
The displacement vectors $\mathbf{e} = \begin{bmatrix} \mathbf{e}_1^\top & \cdots & \mathbf{e}_E^\top \end{bmatrix}^\top$ *are linked to absolute measurements through the following relation:*

$$\mathbf{e} = (\mathbf{E}^\top \otimes \mathbf{I}_M) \begin{bmatrix} \mathbf{p}_1^\top & \cdots & \mathbf{p}_n^\top \end{bmatrix}^\top = (\mathbf{E}^\top \otimes \mathbf{I}_M)\mathbf{p}. \tag{B.33}$$

Definition B.4.9 (Rigidity function)**.**
Let $\mathbf{E} \in \mathbb{R}^{N \times E}$ be the incidence matrix associated to the graph of a framework $(\mathcal{G}, \mathbf{p})$, such that $N = Mn$ can be seen as the degrees of freedom for the underlying framework.

[6] To the purposes of this thesis $\mathcal{G}$ is undirected but this assumption can easily be relaxed for a more general context.

With the same ordering for the edges of $\mathcal{E}$ assigned to the columns of $\mathbf{E}$, the *rigidity function* $r_{\mathcal{G}} : \mathbb{R}^N \to \mathbb{R}^E$ associated to the framework $(\mathcal{G}, \mathbf{p})$ is defined as

$$r_{\mathcal{G}}(\mathbf{p}) = \frac{1}{2}\begin{bmatrix} \cdots & \mathbf{e}_{ij}^\top \mathbf{e}_{ij} & \cdots \end{bmatrix}^\top, \quad \forall e_{ij} \in \mathcal{E}. \tag{B.34}$$

Definition B.4.10 (Rigidity matrix)**.**
One useful tool to characterize the rigidity property of a framework is the *rigidity matrix* $\boldsymbol{\mathcal{R}} \in \mathbb{R}^{E \times N}$, which is defined as the Jacobian of the rigidity function w.r.t. the spacial quantity considered in the relative framework. In formulas, the rigidity matrix for framework a $(\mathcal{G}, \mathbf{p})$ is yielded by

$$\boldsymbol{\mathcal{R}}(\mathbf{p}) = \frac{\partial r_{\mathcal{G}}(\mathbf{p})}{\partial \mathbf{p}}. \tag{B.35}$$

Definition B.4.11 (Diagonal edge operator)**.**
The *diagonal edge operator* $\mathrm{Diag}_{\mathcal{E}}$ is defined as the operator $\mathrm{Diag}_{\mathcal{E}} : \mathbb{R}^E \to \mathbb{R}^{E \times N} : \mathbf{e} \mapsto \mathrm{Diag}_{\mathcal{E}}\left\{\mathbf{e}_{ij}^\top\right\}$ that maps the set of displacement vectors $\mathbf{e}_{ij}^\top$ ordered as in (B.34) into a diagonal matrix of dimensions $E \times N$, such that its k-th diagonal element of dimensions $1 \times M$ is exactly given by the k-th displacement vector, for $k = 1, \ldots, E$.

Lemma B.4.12 (Decomposition of the rigidity matrix)**.**
The rigidity matrix can be factorized into the product

$$\boldsymbol{\mathcal{R}}(\mathbf{p}) = \mathrm{Diag}_{\mathcal{E}}\left\{\mathbf{e}_{ij}^\top\right\}\left(\mathbf{E}^\top \otimes \mathbf{I}_M\right) \tag{B.36}$$

in which the topological part described by the graph and the dimension of the space is decoupled from the displacement measurements.

Definition B.4.13 (Infinitesimally rigid framework)**.**
A framework $(\mathcal{G}, \mathbf{p})$ is *infinitesimally rigid* in a M-dimensional space with N degrees of freedom if

$$\mathrm{rank}(\boldsymbol{\mathcal{R}}(\mathbf{p})) = N - M(M+1)/2. \tag{B.37}$$

In other works, a framework is infinitesimally rigid if every infinitesimal motion is trivial (e.g. translations, rotations).

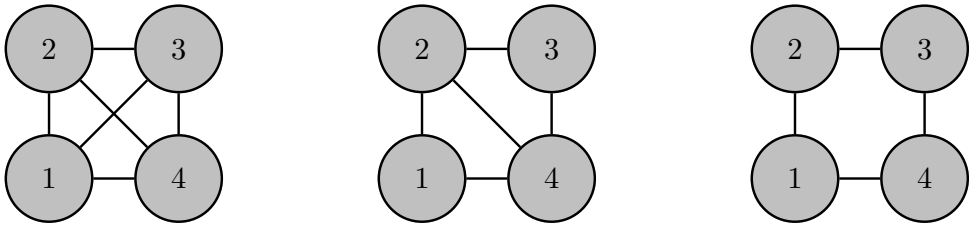
Proposition B.4.14 (Infinitesimal rigidity and rigidity)**.**
Infinitesimal rigidity implies rigidity, but the converse is not true.

Remark B.4.15.
Specifically, the framework $(\mathcal{G}, \mathbf{p})$ with n points is infinitesimally rigid in $\mathbb{R}^2$ (resp. $\mathbb{R}^3$) if and only if $\mathrm{rank}(\boldsymbol{\mathcal{R}}(\mathbf{p})) = 2n - 3$ (resp. $\mathrm{rank}(\boldsymbol{\mathcal{R}}(\mathbf{p})) = 3n - 6$). Obviously, in order to have at least $2n - 3$ (resp. $3n - 6$) edges in $\mathbb{R}^2$ (resp. $\mathbb{R}^3$). If the framework is infinitesimally rigid in $\mathbb{R}^2$ (resp. $\mathbb{R}^3$) and has exactly $2n - 3$ (resp. $3n - 6$) edges, then it is called a *minimally infinitesimally rigid* framework and it is defined as follows.

Definition B.4.16 (Minimally rigid framework)**.**
A *minimally rigid framework* is a rigid framework $(\mathcal{G}, \mathbf{p})$ such that the removal of any edge in $\mathcal{G}$ results in a nonrigid framework.

Lemma B.4.17 (Identification of minimally rigid frameworks)**.**
If the framework $(\mathcal{G}, \mathbf{p})$ *is minimally and infinitesimally rigid in the* M*-dimensional space, then the matrix* $\boldsymbol{\mathcal{R}}(\mathbf{p})\boldsymbol{\mathcal{R}}(\mathbf{p})^\top$ *is positive definite.*

Example B.4.18 (Categories of ridigity)**.**
Fig. B.6 illustrates some differences between the rigidity categories discussed so far.



(a) (Globally) rigid

(b) Minimally and infinitesimally rigid (not globally rigid)

(c) Not rigid

Figure B.6. Various rigidity aspects for different configurations of a graph with 4 nodes describing a squared shape in $\mathbb{R}^2$.

B.5 Elements of Calculus of Variations

In this fifth and last section of the appendix, some basic notions for the Calculus of Variations (CV) are introduced to classify how the online distributed control law in Chap. 4 has been devised and illustrate the deep roots in which the numerical optimization framework PRONTO is based on. For further details, see also Van Brunt (2004); Vinter (2010).

Definition B.5.1 (Functional)**.**
A *functional* is a law or application h that associates every element $\mathbf{x} \in X \subseteq \bar{X}$ to a real number.

Example B.5.2 (Mean of a function)**.**
Let $\bar{X}$ be the set of real valued functions, such that a function $x : \mathbb{R} \to \mathbb{R}$ is an element of $X \subset \bar{X}$. The average value $\bar{h}$ of a function x on the interval $[\omega_0, \omega_1] \subset \mathbb{R}$

$$\bar{h}(x) = \frac{1}{\omega_1 - \omega_0} \int_{\omega_0}^{\omega_1} x(\omega) d\omega \tag{B.38}$$

is a functional.

Definition B.5.3 (Variation of an element)**.**
The *variation* $\delta\mathbf{x}$ *of an element* $\mathbf{x} \in X \subseteq \bar{X}$ is defined as the collection of the variations for all its components $\delta\mathbf{x}_i$, such that $\delta\mathbf{x} = (\delta\mathbf{x}_1, \delta\mathbf{x}_2, \ldots)$ is arbitrary chosen in $\bar{X}$.

Remark B.5.4.
The variation $\delta\mathbf{x}_i$ of the i-th component is specified from the field of application, e.g. if $\bar{X} = \mathbb{R}^M$ then $\delta\mathbf{x}$ is an arbitrary vector in $\mathbb{R}^M$.

Definition B.5.5 (Gateaux differential and Gateaux differentiability)**.**
Let ϑ be a real number chosen in a neighborhood of the origin. The *Gateaux differential* $\delta h(\mathbf{x}, \delta\mathbf{x})$ of functional h w.r.t. $(\mathbf{x}, \delta\mathbf{x})$ is defined as

$$\delta h(\mathbf{x}, \delta\mathbf{x}) = \lim_{\vartheta \to 0} \frac{h(\mathbf{x} + \vartheta\delta\mathbf{x}) - h(\mathbf{x})}{\vartheta} = \frac{d}{d\vartheta} h(\mathbf{x} + \vartheta\delta\mathbf{x})\Big|_{\vartheta=0} \tag{B.39}$$

If the limit in (B.39) exists for all $\delta\mathbf{x} \in \bar{X}$, then h is said *Gateaux differetiable* at $\mathbf{x}$.

Remark B.5.6.
The Gateaux differential extends the concept of differential in a generalized domain $X \subset \bar{X}$ starting from the idea of directional derivative of a real function. This means that the Gateaux differential may not exist in some pathological cases similar to the situations where all the directional derivatives of a real function are well-defined but its differential does not exist.

Definition B.5.7 (Fréchet differentiability)**.**
Let $\bar{X}$ and Y be normed vector spaces such that $\mathbf{0}_{\bar{X}}$ is the identity element of $\bar{X}$, assuming that $\mathbf{x} \in X \subseteq \bar{X}$ is such that functional h is an application from X to Y. If there exists a bounded linear operator $\mathcal{Y} : \bar{X} \to Y$ satisfying

$$\frac{\|h(\mathbf{x} + \delta\mathbf{x}) - h(x) - \mathcal{Y}\delta\mathbf{x}\|_Y}{\|\delta\mathbf{x}\|_{\bar{X}}} = 0, \quad \text{as } \|\delta\mathbf{x}\|_{\bar{X}} \to \mathbf{0}_{\bar{X}} \tag{B.40}$$

then h is said *Fréchet differentiable* at $\mathbf{x}$.

Remark B.5.8.
The definition of Fréchet differentiability represents the starting point to extend the concept of differential for a real function. Indeed, the Fréchet differentiability transfers the gist of best linear approximation to the domain of functionals. However, the framework in which the Fréchet differentiation is defined is less general than that of Gateaux differentiation, since a metric and the existence of a bounded linear operator are required in the former.

Definition B.5.9 (Fréchet differential)**.**
If a functional h is *Fréchet differentiable* at $\mathbf{x}$ then the bounded linear operator $\mathcal{Y}$ defines the best linear approximation $\mathcal{Y}\delta\mathbf{x}$ for h, i.e. its *Fréchet differential*. This is also denoted with $\mathcal{Y}\delta\mathbf{x} = D\mathbf{x} \cdot \delta\mathbf{x}$.

Proposition B.5.10 (Relation between Gateaux and Fréchet differentiability)**.**
If h is Fréchet differentiable at $\mathbf{x}$, it is also Gateaux differentiable there, and $\delta h(\mathbf{x}, \delta\mathbf{x})$ is just the linear operator $D\mathbf{x}$ applied on $\delta\mathbf{x}$.

Definition B.5.11 (Variation of a functional)**.**
The *variation* $\delta h(\mathbf{x}, \delta\mathbf{x})$ of a functional h relative to the variation $\delta\mathbf{x}$ can be defined using either the Gateaux differential in B.39 or the Fréchet differential in Def. B.5.9. Also, $\delta h(\mathbf{x}, \delta\mathbf{x})$ is called Gateaux (or Fréchet) derivative of h.

Definition B.5.12 (Generalized integral action)**.**
Let $h_{2\kappa} : X(\omega_0, \omega_1) \to \mathbb{R}$ be a functional of the form

$$h_{2\kappa} = h\left(\mathbf{x}, \frac{d\mathbf{x}}{d\omega}, \ldots, \frac{d^\kappa \mathbf{x}}{d\omega^\kappa}\right) = \int_{\omega_0}^{\omega_1} l\left(\mathbf{x}(\omega), \frac{d\mathbf{x}(\omega)}{d\omega}, \ldots, \frac{d^\kappa \mathbf{x}(\omega)}{d\omega^\kappa}, \omega\right) d\omega \tag{B.41}$$

where $X(\omega_0, \omega_1)$ contains the real valued function $\mathbf{x} : \mathbb{R} \to \Omega \subseteq \underbrace{\mathbb{R}^M \times \cdots \times \mathbb{R}^M}_{n \text{ times}}$, and its derivatives up to the order $\kappa \geq 0$, and such that $l : \mathbb{R}^{(\kappa+1)nM+1} \to \mathbb{R}$ be a (regular) function in its variables. Let us also assume that values $\frac{d^j \mathbf{x}(\omega_0)}{d\omega^j}$ and $\frac{d^j \mathbf{x}(\omega_1)}{d\omega^j}$ are fixed and given for $j = 0, 1, \ldots, \kappa$. Functional $h_{2\kappa}$ in B.41 is called *generalized integral action* of order 2κ.

Remark B.5.13.
The second order integral action h_2 is a well-known functional studied in Mechanics, since it represents the Hamiltonian action, defined as the definite integral of the Lagrangian.

Proposition B.5.14 (Variation of the generalized integral action)**.**
The variation $\delta h_{2\kappa}$ of the generalized integral action $h_{2\kappa}$ is given by

$$\delta h_{2\kappa} = \sum_{j=0}^{\kappa} \int_{\omega_0}^{\omega_1} \sum_{i=1}^{n} (-1)^j \frac{d^j}{d\omega^j} \frac{\partial l(\mathbf{x}, \mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(\kappa)}, \omega)}{\partial \mathbf{x}_i^{(j)}} \cdot \delta \mathbf{x}_i^{(j)} d\omega. \tag{B.42}$$

Definition B.5.15 (Stationary functional)**.**
A functional h on $X(\boldsymbol{\omega}_0, \boldsymbol{\omega}_1)$ with fixed[7] extrema $\boldsymbol{\omega}_0, \boldsymbol{\omega}_1 \in \Omega$, is said to be *stationary* at $\bar{\mathbf{x}} \in X(\boldsymbol{\omega}_0, \boldsymbol{\omega}_1)$ if, for any variation $\delta \mathbf{x} \in \bar{X}$, the following condition holds:

$$\delta h(\bar{\mathbf{x}}, \delta \mathbf{x}) = 0. \tag{B.43}$$

Equation (B.43) is also known as *first-order necessary optimality condition.*

Lemma B.5.16 (Generalization of the Euler-Lagrange equations)**.**
Let us considering the functional $h_{2\kappa}$ in Def. B.5.12. Necessary and sufficient condition for $\bar{\mathbf{x}} \in X(\omega_0, \omega_1)$ to be stationary point of $h_{2\kappa}$ is that the following system of differential equations be satisfied:

$$\sum_{j=0}^{\kappa} (-1)^j \frac{d^j}{d\omega^j} \frac{\partial l(\mathbf{x}, \mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(\kappa)}, \omega)}{\partial \mathbf{x}_i^{(j)}} = \mathbf{0}_M, \qquad \text{for } i = 1, \ldots, n. \tag{B.44}$$

Expressions in (B.44) are called generalized Euler-Lagrange equations.

Remark B.5.17.
Lemma B.5.16 gives information on the stationarity of a functional of the type $h_{2\kappa}$ and represents a first-order necessary optimality condition to be satisfied in order to minimize or maximize $h_{2\kappa}$. There exist second-order necessary and sufficient optimality conditions to satisfy with the purpose to explore the nature of the stationary points for $h_{2\kappa}$.

[7] This generally implies that $\delta \mathbf{x}(\boldsymbol{\omega}_i) = \mathbf{0}_{\bar{X}}$ for $i = 1, 2$.

Bibliography

Abramowitz M. and Stegun I. A. *Handbook of Mathematical Functions With Formulas, Graphs, and Mathematical Tables*, volume 55. National Bureau of Standards Applied Mathematics Series, Washington D.C., US, 1972. pag. 17,75.

Afsar M. M. and Tayarani-N M.-H. Clustering in sensor networks: A literature survey. *Journal of Network and Computer Applications*, 46:198 – 226, 2014. ISSN 1084-8045. URL http://www.sciencedirect.com/science/article/pii/S1084804514002124.

Aguiar A. P., Bayer F. A., Hauser J., Häusler A. J., Notarstefano G., Pascoal A. M., Rucco A., and Saccon A. *Constrained Optimal Motion Planning for Autonomous Vehicles Using PRONTO*, pages 207–226. Springer International Publishing, Cham, 2017. ISBN 978-3-319-55372-6. URL https://doi.org/10.1007/978-3-319-55372-6_10.

Al-Turjman F. and Alturjman S. 5g/iot-enabled uavs for multimedia delivery in industry-oriented applications. *Multimedia Tools and Applications*, Jun 2018. ISSN 1573-7721. URL https://doi.org/10.1007/s11042-018-6288-7.

Alpago D., Zorzi M., and Ferrante A. Identification of sparse reciprocal graphical models. *IEEE Control Systems Letters*, 2(4):659–664, Oct 2018. ISSN 2475-1456.

Amelina N., Fradkov A., Jiang Y., and Vergados D. J. Approximate consensus in stochastic networks with application to load balancing. *IEEE Transactions on Information Theory*, 61(4):1739–1752, April 2015. ISSN 0018-9448.

Anderson B. D. O. Morse theory and formation control. In *2011 19th Mediterranean Conference on Control Automation (MED)*, pages 656–661, June 2011.

Anderson B. D. O., Yu C., Fidan B., and Hendrickx J. M. Rigid graph control architectures for autonomous formations. *IEEE Control Systems Magazine*, 28(6): 48–63, Dec 2008. ISSN 1066-033X.

Andrade E., de Freitas M. A. A., Robbiano M., and Rodríguez J. New lower bounds for the randić spread. *Linear Algebra and its Applications*, 544:254 – 272, 2018. ISSN 0024-3795. URL http://www.sciencedirect.com/science/article/pii/S0024379518300193.

Antholzer S., Wolf C., Sandbichler M., Dielacher M., and Haltmeier M. Compressive time-of-flight 3d imaging using block-structured sensing matrices. *Inverse Problems*, 35(4):045004, mar 2019. URL https://doi.org/10.1088%2F1361-6420%2Faafce3.

Antonelli G. Interconnected dynamic systems: An overview on distributed control. *IEEE Control Systems Magazine*, 33(1):76–88, Feb 2013. ISSN 1066-033X.

Arcak M. Passivity as a design tool for group coordination. *IEEE Transactions on Automatic Control*, 52(8):1380–1390, Aug 2007.

Aster R. C., Borchers B., and Thurber C. H. *Parameter Estimation and Inverse Problems, Third Edition*. Elsevier, 2019.

Aysal T. C., Yildiz M. E., Sarwate A. D., and Scaglione A. Broadcast gossip algorithms for consensus. *IEEE Transactions on Signal Processing*, 57(7):2748–2761, July 2009. ISSN 1941-0476.

Banerjee A. and Mehatari R. An eigenvalue localization theorem for stochastic matrices and its application to randić matrices. *Linear Algebra and its Applications*, 505:85 – 96, 2016. ISSN 0024-3795. URL http://www.sciencedirect.com/science/article/pii/S0024379516301227.

Barooah P. and Hespanha J. P. Distributed estimation from relative measurements in sensor networks. In *2005 3rd International Conference on Intelligent Sensing and Information Processing*, pages 226–231, Dec 2005.

Barooah P. and Hespanha J. P. Estimation on graphs from relative measurements. *IEEE Control Systems*, 27(4):57–74, 2007.

Barooah P. and Hespanha J. P. Error scaling laws for linear optimal estimation from relative measurements. *IEEE Transactions on Information Theory*, 55(12):5661–5673, 2009.

Bauschke H. H. and Combettes P. L. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. CMS Books in Mathematics. Springer-Verlag, 2nd edition, 2011. ISBN 978-1-4419-9466-0.

Beineke L. W. and Wilson R. J. *Graph Connections: Relationships between Graph Theory and Other Areas of Mathematics*. Oxford University Press, Oxford, UK, 1997. URL https://EconPapers.repec.org/RePEc:oxp:obooks:9780198514978.

Bekkerman R., Bilenko M., and Langford J. *Scaling up Machine Learning: Parallel and Distributed Approaches*. Cambridge University Press, Cambridge, UK, 2012.

Belbachir A., Fallah-Seghrouchni A. E., Casals A., and Pasin M. Smart mobility using multi-agent system. *Procedia Computer Science*, 151:447 – 454, 2019. ISSN 1877-0509. URL http://www.sciencedirect.com/science/article/pii/S187705091930523X. The 10th International Conference on Ambient Systems, Networks and Technologies (ANT 2019) / The 2nd International Conference on Emerging Data and Industry 4.0 (EDI40 2019) / Affiliated Workshops.

Berman A. and Plemmons R. J. *Nonnegative matrices in the mathematical sciences*, volume 9. Siam, 1994.

Berman A. and Zhang X.-D. Lower bounds for the eigenvalues of laplacian matrices. *Linear Algebra and its Applications*, 316(1):13 – 20, 2000. ISSN 0024-3795. URL http://www.sciencedirect.com/science/article/pii/S0024379599002645. Special Issue: Conference celebrating the 60th birthday of Robert J. Plemmons.

Bhanu K., Reddy T. B., and Hanumanthappa M. Multi-agent based context aware information gathering for agriculture using wireless multimedia sensor networks.

Egyptian Informatics Journal, 20(1):33 – 44, 2019. ISSN 1110-8665. URL http://www.sciencedirect.com/science/article/pii/S1110866517301858.

Bonin-Font F., Ortiz A., and Oliver G. Visual navigation for mobile robots: A survey. *Journal of Intelligent and Robotic Systems*, 53(3):263, May 2008. ISSN 1573-0409. URL https://doi.org/10.1007/s10846-008-9235-4.

Brin S. and Page L. The anatomy of a large-scale hypertextual web search engine. 1998. In *Proceedings of the Seventh World Wide Web Conference*, 2017.

Brooks R. Distributed sensor networks: A multiagent perspective. *International Journal of Distributed Sensor Networks*, 4, 01 2008.

Brunckner A. M., Brunckner J. B., and Thomson B. S. *Real Analysis*. Pearson Prentice Hall, Upper Saddle River, NJ, US, 1997. pag. 595-596.

Bryson A. E. Optimal control-1950 to 1985. *IEEE Control Systems Magazine*, 16(3): 26–33, June 1996. ISSN 1066-033X.

Camarinha-Matos L. M. Collaborative smart grids – a survey on trends. *Renewable and Sustainable Energy Reviews*, 65:283 – 294, 2016. ISSN 1364-0321. URL http://www.sciencedirect.com/science/article/pii/S1364032116303264.

Cao Y. U., Fukunaga A. S., and Kahng A. Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots*, 4(1):7–27, Mar 1997. ISSN 1573-7527. URL https://doi.org/10.1023/A:1008855018923.

Carron A., Todescato M., Carli R., and Schenato L. An asynchronous consensus-based algorithm for estimation from noisy relative measurements. *IEEE Transactions on Control of Network Systems*, 1(3):283–295, 2014.

Cenedese A., Favaretto C., and Occioni G. Multi-agent swarm control through kuramoto modeling. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 1820–1825, Dec 2016.

Chandrashekar G. and Sahin F. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16 – 28, 2014. ISSN 0045-7906. URL http://www.sciencedirect.com/science/article/pii/S0045790613003066. 40th-year commemorative issue.

Chang H.-C. and Wang L.-C. A Simple Proof of Thue's Theorem on Circle Packing. *arXiv e-prints*, page arXiv:1009.4322, Sep 2010.

Cheeger J. A lower bound for the smallest eigenvalue of the laplacian. In *Proceedings of the Princeton conference in honor of Professor S. Bochner*, pages 195–199, 1969.

Chen F. and Ren W. *On the Control of Multi-Agent Systems: A Survey*. now, 2019. ISBN 9781680835823. URL https://ieeexplore.ieee.org/document/8755265.

Chen Q., Yin C., Zhou J., Wang Y., Wang X., and Chen C. Hybrid consensus-based cubature kalman filtering for distributed state estimation in sensor networks. *IEEE Sensors Journal*, 18(11):4561–4569, June 2018. ISSN 1530-437X.

Chen W., Ding D., Mao J., Liu H., and Hou N. Dynamical performance analysis of communication-embedded neural networks: A survey. *Neurocomputing*, 346:3 – 11, 2019. ISSN 0925-2312. URL http://www.sciencedirect.com/science/article/pii/S0925231219301122. Protocol-Based Performance Analysis of Artificial Neural Networks and Their Applications.

Chmaj G. and Selvaraj H. Distributed processing applications for uav/drones: a survey. In *Progress in Systems Engineering*, pages 449–454. Springer, 2015.

Chu X., Peng Z., Wen G., and Rahmani A. Distributed formation tracking of multi-robot systems with nonholonomic constraint via event-triggered approach. *Neurocomputing*, 275:121 – 131, 2018. ISSN 0925-2312. URL http://www.sciencedirect.com/science/article/pii/S0925231217308147.

Chuang Y., Huang Y. R., D'Orsogna M. R., and Bertozzi A. L. Multi-vehicle flocking: Scalability of cooperative control algorithms using pairwise potentials. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 2292–2299, April 2007.

Chung F. and Zhao W. *PageRank and Random Walks on Graphs*, pages 43–62. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. ISBN 978-3-642-13580-4. URL https://doi.org/10.1007/978-3-642-13580-4_3.

Chung F. R. K. *Spectral Graph Theory (CBMS Regional Conference Series in Mathematics, No. 92).* American Mathematical Society, 1997. ISBN 0821803158. URL http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0821803158.

Cintuglu M. H. and Ishchenko D. Secure distributed state estimation for networked microgrids. *IEEE Internet of Things Journal*, pages 1–1, 2019. ISSN 2327-4662.

Cook D. J. and Holder L. B. *Mining Graph Data.* Wiley, New Jersey, US, 2007.

Cortes J., Martinez S., Karatas T., and Bullo F. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*, 20(2):243–255, April 2004. ISSN 1042-296X.

Coxeter H. S. M. *Regular polytopes.* Methuen, Dover Publications, London, 2nd edition, 1947. ISBN 0-486-61480-8.

Cui P., Wang X., Pei J., and Zhu W. A survey on network embedding. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1, 2018. ISSN 1041-4347.

de Silva V. and Ghrist R. Coordinate-free coverage in sensor networks with controlled boundaries via homology. *The International Journal of Robotic Research*, 25(12):1205–1222, 2006.

de Silva V. and Ghrist R. Coverage in sensor networks via persistent homology. *Algebraic & Geometric Topology*, 7:339–358, 2007. ISSN 1472-2739.

Dehghanpour K., Wang Z., Wang J., Yuan Y., and Bu F. A survey on state estimation techniques and challenges in smart distribution systems. *IEEE Transactions on Smart Grid*, 10(2):2312–2322, March 2019. ISSN 1949-3053.

Desai J. P. A graph theoretic approach for modeling mobile robot team formations. *Journal of Robotic Systems*, 19(11):511–525, 2002. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.10057.

Dong H., Wang Z., Alsaadi F. E., and Ahmad B. Event-triggered robust distributed state estimation for sensor networks with state-dependent noises. *International Journal of General Systems*, 44(2):254–266, 2015. URL https://doi.org/10.1080/03081079.2014.973726.

Dornhege C., Kleiner A., Hertle A., and Kolling A. Multirobot coverage search in three dimensions. *Journal of Field Robotics*, 33(4):537–558, 2016. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21573.

D'Orsogna M., Chuang Y.-l., Bertozzi A., and Chayes L. *Pattern Formation Stability and Collapse in 2D Driven Particle Systems*, pages 103–113. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006. ISBN 978-3-540-33878-9. URL https://doi.org/10.1007/3-540-33878-0_8.

Du D., Li X., Li W., Chen R., Fei M., and Wu L. Admm-based distributed state estimation of smart grid under data deception and denial of service attacks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pages 1–14, 2019. ISSN 2168-2216.

Du Q., Faber V., and Gunzburger M. Centroidal voronoi tessellations: Applications and algorithms. *SIAM Review*, 41(4):637–676, 1999. URL https://doi.org/10.1137/S0036144599352836.

Du S.-L., Sun X.-M., Cao M., and Wang W. Pursuing an evader through cooperative relaying in multi-agent surveillance networks. *Automatica*, 83:155 – 161, 2017. ISSN 0005-1098. URL http://www.sciencedirect.com/science/article/pii/S0005109817303059.

Dudek G., Jenkin M., Milios E., and Wilkes D. Robotic exploration as graph construction. *IEEE Transactions on Robotics and Automation*, 7(6):859–865, Dec 1991. ISSN 1042-296X.

Dörfler F. and Bullo F. Synchronization in complex networks of phase oscillators: A survey. *Automatica*, 50(6):1539 – 1564, 2014. ISSN 0005-1098. URL http://www.sciencedirect.com/science/article/pii/S0005109814001423.

Fabris M., Cenedese A., and Hauser J. Optimal time-invariant formation tracking for a second-order multi-agent system. In *2019 18th European Control Conference (ECC)*, pages 1556–1561, June 2019a.

Fabris M., Michieletto G., and Cenedese A. On the distributed estimation from relative measurements: a graph-based convergence analysis. In *2019 18th European Control Conference (ECC)*, pages 1550–1555, June 2019b.

Fabris M. and Cenedese A. Distributed strategies for dynamic coverage with limited sensing capabilities. In *Mediterranean Conference on Control and Automation 2019*, 2019.

Fan Y.-Z., Xu J., Wang Y., and Liang D. The Laplacian spread of a tree. *Discrete Mathematics and Theoretical Computer Science*, 10(1):79–86, 2008. URL https://hal.inria.fr/hal-00972305. Graphs and Algorithms.

Fathian K., Doucette E. A., Curtis J. W., and Gans N. R. Vision-based distributed formation control of unmanned aerial vehicles. *CoRR*, abs/1809.00096, 2018. URL http://arxiv.org/abs/1809.00096.

Fiedler M. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23 (2):298–305, 1973.

Fornasini E. *"APPUNTI DI TEORIA DEI SISTEMI" (in italian)*. Edizioni Libreria Progetto, Padova, 01 2013. ISBN 978-88-96477-32-8.

Fouad H. and Moskowitz I. S. Chapter 14 - meta-agents: Using multi-agent networks to manage dynamic changes in the internet of things. In **Lawless W., Mittu R., Sofge D., Moskowitz I. S., and Russell S.**, editors, *Artificial Intelligence for the Internet of Everything*, pages 271 – 281. Academic Press, 2019. ISBN 978-0-12-817636-8. URL http://www.sciencedirect.com/science/article/pii/B9780128176368000144.

Franceschelli M., Gasparri A., Giua A., and Seatzu C. Decentralized estimation of laplacian eigenvalues in multi-agent systems. *Automatica*, 49(4):1031 – 1036, 2013. ISSN 0005-1098. URL http://www.sciencedirect.com/science/article/pii/S0005109813000307.

Frigo G., Fabris M., Galli A., Gambarin F., Marsili I. A., Narduzzi C., and Giorgi G. Efficient tracking of heart rate under physical exercise from photoplethysmographic signals. In *2015 IEEE 1st International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI)*, pages 306–311, Sep. 2015.

Frommer A. and Szyld D. B. On asynchronous iterations. *Journal of Computational and Applied Mathematics*, 123(1):201 – 216, 2000. ISSN 0377-0427. URL http://www.sciencedirect.com/science/article/pii/S037704270000409X. Numerical Analysis 2000. Vol. III: Linear Algebra.

Fu J., Wang Q., and Wang J. Robust finite-time consensus tracking for second-order multi-agent systems with input saturation under general directed communication graphs. *International Journal of Control*, 92(8):1785–1795, 2019. URL https://doi.org/10.1080/00207179.2017.1411609.

Gao J., Xiao Y., Liu J., Liang W., and Chen C. P. A survey of communication/networking in smart grids. *Future Generation Computer Systems*, 28(2):391 – 404, 2012. ISSN 0167-739X. URL http://www.sciencedirect.com/science/article/pii/S0167739X11000653.

Garin F. and Schenato L. *A Survey on Distributed Estimation and Control Applications Using Linear Consensus Algorithms*, pages 75–107. Springer London, London, 2010. ISBN 978-0-85729-033-5. URL https://doi.org/10.1007/978-0-85729-033-5_3.

Gastpar M. and Vetterli M. Power, spatio-temporal bandwidth, and distortion in large sensor networks. *IEEE Journal on Selected Areas in Communications*, 23(4): 745–754, April 2005. ISSN 0733-8716.

Ge X., Yang F., and Han Q.-L. Distributed networked control systems: A brief overview. *Information Sciences*, 380:117 – 131, 2017. ISSN 0020-0255. URL http://www.sciencedirect.com/science/article/pii/S0020025515005551.

Geng M., Zhou X., Ding B., Wang H., and Zhang L. Learning to cooperate in decentralized multi-robot exploration of dynamic environments. In **Cheng L., Leung A. C. S., and Ozawa S.**, editors, *Neural Information Processing*, pages 40–51, Cham, 2018. Springer International Publishing. ISBN 978-3-030-04239-4.

Ghrist R., Lipsky D., Derenick J., and Speranzon A. Topological landmark-based navigation and mapping, technical report, 2012. URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.309.5560.

Goldberg K. Robots and the return to collaborative intelligence. *Nature Machine Intelligence*, 1(1):2, 2019.

Goodrich M. T. and Tamassia R. *Data Structure & Algorithms in Java*. John Wiley & Sons, Asia, fifth edition edition, 2011.

Gray R. M. *Toeplitz and Circulant Matrices: A Review*, volume 2:3. Now - Foundations and Trends in Communications and Information Theory, Boston - Delf, MA, US, 2005. pag. 34.

Grönkvist J. and Hansson A. Comparison between graph-based and interference-based stdma scheduling. In *Proceedings of the 2Nd ACM International Symposium on Mobile Ad Hoc Networking &Amp; Computing*, MobiHoc '01, pages 255–258, New York, NY, USA, 2001. ACM. ISBN 1-58113-428-2. URL http://doi.acm.org/10.1145/501449.501453.

Gutman I., Furtula B., and Bozkurt . B. On randić energy. *Linear Algebra and its Applications*, 442:50 – 57, 2014. ISSN 0024-3795. URL http://www.sciencedirect.com/science/article/pii/S0024379513003984. Special Issue on Spectral Graph Theory on the occasion of the Latin Ibero-American Spectral Graph Theory Workshop (Rio de Janeiro, 27-28 September 2012).

Hammond R. A. and Axelrod R. The evolution of ethnocentrism. *Journal of Conflict Resolution*, 50(6):926–936, 2006. URL https://doi.org/10.1177/0022002706293470.

Hatcher A. *Algebraic Topology*. Cambridge Univ. Press, 1st edition, 2002. ISBN 0-521-79540-0. URL https://pi.math.cornell.edu/~hatcher/AT/AT.pdf.

Hauser J. A projection operator approach to the optimization of trajectory functionals. *IFAC Proceedings Volumes*, 35(1):377 – 382, 2002. ISSN 1474-6670. URL http://www.sciencedirect.com/science/article/pii/S1474667015387334. 15th IFAC World Congress.

Hauser J. and Cook J. On the formation rejoin problem**research supported in part by nsf cps grant cns-1446812. *IFAC-PapersOnLine*, 49(18):284 – 289, 2016. ISSN 2405-8963. URL http://www.sciencedirect.com/science/article/pii/S2405896316317578. 10th IFAC Symposium on Nonlinear Control Systems NOLCOS 2016.

Hauser J. and Saccon A. A barrier function method for the optimization of trajectory functionals with constraints. In *Proceedings of the 45th IEEE Conference on Decision and Control, San Diego, CA*, pages 864–869, Dec 2006. URL https://ieeexplore.ieee.org/abstract/document/4178067.

He L., Bai P., Liang X., Zhang J., and Wang W. Feedback formation control of uav swarm with multiple implicit leaders. *Aerospace Science and Technology*, 72: 327 – 334, 2018. ISSN 1270-9638. URL http://www.sciencedirect.com/science/article/pii/S1270963816309816.

Henriques J. F., Caseiro R., Martins P., and Batista J. Exploiting the circulant structure of tracking-by-detection with kernels. In **Fitzgibbon A., Lazebnik S., Perona P., Sato Y., and Schmid C.**, editors, *Computer Vision – ECCV 2012*, pages 702–715, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-33765-9.

Hernández-Martínez E. G. and Aranda-Bricaire E. Convergence and collision avoidance in formation control: A survey of the artificial potential functions approach. In *Multi-agent systems-modeling, control, programming, simulations and applications*. IntechOpen, 2011.

Huang J. and Li S. On the normalised laplacian spectrum, degree-kirchhoff index and spanning trees of graphs. *Bulletin of the Australian Mathematical Society*, 91(3): 353–367, 2015.

Häusler A. Mission planning for multiple cooperative robotic vehicles, 2015. URL https://scholar.google.it/citations?user=FPftGGoAAAAJ&hl=it&oi=sra. Ph.D. thesis.

J. A. K. and M. M. R. *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton University Press, Princeton, NJ, 2008.

Jalalmaab M., Pirani M., Fidan B., and Jeon S. Cooperative estimation of road condition based on dynamic consensus and vehicular communication. *IEEE Transactions on Intelligent Vehicles*, 4(1):90–100, March 2019. ISSN 2379-8904.

Juvan M. and Mohar B. Laplace eigenvalues and bandwidth-type invariants of graphs. *Journal of Graph Theory*, 17(3):393–407, 1993. ISSN 0024-3795. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/jgt.3190170313.

Kamon I., Rivlin E., and Rimon E. A new range-sensor based globally convergent navigation algorithm for mobile robots. In *Proceedings of IEEE International Conference on Robotics and Automation*, volume 1, pages 429–435 vol.1, April 1996.

Kelley C. T. *Iterative Methods for Optimization*. Society for Industrial and Applied Mathematics, 1999. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611970920.

Khodabandeh A. and Teunissen P. J. G. Distributed least-squares estimation applied to GNSS networks. *Measurement Science and Technology*, 30(4):044005, mar 2019. URL https://doi.org/10.1088%2F1361-6501%2Fab034e.

Kim Y. and Mesbahi M. On maximizing the second smallest eigenvalue of a state-dependent graph laplacian. In *Proceedings of the 2005, American Control Conference, 2005.*, volume 1, pages 99–103, June 2005.

Knuth D. E. *The art of computer programming*, volume 3. Addison Wesley Longman, Boston, MA, US, 1998. pag. 409-411.

Kouvaros P. and Lomuscio A. Formal verification of opinion formation in swarms. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, AAMAS '16, pages 1200–1208, Richland, SC, 2016. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 978-1-4503-4239-1. URL http://dl.acm.org/citation.cfm?id=2937029.2937099.

Krick L., Broucke M. E., and Francis B. A. Stabilisation of infinitesimally rigid formations of multi-robot networks. *International Journal of Control*, 82(3):423–439, 2009. URL https://doi.org/10.1080/00207170802108441.

Kuramoto Y. Self-entrainment of a population of coupled non-linear oscillators. In **Araki H.**, editor, *International Symposium on Mathematical Problems in Theoretical Physics*, pages 420–422, Berlin, Heidelberg, 1975. Springer Berlin Heidelberg. ISBN 978-3-540-37509-8.

Landau H. and Odlyzko A. Bounds for eigenvalues of certain stochastic matrices. *Linear algebra and its Applications*, 38:5–15, 1981.

Lee J. R., Gharan S. O., and Trevisan L. Multiway spectral partitioning and higher-order cheeger inequalities. *J. ACM*, 61(6):37:1–37:30, December 2014. ISSN 0004-5411. URL http://doi.acm.org/10.1145/2665063.

Li H., Xie P., and Yan W. Receding horizon formation tracking control of constrained underactuated autonomous underwater vehicles. *IEEE Transactions on Industrial Electronics*, 64(6):5004–5013, June 2017. ISSN 0278-0046.

Li J., Guo J.-M., and Shiu W. C. On the second largest laplacian eigenvalues of graphs. *Linear Algebra and its Applications*, 438(5):2438 – 2446, 2013. ISSN 0024-3795. URL http://www.sciencedirect.com/science/article/pii/S002437951200777X.

Li W., Wang Z., Wei G., Ma L., Hu J., and Ding D. A survey on multisensor fusion and consensus filtering for sensor networks. *Discrete Dynamics in Nature and Society*, 2015, 2015.

Li Z. and Duan Z. *Cooperative Control of Multi-Agent Systems*. CRC Press, Boca Raton, 2015. URL https://www.taylorfrancis.com/books/9781315215976.

Liu G., Liu S., Muhammad K., Sangaiah A. K., and Doctor F. Object tracking in vary lighting conditions for fog based intelligent surveillance of public spaces. *IEEE Access*, 6:29283–29296, 2018. ISSN 2169-3536.

Liu H. and Lu M. Bounds for the laplacian spectral radius of graphs. *Linear and Multilinear Algebra*, 58(1):113–119, 2010. URL https://doi.org/10.1080/03081080802450021.

Liu Y., Chin K., Yang C., and He T. Nodes deployment for coverage in rechargeable wireless sensor networks. *IEEE Transactions on Vehicular Technology*, 68(6):6064–6073, June 2019. ISSN 0018-9545.

Liu Y., Zhao Y., and Chen G. Finite-time formation tracking control for multiple vehicles: A motion planning approach. *International Journal of Robust and Nonlinear Control*, 26(14):3130–3149, 2016. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.3496.

Lovisari E. and Zampieri S. Performance metrics in the average consensus problem: A tutorial. *Annual Reviews in Control*, 36(1):26 – 41, 2012. ISSN 1367-5788. URL http://www.sciencedirect.com/science/article/pii/S1367578812000041.

Luenberger D. G. *Introduction to Dynamic Systems - Theory, Models and Applications*. John Wisley and Sons, Canada, 1st edition, 1979. ISBN 0-471-02594-1.

Luenberger D. G. and Ye Y. *Linear and Nonlinear Programming*. Springer, USA, 3rd edition, 2008. ISBN 978-0-387-74502-2.

Lukic M. and Stojmenovic I. Energy-balanced matching and sequence dispatch of robots to events: Pairwise exchanges and sensor assisted robot coordination. In *2013 IEEE 10th International Conference on Mobile Ad-Hoc and Sensor Systems*, pages 249–253, Oct 2013.

Luo S., Kim J., Parasuraman R., Bae J. H., Matson E. T., and Min B.-C. Multi-robot rendezvous based on bearing-aided hierarchical tracking of network topology. *Ad Hoc Networks*, 86:131 – 143, 2019. ISSN 1570-8705. URL http://www.sciencedirect.com/science/article/pii/S1570870518301100.

Makhdoumi A. and Ozdaglar A. Convergence rate of distributed admm over networks. *IEEE Transactions on Automatic Control*, 62(10):5082–5095, Oct 2017. ISSN 0018-9286.

Marsden A. *Eigenvalues of the Laplacian and their relationship to the connectedness of a graph*. John Wiley & Sons, Chicago, USA, 2013. URL http://math.uchicago.edu/~may/REU2013/REUPapers/Marsden.pdf.

Matula D. W. and Shahrokhi F. Sparsest cuts and bottlenecks in graphs. *Discrete Applied Mathematics*, 27(1):113 – 123, 1990. ISSN 0166-218X. URL http://www.sciencedirect.com/science/article/pii/0166218X9090133W.

Mclurkin J. and Smith J. Distributed algorithms for dispersion in indoor environments using a swarm of autonomous mobile robots. In *in 7th International Symposium on Distributed Autonomous Robotic Systems (DARS*, 2004.

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

Meyer C. D. *Matrix analysis and applied linear algebra*, volume 71. Siam, 2000.

Muhammad A. and Egerstedt M. Connectivity graphs as models of local interactions. *Applied Mathematics and Computation*, 168(1):243 – 269, 2005. ISSN 0096-3003. URL http://www.sciencedirect.com/science/article/pii/S0096300304006022.

Murray R. M. *Control in an information rich world: Report of the panel on future directions in control, dynamics, and systems*. SIAM, 2003.

Muthukrishnan S., Ghosh B., and Schultz M. H. First-and second-order diffusive methods for rapid, coarse, distributed load balancing. *Theory of computing systems*, 31(4):331–354, 1998.

Nakahara M. *Geometry, Topology and Physics*. Institute of Physics Publishing, Bristol and Philadelphia, 2nd edition, 2003. ISBN 0-7503-0606-8. URL http://alpha.sinp.msu.ru/~panov/LibBooks/GRAV/(Graduate_Student_Series_in_Physics)Mikio_Nakahara-Geometry,_Topology_and_Physics,_Second_Edition_(Graduate_Student_Series_in_Physics)-Institute_of_Physics_Publishing(2003).pdf.

Nerode A. and Kohn W. An autonomous systems control theory: an overview. In *IEEE Symposium on Computer-Aided Control System Design*, pages 204–210, March 1992.

Oh K.-K. and Ahn H.-S. Distance-based undirected formations of single-integrator and double-integrator modeled agents in n-dimensional space. *International Journal of Robust and Nonlinear Control*, 24(12):1809–1820, 2014. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.2967.

Oh K.-K., Park M.-C., and Ahn H.-S. A survey of multi-agent formation control. *Automatica*, 53:424 – 440, 2015. ISSN 0005-1098. URL http://www.sciencedirect.com/science/article/pii/S0005109814004038.

Olfati-Saber R. Flocking for multi-agent dynamic systems: algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420, March 2006. ISSN 0018-9286.

Olfati-Saber R. Distributed kalman filtering for sensor networks. In *2007 46th IEEE Conference on Decision and Control*, pages 5492–5498, Dec 2007.

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

Oliva G., Panzieri S., Setola R., and Gasparri A. Gossip algorithm for multi-agent systems via random walk. *Systems & Control Letters*, 128:34 – 40, 2019. ISSN 0167-6911. URL http://www.sciencedirect.com/science/article/pii/S0167691119300647.

Oliva G. and Setola R. Distributed k-means algorithm. *CoRR*, abs/1312.4176, 2013. URL http://arxiv.org/abs/1312.4176.

Ortega A., Frossard P., Kovacević J., Moura J. M. F., and Vandergheynst P. Graph signal processing: Overview, challenges, and applications. *Proceedings of the IEEE*, 106(5):808–828, May 2018. ISSN 0018-9219.

Ozsoyeller D., Beveridge A., and Isler V. Rendezvous in planar environments with obstacles and unknown initial distance. *Artificial Intelligence*, 273:19 – 36, 2019. ISSN 0004-3702. URL http://www.sciencedirect.com/science/article/pii/S0004370218301024.

Parikh N. and Boyd S. Proximal algorithms. *Foundations and Trends in Optimization*, 1(3):127–239, 2014. ISSN 2167-3888. URL http://dx.doi.org/10.1561/2400000003.

Parsegov S. E., Proskurnikov A. V., Tempo R., and Friedkin N. E. A new model of opinion dynamics for social actors with multiple interdependent attitudes and prejudices. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 3475–3480, Dec 2015.

Peng R., Sun H., and Zanetti L. Partitioning well-clustered graphs with k-means and heat kernel. *CoRR*, abs/1411.2021, 2014. URL http://arxiv.org/abs/1411.2021.

Peng X., Sun J., and Geng Z. The geometric convexity on se(3) and its application to the formation tracking in multi-vehicle systems. *International Journal of Control*, 92(3):528–539, 2019. URL https://doi.org/10.1080/00207179.2017.1361043.

Pourbabak H., Luo J., Chen T., and Su W. A novel consensus-based distributed algorithm for economic dispatch based on local estimation of power mismatch. *IEEE Transactions on Smart Grid*, 9(6):5930–5942, Nov 2018. ISSN 1949-3053.

Putra S. A., Trilaksono B. R., Riyansyah M., Laila D. S., Harsoyo A., and Kistijantoro A. I. Intelligent sensing in multiagent-based wireless sensor network for bridge condition monitoring system. *IEEE Internet of Things Journal*, pages 1–1, 2019. ISSN 2327-4662.

Rad A. A., Jalili M., and Hasler M. A lower bound for algebraic connectivity based on the connection-graph-stability method. *Linear Algebra and its Applications*, 435(1): 186 – 192, 2011. ISSN 0024-3795. URL http://www.sciencedirect.com/science/article/pii/S002437951000666X.

Ramaithitima R., Whitzer M., Bhattacharya S., and Kumar V. Sensor coverage robot swarms using local sensing without metric information. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3408–3415, May 2015.

Rao A. V. *Trajectory Optimization: A Survey*, pages 3–21. Springer International Publishing, Cham, 2014. ISBN 978-3-319-05371-4. URL https://doi.org/10.1007/978-3-319-05371-4_1.

Ravazzi C., Chan N. P. K., and Frasca P. Distributed estimation from relative measurements of heterogeneous and uncertain quality. *IEEE Transactions on Signal and Information Processing over Networks*, 5(2):203–217, June 2019. ISSN 2373-776X.

Redhu S. and Hegde R. M. Low complexity landmark-node tracing in wsns using multi-agent random walks. In *Proceedings of the 20th International Conference on Distributed Computing and Networking*, ICDCN '19, pages 261–266, New York, NY, USA, 2019. ACM. ISBN 978-1-4503-6094-4. URL http://doi.acm.org/10.1145/3288599.3288628.

Reynolds C. W. Flocks, herds and schools: A distributed behavioral model. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '87, pages 25–34, New York, NY, USA, 1987. ACM. ISBN 0-89791-227-6. URL http://doi.acm.org/10.1145/37401.37406.

Rocha R. P. and Kudenko D. Session details: Theme: Ai and agents: Irmas - intelligent robotics and multi-agent systems track. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, SAC '19, pages –, New York, NY, USA, 2019. ACM. ISBN 978-1-4503-5933-7. URL http://doi.acm.org/10.1145/3329374.

Rojo O. A nontrivial upper bound on the largest laplacian eigenvalue of weighted graphs. *Linear Algebra and its Applications*, 420(2):625 – 633, 2007. ISSN 0024-3795. URL http://www.sciencedirect.com/science/article/pii/S0024379506003946.

Rossi W. S., Frasca P., and Fagnani F. Distributed estimation from relative and absolute measurements. *IEEE Transactions on Automatic Control*, 62(12):6385–6391, 2017.

Rutishauser S., Correll N., and Martinoli A. Collaborative coverage using a swarm of networked miniature robots. *Robotics and Autonomous Systems*, 57(5):517 – 525, 2009. ISSN 0921-8890. URL http://www.sciencedirect.com/science/article/pii/S0921889008001590.

Santiaguillo-Salinas J. and Aranda-Bricaire E. Time-varying formation tracking with collision avoidance for multi-agent systems. *IFAC-PapersOnLine*, 50(1):309 – 314, 2017. ISSN 2405-8963. URL http://www.sciencedirect.com/science/article/pii/S240589631730068X. 20th IFAC World Congress.

Sargent R. Optimal control. *Journal of Computational and Applied Mathematics*, 124(1): 361 – 371, 2000. ISSN 0377-0427. URL http://www.sciencedirect.com/science/article/pii/S0377042700004180. Numerical Analysis 2000. Vol. IV: Optimization and Nonlinear Equations.

Schiano F., Franchi A., Zelazo D., and Giordano P. R. A rigidity-based decentralized bearing formation controller for groups of quadrotor uavs. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5099–5106, Oct 2016.

Seneta E. *Non-negative matrices and Markov chains*. Springer Science & Business Media, 2006.

Seveso L., Benedetti C., and Paris M. G. A. The walker speaks its graph: global and nearly-local probing of the tunnelling amplitude in continuous-time quantum walks. *Journal of Physics A: Mathematical and Theoretical*, 52(10):105304, feb 2019. URL https://doi.org/10.1088%2F1751-8121%2Fab0195.

Shao Z., Yan F., Zhou Z., and Zhu X. Path planning for multi-uav formation rendezvous based on distributed cooperative particle swarm optimization. *Applied Sciences*, 9(13), 2019. ISSN 2076-3417. URL https://www.mdpi.com/2076-3417/9/13/2621.

Shirani B., Najafi M., and Izadi I. Cooperative load transportation using multiple uavs. *Aerospace Science and Technology*, 84:158 – 169, 2019. ISSN 1270-9638. URL http://www.sciencedirect.com/science/article/pii/S1270963817306697.

Shu J.-L., Hong Y., and Wen-Ren K. A sharp upper bound on the largest eigenvalue of the laplacian matrix of a graph. *Linear Algebra and its Applications*, 347(1):123 – 129, 2002. ISSN 0024-3795. URL http://www.sciencedirect.com/science/article/pii/S0024379501005481.

Shu X. and Ahuja N. Imaging via three-dimensional compressive sampling (3dcs). In *2011 International Conference on Computer Vision*, pages 439–446, Nov 2011.

SinghPal N. and Sharma S. Robot Path Planning using Swarm Intelligence: A Survey. *International Journal of Computer Applications*, 83(12):5–12, December 2013.

Soatti G., Nicoli M., Savazzi S., and Spagnolini U. Consensus-based algorithms for distributed network-state estimation and localization. *IEEE Transactions on Signal and Information Processing over Networks*, 3(2):430–444, June 2017. ISSN 2373-776X.

Sorgun S. Bounds for the largest laplacian eigenvalue of weighted graphs. *International Journal of Combinatorics*, 2013:1–8, 2013.

Spielman D. A. Spectral graph theory and its applications. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pages 29–38, Oct 2007.

Strikwerda J. C. A probabilistic analysis of asynchronous iteration. *Linear Algebra and its Applications*, 349(1):125 – 154, 2002. ISSN 0024-3795. URL http://www.sciencedirect.com/science/article/pii/S0024379502002586.

Sun Z., Mou S., Deghat M., Anderson B., and Morse A. Finite time distance-based rigid formation stabilization and flocking. *IFAC Proceedings Volumes*, 47(3): 9183 – 9189, 2014. ISSN 1474-6670. URL http://www.sciencedirect.com/science/article/pii/S1474667016430648. 19th IFAC World Congress.

Swain A. R. and Hansdah R. A model for the classification and survey of clock synchronization protocols in wsns. *Ad Hoc Networks*, 27:219 – 241, 2015. ISSN 1570-8705. URL http://www.sciencedirect.com/science/article/pii/S1570870514002960.

Tsitsiklis J. and Athans M. Convergence and asymptotic agreement in distributed decision problems. *IEEE Transactions on Automatic Control*, 29(1):42–50, January 1984. ISSN 0018-9286.

Tsochev G., Trifonov R., Yoshinov R., Manolov S., Popov G., and Pavlova G. Some security model based on multi agent systems. In *2018 International Conference on Control, Artificial Intelligence, Robotics Optimization (ICCAIRO)*, pages 32–36, May 2018.

Van Brunt B. *The Calculus of Variations*. Springer-Verlag New York, Inc., 1st edition, 2004. ISBN 0-387-40247-0.

Varotto L., Fabris M., Michieletto G., and Cenedese A. Distributed dual quaternion based localization of visual sensor networks. In *2019 18th European Control Conference (ECC)*, pages 1836–1841, June 2019.

Vijay Kumar , Rus D., and Sanjiv Singh . Robot and sensor networks for first responders. *IEEE Pervasive Computing*, 3(4):24–33, Oct 2004. ISSN 1536-1268.

Vinter R. *Optimal Control*. Springer-Verlag New York, Inc., 1st (reprint of the 2000 edition) edition, 2010. ISBN 978-0-8176-4990-6.

Wang J. and Elia N. Distributed Averaging Algorithms Resilient to Communication Noise and Dropouts. *IEEE Transactions on Signal Processing*, 61(9):2231–2242, May 2013. ISSN 1053-587X, 1941-0476. URL http://ieeexplore.ieee.org/document/6422410/.

Wang S., Djahel S., and McManis J. A multi-agent based vehicles re-routing system for unexpected traffic congestion avoidance. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 2541–2548, Oct 2014.

Wang Y., He L., and Huang C. Adaptive time-varying formation tracking control of unmanned aerial vehicles with quantized input. *ISA Transactions*, 85:76 – 83, 2019. ISSN 0019-0578. URL http://www.sciencedirect.com/science/article/pii/S0019057818303550.

Wasserman S. and Faust K. *Social Network Analysis: Methods and Applications*. Cambridge University Press, Cambridge, UK, 1999.

Wiggins A. *The Minimum of the Dirichlet Kernel*. University of Michigan-Dearborn, 2007. URL http://www-personal.umd.umich.edu/~adwiggin/TeachingFiles/FourierSeries/Resources/DirichletKernel.pdf.

Wittenburg G., Dziengel N., Adler S., Kasmi Z., Ziegert M., and Schiller J. Cooperative event detection in wireless sensor networks. *IEEE Communications Magazine*, 50(12):124–131, December 2012. ISSN 0163-6804.

Xiao F., Wang L., Chen J., and Gao Y. Finite-time formation control for multi-agent systems. *Automatica*, 45(11):2605 – 2611, 2009. ISSN 0005-1098. URL http://www.sciencedirect.com/science/article/pii/S0005109809003586.

Xie J. and Liu C.-C. Multi-agent systems and their applications. *Journal of International Council on Electrical Engineering*, 7(1):188–197, 2017. URL https://doi.org/10.1080/22348972.2017.1348890.

Xu Y., Lu R., Shi P., Li H., and Xie S. Finite-time distributed state estimation over sensor networks with round-robin protocol and fading channels. *IEEE Transactions on Cybernetics*, 48(1):336–345, Jan 2018. ISSN 2168-2267.

Xue X., Wu F., and Yuan C. Robust consensus for linear multi-agent systems with structured uncertainties. *International Journal of Control*, 0(0):1–12, 2019. URL https://doi.org/10.1080/00207179.2019.1612096.

Yan B., Shi P., Lim C.-C., Wu C., and Shi Z. Optimally distributed formation control with obstacle avoidance for mixed-order multi-agent systems under switching topologies. *IET Control Theory & Applications*, 12:1853–1863(10), September 2018. ISSN 1751-8644. URL https://digital-library.theiet.org/content/journals/10.1049/iet-cta.2017.1086.

Yang Q., Cao M., Garcia de Marina H., Fang H., and Chen J. Distributed formation tracking using local coordinate systems. *Systems & Control Letters*, 111: 70 – 78, 2018. ISSN 0167-6911. URL http://www.sciencedirect.com/science/article/pii/S0167691117302128.

Yang Quan Chen and Zhongmin Wang . Formation control: a review and a new consideration. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3181–3186, Aug 2005.

Yong Z., Liyi Z., Jianfeng H., Zhe B., and Yi Y. An indoor gas leakage source localization algorithm using distributed maximum likelihood estimation in sensor networks. *Journal of Ambient Intelligence and Humanized Computing*, 10(5):1703–1712, May 2019. ISSN 1868-5145. URL https://doi.org/10.1007/s12652-017-0624-z.

Zampieri S. Trends in networked control systems. *IFAC Proceedings Volumes*, 41(2): 2886–2894, 2008.

Zelazo D., Franchi A., Bülthoff H. H., and Giordano P. R. Decentralized rigidity maintenance control with range measurements for multi-robot systems. *The International Journal of Robotics Research*, 34(1):105–128, 2015. URL https://doi.org/10.1177/0278364914546173.

Zhang T., Li Z., Wu Q., and Zhou X. Decentralized state estimation of combined heat and power systems using the asynchronous alternating direction method of multipliers. *Applied Energy*, 248:600 – 613, 2019a. ISSN 0306-2619. URL http://www.sciencedirect.com/science/article/pii/S0306261919307330.

Zhang X.-M., Han Q.-L., Ge X., Ding D., Ding L., Yue D., and Peng C. Networked control systems: a survey of trends and techniques. *IEEE/CAA Journal of Automatica Sinica*, 2019b.

Zhang X.-D. The Laplacian eigenvalues of graphs: a survey. *arXiv e-prints*, page arXiv:1111.2897, Nov 2011.

Zhao S. and Zelazo D. Bearing rigidity and almost global bearing-only formation stabilization. *IEEE Transactions on Automatic Control*, 61(5):1255–1268, May 2016. ISSN 0018-9286.

Zhao Y., Duan Z., Wen G., and Zhang Y. Distributed finite-time tracking control for multi-agent systems: An observer-based approach. *Systems & Control Letters*, 62 (1):22 – 28, 2013. ISSN 0167-6911. URL http://www.sciencedirect.com/science/article/pii/S0167691112002095.

Zheng Z., Xie S., Dai H., Chen X., and Wang H. An overview of blockchain technology: Architecture, consensus, and future trends. In *2017 IEEE International Congress on Big Data (BigData Congress)*, pages 557–564, June 2017.

Zhou D., Zhang A., and Bi W. Finite time formation control for multi-agent systems with event-triggered control scheme. In *AIAA Scitech 2019 Forum*, pages 1–13, 2019. URL https://arc.aiaa.org/doi/abs/10.2514/6.2019-0659.

Zomorodian A. Fast construction of the vietoris-rips complex. *Computers and Graphics*, 34(3):263 – 271, 2010. ISSN 0097-8493. URL http://www.sciencedirect.com/science/article/pii/S0097849310000464. Shape Modelling International (SMI) Conference 2010.

Zuo Z. Nonsingular fixed-time consensus tracking for second-order multi-agent networks. *Automatica*, 54:305 – 309, 2015. ISSN 0005-1098. URL http://www.sciencedirect.com/science/article/pii/S000510981500031X.