

TESIS DOCTORAL EN INGENIERÍA

**REDES CONFIGURABLES DE VEHÍCULOS NO
TRIPULADOS**

AUTOR

ING. JUAN FRANCISCO PRESENZA

DIRECTORES

DR. ING. J. IGNACIO ALVAREZ-HAMELIN (FIUBA/CONICET)

DR. ING. IGNACIO MAS (UDESA/CONICET)

JURADOS/AS

DR. ING. RODRIGO CASTRO (FCEyN UBA/CONICET)

DRA. INGA. GLORIA FRONTINI (FI UNMDP/CONICET)

DR. ING. PABLO FIERENS (ITBA/CONICET)

DR. ING. ANÍBAL ZANINI (FIUBA/CONICET)

LUGAR DE TRABAJO

LABORATORIO DE AUTOMÁTICA Y ROBÓTICA, DEPARTAMENTO DE INGENIERÍA

ELECTRÓNICA, FACULTAD DE INGENIERÍA, UNIVERSIDAD DE BUENOS AIRES

TESIS FINANCIADA POR BECA DOCTORAL UBACYT

**FACULTAD DE INGENIERÍA, UNIVERSIDAD DE BUENOS
AIRES**

BUENOS AIRES, JULIO DE 2023

TESIS DOCTORAL EN INGENIERÍA

**REDES CONFIGURABLES DE VEHÍCULOS NO
TRIPULADOS**

CONFIGURABLE NETWORKS OF UNMANNED
VEHICLES

ING. JUAN FRANCISCO PRESENZA

Índice general

Resumen	IX
Abstract	XI
Símbolos	XIII
Glosario	XV
1. Introducción	1
1.1. Antecedentes	1
1.2. Motivación	4
1.3. Contribuciones y Estructura de la Tesis	8
1.4. Alcance	14
1.5. Preliminares	16
1.5.1. Notación	17
1.5.2. Protocolo de Gradiente Descendiente	20
1.5.3. Manejo de la Información en el Control Multirobot	20
2. Introducción a Frameworks	25
2.1. Grafos	25
2.2. Frameworks	29

2.2.1.	Rigidez de Frameworks	30
2.2.2.	Rigidez Genérica	35
2.3.	Frameworks Generalizados	38
2.3.1.	Definición	38
2.3.2.	Control de Formaciones en el Espacio del Cluster	41
2.4.	Resumen	42
3.	Conectividad Algebraica Generalizada	45
3.1.	Preliminares	45
3.2.	Cotas	48
3.2.1.	Grafos Completos	49
3.2.2.	Grafos Arbitrarios	51
3.3.	Influencia del Diámetro del Grafo	56
3.4.	Resumen	60
4.	Rigidez Basada en Subframeworks	61
4.1.	Introducción	61
4.2.	Definición	62
4.3.	Descomposición Densa en Subframeworks	63
4.3.1.	Métricas	67
4.3.2.	Alcance Mínimo de Rigidez	69
4.3.3.	Simulaciones	70
4.4.	Descomposición Dispersa en Subframeworks	77
4.4.1.	Descomposición Óptima	83
4.4.2.	Simulaciones	85
4.5.	Resumen	87

5. Control de Rigidez por Subframeworks	89
5.1. Introducción	89
5.2. Algoritmo de Posicionamiento	92
5.2.1. Formulación	93
5.3. Control de Mantenimiento de la Rigidez	96
5.3.1. Modelo Dinámico de los Robots	99
5.3.2. Modelo Dinámico del Grafo	99
5.3.3. Evasión de colisiones	100
5.3.4. Mantenimiento de la Rigidez	101
5.3.5. Carga en la red	102
5.3.6. Ley de control	103
5.4. Implementación Descentralizada	104
5.4.1. Intercambio de Información	104
5.4.2. Análisis de Estabilidad del Control	108
5.5. Validación	111
5.5.1. Caso 1: Escalabilidad	112
5.5.2. Caso 2: Alcances variables	115
5.5.3. Caso 3: Robustez al retardo temporal	116
5.6. Resumen	120
6. Cinematografía Aérea Multirobot	127
6.1. Introducción	127
6.2. Variables del Cluster para Cinematografía	131
6.2.1. Formación con 2 VANTs	132
6.2.2. Formación con v VANTs	134
6.3. Optimización para Cinematografía basada en CSC	136
6.3.1. Funciones de Costo	138

6.3.2. Evasión de colisiones	142
6.3.3. Evasión de visibilidad mutua	143
6.3.4. Cómputo de la Acción de Control	145
6.4. Validación	146
6.4.1. Ensayos con 2 VANTs	148
6.4.2. Ensayos con v VANTs	154
6.5. Resumen	155
7. Conclusiones	157
7.1. Comentarios Finales	157
7.2. Trabajo Futuro	159

Resumen

El avance en diversos campos científico-tecnológicos ha impulsado el desarrollo de vehículos autónomos que operan en distintos dominios tales como terrestres, acuáticos y aéreos. En general, las tareas que realizan se desarrollan de manera individual y el trabajo en conjunto se limita a la coordinación efectuada por quienes operan tales sistemas. Recientemente, ha crecido el interés de conformar sistemas complejos que puedan hacer uso de varios de estos vehículos robóticos de manera coordinada para aprovechar las ventajas de cada uno, así como también permitir la creación de nuevas capacidades. Una de las problemáticas fundamentales, al controlar grupos de robots móviles, es la coordinación de los agentes: habilidad de los mismos de lograr un objetivo común mediante la cooperación. Una parte crucial de cualquier esquema de cooperación de robots móviles es la comunicación entre los mismos y/o con una base terrestre. La conectividad en este caso es un problema complejo, dadas las características de heterogeneidad y movilidad de los agentes, sumadas a sus limitaciones de potencia, cómputo, etc. Este trabajo busca atacar el problema de comunicación en redes de robots móviles de manera integral, considerando las características de los robots, las especificaciones de formaciones y los requerimientos de conectividad de tareas específicas. Además, resulta de gran importancia desarrollar esquemas que permitan la asignación de recursos y roles, identificando el tipo de robot idóneo para cada una de ellas. Esto permite realizar las tareas de manera tal de minimizar el tiempo total, la energía consumida, o maximizar la precisión, entre otras.

Abstract

The advancement in several scientific and technological fields have impuled the development of unmanned vehicles (also known as mobile robots) that operate in different domains such as terrestrial, marine, and aerial. In general, tasks are performed by one robot and cooperation is limited to the system operators. Recently, there has been a growing interest in conforming complex systems with several robotic vehicles in a coordinated way. This allows to take advantage of each agent's capabilities and also to generate new collective abilities. One of the fundamental topics regarding the control of a group of mobile robots is associated with their coordination: the ability to achieve a common goal through cooperation. A crucial component in any cooperation scheme is the communication between robots, and with a terrestrial base. Therefore, connectivity arises as a complex problem due to robot heterogeneity, mobility, and the limited available energy, computing power, etc. This work aims to address the communication problem in mobile robot networks in an integrated manner, considering the robots attributes, the formation specifications and the connectivity requirements. Also, it is of great importance to develop strategies that allow optimal and dynamic resource allocation and the assignment of roles. This allows to do the prescribed tasks with optimality, e.g., minimize the total elapsed time or the energy consumption, or maximize precision, etc.

Símbolos

$\mathbb{R}, \mathbb{R}_{\geq 0}, \mathbb{R}_{> 0}$	Números reales, reales no negativos y reales positivos
$\mathbb{N}_0, \mathbb{N}_1$	Números naturales no negativos y positivos
\mathcal{S}^n	Potencia cartesiana n -ésima de un conjunto
u, \mathbf{u}	Un vector columna y una concatenación de vectores columna
$u[i]$ ($\mathbf{u}[i]$)	Entrada i -ésima (bloque i -ésimo) de un vector (concatenado)
e_1, \dots, e_n	Base canónica de \mathbb{R}^n
$E_{11}, E_{12}, \dots, E_{nn}$	Base canónica de $\mathbb{R}^{n \times n}$
$\ u\ , \ u\ _\infty, \ u\ _p$	Normas de un vector
\mathcal{S}^\perp	Complemento ortogonal de un conjunto en \mathbb{R}^n
$\text{linsp}\{\cdot\}$ ($\text{affsp}\{\cdot\}$)	Subespacio lineal (afín) generado por un conjunto de vectores
A, \mathbf{A}	Una matriz y una matriz definida por bloques
I_n	Matriz identidad de $n \times n$
$\text{Nul}(A), \text{rg}(A)$	Espacio nulo y rango de una matriz
$A[i, j]$ ($\mathbf{A}[i, j]$)	Entrada (bloque) ij de una matriz
$A[i, \cdot], A[\cdot, j]$	Fila i y columna j de una matriz
$A \otimes B$	Producto de Kronecker entre matrices
$\lambda(A)^{(m)}$	Autovalor de A con multiplicidad m
$\Lambda(A)$	Espectro de una matriz
$\lambda_k(A)$	Autovalor k -ésimo, en orden creciente, de una matriz simétrica

$A > 0$ ($A \geq 0$)	Matriz simétrica (semi) definida positiva
$\mathbb{O}(d)$ ($\mathbb{SO}(d)$)	Grupo de transformaciones ortogonales (especiales)
$\mathbb{E}(d)$ ($\mathbb{SE}(d)$)	Grupo de transformaciones euclídeas (especiales)
$R_w(\theta)$	Rotación, un ángulo θ , sobre $w \in \mathbb{R}^3$, en sentido antihorario
$S(x)$	Matriz antisimétrica del producto vectorial
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	Grafo dado por un conjunto de vértices \mathcal{V} y uno de enlaces \mathcal{E}
$K(\mathcal{V})$	Grafo completo inducido por un conjunto de vértices
$i \sim j$	Dos vértices conectados por un enlace
$\sum_{i \sim j}$	Sumatoria sobre el conjunto de enlaces
\mathcal{N}_i	Conjunto de vecinos del vértice i
$g(i, j)$	Distancia geodésica entre dos vértices
$D(\mathcal{G}), \bar{n}(\mathcal{G})$	Diámetro y grado promedio de un grafo
$\mathcal{F} = (\mathcal{G}, p)$	Framework dado por un grafo y una realización espacial p
$a(\mathcal{G})$ ($a_d(\mathcal{G})$)	Conectividad Algebraica (Generalizada) de un grafo
$\mathcal{C}(p), \mathcal{T}(p)$	Configuraciones congruentes a p y subespacio trivial
$\mathbf{R}(p), \mathbf{L}(p)$	Matriz y Laplaciano de Rigidez de un framework
$\Lambda(\mathcal{G}, p)$	Espectro de un framework
$\lambda_k(\mathcal{G}, p)$	Autovalor k -ésimo, en orden creciente, de un framework
\mathbf{r}, \mathbf{c}	Vectores de estados en el Espacio Robot y del Cluster
$\mathbf{J}_f(r), \mathbf{L}_f(r)$	Incidencia y Laplaciano de un framework generalizado

Glosario

VNT (VANT)	Vehículo (Aéreo) No Tripulado
IMU	<i>Inertial Measurement Unit</i>
RF	Radio-frecuencia
LIDAR	<i>Light Detection and Ranging</i>
GPS, GLONASS, GALILEO	Sistemas Satelitales de Posicionamiento
WSN	<i>Wireless Sensor Networks</i>
TOA (AOA)	<i>Time (Angle) of Arrival</i>
RSSI	<i>Received Signal Strength Indicator</i>
RGB	<i>Red-Green-Blue</i>
UWB	<i>Ultra-Wide Band</i>
OWD	<i>One-way delay</i>
RTD	<i>Round-trip delay</i>
CSC	<i>Cluster Space Control</i>
POI	<i>Point of Interest</i>
DOP	<i>Director of Photography</i>
FOV	<i>Field of View</i>
DOF	<i>Degrees of Freedom</i>
VL	<i>Virtual Leader</i>
FPV	<i>First Person View</i>

Capítulo 1

Introducción

1.1. Antecedentes

La utilización de vehículos no tripulados (VNT), también conocidos como robots móviles, ha crecido de manera vertiginosa durante las últimas dos décadas. El interés por estos sistemas se debe a las ventajas inherentes que presentan, tanto desde el punto de vista de seguridad —evitando la necesidad de poner en riesgo vidas humanas—, como operativo —permitiendo una reducción en el tamaño y costo de los vehículos, etc.

Gracias al desarrollo de nuevos materiales, al avance en la tecnología de los sensores y de los sistemas embebidos de control, los vehículos no tripulados poseen cada vez mayor autonomía. En la actualidad, pueden encontrarse estos sistemas en diferentes actividades, entre las que se pueden mencionar las misiones de búsqueda y rescate [1]; la inspección de ambientes peligrosos y monitoreo ambiental [2]; entrega de bienes, transporte de objetos y manipulación aérea en general [3, 4, 5]; propósitos militares y de vigilancia; cinematografía [6]; etc. En [7, 8] se reporta el progreso en el estudio teórico y práctico de sistemas móviles multi-robóticos.

Los robots terrestres fueron las primeras plataformas móviles en ser desarrolladas y

actualmente se encuentran en un amplio espectro de aplicaciones más allá de los entornos de laboratorio, ya sea en la industria, en tareas como manejo de inventario en depósitos, traslado de piezas en fábricas, etc., así como en ambientes domésticos (aspiradoras, cortadoras de pasto o juguetes). Por otro lado, los robots aéreos han proliferado en los últimos años debido a que los mismos ofrecen ventajas en reducción de riesgos para los pilotos, disminución de costos de operación y son apropiados para tareas de observación remota de alta resolución –como monitoreo de agricultura o de la diversidad biológica– en situaciones en que las imágenes satelitales no permiten suficiente resolución espacial o temporal. Aplicaciones de alto interés, como la adquisición *in-situ* de valores de contaminación en cuencas hidrográficas, el monitoreo de impacto ambiental de la actividad industrial o debido a desastres naturales, pueden beneficiarse también de la utilización de vehículos autónomos marinos de superficie o sumergibles.

Esto, a su vez, atrajo el interés en el control coordinado de sistemas multi-agente, tanto en las comunidades científicas como en la industria. La razón detrás de esta adopción es que la distribución de tareas permite el aumento del rendimiento, la tolerancia a fallas y la posibilidad de habilitar nuevas capacidades. Estos efectos se amplifican cuando el grupo consiste en robots que operan en diferentes dominios (acuático, terrestre, aéreo o espacial) [9] y, por lo tanto, cada uno posee características y capacidades distintivas. Ejemplos de tareas donde se aprovechan las capacidades de sistemas multi-agente que resultan en beneficios significativos incluyen operaciones de búsqueda y rescate [10], agricultura de precisión, extinción de incendios forestales y prevención de incendios [11, 12], cinematografía autónoma [13, 14], vigilancia y monitoreo [15], o escolta de activos y patrullaje [16, 17], entre otras. Lo mismo se ve en tareas de toma de datos ambientales, donde múltiples VNTs abarcan una mayor extensión del espacio, o pueden mejorar la calidad de las capturas al concentrarse sobre un mismo espacio, como se ha hecho con vehículos marinos en [18]. En Fig. 1.1 y Fig. 1.2 se muestran ejemplos de tareas donde múltiples

vehículos han reemplazado a un único vehículo para realizar las mismas tareas de manera más eficiente. En particular, Fig. 1.1 muestra aplicaciones con vehículos aéreos no tripulados (VANTs) en el transporte cooperativo de objetos, mientras que la Fig. 1.2 muestra aplicaciones industriales con sistemas multi-robóticos.

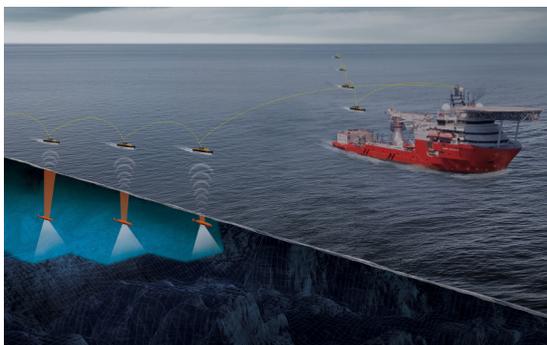


Créditos: Robin Ritz, Mark W. Muller, Markus Hehn, y Raffaello D'Andrea [3].



Créditos: Bruno Gabrich, David Saldaña, Vijay Kumar, y Mark Yim [4].

Figura 1.1: Agarre y transporte cooperativo sustituyendo vehículos grandes por pequeños.



(a) Mapeo del lecho marino con múltiples robots.

Créditos: *Ocean Infinity Group Ltd.*



(b) Edificio desplazado usando patas robóticas.

Créditos: *China Central Television (CCTV).*

Figura 1.2: Aplicaciones industriales utilizando sistemas multi-robóticos.

El método de control utilizado para gobernar el comportamiento de cada robot, de forma tal de obtener un comportamiento deseado para todo el sistema, es un elemento clave en la operación de grupos de robots móviles. Así, el diseño de las leyes de control para

estas aplicaciones se inspira en diversos campos, como la teoría de juegos [19], la biología [20, 21, 22], o las cadenas cinemáticas [23, 24, 25]. Como resultado de la interacción entre estos campos surgen diversas técnicas que incluyen campos potenciales [26, 27, 28], primitivas de comportamiento [29], formulación en espacio del cluster [30], arquitecturas tipo enjambre [31, 32], y configuraciones líder-seguidor [33, 34], entre otras. Las mismas enriquecen el área bien establecida de Control de Sistemas Interconectados (*Control of Network Systems*), la cual se encuentra en la intersección entre los sistemas de control y las ciencias de redes.

Existen diversos criterios para clasificar las técnicas de coordinación de sistemas multirobot. Una posible clasificación se basa en el tipo de acoplamiento requerido entre los vehículos [35, 36, 37], a partir de la cual surge la distinción que clasifica en formaciones y *swarms*. En las formaciones, se desea mantener constante cierta magnitud relativa entre los vehículos, como distancia u orientación, respetando una forma o estructura [38, 39, 40]. En ellas, suelen asignarse roles distintos a los vehículos en función de su ubicación dentro de la formación. Por otro lado, en los *swarms*, generalmente no se imponen restricciones relativas, sino que se definen reglas básicas de comportamiento a partir de las cuales emerge un orden global [41, 42]. Estos arreglos están formados usualmente por decenas de agentes con capacidades básicas y en muchos casos los mismos se consideran dispensables. Resulta interesante analizar los algoritmos de control diseñados en esta tesis bajo esta óptica, lo cual se realizará en los capítulos correspondientes.

1.2. Motivación

Como se ha mencionado, la utilización de sistemas de múltiples robots móviles ha proliferado en los últimos años. A partir del empleo de estrategias de control multirobot descentralizado, surge un concepto de gran importancia: el de coordinación de los agentes, el cual está típicamente relacionado con la habilidad de los mismos de lograr un

objetivo común mediante un protocolo de cooperación [43, 44, 45]. Una parte crucial de cualquier esquema de cooperación de robots móviles es el intercambio de información [46, 47]. Por lo tanto, la topología de la red subyacente, dada por las interacciones entre los agentes, tiene una gran importancia. Cuando el grupo está conformado por pocos agentes, la estructura de la red de comunicación es trivial, ya sea conectando todos los robots entre ellos, o a cada uno con un nodo central a modo de estación terrena centralizada de monitoreo y control [48, 49]. Cuando el número de robots empieza a aumentar —y en consecuencia el volumen de datos y las distancias— la topología de la red comienza a ser un parámetro de diseño en la formación [50]. Resulta entonces apropiado modelar la red multirobot con un grafo, y emplear resultados de dicha teoría en el estudio y el diseño de los algoritmos de control. Más allá de problemas interesantes, tales como contemplar casos en que el grafo no es completo y estudiar como balancear en este caso la transmisión de información [51], el hecho de que los vértices sean móviles implica que el grafo tendrá una naturaleza inherentemente dinámica, con características y requerimientos variantes con el tiempo. Diseñar arquitecturas de control y protocolos de comunicación que puedan contemplar estas características son objetivos de este trabajo.

Además, para potenciar la cooperación, resulta de gran importancia desarrollar esquemas que permitan la asignación de recursos y roles, es decir, dada una aplicación particular, distribuir las tareas a realizar identificando el tipo de robot idóneo para cada una de ellas. Más aún, resulta interesante la reasignación dinámica de estas tareas en función de la evolución de la tarea a desarrollar. Esto permite realizar las tareas de manera tal de minimizar el tiempo total, la energía consumida, o maximizar la precisión, entre otras. Como ejemplo de esta situación, supongamos una tarea realizada de manera autónoma por un grupo de robots, como puede ser la captura de datos ambientales tales como floraciones algales o la evaluación de sitios contaminados. El grupo de robots puede dinámicamente subdividirse en pequeños grupos que realizan distintas subtareas, algunos realizando la

tarea principal y otros realizando tareas de apoyo, tales como cadenas de comunicaciones o protección de perímetro. Los robots pueden tomar distintos roles de manera dinámica [52] dependiendo de distintos factores (distancias, obstáculos, condiciones ambientales, etc.).

Uno de los puntos de partida de los estudios realizados en esta tesis es la localización cooperativa de robots. En la actualidad, las tareas que se desean realizar con robots son cada vez más sofisticadas y por ello se requiere mayor precisión en la estimación del estado de los agentes (posición, orientación, velocidad, etc.). El correcto posicionamiento de los mismos es una funcionalidad indispensable si se quiere que los mismos posean autonomía. La estimación del estado de un único vehículo de manera autónoma es un problema «de larga data», para lo cual se emplean sensores introceptivos: unidades inerciales (IMU), odómetros, sensores de flujo óptico; y exoceptivos: sensores de detección luz (LIDAR), sensores de radio (RF), sistemas de posicionamiento global (GPS, GLO-NASS, GALILEO, etc.) y sistemas de captura de movimiento mediante cámaras RGB. Los sensores introceptivos típicamente proveen información de velocidad y aceleración con una alta tasa de adquisición. Sin embargo, para estimar la posición u orientación del vehículo es necesario integrar dichas observaciones, lo que produce una acumulación de los errores de medición que crece con el transcurso del tiempo. Esto es aceptable solo si se emplean sensores inerciales de alta precisión, los cuales son de costo elevado y de gran peso y tamaño como para ser instaladas en robots de tamaño reducido. Los sensores exoceptivos típicamente proveen información de posición u orientación asociada a una referencia externa, pero a una tasa de muestreo baja y con errores apreciables. Para superar estas limitaciones, la técnica más extendida se basa en la fusión de sensores, que permite combinar las capacidades de los mismos y obtener un posicionamiento robusto y preciso [53, 54, 55]. Si bien esta técnica pueden proveer la información necesaria, en muchos casos, los sistemas de posicionamiento tales como el GPS, o los de captura de movimiento

por imágenes, no se encuentran disponibles. Algunos ejemplos de operación donde generalmente no hay acceso a estos sistemas de posicionamiento son (Fig. 1.3): ambientes interiores, entornos submarinos, del espacio profundo, interiores de minas, bosques frondosos y ciudades altamente edificadas (*urban canyons*).

Cuando se trata de una formación multirobot, existe una gran heterogeneidad en el dominio de los agentes: aéreo, terrestre, acuático, subacuático, interiores, entre otros. Por lo tanto, es deseable diseñar esquemas de localización cooperativa que permitan aprovechar al máximo las capacidades de sensado de cada agente, y reducir la redundancia de sensores, manteniendo cierta robustez [56, 57, 58]. Un esquema que ha generado interés en redes de robots móviles es aquel en que la red posee un subconjunto de vehículos de referencia (*anchor nodes*), que poseen acceso a algún sistema de posicionamiento externo (navegación satelital, antenas de radio, etc.), mientras que los demás (*unknown nodes*) poseen la capacidad de medir información relativa respecto a otros robots cercanos (distancia, dirección relativa, etc.). Este problema se enmarca dentro del área de localización de redes de sensores inalámbricas (*wireless sensor networks* (WSN)) [59, 60], con la particularidad de que cada sensor se reemplaza por un robot con capacidad tanto de sensado y comunicación, como de control de movilidad. Típicamente, los enfoques han utilizado los canales de comunicación entre agentes para estimar distancia y/o dirección relativa entre nodos mediante técnicas tales como *Time of Arrival* (TOA), *Angle of Arrival* (AOA), *Received Signal Strength Indicator* (RSSI), entre otras [61, 62, 63, 64]. Recientemente, se han incorporado esquemas que usan odometría visual mediante cámaras RGB y señales de radio *ultra-wide band* (UWB), entre otros [58, 65, 66]. Resulta entonces interesante estudiar la capacidad del sistema para fusionar estos sensores de forma descentralizada mediante la cooperación, y así resolver la localización de todos los vehículos, aprovechando al máximo los recursos disponibles.

Además de las habilidades presentadas en esta sección, un algoritmo de control de



(a) Océanos.
Créditos: QYSEA FIFISH Ltd.



(b) Minas.
Créditos: US Defense Advanced Research Projects Agency.



(c) Urban Canyons.
Créditos: Fotógrafo desconocido. <http://mowabb.com/aimages/archives/007314.html>

Figura 1.3: Ejemplos de entornos de operación sin acceso al posicionamiento global.

una red de robots móviles debe ser escalable. Esto significa que el mismo debe tener la capacidad de mantener tanto el funcionamiento como el desempeño a medida que aumenta la cantidad de vehículos de la red. Por esta razón, analizar y diseñar sistemas bajo la óptica de la escalabilidad, es un principio rector de esta tesis. En el apartado 1.5.3, se presenta una discusión más profunda sobre la escalabilidad en las técnicas de control.

1.3. Contribuciones y Estructura de la Tesis

En esta sección se destacan las principales contribuciones realizadas al área de estudio de sistemas multirobóticos.

El trabajo realizado busca aportar en el análisis y el diseño de esquemas de coordinación y control de sistemas autónomos multirobot conscientes de las limitaciones en las capacidades de sensado y en las comunicaciones de datos en redes de robots móviles. Para eso, se consideran las características de los robots, las especificaciones de las formaciones y los requerimientos de conectividad de tareas específicas.

Capítulo 2: Introducción a Frameworks

Se realiza un estudio formal de dichos sistemas mediante la utilización de frameworks, objetos matemáticos que permiten modelar el estado de los robots y las interrelaciones entre los mismos. Con base en las motivaciones descritas, se hace énfasis en el uso clásico de los frameworks, llamados aquí frameworks de distancia, que permiten analizar y establecer las condiciones cualitativas y cuantitativas de la capacidad de localizar y/o controlar un conjunto de robots a partir de mediciones de distancia relativa, lo que se conoce como teoría de rigidez de frameworks. No obstante, la propuesta de la utilización de frameworks no se agota aquí, sino que se propone una generalización de los mismos que permite contemplar aquellos casos donde la información disponible sea distinta a la distancia, como por ejemplo de dirección relativa (que ha captado el interés de los investigadores recientemente [67, 68]), e incluso funciones que involucren más de dos agentes. Esto permite extender y aprovechar el estudio formal sobre frameworks de distancia a una multiplicidad de casos más generales encontrados en las aplicaciones.

Capítulo 3: Conectividad Algebraica Generalizada

Continuando con el estudio de frameworks, se presentan resultados teóricos relativos a las propiedades de la conectividad algebraica generalizada, un parámetro propio de los grafos subyacentes que extiende los conceptos de conectividad algebraica clásica. La conectividad algebraica generalizada resulta un parámetro de extrema importancia en el

estudio de la rigidez, por lo que su estudio tiene un interés tanto teórico como práctico. Se realiza una recopilación de los conocimientos existentes sobre la conectividad algebraica generalizada, y se comparan los diferentes resultados existentes a la fecha, a la vez que se contribuye con nuevos teoremas. Además, se presenta una cota superior a la conectividad algebraica generalizada para grafos arbitrarios, en función de la conectividad algebraica clásica, como fue presentado en [69]. Esto supone una contribución al estudio de esta propiedad, ya que provee una cuantificación del concepto conocido de que la conectividad algebraica es condición necesaria para la rigidez. Además, permite emplear la gran cantidad de herramientas y resultados existentes para la conectividad clásica en el campo de la rigidez de grafos, lo cual supone un avance en esta área. A su vez, se estudia el comportamiento de la conectividad algebraica generalizada a medida que el diámetro del grafo crece. Esto es interesante debido a que en una red de robots móviles, la cantidad de conexiones de cada agente es limitada, por lo tanto, el diámetro crece con la cantidad de vértices. Esto resulta de utilidad para evaluar las limitaciones en escalabilidad de los algoritmos existentes de control de rigidez [70, 71], lo que nos lleva a presentar una nueva forma de entender la rigidez en función de subframeworks.

Capítulo 4: Rigidez Basada en Subframeworks

Se presenta una formulación novedosa que permite descomponer un framework en subframeworks. Esto posibilita expresar la propiedad de rigidez de un framework como una propiedad distribuida a lo largo de la red. Se define formalmente el concepto de subframework y se presentan los teoremas que permiten expresar la rigidez en función de los mismos. También, se proponen métricas orientadas a medir el grado de descentralización obtenido en una determinada descomposición, con miras en el control de formaciones de robots. A partir de dichas métricas y de la definición del alcance mínimo de rigidez, se muestran simulaciones para comparar las características de la descomposición densa con

respecto la definición clásica de rigidez.

Capítulo 5: Control de Rigidez por Subframeworks

Se aplica la teoría de rigidez por subframeworks desarrollada, para el control coordinado del mantenimiento de la rigidez de un framework. En primer lugar, se propone una adaptación de los algoritmos de posicionamiento utilizados en redes multirobot, que prescindir de los algoritmos de consenso, mejorando su escalabilidad. También, se propone un esquema de control por optimización y se desarrollan las funciones de costo necesarias para cumplir los múltiples objetivos enunciados. Luego, se presenta un protocolo que permite intercambio de información necesario para poder aplicar dicha ley de control de forma descentralizada por los agentes de la red. El esquema propuesto es ventajoso respecto de protocolos anteriores debido a que no requiere el intercambio de información global, evitando los altos retardos y la gran carga comunicacional, especialmente en redes con gran cantidad de agentes. Para validar dicho esquema se presentan simulaciones computacionales que pretenden evaluar el desempeño del controlador en distintos escenarios de interés. Los resultados desarrollados en los Capítulos 4 y 5 fueron presentados en [72, 73].

Capítulo 6: Cinematografía Aérea Multirobot

Se propone como caso de estudio una aplicación interesante para la utilización, en sistemas multirobóticos, de los frameworks generalizados y de la estrategia de control asociada a los mismos, *Cluster Space Control* (CSC). Se postula una configuración para un sistema de múltiples vehículos aéreos no tripulados (VANT) para la filmación aérea autónoma, el cual fue presentado en [74]. Con este esquema, un sistema de múltiples VANTs con cámaras pan-tilt es concebido como una grúa robótica virtual, capaz de seguir trayectorias generadas durante el vuelo (*on-the-fly*) por un operador de cámara, mientras se

cumplen objetivos específicos de la aplicación. Como las variables de estado del cluster son definidas basándose en la aplicación, los parámetros controlados son intuitivos —y son los mismos que un operador terreno cambiaría para hacer modificaciones a la trayectoria en tiempo de ejecución. Además, el sistema garantiza de forma autónoma que los VANTs filmadores no van a aparecer en el campo de visión de los demás, y que las colisiones son evitadas, con el costo de desviarse de la trayectoria predefinida, liberando al operador de estas tareas. Para validar el esquema propuesto se emplean herramientas avanzadas de simulación, las cuales permiten evaluar el desempeño del sistema con gran precisión.

Capítulo 7: Conclusiones

En este último capítulo se recopilan las contribuciones propuestas en esta tesis y se discuten futuras direcciones en el estudio de Redes de Vehículos No Tripulados.

Publicaciones

Esta tesis se basa en las siguientes publicaciones (dispuestas en el mismo orden en que son presentadas):

A New Upper Bound for the d -dimensional Algebraic Connectivity of Arbitrary Graphs

J. F. Presenza, I. A. Mas, J. I. Giribet, J. I. Alvarez-Hamelin

Repositorio arXiv, 2022

Subframework-Based Rigidity Control in Multirobot Networks

J. F. Presenza, J. I. Alvarez-Hamelin, I. Mas, and J. I. Giribet

American Control Conference, 2022

Control de Rigidez por Subframeworks en Sistemas Multirobot

J. F. Presenza, J. I. Giribet, J. I. Alvarez-Hamelin e I. Mas

Jornadas Argentinas de Robótica, 2022

Aerial Multi-Camera Robotic Jib Crane

P. Moreno, J. F. Presenza, I. Mas, and J. I. Giribet

IEEE/RAS Robotics and Automation Letters, 2021 e *IEEE International Conference of Robotics and Automation, 2021*

Además de las publicaciones mencionadas arriba, durante este doctorado, he contribuido a otros trabajos que, para mantener el foco de esta tesis y que sea de un tamaño razonable, no han sido incluidos. Estos trabajos se encuentran listados en orden cronológico:

Validación de un Sistema de Posicionamiento Ultrasónico para Interiores

J. F. Presenza, C. D. Pose, I. A. Mas, J. I. Giribet

X Jornadas Argentinas de Robótica, 2019

Sistema de control para un Robot Aerodeslizante

E. Pecker Marcosig, J. F. Presenza, J. I. Giribet, I. A. Mas

X Jornadas Argentinas de Robótica, 2019

Trajectory Following with a MAV Under Rotor Fault Conditions

C. D. Pose, F. Presenza, I. Mas, and J. I. Giribet

11th International Micro Air Vehicle Competition and Conference (IMAV), 2019 y *Unmanned Systems, 2020*

Adicionalmente, esta tesis contiene una cantidad significativa de material no publicado, como el análisis de la Conectividad Algebraica Generalizada para grafos completos presentada en el Capítulo 3, y la nueva perspectiva de una descomposición dispersa en subframeworks, desarrollada en el Capítulo 4.

1.4. Alcance

A continuación, presentamos una serie de puntos que fueron tenidos en cuenta para acotar el marco de estudio de este trabajo, y dotar de la profundidad necesaria a los tópicos que son de mayor interés. Las decisiones tomadas, respecto del alcance considerado, pueden ser resumidas en la Fig. 1.4. En la misma, se divide al sistema de control multirobot en tres capas o niveles: (i) el nivel bajo, dado por la física de los vehículos y los sensores, (ii) el nivel medio del preprocesamiento de las mediciones primarias de los sensores y la adecuación de comandos de velocidad en acciones de los actuadores, (iii) el nivel alto, dado por los algoritmos de posicionamiento y control que resuelven la coordinación de los agentes.

En primer lugar, el control de sistemas multirobóticos realizado se centra en aquellas técnicas de alto nivel que resuelvan la coordinación y la concreción de objetivos globales. No se estudian los controles de menor nivel asociados a la estabilización de los vehículos y el seguimiento de comandos por parte de los mismos. Esto se hace en parte porque el tipo de vehículo y su dominio de operación condicionan la dinámica requerida de los esquemas de comunicación y control. Por este motivo, se abstrae el estudio de la tecnología empleada, y se asume que los vehículos tienen capacidad de seguimiento de comandos de velocidad mediante un controlador apropiado, lo cual es usual en este tipo de aplicaciones.

En segundo lugar, si bien existen distintos tipos de sensores y métodos para obtener mediciones tanto relativas entre robots (distancia, dirección relativa, orientación relativa), como absolutas (posición, orientación), no nos detendremos en los principios de funciona-

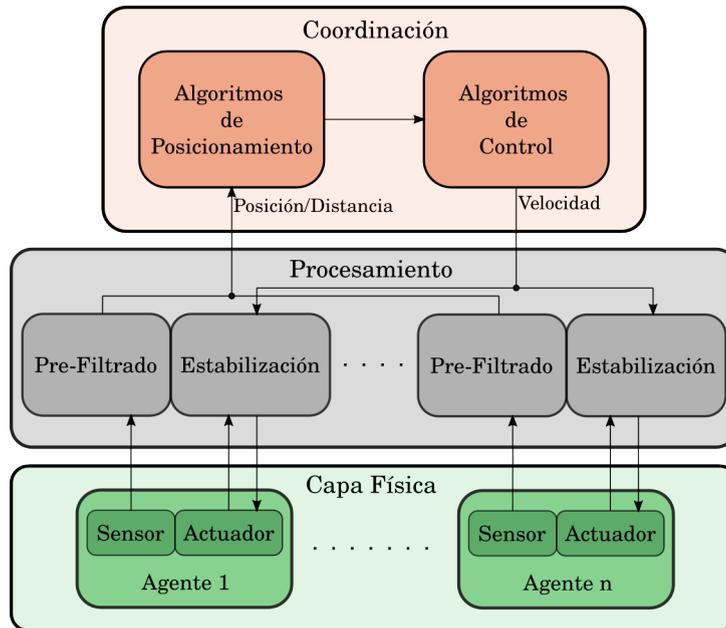


Figura 1.4: Esquema de tres niveles de las estrategias de control. En este trabajo, el nivel intermedio se asume dado y con características ideales.

miento de dichos sensores ni en el procesamiento de los datos primarios. Por simplicidad, y para no perder generalidad, asumiremos que los dispositivos de medición realizan un procesamiento preliminar de dichos datos, cuya salida es la magnitud de interés, afectada por una incerteza. Este es un abordaje adoptado en otros trabajos similares, por ejemplo [66].

En tercer lugar, si bien el paradigma de control por *path-planning*, basado en el diseño *offline* de trayectorias (es decir, previo a la operación) y el seguimiento *online* de las mismas, es un área de intensa investigación en sistemas multirobóticos [75, 76, 77, 78], no es el enfoque adoptado aquí. Esto se debe a que, a pesar de que el mismo permite contemplar múltiples objetivos asociados a la aplicación durante el diseño de la trayectoria, en la práctica existen situaciones imprevistas, generalmente en la forma de perturbaciones externas (errores de posicionamiento, oclusiones visuales, fuerzas ambientales, fallas en las comunicaciones). Las mismas no son manejadas efectivamente por los controles de se-

guimiento de trayectorias, por su naturaleza a lazo abierto. Dependiendo de la aplicación, esto puede ser aceptable o no. En los escenarios de estudio planteados en esta tesis, los objetivos de control tienen requerimientos que deben ser monitoreados durante la misión, como por ejemplo evasión de colisiones, pérdida de enlaces de comunicación, requerimientos artísticos en cinematografía. El incumplimiento de estos objetivos producidos por eventos fortuitos puede echar a perder toda la misión. Por estos motivos, adoptamos un enfoque de control *online* a lazo cerrado, que permite supervisar en tiempo real todas las variables del sistema, a expensas de una limitada capacidad de cómputo a bordo de los vehículos.

Por último, dada la naturaleza de las funciones objetivo planteadas, que dependen típicamente de las posiciones de los robots, los algoritmos de control son diseñados en el dominio de tiempo continuo. No obstante, la validación de dichos esquemas se realizará mediante simulaciones en el dominio de tiempo discreto. El estudio formal del traspaso entre un modelo continuo y uno discreto no es abordado aquí. Se asume que las simulaciones representan apropiadamente la dinámica continua de los robots, si se realizan con un paso de tiempo lo suficientemente pequeño.

1.5. Preliminares

En esta sección presentamos la notación utilizada durante esta tesis. Luego, introducimos el protocolo de gradiente descendiente, que es de gran utilidad en este trabajo. Por último, desarrollamos las ideas básicas detrás del manejo de la información en los sistemas multirobot, el cual será de utilidad en la comprensión de los conceptos desarrollados durante este trabajo.

1.5.1. Notación

Conjuntos

En esta tesis, denotamos $\mathbb{R}, \mathbb{R}_{\geq 0}, \mathbb{R}_{> 0}$ al conjunto de números reales, reales no negativos y reales positivos, respectivamente. El conjunto de números naturales es \mathbb{N}_1 , y si se incluye el cero, \mathbb{N}_0 . El número cardinal de un conjunto \mathcal{S} es $|\mathcal{S}|$. La potencia cartesiana n -ésima de \mathcal{S} la denotamos $\mathcal{S}^n = \{(x_1, \dots, x_n) : x_i \in \mathcal{S} \forall i = 1, \dots, n\}$. El centroide de un conjunto de números o vectores $\{x_1, \dots, x_n\}$ es $\sum_{i=1}^n x_i$.

Vectores

Los vectores $u = [u_1 \ \dots \ u_n]^T \in \mathbb{R}^n$ son considerados vectores columnas; y $u[i]$ se utiliza para extraer el i -ésimo componente $u_i \in \mathbb{R}$. Un vector en negrita se emplea para denotar que es una concatenación vertical de vectores como en $\mathbf{u} = [u_1^T \ \dots \ u_c^T]^T$ con $u_k \in \mathbb{R}^{n_k}$ para todo $k = 1, \dots, c$; y $\mathbf{u}[k]$ denota el vector k -ésimo $u_k \in \mathbb{R}^{n_k}$. Si se quiere explicitar que las n entradas de un vector u toman valores únicamente en un subconjunto $\mathcal{S} \subset \mathbb{R}$, se escribe $u \in \mathcal{S}^n$. Se utiliza el símbolo $\mathbf{1}_n$ para escribir el vector de n entradas todas iguales a la unidad. La base canónica de \mathbb{R}^n es e_1, \dots, e_n . La norma-2 de un vector u se escribe $\|u\| = \sqrt{u_1^2 + \dots + u_n^2}$; y la norma-infinito $\|u\|_\infty = \max\{u_i : 1 \leq i \leq n\}$. La norma inducida por una matriz simétrica definida positiva P , se escribe $\|u\|_P$ y es igual a $u^T P u$. Dado un conjunto de vectores $\{u_1, \dots, u_n\}$ en \mathbb{R}^n , el subespacio lineal generado por los mismos se escribe $\text{linsp}\{u_1, \dots, u_n\}$, mientras que el subespacio afín generado por los mismos se nota $\text{affsp}\{u_1, \dots, u_n\}$.

Matrices

Las matrices se denotan con letra mayúscula $A \in \mathbb{R}^{m \times n}$, y escribimos $A \in \mathcal{S}^{m \times n}$ si las entradas de la matriz se encuentran restringidas a cierto subconjunto $\mathcal{S} \subset \mathbb{R}$. La matriz

identidad de $n \times n$ es I_n . El espacio nulo de una matriz es $\text{Nul}(A)$, y el rango de la misma $\text{rg}(A)$. La entrada i, j de una matriz se obtiene mediante $A[i, j]$, mientras que las fila i y la columna j se escriben $A[i, \cdot]$ y $A[\cdot, j]$ respectivamente. Cuando se escribe una matriz por bloques, se usa la negrita \mathbf{A} y la notación $\mathbf{A}[i, j]$ indica el bloque correspondiente a dichos índices. El producto de kronecker de dos matrices A y B se denota $A \otimes B$. El espectro de una matriz se denota $\Lambda(A) = \{\lambda_1^{(m_1)}, \dots, \lambda_r^{(m_r)}\}$ donde m_k corresponde a la multiplicidad de dicho autovalor. Si A es simétrica, se ordenan los mismos de menor a mayor: $\lambda_1(A) \leq \lambda_2(A) \leq \dots \leq \lambda_n(A)$. Si una matriz simétrica es (semi) definida positiva, denotamos mediante la expresión $(A \geq 0) A > 0$.

Rotaciones e Isometrías

En esta tesis, se hace uso de las isometrías, también llamadas transformaciones euclídeas, debido a que las mismas representan movimientos rígidos de un conjunto de puntos en el espacio: traslaciones compuestas con rotaciones y reflexiones.

Denominamos $\mathbb{O}(d)$ al grupo de transformaciones ortogonales en \mathbb{R}^d , tales que $T(x) = Qx$ con $Q^T Q = I_d$, correspondientes a rotaciones/reflexiones. El subgrupo especial ortogonal $\mathbb{SO}(d)$, de transformaciones con $\det(Q) = 1$, corresponde a rotaciones en el espacio. Tanto $\mathbb{O}(d)$ como $\mathbb{SO}(d)$ tienen dimensión $\binom{d}{2}$. Las matrices de rotación respecto de los

ejes x, y, z en sentido antihorario son, respectivamente,

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix}, \quad (1.1)$$

$$R_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix}, \quad (1.2)$$

$$R_z(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (1.3)$$

El espacio tangente a $\mathbb{SO}(d)$ en el punto I_d es el subespacio de $\mathbb{R}^{d \times d}$ de matrices antisimétricas $\mathbb{A}(d)$, cuya dimensión es $\binom{d}{2}$. Una base del mismo está dada por

$$\{A_{kl} := E_{kl} - E_{lk} : 1 \leq k < l \leq d\}, \quad (1.4)$$

donde $E_{kl} = e_k e_l^T$ es el elemento kl -ésimo de la base canónica de $\mathbb{R}^{d \times d}$. Dado un vector $x = [x_1 \ x_2 \ x_3]^T \in \mathbb{R}^3$, se denota $S(x)$ a la matriz antisimétrica

$$S(x) = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix} \quad (1.5)$$

tal que $S(x)y = x \times y$, para todo $y \in \mathbb{R}^3$.

El grupo euclídeo $\mathbb{E}(d)$ está formado por las isometrías de \mathbb{R}^d , transformaciones que preservan las distancias entre todo par de puntos. Toda transformación en $\mathbb{E}(d)$ consiste de una transformación ortogonal seguida de una traslación: $T(x) = Qx + t$ para todo x en

\mathbb{R}^d . La dimensión de $\mathbb{E}(d)$ es $\binom{d+1}{2}$.

1.5.2. Protocolo de Gradiente Descendiente

En este trabajo, empleamos en numerosas ocasiones la dinámica del gradiente descendiente en los algoritmos de control y localización, por lo que es necesario dar una apropiada formulación. Sea $x \in \mathbb{R}^n$ un vector que modela el estado de cierto sistema y $f : \mathbb{R}^n \rightarrow \mathbb{R}$ una función de x continuamente diferenciable que posee un mínimo global. El protocolo de gradiente descendiente con dinámica continua propone la siguiente trayectoria:

$$\dot{x}(t) = -\frac{\partial f}{\partial x}(x(t)), \quad x(0) = x_0 \quad (1.6)$$

donde x_0 es el estado inicial del sistema. De esta dinámica se sigue que la función f es no creciente con el tiempo

$$\dot{f}(x(t)) = \frac{\partial f}{\partial x} \dot{x} = -\left\| \frac{\partial f}{\partial x} \right\|^2 \leq 0,$$

y se deduce que el protocolo converge a un mínimo local o a un punto de inflexión de f .

Es posible adaptar esta dinámica mediante una función $\mu : \mathbb{R}^n \rightarrow \mathbb{R}^n$ tal que

$$\dot{x}(t) = -\mu \left(\frac{\partial f}{\partial x}(x(t)) \right), \quad (1.7)$$

para obtener algún comportamiento deseado manteniendo las propiedades de convergencia. Para ello, es suficiente que $\mu(y)^T y \geq 0$ para todo $y \in \mathbb{R}^n$.

1.5.3. Manejo de la Información en el Control Multirobot

Existen diferentes enfoques para analizar y comparar los sistemas de control de redes multirobot. La óptica adoptada en esta tesis clasifica los algoritmos de control según el tipo

de comunicaciones necesarias para aplicarlos. Esto se debe a que, debido a las limitaciones existentes en las redes inalámbricas, no se puede asumir que todos los robots pueden intercambiar información entre sí en cualquier instante. Por este motivo, el problema de las comunicaciones debe ser debidamente abordado en el diseño de los controladores. Esto da lugar a dos criterios para clasificar los algoritmos de control de formaciones según: (i) el tipo de información que se debe procesar (global o local), (ii) cómo es el flujo de esa información entre los agentes de la red (centralizado o descentralizado).

En primer lugar, decimos que los requerimientos de comunicación de un esquema son *globales* si, para lograr los objetivos de la aplicación que se esté desarrollado, la red debe intercambiar y procesar información que depende del estado de todos los agentes. Un ejemplo de objetivo que requiere intercambio y procesamiento de información global es el problema de *rendez-vous*. En él, se desea que los agentes se acerquen hasta encontrarse en un punto en común que coincide con el centroide de las posiciones iniciales, suponiendo que no existen colisiones. El lugar donde se ubica el centroide, depende de las posiciones de todos los robots, por lo tanto, cada uno debe recibir directa o indirectamente información de la posición de todos los demás, si quiere dirigirse hacia el encuentro. En cambio, existen aplicaciones donde no es necesario que información de toda la red sea procesada por los agentes de la misma, sino que la información es *local*. Un ejemplo de este caso es la evasión de colisiones entre robots vecinos: si un agente se comunica con otros dentro de un cierto radio de conexión, es suficiente que intercambie y procese información de tales nodos para evitar colisiones con ellos.

En segundo lugar, un enfoque *centralizado* indica la existencia de una estación central de cómputo con los recursos necesarios para controlar un grupo de vehículos en simultáneo. Esencialmente, este tipo de abordaje es una extensión de la filosofía tradicional de control de un único vehículo, donde la red es considerada como una unidad. Si bien la posibilidad de reunir toda la información disponible en un centro de cómputo puede ser

una ventaja, las técnicas centralizadas presentan numerosos inconvenientes. Entre ellos se destacan: altos requerimientos de capacidad computacional del nodo central, efectos de cuello de botella en los canales de comunicación, falta de robustez por la existencia de un punto de falla, entre otras. Todas estas desventajas se traducen en un serio límite a la escalabilidad, entendida como la capacidad de una estrategia de mantener sus propiedades y desempeño a medida que aumenta la cantidad de robots. Por el contrario, en las técnicas *descentralizadas*, las comunicaciones, el cómputo y la decisión se reparten entre los agentes. Si, en particular, este reparto es uniforme para todos los agentes, decimos que el esquema es *distribuido*. En ambos casos, los agentes actúan únicamente en función de la percepción de su entorno (a través de sensores), y de las interacciones locales con otros vehículos (a través de comunicaciones punto-a-punto). Esto permite lidiar mejor con las restricciones de cómputo y energía, los limitados rangos de comunicaciones inalámbricas, la distribución física de los sensores, etc., haciendo de las técnicas de control descentralizadas las de mayor potencial. Fig. 1.5 ejemplifica estas definiciones.

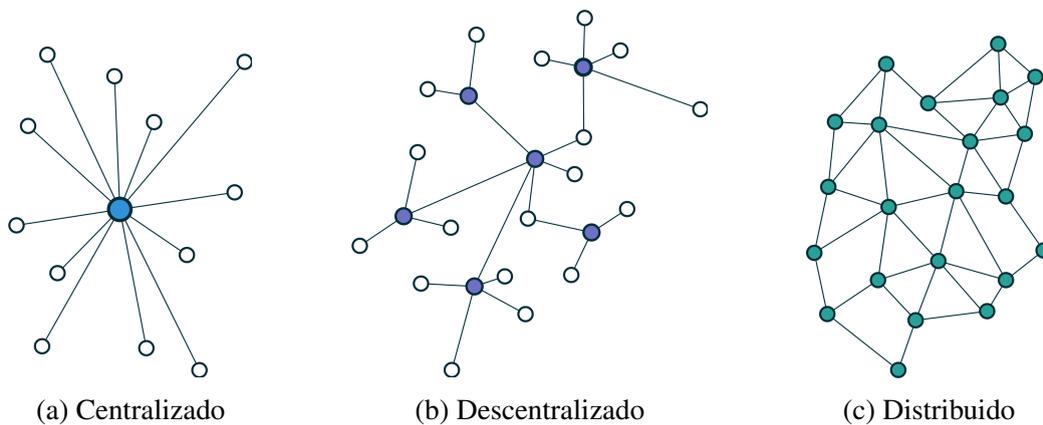


Figura 1.5: Esquemas de control de redes multirobot según el flujo de información empleado.

Es importante notar que los dos criterios de clasificación propuestos están íntimamente relacionados, y que ambas categorías no son mutuamente excluyentes. En general, el

tipo de información que se debe procesar influye en cómo debe ser el flujo de la misma. Por ejemplo, si un esquema de control requiere del procesamiento de algún parámetro que es global a toda la red, entonces su implementación directa requiere de un esquema de comunicaciones centralizado. Sin embargo, en muchos casos se desea implementar el intercambio global de información de forma descentralizada. Para esto, se utilizan algoritmos de consenso que permiten estimar parámetros globales a partir de interacciones locales entre nodos vecinos [79, 80, 53]. Sin embargo, los protocolos de consenso presentan limitaciones en cuanto a la escalabilidad. Esto se debe a que, como la información es global, la misma debe propagarse por toda la red, y esto requiere un tiempo que aumenta con el diámetro de la red [81]. Este análisis sirve para ilustrar las limitaciones inherentes a los esquemas de control coordinado cuando debe procesarse información global, y las correspondientes ventajas de utilizar información de naturaleza local.

Capítulo 2

Introducción a Frameworks

En este capítulo, desarrollamos los conceptos fundamentales relacionados con los frameworks, objetos matemáticos que permiten modelar formalmente una red de robots. En primer lugar, realizamos una breve presentación de la teoría de grafos, debido a su extendida utilización para modelar redes de agentes, y su estrecha relación con los frameworks. Luego presentamos en detalle la definición clásica de frameworks, y su principal propiedad: la rigidez. La misma es crucial para la estimación y el control de redes de robots móviles a partir de las distancias relativas entre los mismos. Por último, extendemos las ideas de frameworks existentes en la literatura para proponer los frameworks generalizados: objetos que permiten formalizar el estudio de cualquier formación de robots, cuyas interrelaciones van más allá de las conexiones punto a punto modeladas con un grafo.

2.1. Grafos

En esta tesis empleamos principalmente los denominados grafos simples, que son aquellos cuyos enlaces no tienen dirección, y tampoco hay lazos propios, es decir, enlaces que conecten a un vértice consigo mismo. Tampoco se permite la existencia de más

de un enlace entre un determinado par de nodos. Estas restricciones serán dejadas de lado para contemplar modelos más complejos en la Sección 2.3.

Un grafo simple $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ es el par compuesto de un conjunto de v vértices $\mathcal{V} = \{1, \dots, v\}$, y un conjunto de e enlaces $\mathcal{E} = \{e_1, \dots, e_e\}$ tal que cada enlace consiste en un par de vértices conectados. Si $\{i, j\} \in \mathcal{E}$ denotamos $i \sim j$. A continuación se presentan algunas definiciones que sientan las bases que necesitamos para operar con grafos.

Definición 2.1.1. La **unión** entre dos grafos $(\mathcal{V}_1, \mathcal{E}_1)$ y $(\mathcal{V}_2, \mathcal{E}_2)$ es el grafo $(\mathcal{V}_1 \cup \mathcal{V}_2, \mathcal{E}_1 \cup \mathcal{E}_2)$. De forma similar se define la **intersección**.

Definición 2.1.2. El grafo **completo** $K(\mathcal{V})$ es aquel que contiene todos los posibles enlaces.

Definición 2.1.3. Sea $(\mathcal{V}, \mathcal{E})$ un grafo. El **subgrafo inducido** por un subconjunto de vértices $\mathcal{V}' \subseteq \mathcal{V}$, es el par $(\mathcal{V}', \mathcal{E}')$ donde $\mathcal{E}' = \{\{i, j\} \in \mathcal{E} : i, j \in \mathcal{V}'\}$.

Definición 2.1.4. El conjunto de vecinos de un vértice $\mathcal{N}_i = \{j \in \mathcal{V} : j \sim i\}$. El **grado** $n(i)$ es la cantidad de vecinos.

Definición 2.1.5. Un **camino simple** sobre \mathcal{G} es una secuencia de vértices distintos $\pi = (i_1, i_2, \dots, i_m)$ tales que $i_k \sim i_{k+1}$ para todo $k = 1, \dots, m-1$. La **longitud** del camino, $|\pi| = m-1$, es decir, la cantidad de enlaces que atraviesa.

Definición 2.1.6. Un grafo es **conexo** si para cada par de vértices existe un camino que los une. Un **componente conexo** de $(\mathcal{V}, \mathcal{E})$ es el subgrafo inducido por un subconjunto de vértices por \mathcal{V}' , tal que el mismo es conexo y no forma parte de ningún otro subgrafo conexo con más vértices o enlaces.

Definición 2.1.7. La **distancia geodésica** $g(i, j)$ entre dos nodos la longitud de un camino más corto que los une

$$g(i, j) = \text{mín}\{|\pi| : \pi \in \Pi_{i,j}\}, \quad (2.1)$$

donde $\Pi_{i,j}$ denota al conjunto de caminos que tienen a i y a j como nodos terminales.

Definición 2.1.8. La *excentricidad* $\varepsilon(i)$ de un vértice es la distancia al nodo más alejado del mismo; mientras que el **diámetro** $D(\mathcal{G})$ de un grafo es la distancia geodésica entre el par de nodos más alejados,

$$\varepsilon(i) = \max\{g(i, j) : j \in \mathcal{V}\}, \quad D(\mathcal{G}) = \max\{g(i, j) : i, j \in \mathcal{V}\}. \quad (2.2)$$

Un nodo se dice **periférico** si su excentricidad equivale al diámetro del grafo.

El límite de Moore [82] indica que un grafo simple \mathcal{G} con grado máximo $n_{\text{máx}}$ y v vértices tiene diámetro

$$D(\mathcal{G}) \geq \begin{cases} \frac{v-1}{2}, & n_{\text{máx}} = 2, \\ \frac{\log((v-1)^{n_{\text{máx}}-2} + 1)}{\log(n_{\text{máx}}-1)}, & n_{\text{máx}} > 2. \end{cases} \quad (2.3)$$

Esto implica que el diámetro crece al menos logarítmicamente con la cantidad de vértices de un grafo.

Matrices de un Grafo

Existen diversas formas de representar información de un grafo utilizando matrices. Entre las más utilizadas se encuentran las detalladas a continuación.

Definición 2.1.9. La *matriz de adyacencia* A de un grafo es la matriz $\{0, 1\}^{v \times v}$ tal que la entrada $A[i, j] = 1$ si $i \sim j$.

Definición 2.1.10. La *matriz de grados* D de un grafo, es la matriz de $v \times v$ diagonal tal que la entrada $D[i, i] = n(i)$.

Definición 2.1.11. La *matriz laplaciana* de un grafo es $L := D - A$.

Es bien sabido que L es una matriz simétrica semi-definida positiva. Las propiedades algebraicas del laplaciano tienen un papel fundamental en el estudio de la difusión de información sobre un grafo, y por consiguiente en su conectividad.

Teorema 2.1.1 (En [83]). *Sea un grafo \mathcal{G} , las siguientes proposiciones son equivalentes:*

1. \mathcal{G} tiene c componentes conexas $(\mathcal{V}_k, \mathcal{E}_k)$, $k = 1, \dots, c$
2. La multiplicidad de $\lambda = 0$ como autovalor de L es igual a c .
3. El conjunto de vectores $u_1, \dots, u_c \in \{0, 1\}^v$ definidos tal que

$$u_k[i] = \begin{cases} 1, & i \in \mathcal{V}_k, \\ 0, & \text{caso contrario,} \end{cases}$$

forman una base ortogonal de $\text{Nul}(L)$.

Definición 2.1.12. *La **conectividad algebraica** del grafo, $a(\mathcal{G}) := \lambda_2(L)$, se define como el segundo menor autovalor del laplaciano.*

Corolario 2.1.1. *Un grafo es conexo si y solo si $a(\mathcal{G}) > 0$.*

La conectividad algebraica es un parámetro de extrema importancia en el estudio espectral de grafos [84]. Esto se debe tanto a su gran influencia en una diversidad de campos de estudio y de aplicación, como por ejemplo los sistemas multirobot. En esta área, la conectividad algebraica tiene un papel fundamental en el desarrollo de algoritmos de control de conectividad [85] y de filtros de consenso para fusión de sensores [53], entre otros. Se recomiendan los textos [86, 87] para un estudio general sobre conceptos de la teoría de grafos; y [88, 89] para un análisis avanzado en la teoría espectral de grafos.

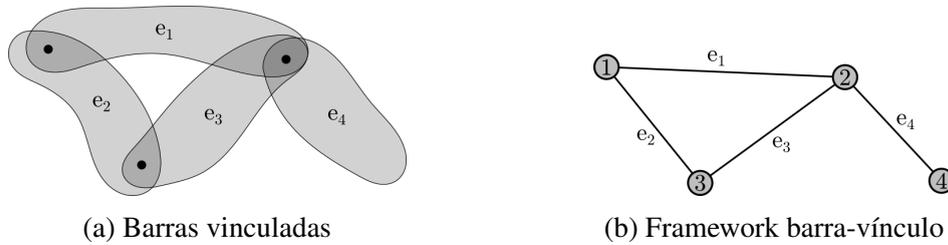


Figura 2.1: Un conjunto de barras vinculadas en el plano modeladas mediante un framework.

2.2. Frameworks

Un framework representa un conjunto de puntos o cuerpos rígidos situados en el espacio y una serie de barras rígidas que fijan las distancias entre dos puntos. Los frameworks han sido empleados durante años en áreas de la matemática como la geometría computacional y algebraica, y en aplicaciones como la ingeniería civil y mecánica, la robótica, la física de los materiales, entre otras. En la literatura se han diferenciado tres tipos: frameworks de barra-vínculo (*bar-joint*) que describen un conjunto de barras rígidas vinculadas a través de puntos de unión que permiten la rotación relativa; los frameworks cuerpo-barra (*body-bar*) compuestos de un conjunto de cuerpos conectados a partir de barras rígidas; y los frameworks cuerpo-bisagra (*body-hinge*) donde los vínculos permiten únicamente la rotación relativa de los cuerpos respecto de su eje. Los últimos dos tipos de frameworks pueden ser reemplazados por frameworks del primer tipo, sustituyendo los cuerpos por frameworks barra-vínculo rígidos, propiedad que analizaremos en detalle en la siguiente sección. Esto hace que los frameworks de barra-vínculo sean los más estudiados en la literatura, y, por lo tanto, los considerados en esta tesis. Un framework barra-vínculo queda determinado por el par (\mathcal{G}, p) , donde los enlaces del grafo corresponden a las barras, y los vértices a los extremos de las mismas (Fig. 2.1), y $p: \mathcal{V} \rightarrow \mathbb{R}^d$ es la función que determina la realización espacial del grafo, asignando una posición a cada vértice.

2.2.1. Rigidez de Frameworks

La Teoría de Rigidez se ha establecido como una de las áreas de más intensa investigación en el estudio de los frameworks [90, 91, 92, 93, 94]. La misma parte del siguiente problema fundamental. Considere un framework (\mathcal{G}, p) de v nodos y e enlaces, el cual tiene asociada la función $f : \mathbb{R}^{dv} \rightarrow \mathbb{R}^e$ tal que $f(p) = [\cdots \|p_i - p_j\| \cdots]^T$ contiene las distancias asociadas a cada enlace. La teoría de rigidez de frameworks estudia la preimagen de $f(p)$, es decir, el conjunto de realizaciones espaciales que inducen las mismas distancias que p , y la influencia del grafo \mathcal{G} en dicho conjunto.

Rigidez Global

Es fácil observar que la preimagen de $f(p)$ no es finita debido a que incluye a toda realización que difiera de p en una transformación euclídea. Es decir, $f^{-1}(f(p))$ contiene al conjunto de configuraciones congruentes a p , que se define como

$$\mathcal{C}(p) := \{[E(p_1)^T \cdots E(p_v)^T]^T : E \in \mathbb{E}(d)\}. \quad (2.4)$$

La rigidez de un framework queda determinada por el tipo de inclusión existente entre estos conjuntos.

Definición 2.2.1. *Se dice que un framework (\mathcal{G}, p) es **globalmente rígido** si $f^{-1}(f(p)) = \mathcal{C}(p)$.*

Conocido el conjunto de distancias, un framework globalmente rígido es unívocamente localizable módulo transformaciones euclídeas. Si bien esto puede ser deseable, resulta una condición muy exigente en muchos casos prácticos. Es por ello que, en muchas situaciones, es suficiente considerar una condición de rigidez en las cercanías del punto p , como se define a continuación.

Rigidez en un Entorno

Definición 2.2.2. *Se dice que un framework (\mathcal{G}, p) es **rígido en un entorno**, o simplemente **rígido**, si existe un entorno abierto \mathcal{U} de p tal que $f^{-1}(f(p)) \cap \mathcal{U} = \mathcal{C}(p) \cap \mathcal{U}$. En caso contrario, se denomina **flexible**.*

Un framework rígido (pero no globalmente rígido) no es unívocamente localizable dadas las distancias entre nodos adyacentes. Sin embargo, existe un conjunto finito de configuraciones no congruentes a p en $f^{-1}(f(p))$. Por último, si un framework es flexible, entonces existe un continuo de configuraciones no congruentes a p en $f^{-1}(f(p))$. En la Fig. 2.2 se dan ejemplos de frameworks globalmente rígidos, rígidos y flexibles, que permiten comparar estos conceptos. Fig. 2.2a muestra un framework globalmente rígido en el plano. Si se elimina el enlace entre los nodos 1 y 4, se obtiene el framework de la Fig. 2.2b, el cual es rígido pero no globalmente rígido. Para observar esto, notar que existe una configuración $p' \in f^{-1}(f(p))$ obtenida de “reflejar” el nodo 4, sin modificar las distancias de los enlaces. Sin embargo, no existe una trayectoria continua de p a p' que preserve dichas distancias. En la Fig. 2.2c se observa el framework obtenido de eliminar el enlace entre 3 y 5 lo que hace que el mismo se vuelva flexible. Se muestra una configuración p'' que preserve las distancias, y a diferencia del caso anterior, la misma se obtiene de una “deformación continua” del framework.

Al momento de escribir este trabajo no se conocen algoritmos eficientes (clase P) que permitan determinar si un framework es rígido o globalmente rígido [95]. Probar la existencia o inexistencia de dichos algoritmos sigue siendo un problema abierto. El problema se vuelve tratable si consideramos frameworks en posición genérica, concepto de gran importancia que será definido en la Sección 2.2.2. Para frameworks genéricos, la rigidez se puede determinar por medio de una aproximación de primer orden al conjunto $f^{-1}(f(p))$, que da lugar al concepto de rigidez infinitesimal.

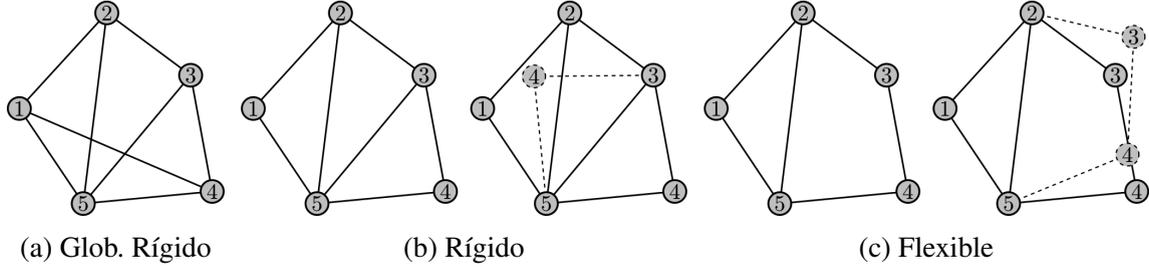


Figura 2.2: Un conjunto de vértices espacialmente distribuidos y un conjunto de enlaces determinados por aquellas distancias que deben preservarse. El conjunto de distancias conocidas determina la rigidez del framework.

Rigidez Infinitesimal

Considere un framework (\mathcal{G}, p) , y el vector de posiciones $\mathbf{p} = [p_1^T \cdots p_v^T]^T$. Se define la Matriz de Rigidez como la matriz jacobiana de f ,

$$\mathbf{R}(p) := \frac{\partial f}{\partial \mathbf{p}}(p) \in \mathbb{R}^{e \times dv}. \quad (2.5)$$

Cada fila de la misma está asociado a un enlace $e_k = \{i, j\}$, tal que

$$\mathbf{R}(p) = \begin{bmatrix} & 1 & & i & & j & & v \\ & 0 & \cdots & \frac{(p_i - p_j)^T}{\|p_i - p_j\|} & \cdots & \frac{(p_j - p_i)^T}{\|p_j - p_i\|} & 0 & \cdots \end{bmatrix} \mathbf{e}_k \quad (2.6)$$

Lema 2.2.1. *La matriz de rigidez posee las siguientes propiedades:*

1. *Invarianza a escala:* para todo $\alpha > 0$, $\mathbf{R}(\alpha p) = \mathbf{R}(p)$.
2. *Invarianza a traslación:* para todo $b \in \mathbb{R}^d$, $\mathbf{R}(p + b) = \mathbf{R}(p)$

3. *Similaridad ante transformaciones ortogonales: para todo $Q \in \mathbb{O}(d)$,*

$$\mathbf{R}(Qp) = \mathbf{R}(p)(I_v \otimes Q)^T$$

La matriz de rigidez es de gran importancia, ya que el espacio nulo de la misma está compuesto por aquellas velocidades asociadas a los vértices que, infinitesimalmente, no producen perturbaciones en el conjunto de distancias relativas. Se define $T(p)$, el espacio tangente a $\mathcal{C}(p)$ en el punto p , denominado **subespacio trivial**, el cual está conformado por las velocidades de roto-traslación del framework. Un sistema de generadores de este subespacio está dado por

$$\{\mathbf{e}_k : 1 \leq k \leq d\} \cup \{\mathbf{A}_{kl}\mathbf{p} : 1 \leq k < l \leq d\}, \quad (2.7)$$

donde $\mathbf{e}_k := \mathbf{1}_v \otimes e_k$ representa velocidades de traslación, y $\mathbf{A}_{kl} := I_v \otimes (E_{kl} - E_{lk})$ y $\mathbf{p} = [p_1^T \cdots p_v^T]^T$ constituyen las velocidades de rotación. Luego, se sigue que

$$T(p) \subseteq \text{Nul}(\mathbf{R}(p)). \quad (2.8)$$

Definición 2.2.3. *Se dice que (\mathcal{G}, p) es infinitesimalmente rígido si*

$$T(p) = \text{Nul}(\mathbf{R}(p)). \quad (2.9)$$

En caso contrario, decimos que (\mathcal{G}, p) es infinitesimalmente flexible.

Un framework infinitesimal rígido es tal que las únicas velocidades asociadas a los vértices que no perturban las distancias establecidas por los enlaces, son las velocidades de roto-traslación. Para evaluar la rigidez infinitesimal de un framework, es de suma importancia el Teorema 2.2.1 presentado a continuación, inicialmente propuesto por [90] y

luego extendido por [96, 97] con la introducción del autovalor de rigidez. Para ello se define la matriz $\mathbf{L}(p) := \mathbf{R}(p)^T \mathbf{R}(p)$ usualmente conocida Matriz Simétrica de Rigidez, denominada Laplaciano de Rigidez en esta tesis por su analogía con el laplaciano de un grafo. El laplaciano de rigidez es una matriz simétrica semi-definida positiva, que posee una estructura de $v \times v$ bloques de tamaño $d \times d$, cada uno asociado a un par de vértices. El mismo puede construirse a partir de una sumatoria de e términos $\mathbf{L} = \sum_{i \sim j} \ell_{ij}$, cada uno correspondiente a un enlace

$$\ell_{ij}(p) = \begin{bmatrix} & & i & & j & & \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ 0 & \cdots & \delta_{ij} \delta_{ij}^T & \cdots & -\delta_{ij} \delta_{ij}^T & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & -\delta_{ij} \delta_{ij}^T & \cdots & \delta_{ij} \delta_{ij}^T & \cdots & 0 \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 \end{bmatrix} \begin{matrix} \\ \\ i \\ \\ j \\ \\ \end{matrix} \quad (2.10)$$

donde

$$\delta_{ij} = \begin{cases} \frac{p_i - p_j}{\|p_i - p_j\|}, & \text{si } p_i \neq p_j, \\ 0, & \text{caso contrario.} \end{cases} \quad (2.11)$$

Definición 2.2.4. Sea $t(p) := \dim(\mathbf{T}(p))$, al autovalor $\lambda_{t(p)+1}(\mathbf{L}(p))$ se lo denomina **autovalor de rigidez**.

Teorema 2.2.1 (En [90, 97]). *Las siguientes proposiciones son equivalentes:*

1. *El framework (\mathcal{G}, p) es infinitesimalmente rígido,*
2. $\text{rg}(\mathbf{R}(p)) = dv - t(p),$
3. $\lambda_{t(p)+1}(\mathbf{L}(p)) > 0.$

El aporte de este teorema es que permite determinar la rigidez infinitesimal de un framework a partir de algoritmos para computar el rango o los autovalores de una matriz. Más aún, $\lambda_{t(p)+1}$ proporciona una medida continua de la rigidez, que permite comparar distintos frameworks en función de su grado de rigidez. Sin embargo, esto requiere el conocimiento de $t(p)$, la dimensión de la variedad de configuraciones congruentes, cuya deducción se puede encontrar en [91]. Allí se define $s := \dim(\text{affsp}\{p_1, \dots, p_v\})$ y se observa que $s \leq \min\{v-1, d\}$. Luego, se muestra que

$$t(p) = \frac{(s+1)(2d-s)}{2}, \quad (2.12)$$

de donde se deduce que $t(p) \leq \binom{d+1}{2}$, y que la igualdad se cumple si y solo si $s = d-1$ o $s = d$.

Lema 2.2.2 (En [91]). *Si (\mathcal{G}, p) es infinitesimalmente rígido, entonces $s = \min\{v-1, d\}$.*

Corolario 2.2.1. *Si (\mathcal{G}, p) es infinitesimalmente rígido y $v \leq d+1$, entonces \mathcal{G} es completo.*

Corolario 2.2.2. *Si (\mathcal{G}, p) es infinitesimalmente rígido, entonces la cantidad de enlaces*

$$e \geq \begin{cases} dv - \frac{v(2d-v+1)}{2}, & v \leq d, \\ dv - \binom{d+1}{2}, & v \geq d+1, \end{cases}$$

Si se cumple la igualdad de la expresión anterior, decimos que el grafo es mínimamente rígido.

2.2.2. Rigidez Genérica

Como se ha presentado en las secciones anteriores, tanto la rigidez como su versión infinitesimal son propiedades de los frameworks. Ambas nociones, si bien íntimamente

relacionadas, no son equivalentes para cualquier realización espacial. Para ver esto, se parte de que

$$T(p) \subseteq F(p) \subseteq \text{Nul}(\mathbf{R}(p)), \quad (2.13)$$

donde $F(p)$ es el espacio tangente a $f^{-1}(f(p))$ en el punto p . La primera inclusión se vuelve una igualdad si y solo si el framework es rígido. La segunda, si y solo si todo vector en $\text{Nul}(\mathbf{R}(p))$ es tangente a $f^{-1}(f(p))$. Cuando esto sucede, el espacio nulo de $\mathbf{R}(p)$ es mínimo, y se dice que la configuración es genérica.

Definición 2.2.5. *Un framework (\mathcal{G}, p) está en **posición genérica** si p maximiza el rango de la matriz de rigidez, es decir, $\text{rg}(\mathbf{R}(q)) \leq \text{rg}(\mathbf{R}(p))$ para toda realización q .*

El siguiente teorema precisa la relación entre los dos tipos de rigidez analizados.

Teorema 2.2.2 (En [91]). *Un framework (\mathcal{G}, p) es infinitesimalmente rígido si y solo si es rígido y está en posición genérica.*

Se puede demostrar que las posiciones genéricas de un grafo forman un conjunto abierto y denso en \mathbb{R}^{dv} [92]. Por lo tanto, ambas nociones de rigidez coinciden en casi todo punto del espacio de realizaciones. Más aún, se deduce que si un grafo tiene una realización que es infinitesimalmente rígida, entonces la misma es genérica, puesto que maximiza el rango de la matriz de rigidez. Entonces, para toda realización genérica, el framework resultante es infinitesimalmente rígido. Por lo tanto, es propicio establecer a la rigidez como una propiedad genérica del grafo subyacente, es decir, que se cumple (o no) para casi toda realización espacial.

Definición 2.2.6. *Un grafo \mathcal{G} es **genéricamente rígido** en \mathbb{R}^d si (\mathcal{G}, p) es rígido para toda realización genérica p en dicho espacio.*

Es importante notar que la rigidez genérica de un grafo depende de la dimensión en la cual se realiza. Por ejemplo, el grafo de la Fig. 2.2b es rígido en \mathbb{R}^2 , pero no en \mathbb{R}^3 .

Determinar la rigidez genérica de un grafo mediante métodos puramente combinatorios, es un problema abierto para $d \geq 3$ [98]. Alternativamente, el siguiente teorema junto con el Teorema 2.2.1 permiten evaluar la rigidez genérica, para todo d , por medio de métodos matriciales.

Teorema 2.2.3. *Un grafo \mathcal{G} es genéricamente rígido en \mathbb{R}^d si y solo si existe al menos una realización tal que (\mathcal{G}, p) que es infinitesimalmente rígido.*

Conectividad Algebraica Generalizada

A partir de lo presentado en esta sección, resulta interesante emplear un parámetro del grafo que permita postular, en términos cuantitativos, la rigidez genérica. Para esto, primero observar que el laplaciano de rigidez de un framework (2.10) es equivalente al laplaciano del grafo asociado, para cualquier realización $p: \mathcal{V} \rightarrow \mathbb{R}^1$. Dado que, además, cuando $d = 1$ se sigue que $t(p) = 1$, por lo tanto, el autovalor de rigidez coincide con la conectividad algebraica. Esto indica que la rigidez infinitesimal de un framework (\mathcal{G}, p) en \mathbb{R}^1 es equivalente a la conectividad algebraica del grafo subyacente. Por lo tanto, es natural extender esta idea a mayores dimensiones [94].

Definición 2.2.7. *La conectividad algebraica d -dimensional de un grafo \mathcal{G} se define como*

$$a_d(\mathcal{G}) := \sup\{\lambda_{t(p)+1}(\mathbf{L}(p)) : p: \mathcal{V} \rightarrow \mathbb{R}^d\}. \quad (2.14)$$

De esta definición y del Teorema 2.2.3 se deduce el siguiente teorema.

Teorema 2.2.4. *\mathcal{G} es genéricamente rígido en \mathbb{R}^d si y solo si $a_d(\mathcal{G}) > 0$.*

Este teorema es poderoso, ya que permite no solo determinar la rigidez genérica de un grafo, sino que además proporciona una medida de rigidez que permite comparar dicha propiedad para distintos grafos. No obstante, el cómputo de $a_d(\mathcal{G})$ es un problema abierto para $d \geq 2$.

2.3. Frameworks Generalizados

2.3.1. Definición

Debido a las ventajas de poder estudiar a los conjuntos de robots interconectados a través de formalismos matemáticos como los frameworks, es deseable contar con una forma de modelar sistemas multirobot con cualquier tipo de interrelación. La idea de extender los frameworks a otras funciones fue explorada por [67, 68] donde los agentes no miden distancia, sino dirección relativa (*bearing*) con respecto a sus vecinos. En dichos trabajos se puede ver la aparición de una estructura algebraica análoga al caso de distancia, lo que permite extender conceptos y herramientas a este nuevo escenario. Por ejemplo, los autores muestran que sobre los frameworks de dirección relativa se puede definir la rigidez, asociada a la posibilidad de estimar las posiciones de los agentes módulo una traslación y un factor de escala, operaciones que no afectan las direcciones relativas entre robots. Además, la rigidez de dirección relativa se puede evaluar mediante las matrices de rigidez y laplaciana definidas en tal contexto. Esto hace que sea muy atractivo poder generalizar esto a otras relaciones entre robots, que sean: (i) híbridas, es decir, que permitan contemplar más de un tipo de función en simultáneo (distancia, dirección); (ii) que no sean necesariamente binarias, es decir, que permitan contemplar interrelaciones entre más de dos agentes (centroide de un subconjunto), o de uno de forma individual (posición absoluta).

Para lograr estos objetivos, en primer lugar debemos extender los conceptos de grafos para poder modelizar relaciones más complejas entre un conjunto de nodos. Para ello empleamos el concepto de hipergrafos. En sistemas multirobot, los hipergrafos pueden ser empleados para incluir en el modelo funciones de un único agente, o de un *cluster* de múltiples robots. Por ejemplo, en la localización multirobot, algunos agentes pueden contar con mediciones absolutas de su posición provistas por un sistema de posicionamiento

global (GPS), lo cual se incorpora al hipergrafo por medio de un lazo propio; mientras que un subconjunto de agentes puede tener la capacidad de estimar su centroide, lo que se modeliza con un hiperenlace que cubre a dicho conjunto.

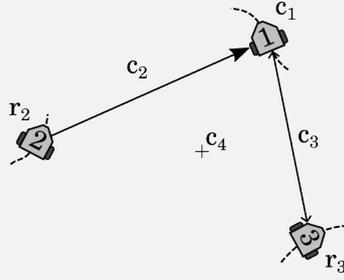
Un hipergrafo $\mathcal{H} = (\mathcal{V}, \mathcal{B})$ está compuesto de un conjunto de vértices \mathcal{V} y un conjunto $\mathcal{B} = \{b_1, \dots, b_b\}$ de subconjuntos de \mathcal{V} no vacíos, que llamamos hiperenlaces o cuerpos. Con esta definición, se permite incorporar al esquema lazos propios ($|b| = 1$) e interacciones entre múltiples nodos ($|b| > 2$). Un framework generalizado, entonces, queda definido por la tripla (\mathcal{H}, r, f) donde $r : \mathcal{V} \rightarrow \mathbb{R}^{d_i}$ asigna un estado a cada robot (típicamente posición, orientación, etc.), y $f(r) = [f_1(r)^T \cdots f_b(r)^T]^T$ donde f_k es una función diferenciable asociada al cuerpo b_k . La matriz de incidencia $\mathbf{J}_f(r)$ del framework es la matriz de derivadas parciales de f respecto de $\mathbf{r} = [r_1^T \cdots r_v^T]^T$, tal que

$$\mathbf{J}_f(r)[k, i] = \frac{\partial f_k(r)}{\partial r_i} \quad (2.15)$$

es el bloque asociado al cuerpo k y al robot i . El Laplaciano de un framework es la matriz simétrica semi-definida positiva definida como $\mathbf{L}_f(r) = \mathbf{J}_f(r)^T \mathbf{J}_f(r)$. La misma está compuesta por $v \times v$ bloques de tamaño $d \times d$, cada uno asociado a un par de vértices. El bloque $\mathbf{L}_f(r)[i, j]$ con $i \neq j$ es distinto de cero solo si existe un cuerpo que contiene a ambos vértices. En el Ejemplo 1 se muestra la definición de un framework generalizado compuesto de 3 robots móviles. En ella se construye la matriz de incidencia del framework, donde puede observarse subyacente la topología del hipergrafo.

Ejemplo 1: Frameworks Generalizados

Considere una formación de 3 vehículos terrestres con $r_i = [p_i^T \ \theta_i]^T$ donde θ_i representa la orientación de los mismos respecto de un sistema de referencia común, como indica la imagen:



Se definen los cuerpos $\mathcal{B} = \{b_1 = \{1\}, b_2 = \{1, 2\}, b_3 = \{1, 3\}, b_4 = \{1, 2, 3\}\}$ tales que

$$c_1 := f_1(r_1) = r_1$$

$$c_2 := f_2(r_1, r_2) = r_1 - r_2$$

$$c_3 := f_3(r_1, r_3) = \|r_1 - r_3\|$$

$$c_4 := f_4(r_2, r_3) = (r_1 + r_2 + r_3)/3$$

La matriz de incidencia toma la forma

$$\mathbf{J}_f(r) = \begin{bmatrix} & 1 & 2 & 3 \\ I_3 & 0 & 0 & \\ I_3 & -I_3 & 0 & \\ (r_1 - r_3)^T/c_3 & 0 & (r_3 - r_1)^T/c_3 & \\ I_3/3 & I_3/3 & I_3/3 & \end{bmatrix} \begin{matrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{matrix}$$

2.3.2. Control de Formaciones en el Espacio del Cluster

Usualmente, en las aplicaciones donde se realizan tareas cooperativas con múltiples robots, el estado de la formación se representa con un vector que agrupa las variables de estados de cada vehículo (e.g., posición, orientación). Sin embargo, en muchos casos resulta beneficioso realizar un cambio de variables, transformando el vector de estados de la formación a otro cuyas variables representan magnitudes geométricas de la formación como un cuerpo articulado, abstrayendo el análisis de los estados individuales de cada robot. El *Cluster Space Control* (CSC) [30, 23] es una estrategia de coordinación de formaciones de robots que permite seleccionar un conjunto de variables de estado, adecuadas para una determinada aplicación, y a partir de ellas realizar la especificación, el control y la supervisión. Al espacio de vectores cuyas coordenadas representan estados individuales de los robots lo llamamos *espacio de los robots*, que incluye típicamente posiciones y orientaciones. Al espacio de vectores que contienen los estados de la formación como un *cluster* o mecanismo virtual, lo llamamos *espacio del cluster*, formado por variables geométricas de la formación, como puede ser centroide y orientación de la formación, distancias relativas entre robots o respecto al centroide, etc.

La técnica de control en espacio del cluster (CSC) se basa en esta idea para diseñar una ley de control que opere la formación en el espacio del cluster, es decir, mediante la especificación y seguimiento de trayectorias en dicho espacio. En consecuencia, se mejora la usabilidad al ofrecer un nivel de abstracción de control por encima de los robots, y relacionando la operación directamente con la tarea que se quiere llevar a cabo. Esta técnica ha sido implementada de forma exitosa en el control de formaciones multirobot en diversos dominios como lo son terrestre, acuático y aéreo [99, 100, 101, 102, 24]. Debido a que la coordinación y el control de la formación sucede en el espacio del cluster, pero la estimación de las variables y el comando de los vehículos se implementan en el espacio de los robots, es necesario contar con un conjunto formal de transformaciones

cinemáticas que relacione ambos espacios.

Formulación general

Dado un sistema de múltiples robots modelado por medio de un framework generalizado, (\mathcal{H}, r, f) donde f es un difeomorfismo que relaciona ambos espacios. La técnica CSC aprovecha la posibilidad de representar el estado del sistema de forma alternativa mediante

$$\mathbf{c} := f(r) = [f_1(r)^T \cdots f_b(r)^T]^T, \quad (2.16)$$

denominado vector de estados en el espacio cluster. El esquema consiste en convertir posición y velocidad del espacio robot $\mathbf{r}, \dot{\mathbf{r}}$ al espacio del cluster $\mathbf{c}, \dot{\mathbf{c}}$, computar la acción de control en dicho espacio $\mathbf{u}_c = \kappa(\mathbf{c}, \dot{\mathbf{c}})$, y luego convertir esta acción al espacio de los robots. Las transformaciones de velocidades entre ambos espacios se llevan a cabo a través de las ecuaciones cinemáticas

$$\dot{\mathbf{c}} = \mathbf{J}_f(r)\dot{\mathbf{r}}, \quad (2.17)$$

$$\dot{\mathbf{r}} = \mathbf{J}_f^{-1}(c)\dot{\mathbf{c}} \quad (2.18)$$

donde \mathbf{J}_f denota la matriz de incidencia del framework. En Fig. 2.3 se muestra el diagrama en bloques que ilustra esta arquitectura de control.

2.4. Resumen

En este capítulo, presentamos los conceptos fundamentales de la teoría de frameworks, necesarios para comprender su aplicación en el control de formaciones multirobot. En primer lugar, dimos un resumen de las ideas más importantes de los grafos, herramienta matemática por excelencia para modelar interacciones entre pares de agentes. En segundo

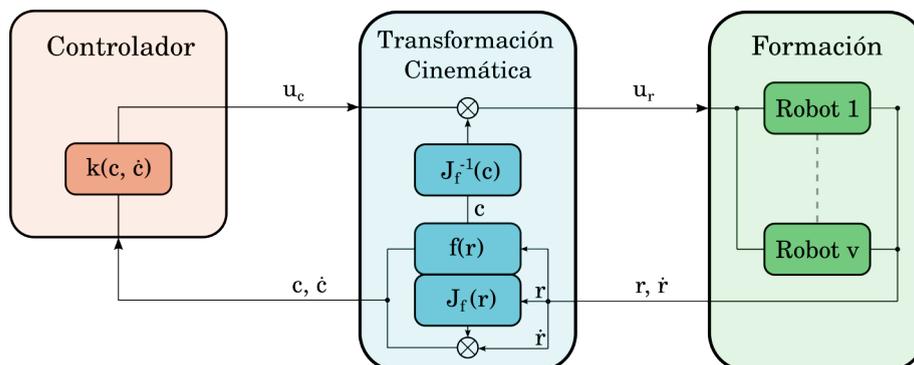


Figura 2.3: Arquitectura del Control en el Espacio del Cluster.

lugar, presentamos el problema de rigidez de frameworks, área que cuenta con una larga tradición y que tiene aplicaciones en un sinnúmero de ramas de la ingeniería y también de otras disciplinas. Vimos que en el caso de las redes multirobot, la rigidez es la propiedad clave que determina si es posible tanto la localización como el control de vehículos a partir de las distancias relativas entre los mismos. Estos conceptos serán utilizados en el esquema de rigidez por subframeworks propuesto en el Capítulo 4 y aplicado al control en el Capítulo 5. A su vez, discutimos que la rigidez es en verdad genérica, en el sentido de que es una propiedad que se puede asignar a los grafos subyacentes, sin distinción de las realizaciones espaciales de los frameworks. Esto posibilita mejorar el estudio teórico de la rigidez, al relacionarla directamente con las propiedades combinatorias de los grafos, lo que permite reutilizar el vasto conocimiento generado para estos últimos.

Luego, propusimos una forma de extender las ideas de frameworks, presentando los frameworks generalizados, herramientas matemáticas que permiten modelar cualquier tipo de interacción entre agentes. Esta idea, está íntimamente relacionada con la técnica de control en el espacio del cluster CSC, la cual será empleada en el contexto de cinematografía aérea autónoma en el Capítulo 6.

Capítulo 3

Conectividad Algebraica Generalizada

En este capítulo presentamos análisis, propiedades y nuevos resultados que extienden la teoría algebraica de rigidez introducida en el Capítulo 2. Se hace especial énfasis en el comportamiento de la conectividad algebraica generalizada en distintas dimensiones, y su relación con propiedades invariantes de un grafo, en particular con el diámetro.

3.1. Preliminares

Por simplicidad en la notación, cuando sea conveniente, utilizaremos $\lambda_k(\mathcal{G}, p)$ y $\Lambda(\mathcal{G}, p)$ para denotar, respectivamente, el k -ésimo autovalor y el espectro del laplaciano de rigidez de un framework. Por simplicidad, y sin pérdida de generalidad, asumimos que el centroide del framework coincide con el origen ($\sum_{i=1}^v p_i = 0$)

Es de utilidad expresar los autovalores y autovectores del laplaciano de rigidez de un framework (\mathcal{G}, p) como

$$\begin{aligned}\lambda_1 = \dots = \lambda_{t(p)} = 0, \quad \text{linsp}\{\mathbf{u}_1, \dots, \mathbf{u}_{t(p)}\} = \mathbf{T}(p) \\ \lambda_k = \min\{\mathbf{x}^T \mathbf{L} \mathbf{x} : \|\mathbf{x}\| = 1, \mathbf{x} \perp \mathcal{U}_{k-1}\} \quad t(p) + 1 \leq k \leq dv\end{aligned}$$

con $\mathcal{U}_k = \{\mathbf{u}_1, \dots, \mathbf{u}_k\}$. Luego, la forma cuadrática asociada

$$\mathbf{x}^T \mathbf{L} \mathbf{x} = \sum_{i \sim j} (\delta_{ij}^T (x_i - x_j))^2, \quad \delta_{ij} = \begin{cases} \frac{p_i - p_j}{\|p_i - p_j\|}, & \text{si } p_i \neq p_j, \\ 0, & \text{caso contrario.} \end{cases} \quad (3.1)$$

resulta de gran importancia para analizar la rigidez de un framework. Esta seminorma cuantifica la deformación infinitesimal en las distancias de los enlaces inducida por las velocidades x_i de los vértices [96]. Un grafo rígido, pero con $\lambda_{t(p)+1} \approx 0$, indica que es posible asignar perturbaciones a las posiciones de los vértices $\mathbf{x} = [x_1^T \cdots x_v^T]^T$, ortogonales a $\mathbf{T}(p)$, tales que la variación que inducen en las distancias $\|p_i + x_i - (p_j + x_j)\| - \|p_i - p_j\| \approx 0$ para todo enlace.

Analizar la rigidez de un framework por medio de las condiciones del Teorema 2.2.1 puede simplificarse si se obtiene una condición equivalente, donde el índice del autovalor a evaluar sea independiente de p . Primero, presentamos el siguiente lema, que es de utilidad en la demostración del Teorema 3.1.1.

Lema 3.1.1. *El laplaciano de rigidez de un framework d -dimensional con v vértices tiene al menos $N := \binom{s+1}{2} + (d-s)v$ autovalores nulos, donde s es la dimensión de $\mathbf{S} := \text{linsp}\{p_1, \dots, p_v\}$.*

Demostración. Sea $\mathcal{B} = \{b_1, \dots, b_s\} \cup \{b_{s+1}, \dots, b_d\}$ la unión de una base de \mathbf{S} y otra de \mathbf{S}^\perp , respectivamente. Se observa que el framework tiene $\binom{s+1}{2}$ velocidades de roto-traslación sobre \mathbf{S} , y $(d-s)v$ velocidades individuales en \mathbf{S}^\perp : $u_{i,k} = e_i \otimes b_k$ con $i = 1, \dots, v$, $k = s+1, \dots, d$ donde e, b son los vectores de las bases estándar y \mathcal{B} . \square

Teorema 3.1.1. *Sea (\mathcal{G}, p) un framework d -dimensional con v vértices. Sean*

$$R := \frac{(d'+1)(2d-d')}{2} + 1, \quad t(p) = \frac{(s+1)(2d-s)}{2}$$

con $d' := \min\{v-1, d\}$ y $s \leq d'$ la dimensión del subespacio lineal S generado por $\{p_1, \dots, p_v\}$. Se cumple que $\lambda_R(\mathcal{G}, p) = \lambda_{t(p)+1}(\mathcal{G}, p)$.

Demostración. Si $s = d'$ la demostración es inmediata. Consideremos ahora $s < d'$, lo cual implica $\lambda_{t(p)+1} = 0$. Por lo tanto, debe probarse que $\lambda_R = 0$. Por el Lema 3.1.1, sabemos que $\lambda_N = 0$. En consecuencia, resta mostrar que $R \leq N$:

$$\begin{aligned} N - (R - 1) &= dv + s \left(\frac{s+1}{2} - v \right) + \left(\frac{s}{2} - d \right) (d' + 1) \\ &> dv + d' \left(\frac{d'+1}{2} - v \right) + \left(\frac{s}{2} - d \right) (d' + 1) = (d - d')((v-1) - d') = 0. \end{aligned}$$

La desigualdad se demuestra como sigue partiendo de que $s < d' < v$

$$\begin{aligned} s \left(\frac{s+1}{2} - v \right) - d' \left(\frac{d'+1}{2} - v \right) &= (d' - s) \left(v - \frac{s+d'+1}{2} \right) \\ &> (d' - s) \left(v - \frac{2(v-1)+1}{2} \right) > 0, \end{aligned}$$

y la igualdad a cero de resulta por definición de d' . □

En la Tabla 3.1 se muestra el valor que toma el parámetro R en función de la cantidad de vértices y la dimensión del espacio.

	$v = 2$	$v = 3$	$v = 4$	$v = 5$	\dots
$d = 1$	2	2	2	2	\dots
$d = 2$	4	4	4	4	\dots
$d = 3$	6	7	7	7	\dots
$d = 4$	8	10	11	11	\dots
\vdots	\vdots	\vdots	\vdots	\vdots	\ddots

Tabla 3.1: Parámetro R como función de d y v .

A continuación presentamos otro resultado que permite obtener los autovalores de un framework contenido en un subespacio de \mathbb{R}^d , y que será de utilidad más adelante.

Lema 3.1.2. *Sea (\mathcal{G}, p) un framework d -dimensional con v vértices que generan un subespacio S de dimensión s . Entonces*

$$\Lambda(\mathcal{G}, p) = \{0^{(d-s)v}\} \cup \Lambda(\mathcal{G}, q)$$

donde $q: \mathcal{V} \rightarrow \mathbb{R}^s$ asigna las coordenadas de cada p_i en una base ortonormal de S .

Demostración. Sea $B \in \mathbb{R}^{d \times s}$ tal que sus columnas forman una base ortonormal de S , luego $q_i = B^T p_i$. Entonces el bloque asociado al enlace ij es

$$\begin{aligned} \mathbf{L}(p)[i, j] &= -\frac{(p_i - p_j)(p_i - p_j)^T}{\|p_i - p_j\|^2} \\ &= -B \frac{(q_i - q_j)(q_i - q_j)^T}{\|q_i - q_j\|^2} B^T \\ &= B \mathbf{L}(q)[i, j] B^T. \end{aligned}$$

Esto implica que $\mathbf{L}(p) = (I_v \otimes B) \mathbf{L}(q) (I_v \otimes B)^T$, lo cual completa la prueba. \square

3.2. Cotas

Como se presentó en la sección 2.2.2, el cómputo de la conectividad algebraica generalizada permite determinar la rigidez genérica de un grafo, independientemente de las realizaciones espaciales del mismo. Estudiar a la rigidez como una extensión geométrica de la teoría espectral de grafos resulta un abordaje prometedor, que permite aprovechar el vasto conocimiento adquirido por esta última a lo largo de los años. Sin embargo, obtener el valor de $a_d(\mathcal{G})$ es tanto o más complejo que decidir si el mismo es genéricamente rígido (condición $a_d > 0$), que a su vez es un problema abierto para $d \geq 3$. No se conoce, a la fecha, un algoritmo que permita calcular la conectividad algebraica generalizada para un grafo arbitrario. Es por ello, que encontrar cotas inferiores y superiores de dicho

parámetro es un área de intensa investigación.

3.2.1. Grafos Completos

A continuación presentamos los resultados más importantes para la conectividad algebraica generalizada de grafos completos.

Teorema 3.2.1 (En [103] y [104]). *Sean $d = 2$, $v \geq 3$ y (\mathcal{K}_v, p°) con $p^\circ \in \mathbb{R}^2$ un posicionamiento de los vértices tal que $\|p_i^\circ\| = 1$ para todo i , y $\sum_i p_i^\circ = 0$. Entonces*

$$\Lambda(\mathcal{K}_v, p^\circ) = \{0^{(3)}, \frac{v}{2}^{(2v-4)}, v^{(1)}\}.$$

Además, p° alcanza el máximo de (2.14), es decir $a_2(\mathcal{K}_v) = \lambda_4(\mathcal{K}_v, p^\circ) = \frac{v}{2}$.

Teorema 3.2.2 (En [104]). *Sea $d \geq 3$ y $(\mathcal{K}_{d+1}, p^\Delta)$ con $p^\Delta \in \mathbb{R}^d$ un posicionamiento de los vértices del grafo sobre los vértices de un d -símplex regular. Entonces*

$$\Lambda(\mathcal{K}_{d+1}, p^\Delta) = \{0^{(d(d+1)/2)}, 1^{((d+1)(d-2)/2)}, \frac{d+1}{2}^{(d)}, d+1^{(1)}\}.$$

Además, p^Δ alcanza el máximo de (2.14), es decir $a_d(\mathcal{K}_{d+1}) = \lambda_{\binom{d+1}{2}+1}(\mathcal{K}_{d+1}, p^\Delta) = 1$.

A continuación presentamos un resultado que permite completar el triángulo inferior de la Tabla 3.2, en la cual se muestra un resumen de los valores de la conectividad algebraica generalizada.

Teorema 3.2.3. *Sea \mathcal{G} un grafo con $d \geq v - 1$, entonces $a_d(\mathcal{G}) = a_{v-1}(\mathcal{G})$.*

Demostración. Si \mathcal{G} no es completo, $a_d(\mathcal{G}) = 0$ para todo $d \geq v - 1$ por Corolario 2.2.1.

Sea \mathcal{G} es completo, entonces por Teoremas 3.2.1 y 3.2.2, para $d = v - 1$

$$\lambda_{\binom{v}{2}+1}(\mathcal{K}_v, p^\Delta) = a_{v-1}(\mathcal{K}_v).$$

Consideremos ahora el caso $d \geq v$, y definimos $\hat{p}_i^\Delta \in \mathbb{R}^d$ tal que las primeras $v-1$ componentes coinciden con p_i^Δ y los restantes iguales a cero. Por Lema 3.1.2, como los vértices se encuentran contenidos en un subespacio S de dimensión $v-1$, entonces

$$\Lambda(\mathcal{K}_v, \hat{p}^\Delta) = \{0^{(dv-v(v-1))}\} \cup \Lambda(\mathcal{K}_v, p^\Delta).$$

Como $R-1 = dv - \frac{v(v-1)}{2}$ si $d \geq v$, se sigue que $\lambda_R(\mathcal{K}_v, \hat{p}^\Delta) = \lambda_{\binom{v}{2}+1}(\mathcal{K}_v, p^\Delta) = a_{v-1}(\mathcal{K}_v)$. Solo resta observar que \hat{p}^Δ maximiza el autovalor de rigidez en \mathbb{R}^{dv} puesto que maximiza dentro el subespacio S , y toda realización se encuentra contenida en dicho subespacio módulo transformaciones euclídeas. \square

Recientemente, fueron presentadas una cota inferior y otra superior, que ayuda a completar el conocimiento sobre $a_d(\mathcal{K}_v)$.

Teorema 3.2.4 (En [104] y [105]). *Sea $d \geq 3$ y $v \geq d+1$, entonces*

$$\frac{1}{2} \left\lfloor \frac{v}{d} \right\rfloor \leq a_d(\mathcal{K}_v) \leq \frac{2v}{3(d-1)} + \frac{1}{3}, \quad (3.2)$$

donde $\lfloor \cdot \rfloor$ es la función piso.

	\mathcal{K}_2	\mathcal{K}_3	\mathcal{K}_4	\mathcal{K}_5	\mathcal{K}_6	\dots
$d=1$	2	3	4	5	6	\dots
$d=2$	2	$\frac{3}{2}$	$\frac{4}{2}$	$\frac{5}{2}$	$\frac{6}{2}$	\dots
$d=3$	2	$\frac{3}{2}$	1	$[\frac{1}{2}, 2]$	$[1, \frac{7}{3}]$	\dots
$d=4$	2	$\frac{3}{2}$	1	1	$[\frac{1}{2}, \frac{5}{3}]$	\dots
$d=5$	2	$\frac{3}{2}$	1	1	1	\dots
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\ddots

Tabla 3.2: Conectividad Algebraica Generalizada para grafos completos.

3.2.2. Grafos Arbitrarios

Los resultados presentados para grafos completos son un primer paso importante en el estudio de la conectividad algebraica generalizada. Sin embargo, resulta relevante estudiar $a_d(\mathcal{G})$ para grafos arbitrarios. Una primera propiedad importante establece que este parámetro es no decreciente con el agregado de enlaces, lo cual se deduce de que el laplaciano de rigidez es una composición de matrices semi-definidas positivas (2.10).

Lema 3.2.1. *Sean $\mathcal{G}, \mathcal{G}^+$ dos grafos con el mismo conjunto de vértices, tales que $\mathcal{E} \subseteq \mathcal{E}^+$. Entonces para todo $d \geq 1$*

$$a_d(\mathcal{G}) \leq a_d(\mathcal{G}^+).$$

Este lema indica que el grafo completo \mathcal{K} maximiza la conectividad algebraica como función de \mathcal{G} , lo cual es útil para acotar por arriba este parámetro. No obstante, la brecha entre $a_d(\mathcal{G})$ y $a_d(\mathcal{K})$ puede ser muy grande. Por este motivo, y dado que existen múltiples técnicas para computar o acotar $a_1(\mathcal{G})$ [106], resulta interesante estudiar como evoluciona $a_d(\mathcal{G})$ con el aumento de d . Siguiendo esta idea, en [94, Teorema 4.2] se muestra que

$$a_1(\mathcal{G}) \geq a_2(\mathcal{G}), \tag{3.3}$$

es decir, el máximo grado de rigidez de un grafo en \mathbb{R}^2 está limitado por su grado de conectividad. No obstante, la técnica propuesta en dicho trabajo para obtener (3.3) no se extiende a mayores dimensiones. A continuación demostraremos que, utilizando herramientas espectrales, es posible extender esta desigualdad a cualquier dimensión $d \geq 3$.

En primer lugar destacamos el siguiente lema, que es de utilidad para probar el teorema principal de esta sección, Teorema 3.2.5.

Lema 3.2.2. *Sea (\mathcal{G}, p) un framework. Sea $W := \text{linsp}\{w\}$ con $\|w\| = 1$ un subespacio en \mathbb{R}^d , $u = [u_1 \cdots u_v]^T$ un vector no nulo en \mathbb{R}^v , $\mathbf{x} = u \otimes w$ una asignación de velocidades a*

los vértices en W . Entonces

$$\mathbf{x}^T \mathbf{L}(p) \mathbf{x} \leq \mathbf{x}^T (L \otimes ww^T) \mathbf{x}. \quad (3.4)$$

La igualdad ocurre si y solo si $p_i - p_j \in W$ y $p_i \neq p_j$ para todo $ij \in \mathcal{E}$.

Demostración. Sea δ_{ij} el vector definido como en (3.1). Como $w, x_1, \dots, x_v \in W$,

$$\begin{aligned} \mathbf{x}^T \mathbf{L}(p) \mathbf{x} &= \sum_{i \sim j} ((x_i - x_j)^T \delta_{ij})^2 \leq \sum_{i \sim j} ((x_i - x_j)^T w)^2 \\ &= \sum_{i \sim j} (u_i - u_j)^2 = u^T L u = \mathbf{x}^T (L \otimes ww^T) \mathbf{x}. \end{aligned}$$

La igualdad sucede si y solo si $\delta_{ij} = \pm w$ para todo enlace. □

Teorema 3.2.5. Para todo grafo \mathcal{G} , $d \geq 2$

$$a_1(\mathcal{G}) \geq a_d(\mathcal{G}). \quad (3.5)$$

Demostración. Sea v la cantidad vértices, y p una realización d -dimensional. Asumamos \mathcal{G} conexo, de lo contrario $a_1 = a_d = 0$. Definimos, para $1 \leq k \leq d$, el subespacio $E_k := \{[\alpha_1 e_k^T \cdots \alpha_v e_k^T]^T : \alpha_i \in \mathbb{R}\}$, de dimensión v . Se observa que $\mathbb{R}^{dv} = E_1 + \dots + E_d$ y que los mismos son mutuamente ortogonales. Luego, para toda matriz $Q \in \mathbb{O}(d)$

$$\lambda_R(\mathcal{G}, p) = \lambda_R(\mathcal{G}, Qp) = \min \left\{ \frac{\mathbf{x}^T \mathbf{L}(Qp) \mathbf{x}}{\mathbf{x}^T \mathbf{x}} : \mathbf{x} \in \mathbb{T}(Qp)^\perp \right\} \quad (3.6)$$

$$\leq \min \left\{ \frac{\mathbf{x}^T \mathbf{L}(Qp) \mathbf{x}}{\mathbf{x}^T \mathbf{x}} : \mathbf{x} \in E_1 \cap \mathbb{T}(Qp)^\perp \right\} \quad (3.7)$$

$$\leq \min \left\{ \frac{\mathbf{x}^T (L \otimes e_1 e_1^T) \mathbf{x}}{\mathbf{x}^T \mathbf{x}} : \mathbf{x} \in E_1 \cap \mathbb{T}(Qp)^\perp \right\} \quad (3.8)$$

$$=: \alpha(Q), \quad (3.9)$$

donde se usó el Lema 3.2.2. La elección de E_1 como subespacio al cual se restringen el vector \mathbf{x} es arbitraria, y podría elegirse cualquier subespacio E_k . Llamemos $A(Q) := E_1 \cap T(Qp)^\perp$, subespacio de búsqueda de (3.8). Por otro lado, recordamos que $a_1(\mathcal{G}) = \lambda_2(L)$ y

$$\lambda_2(L) = \lambda_{(d-1)v+2}(L \otimes e_1 e_1^T) = \min \left\{ \frac{\mathbf{x}^T (L \otimes e_1 e_1^T) \mathbf{x}}{\mathbf{x}^T \mathbf{x}} : \mathbf{x} \in \text{Nul}(L \otimes e_1 e_1^T)^\perp \right\} \quad (3.10)$$

donde $\mathbf{e}_1 := \mathbf{1}_v \otimes e_1$ representa una velocidad de traslación del framework en la dirección de e_1 . Ambas igualdades se siguen de que $\Lambda(e_1 e_1^T) = \{0^{(d-1)}, 1\}$, entonces $\Lambda(L \otimes e_1 e_1^T) = \{0^{((d-1)v)}, \lambda_1(L), \lambda_2(L), \dots, \lambda_v(L)\}$ por propiedad del producto de kronecker. Luego, notamos que $\text{Nul}(L \otimes e_1 e_1^T) = E_1^\perp + \text{linsp}\{\mathbf{e}_1\}$, y llamando $B := \text{Nul}(L \otimes e_1 e_1^T)^\perp$, se observa que $\alpha(Q)$ (3.9) y $\lambda_2(L)$ (3.10) corresponden a mínimos del mismo funcional, pero en subespacios de búsqueda distintos $A(Q)$ y B , respectivamente. En lo que sigue analizaremos cómo se relacionan estos dos subespacios de búsqueda y mostraremos que existe $Q^* \in \mathbb{O}(d)$ tal que $\alpha(Q^*) = \lambda_2(L)$, de donde se sigue (3.5).

Para nuestro análisis, es importante ver que $A(Q) \subset B$. Esto se sigue de que $\mathbf{e}_1 \in T(Qp)$, entonces $B^\perp = E_1^\perp + \text{linsp}\{\mathbf{e}_1\} \subset E_1^\perp + T(Qp) = A(Q)^\perp$. Esto, además, implica que $\alpha(Q) \geq \lambda_2(L)$ para toda Q . Para hallar Q^* , obtenemos primero un vector que minimiza (3.10). Sea $u \in \mathbb{R}^v$ el autovector asociado a $\lambda_2(L)$, el cual cumple que $\sum_i u_i = 0$ por definición. Luego, definimos $\mathbf{u} := u \otimes e_1$ el cual, por propiedades del producto de Kronecker, es un autovector asociado a $\lambda_{(d-1)v+2}(L \otimes e_1 e_1^T)$, por lo tanto, un vector que alcanza el mínimo del funcional en cuestión en B (3.10). Luego, el objetivo es mostrar que existe Q^* tal que $A(Q^*)$ contiene a \mathbf{u} (vector que no depende de Q), lo cual implica que los mínimos de $A(Q^*)$ y B coinciden, es decir, $\alpha(Q^*) = \lambda_2(L)$. Esto se ilustra en Fig. 3.1.

Para que $\mathbf{u} \in A(Q^*)$, debe cumplirse que \mathbf{u} sea ortogonal a $T(Q^*p)$, puesto que ya se cumple que $\mathbf{u} \in E_1$. Para ello, consideremos un sistema de generadores del subespacio

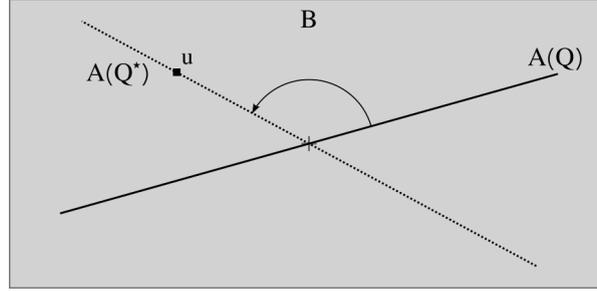


Figura 3.1: Esquema que ilustra cómo se elige la matriz Q^* de forma tal que $\mathbf{u} \in A(Q^*)$.

trivial $T(Q^*p)$

$$\{\mathbf{e}_k : 1 \leq k \leq d\} \cup \{\mathbf{A}_{kl}\mathbf{Q}^*\mathbf{p} : 1 \leq k < l \leq d\}, \quad (3.11)$$

donde $\mathbf{Q}^* := I_v \otimes Q^*$. Para la parte traslacional del subespacio trivial, la ortogonalidad está asegurada, dado que $\mathbf{u} \perp \mathbf{e}_1$ por definición, y $\mathbf{u} \perp \mathbf{e}_k$ para $2 \leq k \leq d$ porque $\mathbf{u} \in E_1$ y $\mathbf{e}_2, \dots, \mathbf{e}_d$ son ortogonales a E_1 . Para la parte rotacional, sucede que el vector $\mathbf{A}_{kl}\mathbf{Q}^*\mathbf{p} \in E_k + E_l$, y como $2 \leq l \leq d$, si $k > 1$ se cumple que \mathbf{u} es ortogonal a dicho vector. Entonces, para hallar \mathbf{Q}^* , es suficiente imponer la condición

$$\mathbf{u}^T \mathbf{A}_{1l}\mathbf{Q}^*\mathbf{p} = 0 \quad \text{para todo } 2 \leq l \leq d. \quad (3.12)$$

Entonces, como $\mathbf{u}^T = [u_1 e_1^T \cdots u_v e_1^T]$, reescribimos (3.12) como

$$\sum_{i=1}^v u_i e_i^T Q^* p_i = 0 \quad \text{para todo } 2 \leq l \leq d,$$

donde usamos que $e_1^T A_{1l} = e_l^T$. Entonces, si definimos la matriz M de $v \times d$ que contiene a las posiciones en sus filas, es decir, $M[i, \cdot] = p_i^T$. Y llamamos q_l a las filas de Q^* , es decir, $Q^*[l, \cdot] = q_l^T$, se sigue que $(Mq_l)[i] = e_i^T Q^* p_i$. Entonces, (3.12) es equivalente a

$$u^T M q_l = 0 \quad \text{para todo } 2 \leq l \leq d.$$

Como no hay condiciones sobre q_1 , elegimos $q_1 := \frac{M^T u}{\|M^T u\|}$ y se definen las filas restantes q_2, \dots, q_d para completar una base ortonormal de \mathbb{R}^d , de forma tal que $Q^* \in \mathbb{O}(d)$. Esto completa la demostración, pues la Q^* seleccionada satisface $\mathbf{u} \in A(Q^*)$. Si sucediera que $M^T u = 0$, entonces cualquier matriz ortogonal sirve como Q^* . \square

Este teorema representa un paso importante en el estudio de la conectividad algebraica generalizada para grafos arbitrarios. El mismo fue demostrado por [105] de forma independiente, usando un método alternativo, lo que muestra el interés de la comunidad en el tema. Además, (3.5) provee una cota ajustada, es decir que existe \mathcal{G} tal que $a_d(\mathcal{G}) = a_1(\mathcal{G})$ para todo $d \geq 1$.

Lema 3.2.3 (Ver Tabla 3.2). *El grafo completo de dos vértices, \mathcal{K}_2 , es tal que $a_d(\mathcal{K}_2) = a_1(\mathcal{K}_2) = 2$ para todo d .*

Esto responde la pregunta planteada en [105, Problema 8.4], donde se propone estudiar el valor de $\sup\{a_d(\mathcal{G})/a_1(\mathcal{G}) : \mathcal{G} \text{ es conexo}\}$, el cual resulta ser igual a 1. No obstante, caracterizar el conjunto de grafos tales que $a_d(\mathcal{G}) = a_1(\mathcal{G})$ para todo d sigue siendo un problema abierto. Entre otros problemas no resueltos, destacamos la monotonía de la conectividad algebraica generalizada como función de d , la cual se conjetura decreciente a partir de un gran número de simulaciones computacionales que los sugieren.

Conjetura 3.2.1. *Para todo grafo \mathcal{G} , $d \geq 1$*

$$a_d(\mathcal{G}) \geq a_{d+1}(\mathcal{G}). \quad (3.13)$$

Para completar el panorama del conocimiento sobre la conectividad algebraica generalizada, empleamos el siguiente teorema [107, Teorema 6].

Teorema 3.2.6 (En [107]). *Si \mathcal{G} es rígido en \mathbb{R}^{d+1} , entonces es globalmente rígido en \mathbb{R}^d .*

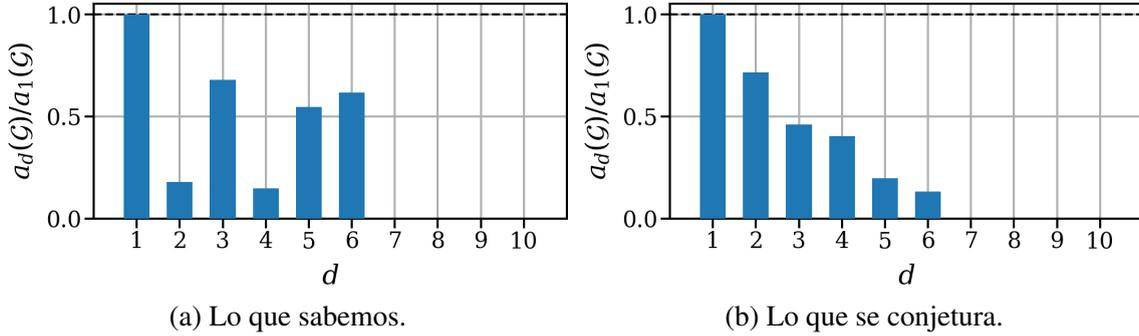


Figura 3.2: Panorama del estado del estudio de $a_d(\mathcal{G})$. Ejemplo basado en un grafo conexo, no completo, con $d_f = 7$.

Esto implica que si $a_{d+1}(\mathcal{G}) > 0$, entonces $a_d(\mathcal{G}) > 0$. Luego, los resultados desarrollados en esta sección se resumen el siguiente corolario, que se ilustra en la Fig. 3.2.

Corolario 3.2.1. *Para todo grafo \mathcal{G} no completo, existe d_f tal que*

1. Si $1 \leq d < d_f$, entonces $a_1(\mathcal{G}) \geq a_d(\mathcal{G}) > 0$.
2. Si $d \geq d_f$, entonces $a_d(\mathcal{G}) = 0$.

Más aún, se cumple que $1 \leq d_f \leq v - 1$.

3.3. Influencia del Diámetro del Grafo

En la sección anterior hemos probado que la conectividad algebraica de un grafo limita por arriba al grado de rigidez del mismo, en cualquier dimensión. Para estudiar este parámetro, es de utilidad escribirlo como

$$a_1(\mathcal{G}) = \min\{x^T Lx : \sum_i x_i = 0, \sum_i x_i^2 = 1\}.$$

Un grafo conexo, pero con $a_1 \approx 0$, indica que es posible asignar un escalar x_i a cada vértice con $\sum_i x_i = 0$ y $\sum_i x_i^2 = 1$ de forma tal que $(x_i - x_j)^2 \approx 0$ para todo par de nodos

adyacentes. Esto significa que $x \in \mathbb{R}^v$ determina dos conjuntos de vértices $\{i : x_i \geq 0\}$ y $\{i : x_i < 0\}$ tales que $x_i \approx x_j$ para dos vértices del mismo grupo. Esto equivale a una degradada conectividad, ya que la existencia de estos dos conjuntos puede considerarse como la emergencia de dos componentes conexos.

Dado un grafo con un diámetro elevado, es esperable un bajo valor de $a_1(\mathcal{G})$. Esto es debido a la existencia de al menos dos nodos periféricos p, q cuya distancia geodésica sea elevada, lo que hace posible definir una función x tal que $x_p < 0 < x_q$ y que varíe suavemente entre ambos vértices. Una completa caracterización de $a_1(\mathcal{G})$ en función del diámetro $D(\mathcal{G})$, sigue siendo un problema abierto. Sin embargo, se conoce la siguiente relación (ver [106])

$$\frac{4}{v \cdot D(\mathcal{G})} \leq a_1(\mathcal{G}) \leq 1 - \gamma + \frac{2}{D(\mathcal{G})}(1 + \gamma),$$

donde $\gamma = \frac{\sqrt{n_{\text{máx}} - 1}}{n_{\text{máx}}}$ es tal que $0 \leq \gamma \leq 0.5$, siendo $n_{\text{máx}}$ el máximo grado de \mathcal{G} , el cual puede considerarse independiente del diámetro. Esta relación, si bien útil, cuando $D(\mathcal{G}) \gg 1$ se llega a $a_1(\mathcal{G}) \leq 0.5$. Es decir, no establece si la conectividad algebraica converge a cero, lo cual es esperable cuando D crece indefinidamente. Para comprender mejor la relación entre la conectividad algebraica y el diámetro, presentamos el siguiente resultado. El mismo permite, en muchos casos, determinar el comportamiento límite de un grafo cuando el mismo crece indefinidamente, como se muestra en el Ejemplo 1 para grafos reticulados.

Teorema 3.3.1. *Sea \mathcal{G} un grafo con e enlaces y diámetro D , entonces*

$$a_1(\mathcal{G}) \leq \frac{12e}{D(D+1)(D+2)}. \quad (3.14)$$

Demostración. Sean p, q dos vértices de \mathcal{G} tales que $g(p, q) = D$, y $\pi_{p,q}$ un camino simple

de longitud D que conecta dichos vértices. Se define el vector $y = [y_1 \cdots y_v]^T$ donde $y_i = g(i, p)/D$ de donde se sigue que $0 \leq y_i \leq 1$. Ahora, sea $\bar{y} := \frac{1}{v} \sum_{i=1}^v y_i$ y definimos $x_i := y_i - \bar{y}$, es claro que $\sum_i x_i = 0$. Observar que, para todo enlace $i \sim j$,

$$(x_i - x_j)^2 = \frac{(g(i, p) - g(j, p))^2}{D^2} = \begin{cases} 0, & g(i, p) = g(j, p) \\ \frac{1}{D^2}, & \text{caso contrario} \end{cases}$$

Entonces, se cumple que

$$\begin{aligned} x^T Lx &= \sum_{i \sim j} (x_i - x_j)^2 \leq \frac{e}{D^2}, \\ x^T x &= \sum_{i=1}^v (y_i - \bar{y})^2 \geq \sum_{i \in \pi_{p,q}} (y_i - \bar{y})^2 \geq \sum_{i \in \pi_{p,q}} (y_i - \frac{1}{2})^2 \end{aligned}$$

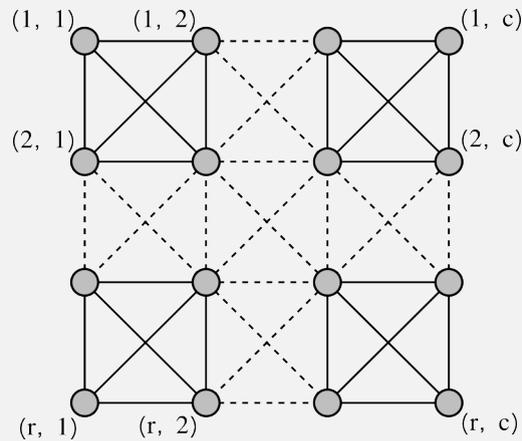
dado que $\frac{1}{2}$ es el centroide del conjunto $\{y_i : i \in \pi_{p,q}\}$. Luego,

$$x^T x \geq \sum_{i \in \pi_{p,q}} (y_i - \frac{1}{2})^2 = \sum_{d=0}^D \left(\frac{d}{D} - \frac{1}{2} \right)^2 = \frac{(D+1)(D+2)}{12D}$$

Por último, la prueba finaliza observando que $a_1(\mathcal{G}) \leq \frac{x^T Lx}{x^T x}$, como consecuencia de que $\sum_i x_i = 0$. \square

Ejemplo 1: Conectividad límite de grafos reticulados

Considere el grafo $\mathcal{G}(r, c)$ definido por $\mathcal{V} = \{(i, j) : 1 \leq i \leq r, 1 \leq j \leq c\}$ y $\mathcal{E} = \{(i, j), (k, l) : |i - k| \leq 1, |j - l| \leq 1\}$ como indica la imagen:



La cantidad de enlaces $e(r, c)$ se obtiene a partir del grado de los vértices

$$\begin{aligned} e(r, c) &= \frac{1}{2} \sum_{i=1}^v n(i) = \frac{1}{2} (4 \cdot 3 + (2r + 2c - 8) \cdot 5 + (rc - 2r - 2c + 4) \cdot 8) \\ &= 4rc - 3(r + c) + 2 \end{aligned}$$

También se puede observar que el diámetro es igual a

$$D(r, c) = g((1, 1), (r, c)) = \max\{r, c\},$$

por lo tanto $a_1(\mathcal{G})$ tiende a cero cuando $D(r, c) \rightarrow \infty$.

Como se mostró en el Teorema 3.2.5, la conectividad algebraica de un grafo es una cota superior para el autovalor de rigidez, para toda realización espacial de dicho grafo. Por lo tanto, el Teorema 3.3.1 también indica que la rigidez de frameworks se degrada a medida que aumenta su diámetro.

3.4. Resumen

En este capítulo se presentaron resultados teóricos relativos a las propiedades de la conectividad algebraica generalizada. Se hizo una revisión de los conocimientos existentes sobre la conectividad algebraica generalizada y se presentó un límite superior para la misma para grafos arbitrarios, el Teorema 3.2.5, el cual fue presentado en [69]. El mismo supone una contribución al estudio de esta propiedad fundamental, pero de reciente aparición, asociada a la rigidez genérica de los grafos. También se presentaron resultados relacionados con la influencia del diámetro en la conectividad algebraica generalizada, Teorema 3.3.1, presentado en [72]. Este ayuda a comprender cómo evoluciona esta propiedad a medida que aumenta la cantidad de vértices de un grafo, si el grado máximo es acotado, lo cual sucede en las redes de robots móviles. Esto será útil en el Capítulo 4 para comprender que la rigidez de los frameworks es una propiedad que no es práctica de utilizar cuando crece el tamaño framework, lo que da pie a presentar una nueva forma de entender la rigidez en función de subframeworks.

Capítulo 4

Rigidez Basada en Subframeworks

En el presente capítulo se presenta una formulación novedosa que permite descomponer la rigidez de un framework en función de un conjunto de subframeworks. Esto representa un aporte valioso por dos motivos: (i) permite estudiar la rigidez desde una nueva óptica, expresándola como una propiedad distribuida a lo largo de la red, (ii) sienta las bases para el diseño de esquemas de control de rigidez multirobot, que dependen únicamente de información local.

4.1. Introducción

Uno de los inconvenientes que presenta la teoría de rigidez de frameworks es la naturaleza global de dicha propiedad. Es decir, la rigidez depende del framework en su totalidad: es función de las posiciones de todos los vértices y de todos los enlaces existentes. Por ejemplo, la eliminación de un solo enlace puede volver flexible a un framework previamente rígido. Por el contrario, el agregado de un solo enlace puede dotar de rigidez a un framework originalmente flexible. Esto hace que la aplicación de esta propiedad en redes de agentes móviles sea poco útil, especialmente cuando el tamaño del framework crece.

Esto se debe, por un lado, a que, como ya se discutió, el procesamiento de información global por parte de una red de robots limita seriamente su escalabilidad. Por otro lado, considere la situación en que un framework de gran tamaño se vuelve flexible por un enlace perdido. Este evento, que es local, afecta a todos los nodos, lo cual resulta inconveniente. En esta situación, sería deseable orientar el estudio de rigidez a subconjuntos de nodos que puedan formar subframeworks localmente rígidos.

Por estas razones, es necesario buscar otras maneras de describir la rigidez del framework, de forma tal de poder definirla en función de variables locales, que no dependan de todo el conjunto de nodos y enlaces, sino de uno o varios subconjuntos. Aquí, nos proponemos superar estas limitaciones mediante una formulación novedosa del subframework, que aporta nuevas direcciones a su actual definición. La misma permite presentar una condición necesaria y suficiente para la rigidez del framework en función de un conjunto de subframeworks, que serán definidos en el apartado 4.2. Esto nos habilita a expresar la rigidez como un conjunto de condiciones locales, que dependen únicamente de un subconjunto de nodos cercanos entre sí, sin necesidad de compartir información global con toda la red. Esto hace que este esquema sea más apropiado para estudiar las propiedades locales de rigidez de grandes frameworks.

4.2. Definición

Sea un framework (\mathcal{G}, p) con $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, y una función $h : \mathcal{V} \rightarrow \mathbb{N}_0$ que asigna, a cada nodo, un valor asociado a una distancia geodésica. Se define el cuerpo con *centro* en i y *alcance* $h(i)$ como

$$\mathcal{V}_i := \{j \in \mathcal{V} : g(i, j) \leq h(i)\}. \quad (4.1)$$

Es decir, \mathcal{V}_i es el conjunto de vértices que se encuentran a una distancia de i menor o igual a $h(i)$. Si $h(i) = 0$ decimos que el cuerpo \mathcal{V}_i es *trivial*, puesto que contiene únicamente a un

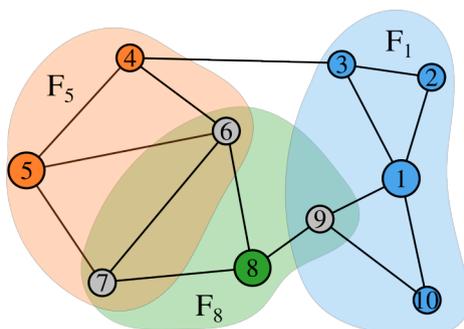


Figura 4.1: Una descomposición de un framework en cuerpos no triviales.

vértice; si $h(i) < \varepsilon(i)$, es decir, si no cubre con todo el framework, decimos que el i -ésimo cuerpo es *propio*. A cada cuerpo \mathcal{V}_i le corresponde un subframework $\mathcal{F}_i = (\mathcal{G}_i, p)$ donde \mathcal{G}_i es el subgrafo inducido por \mathcal{V}_i . Es importante observar que un vértice puede pertenecer a múltiples subframeworks, no obstante es el centro de uno solo. En Fig. 4.1 un framework y tres subframeworks no triviales dados por la función $h(1) = h(5) = h(8) = 1$ y $h(\cdot) = 0$ para los otros vértices.

4.3. Descomposición Densa en Subframeworks

En esta sección, consideramos el caso donde $h(i) > 0$ para todo $i \in \mathcal{V}$, es decir, ningún subframework es trivial. Estos v subframeworks no triviales se superponen entre sí, por lo que decimos que la descomposición en subframeworks es *densa*. En la Sección 4.4 se extenderá el estudio al caso general donde se permite que haya subframeworks triviales, lo que da lugar a una descomposición *dispersa*.

En [108] se presentó por primera vez una descomposición de un frameworks en subframeworks, sin embargo, los mismos son restringidos a tener un alcance unitario, es decir $h(i) = 1$ para todo $i \in \mathcal{V}$. Esto permite obtener una condición suficiente pero no necesaria para la rigidez del framework en términos de sus subframeworks. La definición de subframeworks propuesta aquí, contempla subframeworks con alcances mayores a uno, lo

que nos permite presentar una condición necesaria y suficiente para la rigidez basada en subframeworks.

En primer lugar, presentamos el siguiente lema, que habla de la unión de dos frameworks infinitesimalmente rígidos, es una adaptación del presentado en [109] para frameworks globalmente rígidos.

Lema 4.3.1. *Sean dos frameworks \mathcal{F}_1 y \mathcal{F}_2 infinitesimalmente rígidos en \mathbb{R}^d . El framework $\mathcal{F}_1 \cup \mathcal{F}_2$ es infinitesimalmente rígido si y solo si $\mathcal{P} := p(\mathcal{V}_1 \cap \mathcal{V}_2)$ tiene dimensión afín igual a $d - 1$ o d .*

Demostración. (\Rightarrow) Por contrarrecíproca: si la dimensión afín de \mathcal{P} es $d - 2$ o menos, entonces es posible asignar una rotación relativa entre ambos frameworks que mantenga a \mathcal{P} fijo, por lo tanto, la unión es infinitesimalmente flexible. Para ver esto, asumimos, sin pérdida de generalidad, que $\mathcal{P} \subseteq \text{linsp}\{e_1, \dots, e_{d-2}\}$ donde e_k son los vectores canónicos de \mathbb{R}^d . Luego $\text{linsp}\{e_{d-1}, e_d\} \subseteq \mathcal{P}^\perp$. Por último, la rotación $Q := \text{diag}(I_{d-2}, R)$, con $R \in \mathbb{SO}(2)$, aplicada a uno de los frameworks mientras el otro se mantiene constante, representa la rotación relativa mencionada. (\Leftarrow) Si la dimensión afín de \mathcal{P} es mayor o igual a $d - 1$, entonces la única rotación posible que mantiene a \mathcal{P} fijo es la identidad. Es decir, no puede haber movimientos relativos entre ambos frameworks, por lo tanto, la unión es infinitesimalmente rígida. □

Corolario 4.3.1. *Sean dos grafos \mathcal{G}_1 y \mathcal{G}_2 genéricamente rígidos en \mathbb{R}^d . El grafo $\mathcal{G}_1 \cup \mathcal{G}_2$ es genéricamente rígido si y solo si $|\mathcal{V}_1 \cap \mathcal{V}_2| \geq d$.*

Ahora, introducimos el siguiente lema que determina la dimensión del conjunto de posiciones asociadas a los vértices $\mathcal{B}_i := \{i\} \cup \mathcal{N}_i$, es decir, a un nodo y a sus vecinos. Como se postuló en el Capítulo 2, el grafo completo es el único rígido con $v \leq d + 1$ vértices, por lo que enfocamos el análisis a grafos con $v \geq d + 1$.

Lema 4.3.2. *Sea (\mathcal{G}, p) un framework infinitesimalmente rígido con $v \geq d + 1$ vértices. Luego, para todo $i \in \mathcal{V}$, $p(\mathcal{B}_i)$ tiene dimensión afín igual a d .*

Demostración. Por contrarrecíproca: tomemos un vértice $i \in \mathcal{V}$. Supongamos que $p(\mathcal{B}_i)$ tiene dimensión afín menor a d , es decir, está contenido en un hiperplano $p_i + S$ donde S es un subespacio. Luego, dado un vector $t \neq 0$ ortogonal a S , definimos el vector de velocidades $\mathbf{u} = [u_1^T \cdots u_v^T]^T$ tal que

$$u_j = \begin{cases} t, & j = i, \\ 0, & j \neq i. \end{cases} \quad (4.2)$$

Se sigue que $\mathbf{u} \notin T(p)$, puesto que \mathbf{u} corresponde a una trayectoria tal que el conjunto $p(\mathcal{V} \setminus \{i\})$ es invariante, pero el menor subespacio afín que contiene a este conjunto tiene dimensión $d - 1$ o d , y ninguna roto-traslación distinta de la identidad tiene un subespacio invariante de esa dimensión. Luego, vemos que

$$\mathbf{u}^T \mathbf{L}(p) \mathbf{u} = \sum_{j \sim i} \left(t^T \frac{p_i - p_j}{\|p_i - p_j\|} \right)^2 = 0$$

puesto que $p_i - p_j \in S$. Se concluye que el framework es infinitesimalmente flexible. \square

Corolario 4.3.2. *Sea \mathcal{G} un grafo genéricamente rígido con $v \geq d + 1$ vértices. Luego, para todo $i \in \mathcal{V}$, se cumple que $|\mathcal{B}_i| \geq d + 1$.*

Ahora estamos listos para presentar el primero de los dos teoremas principales de esta sección.

Teorema 4.3.1 (Rigidez Genérica por Descomposición Densa). *Sea \mathcal{G} un grafo conexo. Entonces \mathcal{G} es genéricamente rígido en \mathbb{R}^d si y solo si existe una función positiva $h : \mathcal{V} \rightarrow \mathbb{N}_1$, tal que para todo $i \in \mathcal{V}$ el subgrafo \mathcal{G}_i es genéricamente rígido.*

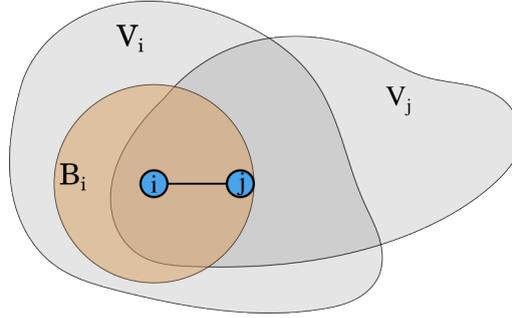


Figura 4.2: Representación esquemática de dos subframeworks adyacentes, y los conjuntos involucrados en la demostración del Teorema 4.3.1.

Demostración. (\Leftarrow) Sea una función h tal que todos los subgrafos son rígidos. Para probar que \mathcal{G} es rígido, primero mostraremos que la unión de dos subgrafos con centros adyacentes cumple las condiciones del Corolario 4.3.1. Luego, como \mathcal{G} es la unión de sus subgrafos, y el mismo es conexo, se llega a que el mismo es rígido. Para ello, tomamos \mathcal{G}_i y \mathcal{G}_j tales que $i \sim j$, como indica la Fig. 4.2. Dado que $h(j) \geq 1$, se sigue $i \in \mathcal{V}_j$, y como \mathcal{G}_j es genéricamente rígido, entonces, por Corolario 4.3.2, i tiene al menos d vecinos en \mathcal{G}_j , es decir, $|\mathcal{B}_i \cap \mathcal{V}_j| \geq d + 1$. Pero $\mathcal{B}_i \subseteq \mathcal{V}_i$ pues $h(i) \geq 1$, entonces $\mathcal{B}_i \cap \mathcal{V}_j \subseteq \mathcal{V}_i \cap \mathcal{V}_j$. Esto implica que $\mathcal{V}_i \cap \mathcal{V}_j$ tiene al menos $d + 1$ elementos. (\Rightarrow) Se demuestra tomando $h(i) = \varepsilon(i)$ para todo $i \in \mathcal{V}$. \square

La contribución de este teorema es que provee de una forma de evaluar la rigidez genérica de un grafo, que es una propiedad global, por medio de un conjunto de condiciones locales distribuidas a lo largo del mismo.

Al expresar la descomposición por subframeworks en términos genéricos, es decir, para los grafos subyacentes, se abstrae el análisis de las posibles realizaciones, simplificando la postulación. Esto resulta muy útil desde el punto de vista teórico. Sin embargo, para realizar un análisis cuantitativo de la rigidez de los frameworks, es necesario tener en cuenta las realizaciones del grafo. Por ello, presentamos a continuación el segundo teorema de esta sección que establece la condición de rigidez infinitesimal de un framework a

partir de sus subframeworks.

Teorema 4.3.2 (Rigidez infinitesimal por Descomposición Densa). *Sea (\mathcal{G}, p) con \mathcal{G} conexo. Entonces $\mathcal{F} = (\mathcal{G}, p)$ es infinitesimalmente rígido si y solo si existe una función positiva $h : \mathcal{V} \rightarrow \mathbb{N}_1$, tal que para todo $i \in \mathcal{V}$ el subframework \mathcal{F}_i es infinitesimalmente rígido.*

Demostración. Se procede de forma similar a la demostración del Teorema 4.3.1, utilizando los Lemas 4.3.1 y 4.3.2.

□

4.3.1. Métricas

En un espacio d -dimensional, definimos el conjunto de funciones de alcance

$$\mathcal{H}_1(\mathcal{G}) := \{h : \text{cumple las condiciones del Teorema 4.3.1}\}. \quad (4.3)$$

el cual es no vacío si y solo si \mathcal{G} es rígido. Para comparar de forma cuantitativa las descomposiciones obtenidas para las distintas funciones de \mathcal{H}_1 , es necesario definir métricas sobre las mismas, que permitan evaluar algún comportamiento deseado. En este sentido, resulta apropiado elegir h tal que los subframeworks resultantes tengan corto alcance, lo cual se debe principalmente a dos factores.

En primer lugar, para implementar un algoritmo de rigidez de forma descentralizada, los agentes deben intercambiar paquetes de información que recorren caminos de distinta longitud hasta llegar a su destino. Por lo tanto, es necesario considerar el tiempo transcurrido durante ese transporte, el cual hace que los agentes reciban información desactualizada. Para cuantificar esto, notamos que en una descomposición por subframeworks, es natural que todos los nodos de un mismo subframework se comuniquen entre sí enrutando mensajes a través del centro, resultando en caminos de longitud menor o igual al doble del

alcance. Esto se observa en Fig. 4.3a. Por ello, definimos el *máximo retardo* presente en una descomposición h como

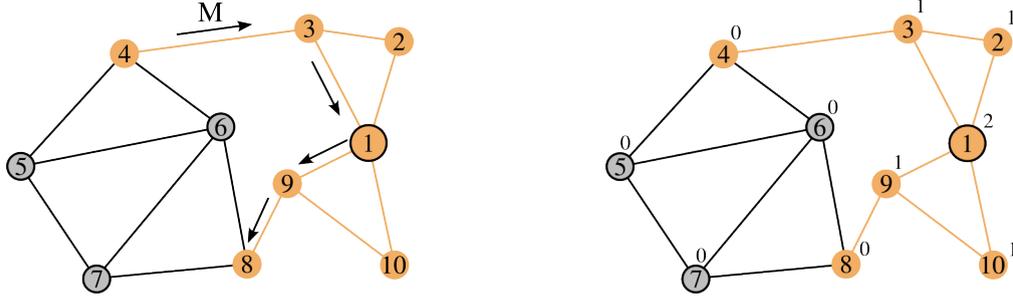
$$\mathcal{D}(h) := 2 \max\{h(i) : i \in \mathcal{V}\}. \quad (4.4)$$

Notar que $2 \leq \mathcal{D}(h) \leq 2D(\mathcal{G})$, y que el límite inferior se alcanza cuando todos los subframeworks tienen alcance unitario, mientras que el límite superior ocurre cuando existe un subframework que cubre todo el framework.

En segundo lugar, el intercambio de información bidireccional entre los nodos de un subframework y su centro implica cierto nivel de utilización de los canales de comunicación. A medida que el alcance de un subframework crece, el centro debe transmitir mensajes a una mayor cantidad de vértices, lo que produce un aumento de la carga comunicacional, especialmente en los enlaces cercanos a dicho centro. Para cuantificar la *carga total de comunicación* sobre \mathcal{G} , proponemos la siguiente métrica

$$\mathcal{L}(h) := \frac{1}{v} \sum_{i \sim j} (c_i + c_j), \quad \text{donde } c_i := \sum_{j \in \mathcal{V}} \max\{0, h(j) - g(i, j)\}. \quad (4.5)$$

Vemos que (4.5) es una suma de términos asociados a cada enlace ij , afectados por coeficientes que les dan mayor o menor importancia relativa. El *coeficiente de carga* c_i mide la cantidad de información que debe transmitir cada nodo. Notar que $\max\{0, h(j) - g(i, j)\} > 0$ si y solo si $i \in \mathcal{V}_j$, por lo tanto, c_i es función únicamente del conjunto de subframeworks que contienen a i . Cada término de c_i da cuenta de la cercanía del nodo i al centro j : equivale a $h(j)$ si $i = j$ y decrece linealmente hasta llegar a los nodos en la frontera ($g(i, j) = h(j)$) para los cuales vale cero, dado que estos ya no propagan los mensajes provenientes de dicho centro. Esto se observa en Fig. 4.3b. Notar que si $h(i) = 1$ para todo i , entonces $c_i = 1$ para todo i , y se cumple que $\mathcal{L}(h) = \bar{n}(\mathcal{G})$ donde $\bar{n} = 2e/v$ es el grado promedio de \mathcal{G} . Dado que $c_i \geq 1$, se observa que $\mathcal{L}(h) \geq \bar{n}(\mathcal{G})$.



(a) Trayectoria de un mensaje M de $4 \rightarrow 8$ en \mathcal{G}_1 . (b) Coeficientes en \mathcal{G}_1 : $\max\{0, h(1) - g(1, i)\}$.

Figura 4.3: Métricas para comparar distintas descomposiciones en subframeworks. En amarillo se muestra el subframework \mathcal{F}_1 con $h(1) = 2$.

4.3.2. Alcance Mínimo de Rigidez

Puede observarse que (4.4) es no decreciente y (4.5) es estrictamente creciente respecto del alcance de los subframeworks. Entonces llamamos *alcance mínimo de rigidez* a la función

$$h_r(i) := \min\{h(i) : \mathcal{G}_i \text{ es rígido}\}. \quad (4.6)$$

En consecuencia, h_r es la descomposición en subframeworks que minimiza $\mathcal{D}(h)$ y $\mathcal{L}(h)$ en \mathcal{H}_1 . Es, por lo tanto, la función de alcance elegida de aquí en adelante.

En Fig. 4.4 se ilustra esta definición. El subgrafo \mathcal{G}_1 con $h(1) = 1$ es flexible, pero con $h(1) = 2$ es rígido, por lo tanto $h_r(1) = 2$. El alcance mínimo de rigidez impone un límite inferior al tamaño de los subframeworks, el cual mide el grado de descentralización lograda por una determinada descomposición en subframeworks. En Fig. 4.5 se muestran ejemplos de framework rígidos y los alcances de rigidez correspondientes. Notar que los alcances mínimos de rigidez tienden a ser mayores en aquellas zonas de menor densidad de enlaces.

Como veremos en la Sección 4.4, si se extiende el conjunto de funciones válidas eliminando la condición $h(i) > 0$, es posible hallar descomposiciones en subframeworks que

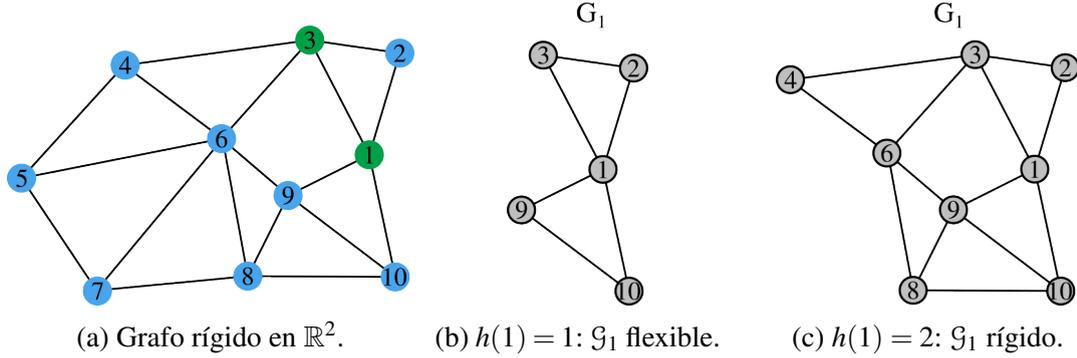


Figura 4.4: El alcance mínimo de rigidez de los nodos un grafo, diferenciado por colores: $h_r(1) = h_r(3) = 2$ y $h_r(\cdot) = 1$ para los restantes.

reduzcan aún más las métricas presentadas.

4.3.3. Simulaciones

Modelo de Framework Aleatorio

Para evaluar el comportamiento del esquema de descomposición en subframeworks es necesario generar una gran cantidad de frameworks rígidos. Para ello, proponemos un modelo aleatorio de framework dado por la tripla (\mathcal{R}, v, ρ) donde $\mathcal{R} \subset \mathbb{R}^d$ es una región acotada, v es la cantidad de vértices y $\rho \in \mathbb{R}_{>0}$ el rango máximo de conexión entre los mismos. Con este modelo, se obtiene un framework aleatorio (\mathcal{G}_ρ, p) mediante

$$p_i \sim \mathcal{U}(\mathcal{R}) \quad \forall i = 1, \dots, v, \quad (4.7)$$

$$\mathcal{E} = \{\{i, j\} : \|p_i - p_j\| \leq \rho\},$$

donde \mathcal{U} es la distribución uniforme sobre la región correspondiente. El esquema de enlaces por proximidad dado en (4.7) resulta apropiado para modelar una red de agentes distribuidos espacialmente, que cuentan con un rango limitado para establecer conexiones punto a punto.

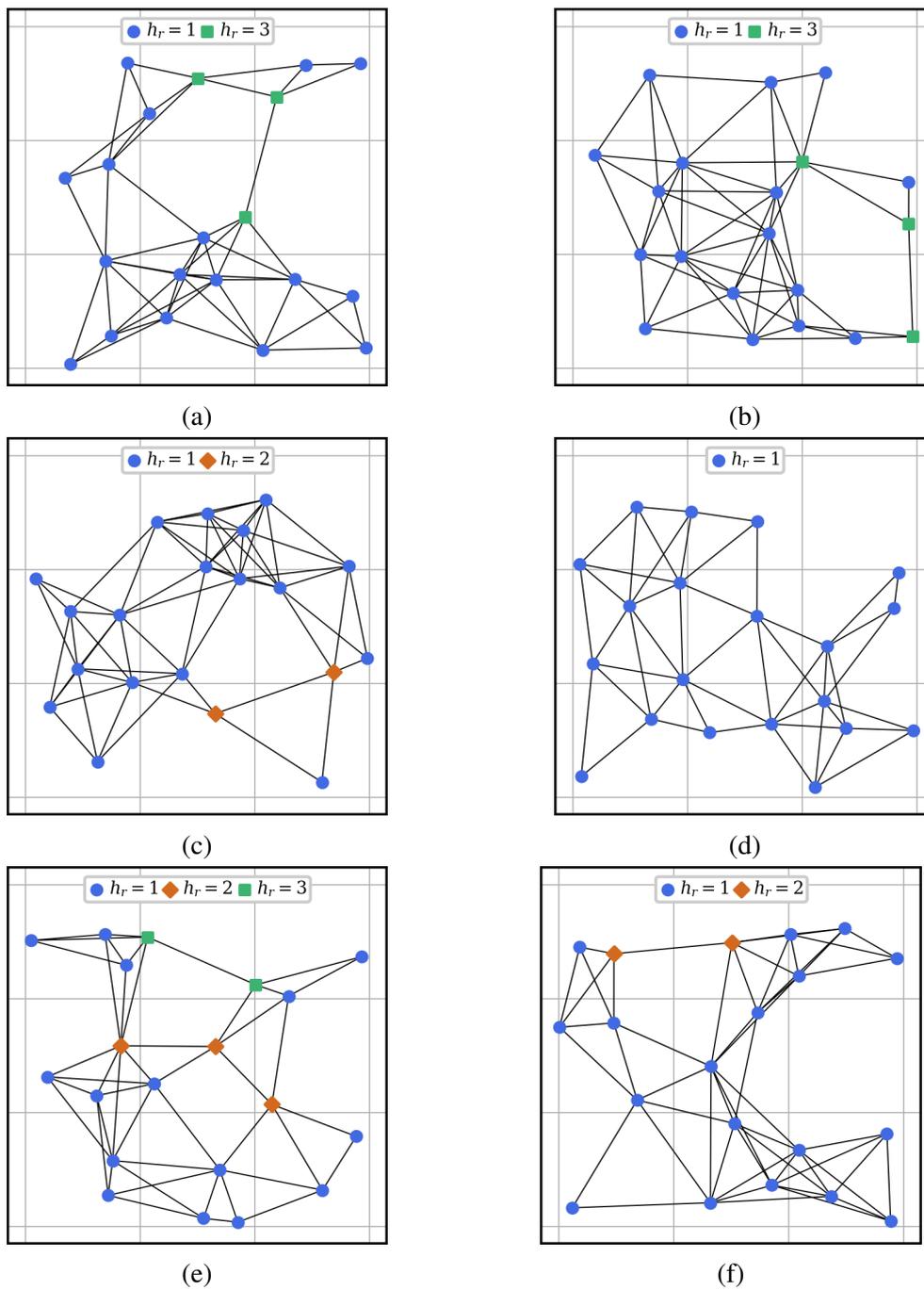


Figura 4.5: Alcances mínimos de rigidez para frameworks con $\nu = 20$.

Para forzar al modelo a producir frameworks rígidos se procede de la siguiente manera. Dada una asignación de posiciones p sobre \mathbb{R}^d , se definen

$$\rho_0 := \text{mín}\{\rho : (\mathcal{G}_\rho, p) \text{ es infinitesimalmente rígido}\}, \quad (4.8)$$

$$\rho_1 := \text{máx}\{\|p_i - p_j\| : i, j = 1, \dots, v\}. \quad (4.9)$$

Es decir, ρ_0 es el mínimo rango que induce un framework infinitesimalmente rígido, y ρ_1 el mínimo rango que genera el grafo completo. Notar que $\rho_0 \leq \rho_1$. En Fig. 4.6 se muestran los resultados obtenidos para numerosas muestras de frameworks rígidos en \mathbb{R}^2 , utilizando distintos números de vértices $4 \leq v \leq 100$ situados sobre la región $\mathcal{R} = [0, 1] \times [0, 1]$. Se observa como el rango ρ_0 disminuye con el aumento de vértices, debido a que los nodos encuentran mayor cantidad de vecinos en sus cercanías. Por otro lado, el valor de ρ_1 aumenta con v y se encuentra acotado por $\sqrt{2}$, la máxima distancia entre puntos de la región.

Se define entonces el modelo (\mathcal{R}, v, α) , que produce frameworks $(\mathcal{G}_{\rho(\alpha)}, p)$ donde $\rho(\alpha) = (1 - \alpha)\rho_0 + \alpha\rho_1$, con $\alpha \in [0, 1]$, el cual es infinitesimalmente rígido por definición. En Fig. 4.7 se muestran tres frameworks rígidos obtenidos para el mismo conjunto de posiciones pero para diferentes valores de α . Se observa que para $\alpha = 0$, aparecen subframeworks de alcance 2 y 3 en la descomposición. Sin embargo, con pequeños aumentos de parámetro α , se comienzan a agregar nuevos enlaces que hacen posible la reducción del alcance de los subframeworks hasta llegar a que todos sean unitarios. En este ejemplo, esto se logra para $\alpha = 0.1$. Para visualizar este fenómeno con mayor claridad, proponemos lo siguiente. Dada una posición p , el grafo $\mathcal{G}_{\rho(0)}$ es infinitesimalmente rígido, pero h_r puede tomar valores mayores a 1 para uno o más vértices. Entonces, definimos el parámetro $\alpha^* \in [0, 1]$ como el mínimo valor tal que todos los subgrafos del grafo $\mathcal{G}_{\rho(\alpha^*)}$ tienen alcance de rigidez igual a 1. En Fig. 4.8 se visualizan los valores de α^* obtenidos mediante simulaciones, para distintas cantidades de vértices. Se observa que, luego de un

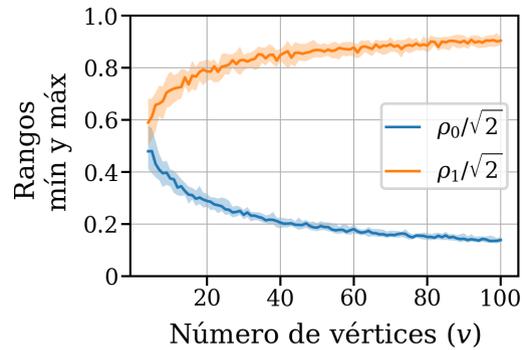


Figura 4.6: Los rangos ρ_0 y ρ_1 versus el número de vértices. Valores obtenidos para 50 muestras por cada valor de v , dando un total de ≈ 5000 frameworks aleatorios. La línea sólida indica la mediana y el área coloreada representa la zona delimitada por los cuartiles $Q_1(25\%)$ y $Q_3(75\%)$.

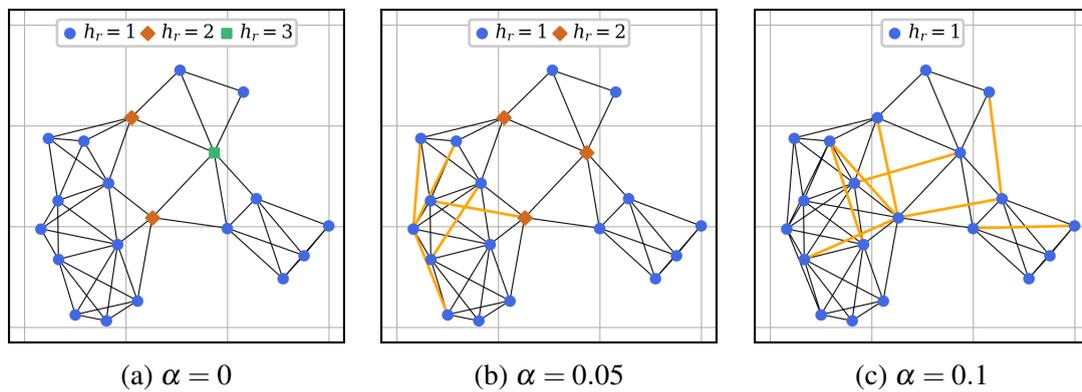


Figura 4.7: Tres frameworks inducidos por el mismo conjunto de posiciones. Los nodos son diferenciados según su alcance de rigidez h_r . En naranja se muestran los enlaces agregados por el incremento de α .

aumento inicial, a partir de $v \approx 30$ el valor de α^* se estabiliza, con una mediana de ≈ 0.04 . Este comportamiento indica que un pequeño aumento del rango de conexión respecto de ρ_0 , es suficiente para lograr subframeworks de alcance unitario.

Análisis Cuantitativo

Para evaluar la escalabilidad que presenta la técnica de descomposición en subframeworks, se simuló los valores que toman las métricas para un número creciente de

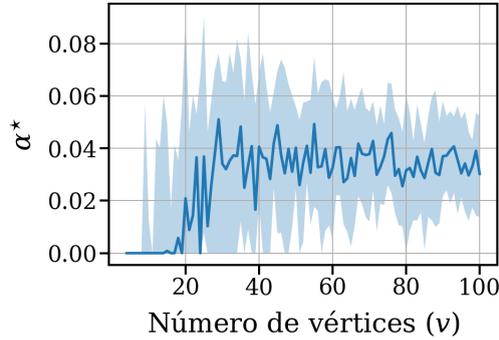


Figura 4.8: Mínimo α tal que todos los subframeworks tienen alcance de rigidez unitario. Valores obtenidos para 50 muestras por cada valor de v . La línea sólida indica la mediana y el área coloreada representa la zona delimitada por los cuartiles Q_1 y Q_3 .

vértices entre $4 \leq v \leq 100$. Se analizaron un total de 50 frameworks rígidos por cada valor de v , generados con el modelo de framework aleatorio propuesto. En Fig. 4.9 y Fig. 4.10 se muestran los resultados obtenidos para distintos valores del parámetro $\alpha \in \{0, 0.05, 0.1\}$.

En Fig. 4.9 se comparan las dependencias con v del máximo retardo $\mathcal{D}(h_r)$ y del diámetro D del grafo. Resulta apropiado comparar con este último, debido a que representa una cota inferior del retardo existente en el intercambio de información por consenso. En Fig. 4.9a se observa cómo para $\alpha = 0$, el máximo retardo $\mathcal{D}(h_r)$ sigue aproximadamente el valor del diámetro del grafo, apenas despegándose por debajo del mismo a partir de valores mayores a los ≈ 60 vértices. Esto indica que, en el mayor subframework, el retardo asociado al intercambio de información es del orden del diámetro, indicando una virtual equivalencia con las técnicas existentes. Sin embargo, Fig. 4.9b y Fig. 4.9c muestran que, para pequeños valores de α , si bien el diámetro disminuye, el retardo \mathcal{D} se reduce en mayor medida, haciendo que ambas curvas se despeguen. Más aún, se observa que el máximo retardo es prácticamente independiente de v , a diferencia del diámetro, que crece al menos logarítmicamente. Esto muestra que la descomposición en subframeworks es una técnica escalable.

En cuanto a la utilización de los canales de comunicación, observamos en Fig. 4.10 la

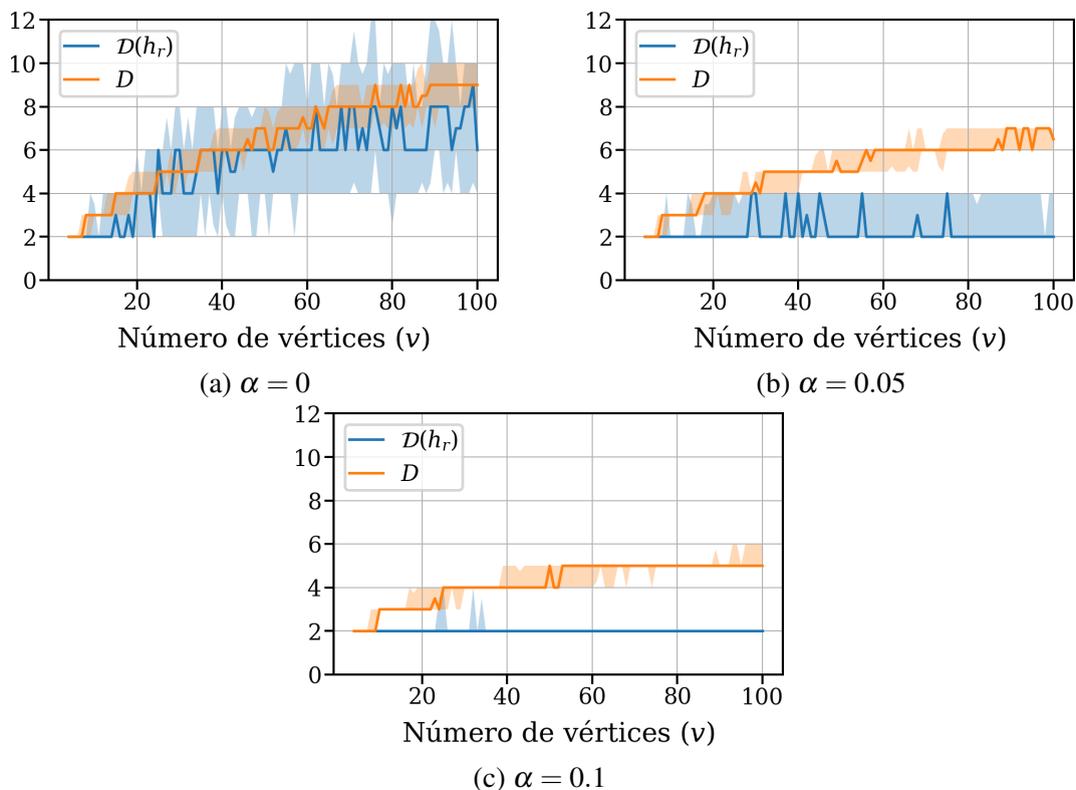


Figura 4.9: El máximo retardo \mathcal{D} y el diámetro D versus el número de vértices. La línea sólida indica la mediana del conjunto de muestras y el área coloreada representa la zona delimitada por los cuartiles Q_1 y Q_3 .

evolución, con la cantidad de vértices, de la carga $\mathcal{L}(h_r)$ y del grado promedio \bar{n} . Este último representa la utilización de los canales en un esquema distribuido y es cota inferior del primero. En Fig. 4.10a podemos observar que, para $\alpha = 0$, el esquema de subframeworks implica una mayor carga que un esquema distribuido, especialmente cuando la cantidad de vértices aumenta. Esto podría representar un problema para la escalabilidad de la descomposición por subframeworks. Sin embargo, Fig. 4.10b vemos que para $\alpha = 0.05$ la carga acompaña el crecimiento del grado promedio, apenas por encima del mismo. Más aún, en Fig. 4.10c se observa que para $\alpha = 0.1$ ambos parámetros son prácticamente iguales. Esto se debe a que con pequeños valores de α , los subframeworks tienden a tener alcance

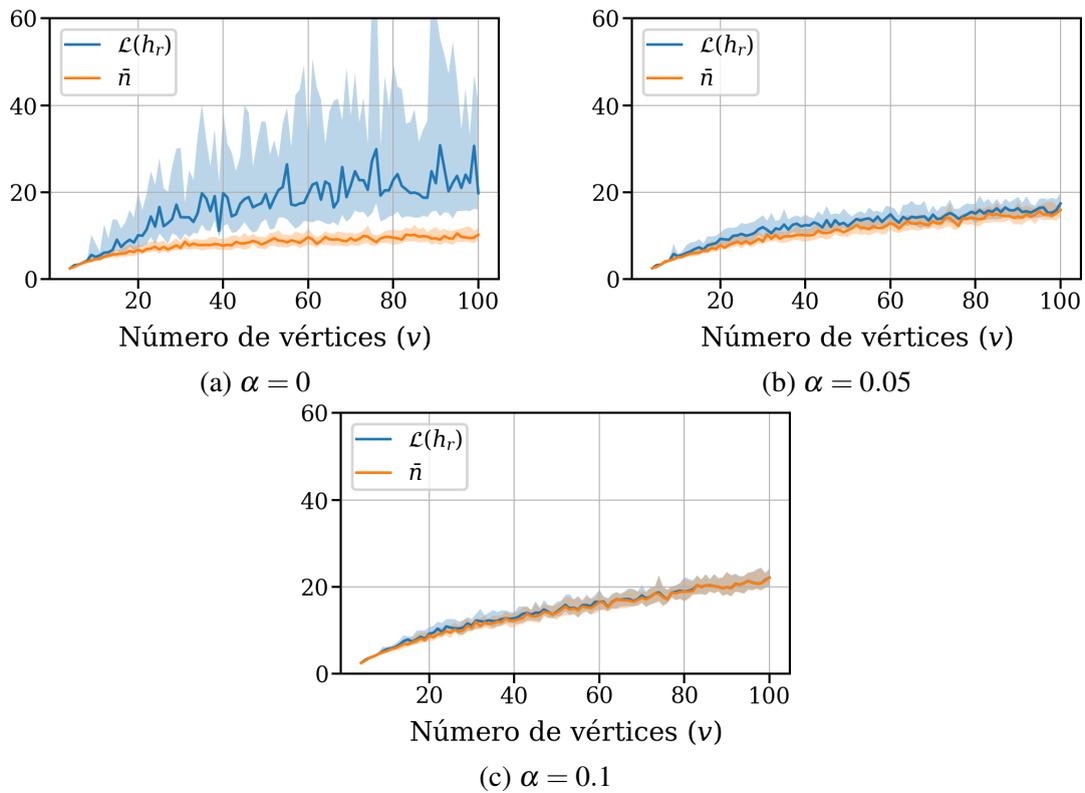


Figura 4.10: La carga de comunicaciones \mathcal{L} y el grado promedio del grafo versus el número de vértices.

unitario, haciendo de la descomposición de subframeworks una estrategia prácticamente equivalente a una ley distribuida.

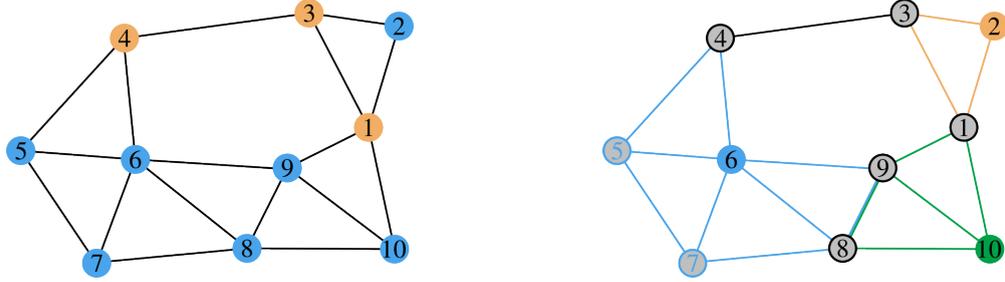
El análisis expuesto muestra las ventajas de la estrategia de descomposición de un framework en subframeworks. Se definió el parámetro α , el cual mostró ser muy útil en la descripción de este sistema. Se observó que para pequeños valores de este parámetro, lo que equivale a frameworks con rango de conexión apenas superior al mínimo de rigidez, la descomposición en subframeworks se vuelve una estrategia cualitativamente mejor que las existentes.

4.4. Descomposición Dispersa en Subframeworks

En esta sección desarrollamos el caso donde se permite que haya subframeworks triviales. Debido a que los mismos no tienen relevancia en la rigidez del framework, la descomposición resultante está gobernada por los no triviales, motivo por el cual decimos que la descomposición es *dispersa*. El uso de un menor número de subframeworks permite reducir el retardo máximo y la carga de comunicaciones, y de esta forma mejorar las características de la descomposición.

Considere el grafo rígido de Fig. 4.11a. El mismo posee baja cantidad de enlaces, por lo tanto, los alcances mínimos de rigidez tienden a ser altos, como fue observado en la sección anterior. En este ejemplo, el alcance mínimo de rigidez está dado por $h_r(1) = h_r(3) = h_r(4) = 3$ y $h_r(\cdot) = 1$ para el resto. Se puede observar que los subgrafos \mathcal{G}_1 , \mathcal{G}_3 y \mathcal{G}_4 son impropios. Por consiguiente, es redundante analizar la rigidez de todos los subgrafos para determinar la rigidez de \mathcal{G} . Esta superposición de subgrafos se evidencia en los altos valores que toman las métricas: $\mathcal{D}(h_r) = 1.5D$ y $\mathcal{L}(h_r) \approx 4.7\bar{n}$. Para reducir esta redundancia, eliminamos la condición de positividad de la función h que define la descomposición. Entonces es suficiente tomar, por ejemplo, $\tilde{h}(1) = 3$ y $\tilde{h}(\cdot) = 0$ para el resto. De esta forma se elimina la superposición existente entre los subgrafos, obteniéndose $\mathcal{D}(\tilde{h}) = 1.5D$ y $\mathcal{L}(\tilde{h}) \approx 1.4\bar{n}$. Es interesante observar que \tilde{h} permite reducir notablemente la carga en las comunicaciones. No obstante, el retardo máximo permanece constante, como consecuencia de que dicha descomposición corresponde a un esquema centralizado en el nodo 1, que tampoco es deseable. Por estas razones, es apropiado preguntarse qué subgrafos es posible remover, especialmente los de mayor alcance, y emplear únicamente los necesarios para garantizar que la descomposición permita determinar la rigidez del grafo.

Para responder a esta pregunta, comenzamos definiendo el conjunto de centros no



(a) El color de los nodos según su alcance de rigidez. (b) Una descomposición dispersa dada por $\hat{h}(2) = \hat{h}(6) = \hat{h}(10) = 1$ y $\hat{h}(\cdot) = 0$ para el resto.

Figura 4.11: Alcances de rigidez y descomposición dispersa de un framework rígido.

triviales $\mathcal{C} = \{i \in \mathcal{V} : h(i) > 0\}$. En Fig. 4.11b se muestra una descomposición del grafo analizado en 3 subgrafos con $\mathcal{C} = \{2, 6, 10\}$. En este caso, se obtienen $\mathcal{D}(\hat{h}) = 0.5D$ y $\mathcal{L}(\hat{h}) \approx 0.3\bar{n}$, es decir, se reducen sustancialmente ambas métricas, e incluso se logra reducir la carga \mathcal{L} por debajo del grado promedio, que es la cota inferior si no se permiten subgrafos triviales. Para que esta descomposición dispersa sea válida, es decir, que permita determinar la rigidez del grafo, es necesario reformular los teoremas 4.3.1 y 4.3.2. En primer lugar, dada una función h definimos

$$\mathcal{V}_U := \bigcup_{i \in \mathcal{C}} \mathcal{V}_i, \quad \mathcal{E}_U := \bigcup_{i \in \mathcal{C}} \mathcal{E}_i, \quad (4.10)$$

entonces, es necesario considerar únicamente descomposiciones tales que $\mathcal{V}_U = \mathcal{V}$. No obstante, en general se tiene $\mathcal{E}_U \subseteq \mathcal{E}$ debido a que puede haber enlaces que no queden incluidos en ningún subgrafo. El conjunto $\mathcal{E} \setminus \mathcal{E}_U$ de enlaces que no quedan contenidos en ningún subgrafo, son denominados *enlaces vinculantes*, dado que conectan subgrafos distintos. En Fig. 4.11b el enlace $\{3, 4\}$ es vinculante. Como se verá más adelante, estos enlaces tienen un papel importante en la descomposición de la rigidez.

En segundo lugar, presentamos la siguiente noción, preponderante en nuestro análisis.

Definición 4.4.1. Decimos que el conjunto de subgrafos $\{\mathcal{G}_i : i \in \mathcal{C}\}$ está rigidamente

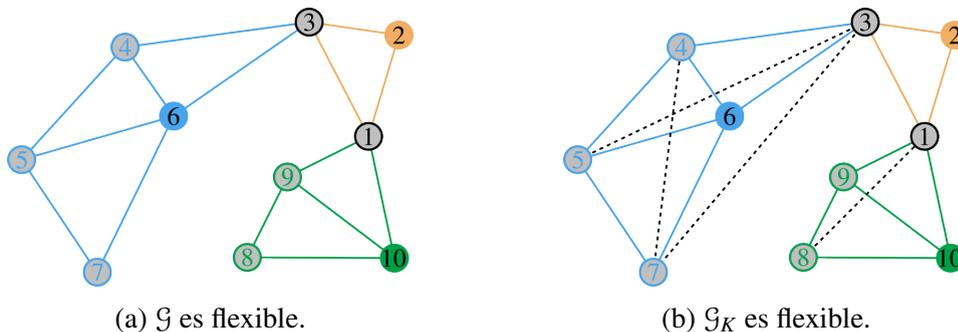


Figura 4.12: Los subgrafos 2, 6 y 10 son rígidos, pero no están rígidamente vinculados.

vinculado si el grafo \mathcal{G}_K es genéricamente rígido, donde \mathcal{G}_K se obtiene a partir de \mathcal{G} reemplazando cada subgrafo \mathcal{G}_i por su versión completa $K(\mathcal{V}_i)$.

La definición de \mathcal{G}_K es útil para separar la rigidez por subgrafos en dos condiciones: la primera referida a la rigidez de los subgrafos y la segunda a la forma en la que los mismos se vinculan. Es decir, si un grafo es genéricamente flexible y tiene todos sus subgrafos rígidos, entonces los mismos no pueden estar rígidamente vinculados. Esto se ilustra en Fig. 4.12, ya que si se transforma \mathcal{G}_i en $K(\mathcal{V}_i)$ para $i = 2, 6, 10$, el grafo resultante \mathcal{G}_K no es rígido. En cambio, un grafo flexible tal que el conjunto de subgrafos está rígidamente vinculado, entonces la flexibilidad debe provenir del interior de algún subgrafo. En Fig. 4.13 vemos un ejemplo de esto, donde si se reemplazan los subgrafos por los completos correspondientes, el grafo resultante es rígido, por lo tanto, decimos que están rígidamente vinculados. Es importante notar dos casos en que la condición de vinculación rígida (4.4.1) se satisface de forma automática: en primer lugar, si al menos un subgrafo es impropio; en segundo lugar, cuando la descomposición es densa y el grafo es conexo.

Evaluar la rigidez del grafo vinculante \mathcal{G}_K es de complejidad equivalente a la del grafo original, de forma tal que debemos hallar otra manera de determinarla si queremos emplear la descomposición dispersa en un sistema multirobot descentralizado. Para ello, notamos que los subgrafos pueden estar vinculados de dos formas: mediante nodos en

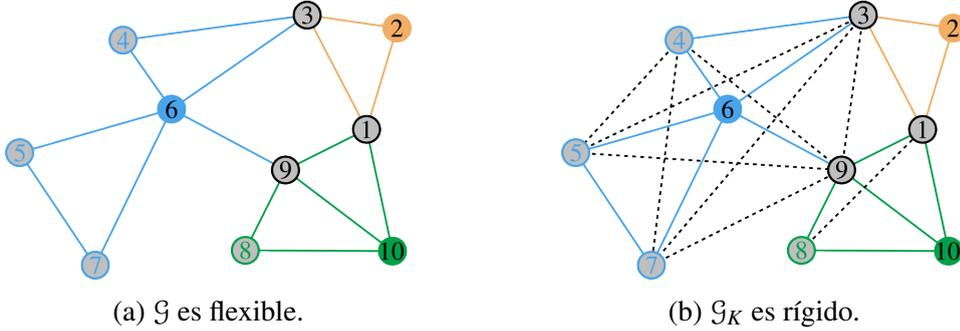


Figura 4.13: \mathcal{G} es flexible, pero los subgrafos están rígidamente vinculados. La flexibilidad proviene de \mathcal{G}_6 .

común o mediante enlaces vinculantes. Esto se observa en Fig. 4.14a, donde los vértices 1, 3, 4, 8, 9 y el enlace $\{3, 4\}$ son vinculantes. Para formalizar esto, dada una asignación h , definimos el subconjunto de *vértices vinculantes* $\mathcal{L} \subseteq \mathcal{V}$ como aquellos contenidos en más de un subgrafo, o incidentes a enlaces vinculantes. Es decir,

$$\mathcal{L} := \{i : |\{j : i \in \mathcal{V}_j\}| \geq 2\} \cup \{i : i \in e, e \in \mathcal{E} \setminus \mathcal{E}_U\}. \quad (4.11)$$

Luego, si llamamos $\mathcal{L}_i := \mathcal{L} \cap \mathcal{V}_i$, entonces el grafo vinculante \mathcal{G}_L se define transformando \mathcal{G}_i en $K(\mathcal{L}_i)$ y manteniendo los enlaces vinculantes,

$$\mathcal{G}_L := \bigcup_{i \in \mathcal{C}} K(\mathcal{L}_i) \cup (\mathcal{L}, \mathcal{E} \setminus \mathcal{E}_U). \quad (4.12)$$

En Fig. 4.14b se muestra el grafo vinculante del ejemplo tratado. Notar que en el grafo vinculante pueden aparecer enlaces que no estaban presentes en el grafo original. A continuación presentamos el rol que juega el mismo.

Teorema 4.4.1. *Sea una descomposición de un grafo en dos o más subgrafos. El conjunto $\{\mathcal{G}_i : i \in \mathcal{C}\}$ está rígidamente vinculado si y solo si $|\mathcal{L}_i| \geq d$ y \mathcal{G}_L es rígido.*

Demostración. Ilustrado en Fig. 4.15. (\Leftarrow) Se parte de \mathcal{G}_L y se agregan los nodos no

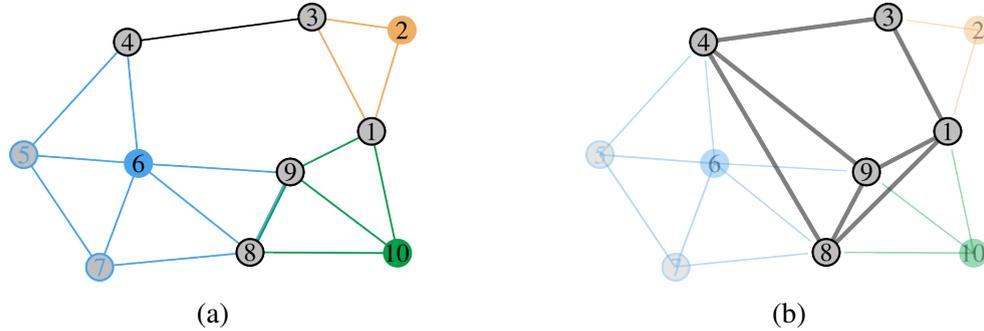


Figura 4.14: Vinculación de los subgrafos \mathcal{G}_2 , \mathcal{G}_6 y \mathcal{G}_{10} . Los nodos vinculantes $\mathcal{L}_2 = \{1, 3\}$, $\mathcal{L}_6 = \{4, 8, 9\}$ y $\mathcal{L}_{10} = \{1, 8, 9\}$ se muestran en color gris. En (b) se muestra en gris el grafo vinculante.

vinculantes y todos los enlaces dentro de cada subgrafo de forma tal de transformar $K(\mathcal{L}_i)$ en $K(\mathcal{V}_i)$. Como resultado se obtiene \mathcal{G}_K , y dado que todo vértice agregado se conecta a d nodos en \mathcal{L}_i , el agregado de los nodos no vinculantes preserva la rigidez. (\Rightarrow) \mathcal{G}_K rígido implica $|\mathcal{L}_i| \geq d$ para todo $i \in \mathcal{C}$, puesto que de no ser así, al menos un subgrafo no estaría rígidamente vinculado al resto. Luego, se eliminan los vértices no vinculantes, manteniendo estos últimos de forma tal de transformar $K(\mathcal{V}_i)$ en $K(\mathcal{L}_i)$. Este proceso, que culmina cuando se obtiene \mathcal{G}_L , preserva la rigidez, dado que cada subgrafo resultante es $K(\mathcal{L}_i)$, el cual es rígido, y no se elimina ningún enlace vinculante. \square

Con estas definiciones estamos en condiciones de presentar los teoremas principales de esta sección, que nos permiten establecer la rigidez genérica de un grafo por medio de una descomposición dispersa, y una condición suficiente para la genericidad de la realización de un framework.

Teorema 4.4.2 (Rigidez Genérica por Descomposición Dispersa). *Un grafo \mathcal{G} es genéricamente rígido en \mathbb{R}^d si y solo si existe una función $h : \mathcal{V} \rightarrow \mathbb{N}_0$, tal que los subgrafos $\{\mathcal{G}_i : i \in \mathcal{C}\}$ cubren todos los vértices, son genéricamente rígidos y están rígidamente vinculados.*

Demostración. (\Leftarrow) Se comienza con \mathcal{G} (Fig. 4.15a), y se agregan los enlaces faltantes en

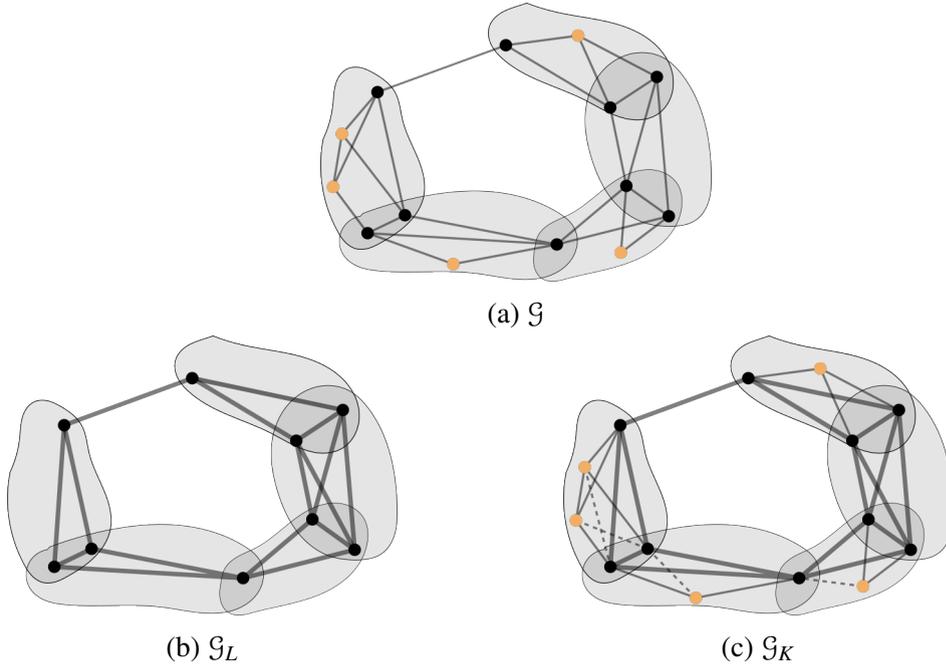


Figura 4.15: Grafo descompuesto en subgrafos dispersos, representados cada uno con un cuerpo rígido. Los nodos vinculantes se muestran en negro, y en amarillo los no vinculantes.

cada subgrafo de forma tal de transformar \mathcal{G}_i en $K(\mathcal{V}_i)$. Como \mathcal{G}_i es rígido, este proceso no altera la rigidez de cada subgrafo, pero tampoco la forma en que están vinculados los mismos. El proceso termina cuando se obtiene \mathcal{G}_K , el cual es rígido por hipótesis, y se sigue que el \mathcal{G} también lo es. (\Rightarrow) Se deduce eligiendo un subgrafo impropio, el cual es rígido porque coincide con \mathcal{G} y rígidamente vinculado por definición. \square

Como es visible, el uso de la rigidez genérica resulta de gran utilidad para determinar las condiciones de la rigidez por descomposición sin tener que considerar las posibles realizaciones espaciales. Sin embargo, por su importancia práctica, presentamos la condición correspondiente al estudio de frameworks.

Teorema 4.4.3 (Rigidez Infinitesimal por Descomposición Dispersa). *Un framework \mathcal{F} es infinitesimalmente rígido en \mathbb{R}^d si existe una función $h : \mathcal{V} \rightarrow \mathbb{N}_0$, tal que los subframe-*

works $\{\mathcal{F}_i : i \in \mathcal{C}\}$ cubren todos los vértices y son infinitesimalmente rígidos, la dimensión afín de $p(\mathcal{L}_i)$ es mayor o igual a $d - 1$, y \mathcal{F}_L es infinitesimalmente rígido.

Demostración. Se deduce que $\mathcal{F}_L \cup \mathcal{F}_i$ es infinitesimalmente rígido por unión de subframeworks (Lema 4.3.1), dado que los puntos en la intersección corresponden a $p(\mathcal{L}_i)$. Debido a que \mathcal{F}_i es infinitesimalmente rígido, se pueden eliminar los enlaces entre vértices de \mathcal{L}_i que existen en \mathcal{F}_L pero no en \mathcal{F}_i , sin afectar la rigidez. Repitiendo este proceso para todo $i \in \mathcal{C}$ se llega a que $\mathcal{F} = \mathcal{F}_L \cup \mathcal{F}_i \dots \cup \mathcal{F}_{|\mathcal{C}|}$ es infinitesimalmente rígido. \square

4.4.1. Descomposición Óptima

En el caso de descomposición densa, se vio que el alcance mínimo de rigidez (h_r) minimiza ambas métricas en \mathcal{H}_1 . Sin embargo, en el esquema de descomposición dispersa, es esperable que h_r no sea el mínimo, debido a que el conjunto de funciones válidas

$$\mathcal{H}_0(\mathcal{G}) := \{h : \text{cumple las condiciones del Teorema 4.4.2}\} \quad (4.13)$$

contiene a \mathcal{H}_1 . Surge naturalmente entonces el problema de seleccionar una función de alcance de forma óptima en el conjunto $\mathcal{H}_0(\mathcal{G})$. Es importante notar que no necesariamente existe $h \in \mathcal{H}_0$ que minimice ambas métricas \mathcal{D} y \mathcal{L} en simultáneo, como sí ocurre en \mathcal{H}_1 . Por ello, debemos proponer un criterio de optimalidad que pondere ambos objetivos. En primer lugar, es propicio restringir las funciones h a aquellas tales que $\mathcal{D}(h) \leq \mathcal{D}(h_r)$, de modo de no aumentar el tamaño máximo de los subgrafos. En segundo lugar, es deseable minimizar la carga \mathcal{L} , por lo tanto, proponemos como alcance óptimo a

$$h^* := \arg \min \{\mathcal{L}(h) : h \in \mathcal{H}_0, \mathcal{D}(h) \leq \mathcal{D}(h_r)\}. \quad (4.14)$$

Esta definición es conceptualmente útil, sin embargo, hallar h^* implica resolver un problema de optimización en un dominio discreto. Buscar de forma exhaustiva aquellas funcio-

nes que cumplen las condiciones del Teorema 4.4.2 en el conjunto $\{h : \mathcal{D}(h) \leq \mathcal{D}(h_r)\}$, no resulta un abordaje escalable. Este conjunto de búsqueda tiene $(\frac{1}{2}\mathcal{D}(h_r) + 1)^v$ elementos, es decir, su tamaño crece exponencialmente con la cantidad de vértices. Debido a la complejidad computacional que supone hallar h^* , proponemos la utilización de una descomposición subóptima \hat{h} . La misma se obtiene, partiendo de h_r , y eliminando iterativamente el subgrafo que provoca el mayor decrecimiento de la función \mathcal{L} en el conjunto de búsqueda. El procedimiento se muestra en el Algoritmo 1, que aproxima el valor de h^* en menos de $v(v+1)/2$ pasos. En Fig. 4.16 se muestra el alcance subóptimo obtenido para los frameworks de Fig. 4.5. Comparando ambas imágenes se observa como la descomposición dispersa es capaz de explicar la rigidez del grafo a partir de un conjunto reducido de subgrafos, lo que disminuye radicalmente la superposición de los mismos.

Algoritmo 1: Búsqueda de descomposición subóptima.

Input: \mathcal{G} (grafo rígido), h_r (alcance mínimo de rigidez)

Output: \hat{h} (alcance subóptimo)

Comment: La función `descomposicion_valida(h)` verifica si los subgrafos cubren todo el grafo, y si los mismos están rígidamente vinculados.

$\hat{h} \leftarrow h_r$;

terminar \leftarrow **False**;

while not terminar **do**

$\min \leftarrow \infty$;

 trivial $\leftarrow \{\}$;

for $i = 1 : v$ **do**

if $\hat{h}(i) > 0$ **then**

$s \leftarrow \hat{h}$;

$s(i) \leftarrow 0$;

if `descomposicion_valida(s)` **then**

$carga \leftarrow \mathcal{L}(s)$;

if $carga < \min$ **then**

$\min \leftarrow carga$;

 trivial $\leftarrow i$;

if $\min < \infty$ **then**

$\hat{h}(\text{trivial}) \leftarrow 0$;

else

 terminar \leftarrow **True**

4.4.2. Simulaciones

Resulta importante cuantificar la disminución de las métricas obtenidas con la descomposición dispersa, para lo cual realizamos simulaciones de forma tal de obtener gráficas análogas a la Fig. 4.9 y a la Fig. 4.10. Para ello, utilizamos el modelo de framework aleatorio propuesto, con $\alpha = 0$, y el Algoritmo 1 para la obtención de la descomposición

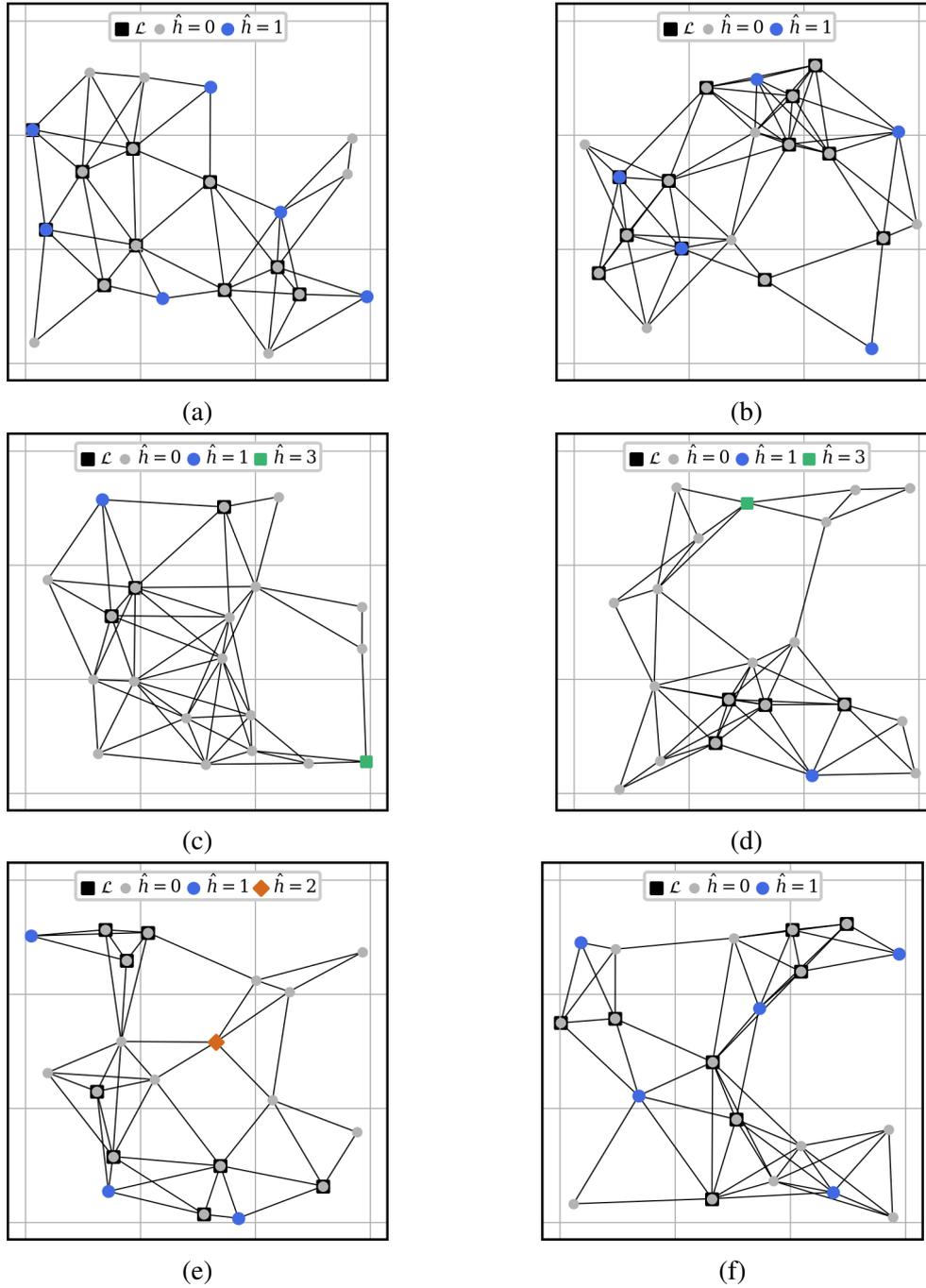


Figura 4.16: Alcances mínimos de rigidez. Los nodos son identificados según su alcance subóptimo, y según si pertenecen al conjunto de vínculos \mathcal{L} .

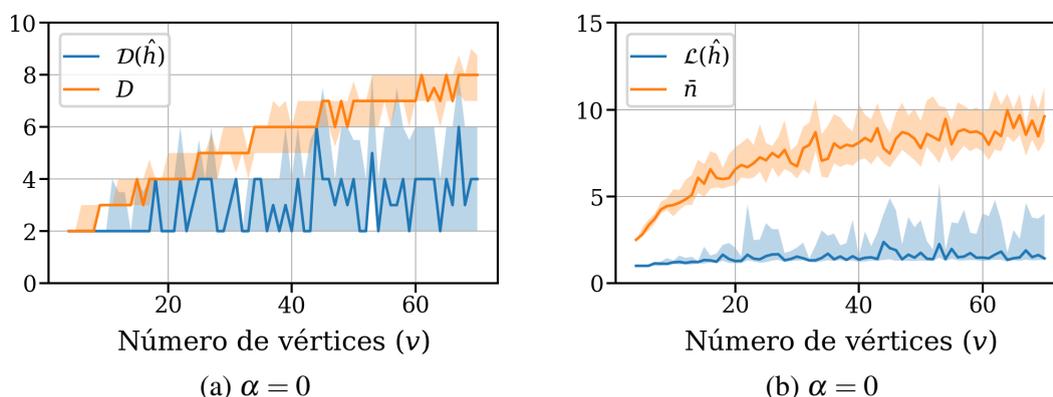


Figura 4.17: Comparación, con sus parámetros de referencia, del máximo retardo \mathcal{D} y de la carga de comunicaciones \mathcal{L} asociados a una descomposición subóptima. La línea sólida indica la mediana del conjunto de 50 muestras y el área coloreada representa la zona delimitada por los cuartiles Q_1 y Q_3 .

dispersa. Los resultados se muestran en Fig. 4.17. Como podemos observar, tanto el máximo retardo, como la carga en los enlaces, se encuentran por debajo de los parámetros de referencia D y \bar{n} , respectivamente. Es decir, no solo se logra que el máximo retardo sea, en general, menor al diámetro, sino que además, la utilización de los canales de comunicación resulta incluso menor que la correspondiente a una estrategia distribuida, lo cual es imposible con la descomposición densa. Más aún, se nota como las métricas de la descomposición tienen un crecimiento, con respecto a la cantidad de vértices, menor a los parámetros de referencia, indicando la escalabilidad del esquema. Además, esto se logra para $\alpha = 0$, es decir, sin necesidad de aumentar el rango de conexión, y por lo tanto, la cantidad de enlaces.

4.5. Resumen

En este capítulo presentamos una formulación novedosa de descomposición en sub-frameworks que permite expresar la propiedad de rigidez de un framework como una propiedad distribuida a lo largo de la red. En primer lugar, mostramos las limitaciones

que tiene expresar la rigidez en función del framework, especialmente cuando el tamaño del mismo crece. Luego definimos formalmente el concepto de subframework y presentamos los teoremas que permiten expresar la rigidez en función de los mismos, cuando la descomposición del framework es densa. Más adelante, propusimos métricas orientadas a medir el grado de descentralización obtenido en una determinada descomposición, con miras en el control de redes de robots. Para ello tuvimos en cuenta las características de la red, y propiedades fundamentales de las mismas, como el *delay* y la carga comunicacional, claves en el diseño de los esquemas de control y de los protocolos de comunicación, como se detallará en el Capítulo 5. A partir de dichas métricas y de la definición del alcance mínimo de rigidez, realizamos simulaciones para comparar las características de la descomposición densa con respecto la definición clásica de rigidez. Pudimos ver que en la mayoría de los frameworks evaluados el tamaño de los subframeworks es visiblemente menor que el diámetro del grafo, y que, por lo tanto, el retardo asociado al intercambio de información suele ser menor en el primer caso. Más aún, pudimos ver que el tamaño de los subframeworks tiende a permanecer constante con la cantidad de vértices, a diferencia del diámetro que crece al menos de forma logarítmica. Este resultado es muy prometedor en cuanto a la escalabilidad del esquema de subframeworks. Los Teoremas 4.3.1 y 4.3.2, y el Lema 4.3.2, resultados relativos a la descomposición densa, fueron presentados en [72].

Finalmente, presentamos una modificación del esquema, que llamamos descomposición dispersa, la cual permite reducir el número necesario de subframeworks para determinar la rigidez del conjunto. Esto permite, no solo la reducción tanto del retardo máximo y de la carga en las comunicaciones, sino que además libera robots de la tarea de tener asociado un subframework. Esto puede ser aprovechado en la práctica, mediante la asignación de otras tareas a estos agentes, optimizando la utilización de los recursos.

Capítulo 5

Control de Rigidez por Subframeworks

En este capítulo se aplica la teoría de rigidez por subframeworks desarrollada, para el control coordinado del mantenimiento de la rigidez de una red multirobot. En primer lugar, se propone una adaptación de los algoritmos de posicionamiento de redes, que prescinde de los esquemas de consenso, mejorando su escalabilidad. También, se propone un esquema de control de tipo *swarm* por optimización y se desarrollan las funciones de costo necesarias para cumplir los múltiples objetivos enunciados. Luego, se presenta un protocolo que permite intercambio de información necesario para poder aplicar dicha ley de control de forma descentralizada por los agentes de la red. Por último, se valida el esquema propuesto mediante simulaciones computacionales que pretenden emular distintos escenarios de interés.

5.1. Introducción

Una de las áreas donde más ha proliferado la utilización de frameworks es en el control de redes multirobot, donde los agentes son modelados con vértices en el espacio, y las aristas del grafo representan enlaces de comunicación entre los mismos. Este esquema es

apto para formular dos problemas de naturaleza análoga: (i) el control de una formación de robots móviles de posición conocida, a partir de la especificación de un subconjunto de distancias relativas [110], y (ii) el posicionamiento de los robots a partir de un subconjunto de mediciones de distancia relativa [111, 112]. En ambos problemas, la rigidez del framework tiene un rol fundamental, ya que es la propiedad que permite sus resoluciones. En este capítulo nos enfocaremos en el problema de posicionamiento.

En una red de agentes móviles, las distancias relativas se pueden estimar aprovechando el intercambio de señales entre dichos agentes, a través de canales de comunicación inalámbricos. Cuando la topología del grafo determinado por el conjunto de enlaces de comunicación es invariante en el tiempo, el posicionamiento se puede resolver de la siguiente manera. En primer lugar, se debe asegurar que el framework inducido por los agentes y sus enlaces sea rígido, luego se aplica una técnica de estimación de las distancias relativas (TOA, AOA, RSSI) [113, 114], y a partir de las mismas un algoritmo de posicionamiento colaborativo [115, 66]. Sin embargo, en el campo de la robótica móvil esto no ocurre. Esto se debe a que a medida que los robots se mueven para realizar una misión, los enlaces de comunicación pueden crearse o destruirse. La eliminación de enlaces puede ocasionar la pérdida de rigidez. Por lo tanto, para que el algoritmo de posicionamiento funcione, es necesario incorporar el mantenimiento de la rigidez a los objetivos del esquema de control. En este sentido, se han realizado importantes esfuerzos para diseñar estrategias de control que permitan mantener o recuperar la rigidez en redes con topología dinámica. Las técnicas propuestas en la literatura, pueden ser clasificadas como de *movilidad* [97, 71, 70] o de *conectividad* [116, 108]. En las primeras, la estrategia adoptada se basa en el control de la posición de los robots, modelando la pérdida de rigidez como un obstáculo en el espacio que debe ser evadido. En las segundas, se controla la creación y eliminación de enlaces por métodos combinatorios. En ambos casos, desarrollar algoritmos completamente descentralizados es una dificultad, debido a la naturaleza global de la

rigidez.

Para el mantenimiento de la rigidez, las estrategias de movilidad proponen la utilización del autovalor de rigidez como medida del grado de rigidez de un framework. Un controlador de gradiente descendiente guía el movimiento de los robots de forma tal de mantener positivo el autovalor de rigidez, permitiendo cambios en la topología de la red. La estrategia común adoptada para la implementación descentralizada se basa en la utilización de filtros de consenso que permiten obtener localmente una estimación del autovalor de rigidez y del correspondiente autovector, parámetros necesarios para computar el gradiente. El desempeño del control depende, en gran medida, de la capacidad de estimar rápidamente el autovalor y el autovector de rigidez, que son propiedades globales de toda la red. No obstante, el mínimo número de iteraciones necesarias para la convergencia de estos estimadores es, en el mejor caso, igual al diámetro de la red [81]. Este, a su vez, aumenta con la cantidad de vértices, lo que configura la principal limitación a la escalabilidad de las técnicas de movilidad.

Alternativamente, en [116] se desarrolla un abordaje de conectividad para el mantenimiento de rigidez. Reglas locales para la adición y remoción de enlaces entre agentes son empleadas para dictar los movimientos de los robots. La naturaleza combinatoria del esquema lo dota de mayor eficiencia, dado que únicamente se requiere realizar cálculos e intercambio de información durante las transiciones en la topología de la red. No obstante, la teoría empleada en dicho esquema es válida únicamente para frameworks realizados en el espacio bidimensional, limitando su aplicación en escenarios reales. Un trabajo más reciente [108], presenta una técnica distribuida para recuperar la falta de rigidez ocasionada por la pérdida de conexión entre un par de agentes. En este trabajo se define, para cada vértice, un subframework de alcance unitario, y se prueba que la rigidez de los mismos implica la rigidez del framework. Cuando ocurre una pérdida de conexión que afecta la rigidez de un subframework, una nueva configuración es elegida para recuperar localmente

la rigidez del mismo, a partir de la creación de nuevos enlaces con agentes dentro del rango de conexión. La condición de alcance unitario impuesta a los subframeworks restringe la aplicabilidad del esquema, debido a que solo es válida para frameworks con alcance mínimo de rigidez unitario. Los abordajes de conectividad poseen mejor escalabilidad que los métodos de movilidad, dado que las acciones de control se toman con datos locales, sin requerir intercambio de información global. No obstante, son conservativos en cuanto a la cantidad de enlaces necesarios, debido a que conexiones redundantes pueden no ser detectadas de forma local. Estos esquemas tampoco consideran una métrica de rigidez asociada al framework, por lo tanto, no consideran un criterio de optimización asociado a la posición de los agentes en la formación.

Las limitaciones que presentan estas estrategias pueden ser superadas aprovechando los esquemas de descomposición por subframeworks presentados en el Capítulo 4. Los mismos permiten expresar la rigidez del framework en términos locales a través de la rigidez de sus subframeworks. Los autovalores de rigidez asociados a los mismos surgen naturalmente como métricas locales de rigidez. En este capítulo presentamos estrategias de control descentralizado de mantenimiento de la rigidez que: (i) no requieren el intercambio global de información, (ii) optimizan las posiciones de los robots, y (iii) son aplicables a cualquier framework.

5.2. Algoritmo de Posicionamiento

Para implementar un algoritmo de control, cada robot debe tener una estimación de su posición, para lo cual se cuenta con mediciones de distancia relativa entre nodos vecinos. Es claro que solo es posible localizar los robots de forma unívoca (módulo una transformación euclídea) si el framework es globalmente rígido. Sin embargo, este problema es clase NP-difícil y no se conoce un método eficiente para su solución con convergencia global [117]. Por lo tanto, se asume que se tiene una estimación de las posiciones iniciales de los

agentes, entonces se puede garantizar convergencia si el framework es infinitesimalmente rígido.

De los algoritmos de control de rigidez mencionados, únicamente [71] considera en simultáneo el problema de localización de los robots. El algoritmo de posicionamiento utilizado allí, se basa en el propuesto por [115]. En él, cada robot intercambia las posiciones estimadas con sus vecinos, y se aplica un protocolo de gradiente descendiente (GD) utilizando el factor de escala de Barzilai-Borwein (α_{BB}), ampliamente utilizado en las técnicas de optimización. Este factor, que va evolucionando a medida que avanza la misión, es único para todos los agentes, por lo que es obtenido de forma descentralizada, mediante un algoritmo de consenso, lo que afecta la escalabilidad. Para evitar esto, presentamos una adaptación de este algoritmo de posicionamiento, que no requiere el intercambio de información global, por lo tanto, su desempeño no depende de la cantidad de vehículos en la red.

5.2.1. Formulación

Sea (\mathcal{G}, p) un framework, $f : \mathbb{R}^{dv} \rightarrow \mathbb{R}^e$ la función que computa el vector de distancias relativas, y sea $z \in \mathbb{R}^e$ el vector de mediciones asociado. Se desea hallar una estimación de la posición \hat{p} que minimice el error cuadrático $\xi(\hat{p}) = \frac{1}{2} \|z - f(\hat{p})\|^2$. Aplicando el algoritmo de gradiente descendiente se obtiene $\dot{\hat{p}} = \alpha_{BB} \mathbf{R}(\hat{p})^T (z - f(\hat{p}))$, el cual puede reescribirse como

$$\forall i \in \mathcal{V} : \quad \dot{\hat{p}}_i = \alpha_{BB} R_i(\hat{p})^T (z_i - f_i(\hat{p})), \quad \hat{p}_i(t_0) = \hat{p}_{i,t_0} \quad (5.1)$$

donde $z_i, f_i(\hat{p}) \in \mathbb{R}^{n(i)}$ corresponden a las distancias incidentes a i únicamente, y R_i es la matriz $n(i) \times d$ de derivadas parciales de $f_i(\hat{p})$ respecto de \hat{p}_i . En este trabajo, proponemos

la siguiente modificación de (5.1)

$$\forall i \in \mathcal{V} : \begin{cases} \dot{\hat{p}}_i = K_i(\hat{p})(z_i - f_i(\hat{p})), & \hat{p}_i(t_0) = \hat{p}_{i,t_0} \\ \dot{P}_i = -K_i(\hat{p})R_i(\hat{p})P_i, & P_i(t_0) = P_{i,t_0} \end{cases} \quad (5.2)$$

donde $K_i = P_i R_i^T (R_i P_i R_i^T + \sigma^2 I_{n(i)})^{-1}$ es la ganancia del filtro, P_i la matriz de covarianza asociada a la estimación \hat{p}_i y σ^2 es la varianza asociada al ruido de medición. Notar que (5.2) es de naturaleza distribuida y puede ser implementada de forma independiente por cada robot únicamente conociendo la posición estimada de sus vecinos. El esquema propuesto configura un Filtro de Kalman aproximado, que indicamos según las siglas FKA. Esto se debe a que el mismo emplea una versión aproximada de la matriz de covarianza de $dv \times dv$ de la posición \hat{p} , la cual desestima las covarianzas cruzadas, es decir, los bloques no diagonales de la misma son nulos. Por lo tanto, cada robot no es capaz de incluir en su filtro el error inducido por la incerteza en las posiciones que recibe de sus vecinos. No obstante, esta aproximación es la que permite su implementación de forma distribuida. A continuación probamos que este filtro converge al mismo conjunto que el gradiente descendiente.

Teorema 5.2.1. *La dinámica (5.2) converge a un mínimo local de ξ , si la estimación inicial se encuentra lo suficientemente cerca.*

Demostración. En primer lugar, analizamos la dinámica del error inducida por (5.2)

$$\dot{\xi}(\hat{p}) = \sum_{i=1}^v \frac{\partial \xi}{\partial \hat{p}_i} \dot{\hat{p}}_i = -\frac{1}{2} \sum_{i=1}^v (z_i - f_i(\hat{p}))^T R_i(\hat{p}) K_i(\hat{p}) (z_i - f_i(\hat{p})) \leq 0.$$

Para ver por qué $\dot{\xi}(\hat{p}) \leq 0$, en primer lugar observamos que $R_i P_i R_i^T \geq 0$, y que $\text{Nul}(R_i P_i R_i^T) = \text{Nul}(R_i^T)$, ya que dado un $x \in \mathbb{R}^{n(i)}$, se tiene $x^T R_i P_i R_i^T x = 0$ si y solo si $R_i^T x = 0$, por ser $P_i > 0$. Además, se observa que $R_i K_i \geq 0$ y que $\text{Nul}(R_i K_i) = \text{Nul}(R_i P_i R_i^T)$, puesto que sus

autovalores se relacionan por medio de

$$\lambda(R_i K_i) = \frac{\lambda(R_i P_i R_i^T)}{\lambda(R_i P_i R_i^T) + \sigma^2}.$$

Entonces, se llega a que $\text{Nul}(R_i K_i) = \text{Nul}(R_i^T)$. Luego $\dot{\xi}(\hat{p}) = 0$ si y solo si \hat{p} es punto de equilibrio de la dinámica colectiva dada por (5.1). Por lo tanto, este protocolo, posee el mismo conjunto de convergencia que el gradiente descendiente. \square

En un escenario sin ruido de medición, la actualización (5.2) asegura que \hat{p} converge a una formación coincidente con las verdaderas posiciones (módulo transformaciones euclídeas), siempre y cuando el framework sea infinitesimalmente rígido y la estimación inicial $\hat{p}(t_0)$ sea lo suficientemente cercana a la verdadera. En Fig. 5.1 y Fig. 5.2 se muestra el desempeño del algoritmo de posicionamiento propuesto mediante una implementación discreta de las dinámicas (5.1) y (5.2). Se utiliza, como ejemplo, un framework infinitesimalmente rígido de $v = 7$ agentes, $e = 14$ enlaces, situados en una región de $20\text{m} \times 20\text{m}$ y un ruido de medición con desviación estándar $\sigma = 3\text{m}$. En la primera figura se ilustra el framework y las trayectorias de \hat{p} obtenidas, comparando respecto de un factor de escala fijo $\alpha = 0.15$ ajustado manualmente, y respecto de α_{BB} . En la segunda se muestra la evolución error cuadrático medio (ξ/e) obtenido. Se observa que el Filtro de Kalman aproximado presenta un desempeño superior tanto en velocidad de convergencia como en suavidad de la trayectoria. Además, notamos una excesiva sensibilidad del factor de Barzilai-Borwein ante la presencia de ruido, lo que puede ser comprometedor para su aplicación en redes de robots.

Es claro que el filtro propuesto solo es capaz de estimar correctamente la posición de los vehículos módulo una roto-traslación del framework. Por lo tanto, información adicional debe ser provista a los mismos. La solución propuesta es dotar a d robots de mediciones de posición absoluta, lo que permite completar el esquema de localización en

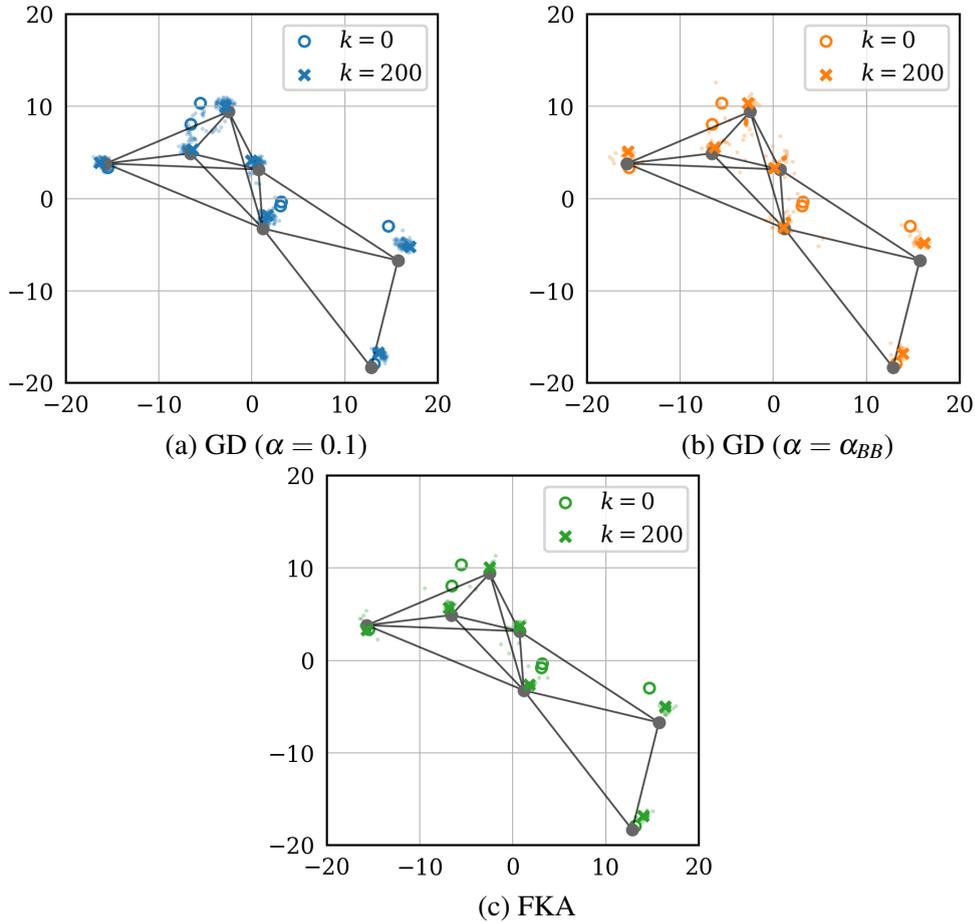


Figura 5.1: Evolución de la estimación entre los instantes $k = 0$ y $k = 200$.

el caso d -dimensional. Hallar, de forma óptima y mediante técnicas descentralizadas, el subconjunto de robots a los cuales proveer de información de posición, es un problema en sí mismo que no fue abordado en esta tesis.

5.3. Control de Mantenimiento de la Rigidez

Para validar el enfoque de rigidez por subframeworks, proponemos un esquema descentralizado de control de movilidad, definido como un problema de optimización similar al presentado en [97] y luego validado en [71]. La principal novedad del esquema aquí

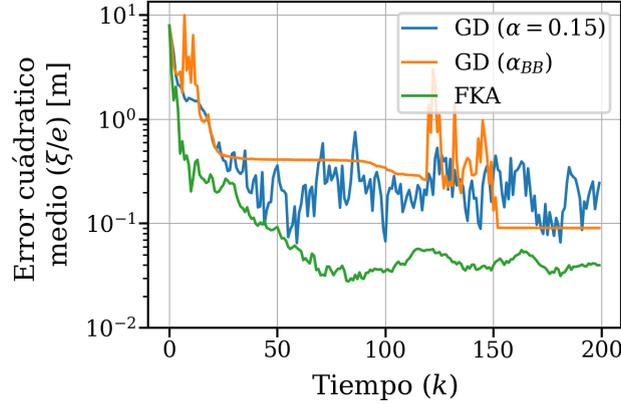


Figura 5.2: Comparación de desempeño del Filtro de Kalman Aproximado y el Gradiente Descendiente.

propuesto es que el mismo se vale de los aportes teóricos presentados en el Capítulo 4 para aplicar una técnica de control de mantenimiento de la rigidez por subframeworks, la cual es de naturaleza descentralizada. La ventaja más importante de este esquema es que permite realizar el control de la rigidez, sin implementar algoritmos de consenso, que como ya se discutió, limitan la escalabilidad.

La ley de control permite comandar la red de robots móviles cuyo posicionamiento se logra utilizando mediciones de distancia entre nodos adyacentes. El controlador tiene dos objetivos obligatorios: (i) la evasión de colisiones entre robots, y (ii) el mantenimiento de la rigidez del framework, condición necesaria para la correcta localización de los vehículos. Además, puede contar con metas específicas de la misión a desarrollar. Los múltiples objetivos son incluidos en el esquema mediante funciones de costo que dependen de la posición de los robots p_i . Si $\mathcal{J}_c, \mathcal{J}_r : \mathbb{R}^{dv} \rightarrow \mathbb{R}$ son funciones que dependen del estado del sistema en el tiempo t , $p(t)$, y que respectivamente crecen sin límite a medida que se pierde la rigidez o dos vehículos se aproximan entre sí, entonces es posible controlar estos objetivos mediante la minimización de ambas funciones. El objetivo de control se propone entonces como el guiado de los robots a seguir una trayectoria suave $p(t)$ con $t \in [t_0, t_f]$, de forma tal de minimizar el costo total $\mathcal{J}(p) := \mathcal{J}_c(p) + \mathcal{J}_r(p) + \mathcal{J}_a(p)$,

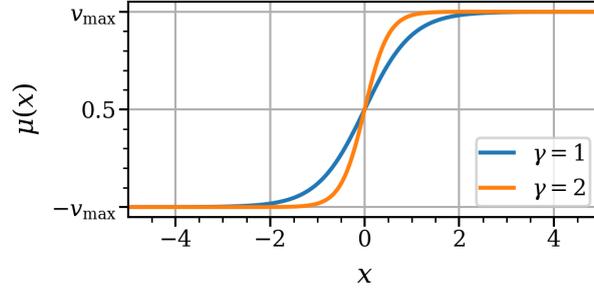


Figura 5.3: Modelo de saturación dado por la sigmoide logística.

donde \mathcal{J}_a representa un costo extra, que depende de la aplicación, diseñado para cumplir un objetivo específico. La estrategia adoptada en [97, 70] es el empleo del protocolo continuo de gradiente descendiente, de forma tal de guiar al conjunto de robots hacia un mínimo local de $\mathcal{J}(p)$. Aquí, proponemos una adaptación del mismo:

$$\dot{p}_i(t) = -\mu \left(\frac{\partial \mathcal{J}}{\partial p_i}(p(t)) \right), \quad p_i(t_0) = p_{i,t_0} \quad (5.3)$$

donde $\mu : \mathbb{R}^d \rightarrow \mathbb{R}^d$ es una función de saturación que mantiene el vector velocidad dentro de los límites aceptables. En este trabajo empleamos la función de saturación elemento-a-elemento dada por la sigmoide (Fig. 5.3):

$$\mu(x) = v_{\max} \frac{1 - e^{-\gamma x}}{1 + e^{-\gamma x}}, \quad (5.4)$$

donde $\gamma > 0$ regula la pendiente. Esta función de saturación tiene el efecto de acotar el valor absoluto de la velocidad en cada componente, lo que se traduce en un límite para la norma-infinito del vector velocidad $\|\dot{p}_i\|_\infty$.

A continuación desarrollamos los modelos necesarios para poder escribir \mathcal{J} en términos de p y computar (5.3) de forma descentralizada.

5.3.1. Modelo Dinámico de los Robots

Aplicar la ley (5.3) directamente es posible si los vehículos disponen de un controlador de seguimiento de velocidad [118, 119], una funcionalidad típicamente incluida en los vehículos robóticos comerciales. En pocas palabras, estos controladores de menor nivel, resuelven el problema de hallar la acción u_i tal que $\dot{p}_i = F(p_i, u_i)$ sea igual a (5.3), donde $F : \mathbb{R}^d \times \mathbb{R}^{d_u} \rightarrow \mathbb{R}^d$ es el modelo dinámico de los vehículos. Nuestro abordaje se enfoca en robots móviles con estas capacidades incluidas, lo que nos permite modelar la dinámica de la posición de los agentes como un integrador simple: $\dot{p}_i = u_i$.

5.3.2. Modelo Dinámico del Grafo

Debido a que los rangos de conexión son limitados, a medida que los robots se mueven, los enlaces de comunicación existentes pueden perderse, a la vez que pueden crearse otros nuevos. El esquema debe ser capaz de adaptarse a estas topologías variantes. Para ello, adoptamos un abordaje clásico para asignar pesos $w : \mathcal{E} \rightarrow [0, 1]$ a los enlaces del grafo. La función w debe ser definida de tal forma que, si ρ es el radio de conexión entre agentes,

$$w_{ij} \approx \begin{cases} 1, & \|p_i - p_j\| \ll \rho, \\ 0, & \|p_i - p_j\| \gg \rho. \end{cases}$$

Con el fin de obtener este comportamiento mediante una función suave, elegimos la función sigmoide logística dada por

$$w_{ij} := \frac{1}{1 + e^{-\beta(\tilde{\rho} - \|p_i - p_j\|)}}, \quad (5.5)$$

donde $\tilde{\rho} \approx \rho$, y $\beta > 0$ es el parámetro que regula la rapidez del decaimiento. El parámetro $\tilde{\rho}$ se emplea de la siguiente forma. Para el mantenimiento de la rigidez, se desea que

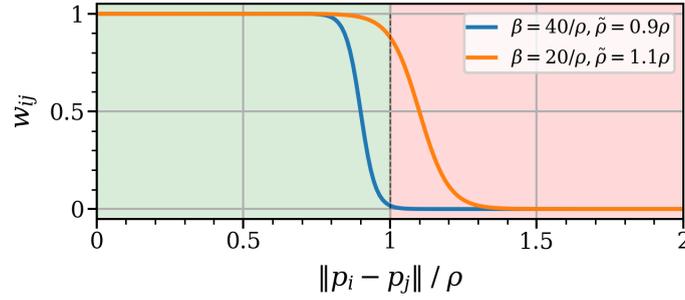


Figura 5.4: Modelo de conexión y desconexión dados por la sigmoide logística.

que controlador acuse la inminente pérdida de un enlace ij antes de que $\|p_i - p_j\| > \rho$, es decir, dentro de la zona verde, como se muestra en Fig. 5.4. Por otro lado, cuando se quiere propiciar la pérdida de enlaces, por ejemplo para reducir la carga de la red, es necesario que el controlador aleje a los nodos i, j entre sí, hasta que $\|p_i - p_j\| > \rho$. Para ello, es necesario que el costo asociado al enlace solo sea cercano a cero cuando el mismo ya se encuentra desconectado. Según los resultados obtenidos por simulación, es apropiado situar $\tilde{\rho} \approx 1.1\rho$, $\beta \approx 20/\rho$ para evitar la desconexión y $\tilde{\rho} \approx 0.9\rho$, $\beta \approx 40/\rho$ para propiciar la misma. Los pesos w_{ij} son incorporados al esquema de control como se detallará en las siguientes subsecciones.

5.3.3. Evasión de colisiones

La evasión de colisiones entre agentes es obligatoria en cualquier aplicación real, por razones evidentes. Para ello, es suficiente considerar únicamente nodos vecinos, ya que el grafo de conexiones queda determinado por la distancia euclídea. Para diseñar la acción de control que permite la evasión, utilizamos la función potencial

$$\mathcal{J}_c(p) := \sum_{i \sim j} \|p_i - p_j\|^{-a}, \quad a > 0 \quad (5.6)$$

que crece indefinidamente cuando dos agentes se acercan, y tiende a cero cuando aumenta la distancia entre todos ellos. En base a lo obtenido mediante simulaciones, el valor $a \approx 1$ presenta buenos resultados. El negativo del gradiente de $\mathcal{J}_c(p)$ respecto de la posición del i -ésimo robot

$$-\frac{\partial \mathcal{J}_c}{\partial p_i} = \sum_{j: j \sim i} a \|p_i - p_j\|^{-(a+2)} (p_i - p_j), \quad (5.7)$$

representa una acción de repulsión entre agentes. La misma puede ser computada por cada nodo únicamente a través de interacciones con sus vecinos.

5.3.4. Mantenimiento de la Rigidez

Para mantener la rigidez del framework durante la misión, es necesario restringir el movimiento de los robots. La estrategia adoptada es mantener la rigidez de cada subframework mediante el guiado los agentes. Para ello, definimos una versión pesada del laplaciano corresponde al i -ésimo subframework como

$$\mathbf{L}_i := \sum_{j, k \in \mathcal{V}_i: j \sim k} w_{jk} \ell_{jk} \quad (5.8)$$

donde la matriz ℓ_{jk} es definida como en (2.10). Entonces, para controlar la rigidez de i -ésimo subframework, se mantiene la positividad del autovalor de rigidez λ_i asociado a \mathbf{L}_i . Luego, definimos la siguiente función que crece indefinidamente si el autovalor de rigidez de algún subframework se aproxima a cero

$$\mathcal{J}_r(p) := \sum_{i \in \mathcal{V}} \mathcal{J}_{r,i}(p), \quad \mathcal{J}_{r,i} := \lambda_i^{-b}, \quad b > 0. \quad (5.9)$$

Se pudo observar que $b \in [1/3, 1/2]$ es un buen rango para ajustar este parámetro. Por lo tanto, la rigidez del framework en su totalidad es mantenida si \mathcal{J}_r se mantiene acotado,

como consecuencia del Teorema 4.3.2. Luego, el gradiente descendiente resulta

$$-\frac{\partial \mathcal{J}_r}{\partial p_i} = - \sum_{j: i \in \mathcal{V}_j} \frac{\partial \mathcal{J}_{r,j}}{\partial p_i} = \sum_{j: i \in \mathcal{V}_j} b \lambda_j^{-(b+1)} \boldsymbol{\nu}_j^T \frac{\partial \mathbf{L}_j}{\partial p_i} \boldsymbol{\nu}_j, \quad (5.10)$$

donde $\boldsymbol{\nu}$ es un autovector de norma unitaria asociado al autovalor de rigidez. Para llegar a esta expresión se utilizó la fórmula de la derivada de los autovalores de una matriz simétrica, cuyos detalles se pueden ver en [97, 70]. Notar que (5.10) contiene únicamente términos correspondientes a los subframeworks que incluyen al nodo i , dado que la rigidez de los restantes no depende del estado de dicho robot.

La acción de control empleada en [97, 71] es idéntica a (5.10) pero con un único término correspondiente a todo el framework. Como se discutió en el Capítulo 3, el autovalor de rigidez tiende a acercarse a cero a medida que aumenta el diámetro de la red. Esto significa que la evasión de la rigidez mediante (5.10) implica la tendencia del controlador a reducir el diámetro del grafo. Esto representa otra limitación a la escalabilidad del control presentado en dichos trabajos, dado que al aumentar la cantidad de agentes el diámetro tiende a crecer, en consecuencia, la acción de control fuerza a los mismos a acercarse entre sí para aumentar la cantidad de conexiones. Esto evita que se pueda usar dicha técnica para controlar un grupo numeroso de robots. Esta limitación se resuelve empleando el esquema de subframeworks, si los mismos tienen un alcance que no aumenta con la cantidad de vértices.

5.3.5. Carga en la red

Como fue analizado en el Capítulo 4, la carga sobre los enlaces de comunicación puede ser excesiva cuando los mismos se utilizan para transmitir mensajes en un esquema de subframeworks. Para minimizar esta carga mediante acciones de control, proponemos la utilización de la métrica \mathcal{L} (4.5) como una función de costo. La misma es adaptada

para incorporar los pesos asignados a los enlaces

$$\mathcal{J}_l(p) := \frac{1}{v} \sum_{i \sim j} w_{ij} (c_i + c_j). \quad (5.11)$$

Luego, el anti-gradiente resulta

$$-\frac{\partial \mathcal{J}_l}{\partial p_i} = -\frac{1}{v} \sum_{j: j \sim i} (c_i + c_j) \frac{\partial w_{ij}}{\partial p_i}, \quad (5.12)$$

Notar que, si bien los coeficientes c_i, c_j dependen de la topología de la red, y esta, de las posiciones, no consideramos esta dependencia en (5.12). Esto se hace debido a que escribir los coeficientes c_i, c_j en función de p , implica escribir las distancias geodésicas $g(i, j)$ en función de las posiciones, lo que agrega una complejidad excesiva al esquema. Dado que (5.11) es el grado promedio de la red, afectado por coeficientes $c_i + c_j$, la ley (5.12) equivale a la aplicación del gradiente descendiente de $(1/v) \sum_{i \sim j} w_{ij}$, afectando cada término por un peso relativo $c_i + c_j$, cuyo efecto es priorizar la eliminación de los enlaces más cargados. Dichos coeficientes, pueden ser obtenidos de forma descentralizada por cada robot aprovechando el esquema de intercambio de información presentado en la Subsección 5.4.1.

5.3.6. Ley de control

Con base en lo presentado, definimos la acción de control, como el anti-gradiente de la función de costo $\mathcal{J}(p) := \mathcal{J}_c(p) + \mathcal{J}_r(p) + \mathcal{J}_l(p)$. Es decir, cada robot sigue la ley

$$\begin{aligned} \dot{p}_i = -\frac{\partial \mathcal{J}}{\partial p_i} = & \sum_{j: j \sim i} \left(a \frac{p_i - p_j}{\|p_i - p_j\|^{(a+2)}} - \frac{c_i + c_j}{v} \frac{\partial w_{ij}}{\partial p_i} \right) + \\ & \sum_{j: i \in \mathcal{V}_j} b \lambda_j^{-(b+1)} \nu_j^T \frac{\partial \mathbf{L}_j}{\partial p_i} \nu_j, \end{aligned} \quad (5.13)$$

donde se emite la función de saturación μ .

Como consecuencia de la utilización de subframeworks, (5.13) puede ser computada de forma descentralizada por los nodos, a partir del intercambio de información dentro de cada subframework. Es interesante notar que, si bien los alcances de los subframeworks son considerados constantes, esta ley permite que los robots puedan entrar y salir de los subframeworks, siempre y cuando eso no haga que alguno de ellos pierda rigidez. En la Sección 5.4 presentamos el intercambio de información necesario para que cada agente pueda computar esta ley de control de forma descentralizada y discutimos los detalles de la estabilidad de la misma.

5.4. Implementación Descentralizada

5.4.1. Intercambio de Información

Un esquema de intercambio de mensajes debe implementarse para que los robots accedan a la información necesaria para aplicar el término correspondiente a la rigidez. Para ello se asume que el grafo es rígido en el instante inicial t_0 .

Inicialización

El protocolo comienza con un proceso donde los robots no se mueven y realizan un intercambio de mensajes cuyo objetivo es que puedan determinar el alcance mínimo de rigidez h_r de forma descentralizada. Para ello, los robots envían a sus vecinos, por única vez, un mensaje de estado:

$$\text{state_init_msg}(i) = (i, \eta_T, \eta_{\text{máx}}, \hat{p}_i) \quad (5.14)$$

donde η_T (inicializado en 1) es la cantidad de enlaces que atravesó el mensaje, y $\eta_{\text{máx}}$ es la cantidad de enlaces que debe recorrer, y la estimación de la posición \hat{p}_i . El parámetro $\eta_{\text{máx}}$ puede ser inicializado igual al diámetro, si este es conocido; en caso contrario es inicializado en $v - 1$, cota superior para el diámetro de grafos conexos. Cada vez que un robot j recibe un mensaje $\text{state_init_msg}(i)$, extrae (i, \hat{p}_i) y reenvía el mensaje $(i, \eta_T + 1, \eta_{\text{máx}}, \hat{p}_i)$ actualizando la cantidad de enlaces que atravesó el mismo. En caso de que el mensaje recorra toda la red, el valor de η_T será igual a $\eta_{\text{máx}}$, y cuando eso sucede el mismo ya no es reenviado. A medida que transcurre este intercambio, que finaliza como máximo en $\eta_{\text{máx}}$ pasos, los agentes reciben mensajes de posición de robots cada vez más lejanos. A partir de los pares (i, \hat{p}_i) recibidos, los mismos construyen el subframework del cual son centro, agregando los enlaces existentes, dado que el rango de conexión ρ es conocido. Una vez que el subframework construido es rígido, define $h_r(i)$ igual a la distancia al nodo más alejado en dicho subframework. Entonces, el nodo deja de extraer la posición de los mensajes de estado que recibe, y solo los retransmite.

Cuando finaliza esta primera parte, los agentes tienen construido su subframework, definido con el alcance mínimo de rigidez. Entonces, cada uno envía, por única vez, un mensaje que llamamos de acción:

$$\text{action_init_msg}(i) = (i, \eta_T, h_r(i)), \quad (5.15)$$

donde η_T es inicializado en 1. El mensaje $(i, \eta_T, h_r(i))$ es retransmitido por los agentes actualizando el alcance $(i, \eta_T + 1, h_r(i))$ hasta que $\eta_T = h_r(i)$. De esta manera, el mensaje de acción llega únicamente a los agentes en el i -ésimo subframework. Una vez que finaliza este proceso, cada agente conoce el conjunto de subframeworks al que pertenece, y la distancia a los correspondientes centros, contando la mínima cantidad de enlaces atravesados por cada mensaje de acción recibido de un mismo nodo. Con esta información, cada nodo es capaz de obtener su propio coeficiente de carga c_i (5.11). Este proceso luego

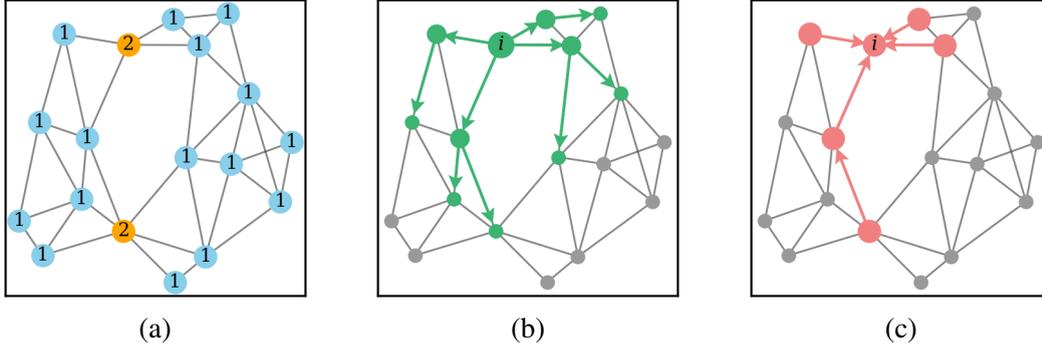


Figura 5.5: En (a) se etiquetan los nodos según su alcance de rigidez. En (b) se muestra en verde el conjunto \mathcal{V}_i , esto es, los agentes que siguen comandos de i . En (c) se ilustra en rojo el grupo de centros que incluyen a i .

es repetido de forma sucesiva durante la etapa de control, lo que permite a los agentes mantener actualizados dichos coeficientes. Además, como se puede observar, el tiempo requerido en el cómputo de los mismos no se ve afectada por el tamaño de la red, sino por el tamaño de los subframeworks, resultando en una técnica escalable. Luego, cada agente i define $m(i) := \max\{h_r(j) : i \in \mathcal{V}_j\}$ que es la distancia al centro más lejano, por lo tanto, la distancia a la que debe enviar sus mensajes de estado en la etapa de control.

En Fig. 5.5 se muestra un grafo, rígido y los alcances mínimos de rigidez de cada agente. En Fig. 5.5b se observa el subframework centrado en $i = 2$, y en Fig. 5.5c el conjunto de centros cuyo subframework contiene a dicho vértice.

Etapa de Control

Una vez finalizada la etapa de inicialización, todo agente envía en cada instante k el mensaje

$$\text{state_msg}(i, k) = (i, k, \eta_T, m(i), \hat{p}_i) \quad (5.16)$$

con $\eta_T \leftarrow 1$, el cual es retransmitido hasta llegar a los centros que incluyen a i . Cada centro recolecta mensajes de estado de los nodos de su subframework, computa el gradiente

correspondiente al término de rigidez y construye el mensaje

$$\text{action_msg}(i, k) = \left(i, k, \eta_T, h_r(i), \frac{\partial \mathcal{J}_{r,i}}{\partial p_{j_1}}, \dots, \frac{\partial \mathcal{J}_{r,i}}{\partial p_{j_n}} \right) \quad (5.17)$$

donde $\{j_1, \dots, j_n\} = \mathcal{V}_i \setminus \{i\}$. Aquellos nodos que reciben $\text{action_msg}(i, k)$ extraen el gradiente que les corresponde y retransmiten el mensaje de forma tal que el mensaje viaje exactamente una distancia $h_r(i)$.

Con este esquema, se produce el intercambio necesario para que cada agente aplique el término de rigidez de (5.10).

El protocolo de comunicaciones presentado es completado en un número finito de iteraciones igual a $\mathcal{D}(h_r)$. Esta es una cota de peor caso, debido a que aquellos subframeworks que cuenten con un alcance menor, terminarán sus intercambios de información en un menor número de iteraciones. Cabe mencionar que a pesar de que el esquema asegura el intercambio de información necesaria, los agentes acceden a la información con retrasos en el tiempo, los cuales no son uniformes a lo largo de la red. Esto se debe a que el gradiente $\partial \mathcal{J}_{r,i} / \partial p_{j_1}$ que recibe el nodo j_1 tiene que viajar $g(i, j_1)$ enlaces. Y dicho gradiente es calculado en función de las posiciones de los agentes en $\mathcal{V}_i = \{i, j_1, j_2, \dots, j_n\}$, los cuales fueron transmitidos desde estos hacia el centro i . Si definimos τ como el tiempo de retardo existente en la transmisión de datos de un nodo a sus vecinos, entonces la latencia de envío de mensajes (*one-way delay* (OWD)) de i a j es $T_{ij} := \tau g(i, j)$. De esta forma, el gradiente $\partial \mathcal{J}_{r,i} / \partial p_{j_1}$ que aplica el nodo j_1 en el instante t se computa con las posiciones $\{p_i(t - T_{ij_1}), p_{j_1}(t - 2T_{ij_1}), p_{j_2}(t - T_{j_2i} - T_{ij_1}), \dots\}$. Observar que, si bien los retrasos no son uniformes, se encuentran acotados $T_{ij} \leq \tau \mathcal{D}(h_r) / 2$, razón por la cual es importante mantener el alcance de los subframeworks lo más reducidos posible. Otra razón para esto, es que el tamaño de los subframeworks también afecta la carga computacional sobre su centro, que debe procesar la información y computar los comandos correspondientes a los nodos del mismo.

5.4.2. Análisis de Estabilidad del Control

Resulta interesante discutir la naturaleza de la señal de control propuesta. La misma se obtiene mediante una acción que está inspirada en un protocolo de gradiente descendiente con dinámica continua. Si la función de costo \mathcal{J} es función únicamente de la posición de los robots p , este protocolo presenta garantías de estabilidad. Sin embargo, algunos parámetros de la función \mathcal{J} se asumieron independientes de p para el cómputo del gradiente. Esta simplificación fue realizada para que la señal pueda ser obtenida de forma descentralizada. A continuación, discutiremos los efectos causados, por dicha aproximación, en la señal de control y en la estabilidad del sistema.

En primer lugar, se observa que las funciones de costo (5.6), (5.11) y (5.8) dependen del conjunto de enlaces existentes. Sin embargo, al computar los gradientes, no se tuvo en cuenta que el conjunto de enlaces es función de p , debido al rango de conexión limitado. Los cambios en dicho conjunto producen los siguientes efectos en las funciones de costo: (i) agregan o quitan términos en las sumatorias y (ii) modifican los valores de los coeficientes c_i . La pérdida de enlaces es monitoreada por el controlador, debido a que este contempla todas las conexiones existentes mediante el uso de pesos. Por lo tanto, la eliminación de enlaces es controlada y no produce inconvenientes. Sin embargo, la creación de nuevos enlaces, producto del movimiento de los robots, no es supervisada por el controlador. Esto significa que, desde el punto de vista del mismo, un nuevo enlace produce una discontinuidad en los gradientes obtenidos, y, por lo tanto, en la señal de control. Esto puede modelarse como si los costos fueran variantes en el tiempo, es decir,

$$\mathcal{J}(p, t) := \mathcal{J}_c(p, t) + \mathcal{J}_r(p, t) + \mathcal{J}_l(p, t) \quad (5.18)$$

donde la dependencia con el tiempo es constante por tramos, y las discontinuidades ocurren durante la creación de enlaces. Es importante notar que este fenómeno, producido

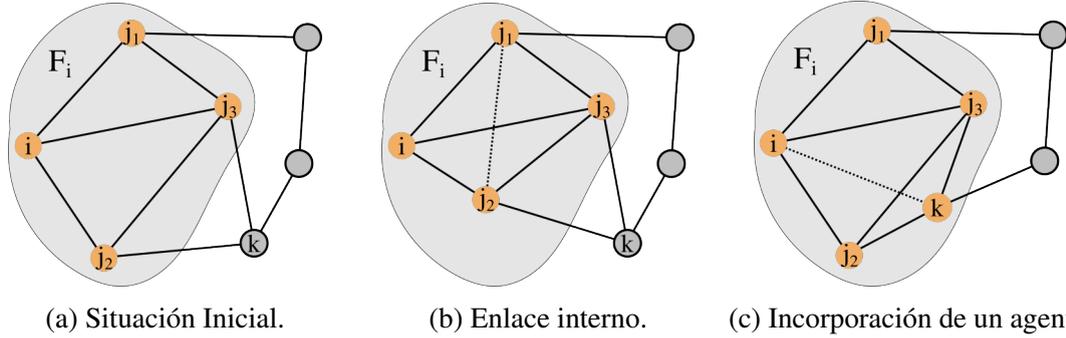


Figura 5.6: Tipos de enlaces creados: (b) enlace interno, (c) enlace que incorpora un agente al subframework.

por la incapacidad del controlador de prever la creación de enlaces, se presenta también en los trabajos previos a esta tesis [71, 70].

Para comprender mejor cuáles son los efectos de la situación planteada, realizamos el siguiente análisis. En primer lugar, consideramos un subframework \mathcal{F}_i y separamos el análisis según el tipo de enlace creado (Fig. 5.6): (i) entre dos nodos j_1, j_2 contenidos en el subframework ($g(i, j_1) \leq h(i)$ y $g(i, j_2) \leq h(i)$), y (ii) entre un nodo interior j ($g(i, j) < h(i)$) y uno exterior k ($g(i, k) > h(i)$), que tiene como efecto incorporar k a \mathcal{F}_i .

El primer tipo de enlace induce discontinuidades, acotadas en magnitud, en los gradientes (5.7), (5.12) y (5.10). Pero, a diferencia de los trabajos previos, gracias al esquema de control e intercambio de información propuesto aquí, este tipo de discontinuidades pueden ser eliminadas, dado que cada agente posee conocimiento de las posiciones de los robots contenidos en su subframework. Para ello, es posible modificar (5.6), (5.11) y (5.8)

de forma tal de contemplar los enlaces no existentes dentro de cada subframework

$$\tilde{\mathcal{J}}_c(p) := \frac{1}{2} \sum_{i \in \mathcal{V}} \sum_{j: j \in \mathcal{V}_i} w_{ij} \|p_i - p_j\|^{-a}, \quad (5.19)$$

$$\tilde{\mathcal{J}}_l(p) := \frac{1}{2\nu} \sum_{i \in \mathcal{V}} \sum_{j: j \in \mathcal{V}_i} w_{ij} (c_i + c_j) \quad (5.20)$$

$$\tilde{\mathbf{L}}_i := \sum_{j, k \in \mathcal{V}_i} w_{jk} \ell_{jk} \quad (5.21)$$

Notar que (5.19), (5.20) y (5.21) equivalen a las funciones originales si $w_{ij} = 1$ si $i \sim j$ y $w_{ij} = 0$ en caso contrario. Esto permite al controlador monitorear todas las variables internas a cada subframework, y supervisar la creación de enlaces del primer tipo. Por simplicidad, y por no haber sido necesario, en las simulaciones realizadas para validar el esquema de control, Sección 5.5, se utilizaron las funciones de costo originales.

No obstante, los efectos de la creación de una conexión del segundo tipo no pueden ser evitados con este esquema. Respecto de las funciones de evasión de colisiones y de carga en la red, la incorporación de un nuevo agente a un subframework puede traer discontinuidades acotadas, las cuales pueden ser toleradas. Sin embargo, en cuanto al mantenimiento de la rigidez, este tipo de enlaces puede generar acciones de control excesivamente grandes. Esto se ilustra en la Fig. 5.7: si el agente k cuenta con menos de d vecinos en \mathcal{F}_i luego de ser incorporado, entonces el ingreso del mismo a \mathcal{F}_i hace que este pierda su rigidez. Esto se debe a que el autovalor de rigidez λ_i tendrá un valor igual o muy cercano a cero (dependiendo de si se emplea (5.8) o (5.21)), haciendo que el gradiente (5.10) tome valores demasiado altos, que pueden comprometer al sistema.

Es interesante observar que la situación planteada no altera la rigidez del framework como tal, solo a algunos de sus subframeworks. Por lo tanto, gracias a los teoremas de descomposición en subframeworks, siempre es posible encontrar una nueva descomposición que permita evitar este problema. En el ejemplo de la Fig. 5.7, el problema se resuelve

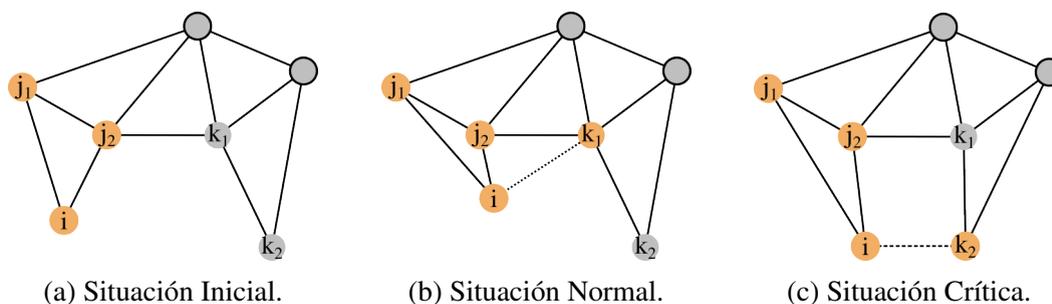


Figura 5.7: Situación crítica para el control de rigidez por subframeworks: la incorporación de un nuevo nodo sin los suficientes vecinos.

si el agente i , luego del ingreso de k_2 , actualiza su alcance, tal que $h(i) \leftarrow h(i) + 1$. Para lograr esto, el esquema de control debe ser capaz de lidiar con alcances de subframeworks crecientes, lo cual excede al estudio de esta tesis. Por último, es interesante notar que, para que la situación crítica planteada ocurra, tiene que darse un contexto donde dos nodos alejados se acerquen y se encuentren. Las aplicaciones propuestas para la validación de la ley de control mediante simulaciones (Sección 5.5), contemplan la existencia de una fuerza repulsiva que tiende a alejar los robots, por lo que la situación crítica es más difícil que ocurra.

5.5. Validación

Para validar el esquema de control de mantenimiento de rigidez, se presentan resultados de simulación. Las funciones empleadas para computar la ley de control, para simular la dinámica de los robots y del grafo subyacente, fueron escritas específicamente para este trabajo. Todas ellas fueron implementadas en PYTHON, y pueden encontrarse en https://gitlab.com/fpresenza/uv_network/-/tree/simu_tesis?ref_type=heads. La elección de emplear código propio para modelar las dinámicas de los vehículos y de los enlaces de comunicación, se debe a la simplicidad de los modelos adoptados. En el futuro esperamos incluir modelos que contemplen dinámicas más realistas, para lo cuales se

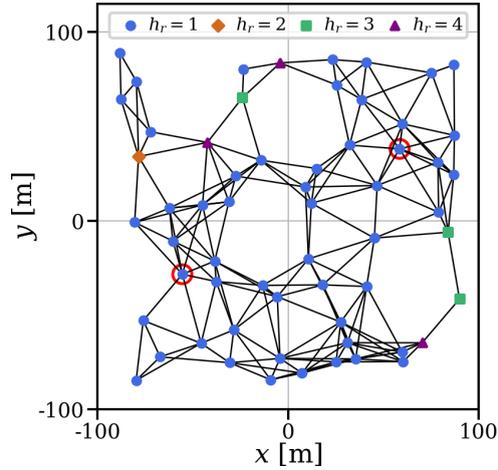


Figura 5.8: Estado inicial de la red de 60 agentes. Los agentes remarcados con circunferencias rojas son aquellos dotados con mediciones de posición absoluta.

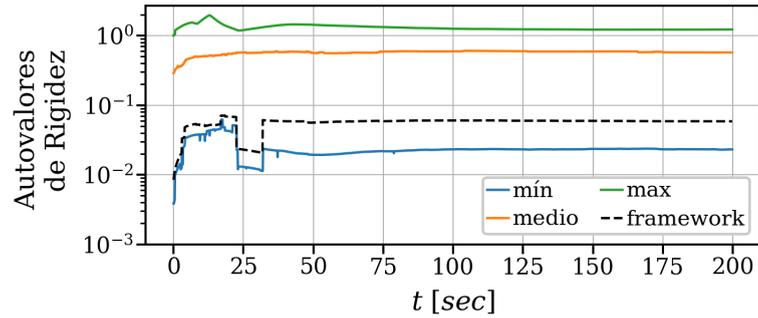
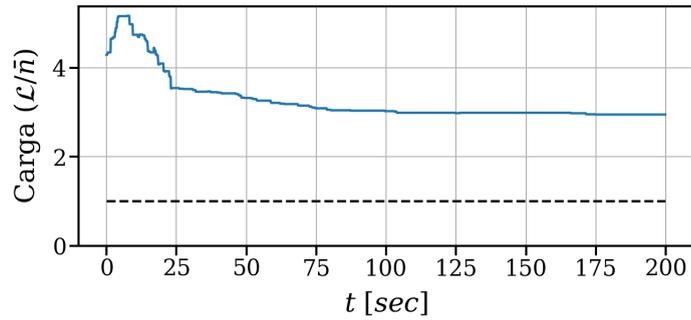
evaluarán los simuladores disponibles en el área de agentes móviles. Videos explicativos de las simulaciones presentadas pueden ser vistos en <https://youtu.be/xNsZodETsYI> y <https://youtu.be/m6na7g2UqFg>.

5.5.1. Caso 1: Escalabilidad

En primer lugar, queremos evaluar la escalabilidad de la estrategia presentada. Para esto, en este caso de validación, se asume el tiempo de transmisión de datos $\tau = 0$, para eliminar los efectos producidos por el mismo. Además, se consideran las extensiones de los subframeworks constantes, de modo que la reducción de la carga de la red se obtiene únicamente a partir de la inclusión de la métrica \mathcal{L} en la función objetivo de control.

Realizamos simulaciones para un framework de $v = 60$ agentes generado aleatoriamente mediante el modelo (\mathcal{R}, v, ρ) (4.7). El conjunto de robots se sitúa inicialmente en una región \mathcal{R} de $200\text{m} \times 200\text{m}$. El rango de conexión es $\rho = 40\text{m}$, lo cual da lugar al framework mostrado en Fig. 5.8.

En Fig. 5.9 y Fig. 5.10 se muestran los resultados obtenidos al comandar los agentes

(a) Evolución de los valores mínimos, promedio y máximo de los ν autovalores de rigidez.

(b) Carga de comunicaciones. La línea punteada indica el mínimo valor posible.

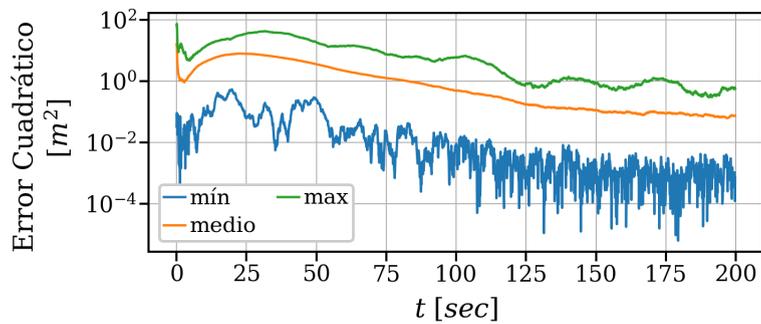
(c) Error cuadrático mínimo, promedio y máximo en el posicionamiento del conjunto de ν robots.

Figura 5.9

por medio de la acción de control (5.13). La Fig. 5.9a muestra la evolución de los autovalores de rigidez de los subframeworks, que son mantenidos positivos en todo instante. La Fig. 5.9b muestra la evolución de la relación $\mathcal{J}_1/\bar{n}(\mathcal{G})$, que mide la carga en los enlaces de la red. Por último (5.9c) muestra el error cuadrático en el posicionamiento de los robots.

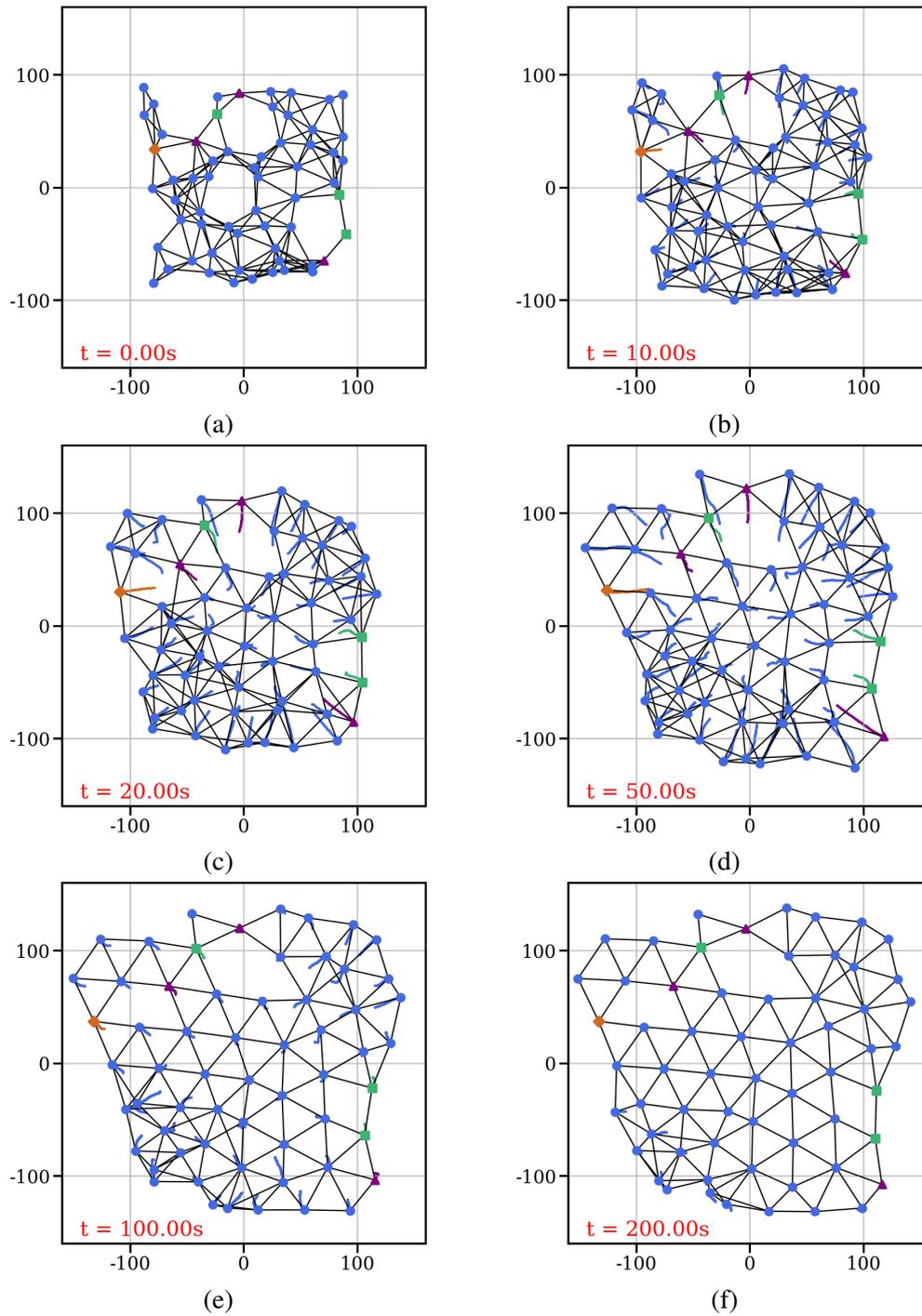


Figura 5.10: Estado de la red en 0s, 10s, 20s, 50s, 100s, y 200s.

Se observa que en los primeros 25s, de forma coordinada, los controladores logran aumentar rápidamente el autovalor de rigidez del framework, mediante la optimización de la posición de los robots y la creación de nuevos enlaces. Esto trae un correspondiente incremento en la carga comunicacional, y a partir de allí empieza a tener efecto la minimización del costo \mathcal{J}_1 . La misma actúa como una fuerza repulsiva, alejando los robots entre sí de forma tal de eliminar los enlaces más cargados. Esta fuerza es contraria al mantenimiento de la rigidez del framework, resultando en un compromiso entre dos objetivos con el cual el controlador debe lidiar. No obstante, la ley de control es eficaz en mantener positivos los ν autovalores de rigidez de los subframeworks. El autovalor de rigidez del framework, aunque no es incluido directamente en el esquema de control, se mantiene positivo en todo momento, como predice el Teorema 4.3.2. Es relevante notar que mientras la carga de red disminuye (hasta aproximadamente un 40% del valor inicial), los autovalores de rigidez no se degradan. Esto muestra la versatilidad de la ley de control para combinar estos múltiples objetivos.

Por último, es importante mencionar la aparición de una propiedad emergente del conjunto de robots, a pesar de no haber sido postulado como un objetivo. En Fig. 5.10 se observa, a medida que va transcurriendo la misión, el surgimiento de un patrón regular, similar a una retícula triangular. Esto es consecuencia del esfuerzo del control para maximizar los autovalores de rigidez, que priorizan estructuras regulares. Resulta interesante observar que, esta estructura triangular es propia del espacio bidimensional utilizado en las simulaciones. En general, en \mathbb{R}^d , una estructura rígida regular podría estar compuesta por una red de d -símplices.

5.5.2. Caso 2: Alcances variables

En este caso de estudio, diseñamos un esquema para que los agentes puedan reducir dinámicamente las extensiones de los subframeworks a medida que se desarrolla la apli-

cación. Esto permite reducir tanto el retardo máximo como la carga de comunicaciones.

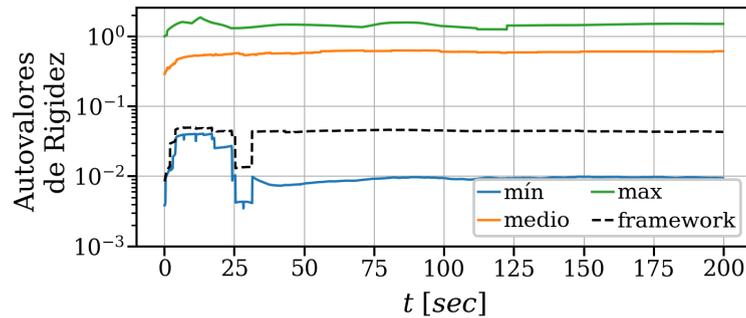
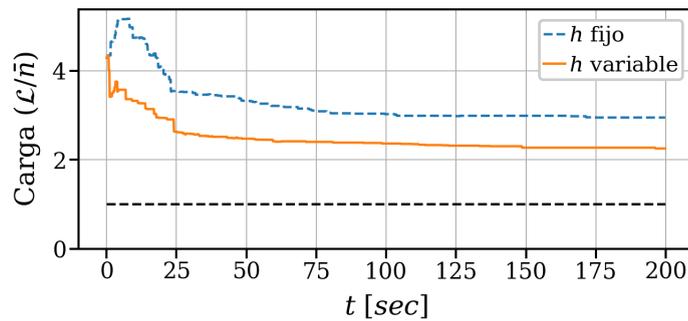
Para esto, se define el alcance de cada subframework en el tiempo k como el parámetro $h(i, k)$ donde $h(i, 0) = h_r(i)$. El mismo evoluciona según

$$h(i, k) = \begin{cases} h(i, k-1) - 1, & \text{si } \lambda_i^-(k) \geq \alpha \\ h(i, k-1), & \text{caso contrario} \end{cases} \quad (5.22)$$

donde λ_i^- denota el autovalor de rigidez del subframework inducido por el alcance reducido en uno, y $\alpha > 0$ es un parámetro de ajuste, que en nuestro fue configurado en 0.05. Dicho parámetro determina cuál es el mínimo valor aceptable del autovalor de rigidez luego de reducir el alcance. Esta ley de actualización, asegura que las extensiones solo pueden reducirse, eventualmente llegando a ser iguales a 1, en caso de ser posible. Para validar este esquema, se agregó (5.22) al control de la misma red empleada en el Caso 1. Los resultados se observan en Fig. 5.11, donde se puede ver cómo la carga de comunicaciones sobre los enlaces presenta una reducción importante en los primeros segundos, que es mantenida durante toda la misión. Esto es posible gracias a la creación de nuevos enlaces en los primeros 25s, que permite la reducción de los alcances mínimos de rigidez, pero que no era aprovechada en el Caso 1.

5.5.3. Caso 3: Robustez al retardo temporal

En este caso de estudio, queremos evaluar la capacidad del esquema de control para cumplir con los objetivos en la presencia de retardos en el tiempo en la transmisión de mensajes. Para esto, realizamos simulaciones con una red de $v = 10$ agentes dispuestos inicialmente sobre una región de $80\text{m} \times 80\text{m}$ con rango de conexión $\rho = 8.5\text{m}$, formando un framework rígido como indica Fig. 5.12. Notar que el ejemplo elegido configura un caso comprometedor para el controlador. Esto se debe a que, por un lado, las distancias

(a) Evolución de los valores mínimos, promedio y máximo de los v autovalores de rigidez.

(b) Carga de comunicaciones, la línea punteada indica el mínimo valor posible.

Figura 5.11

entre vecinos son bajas (aproximadamente 5 – 8m), y, por otro lado, el alcance de los subframeworks es grande ($\mathcal{D}(h_r)$ es igual al diámetro). Esto hace que un pequeño retardo pueda producir errores en el control de posición de unos pocos metros, lo suficiente para romper enlaces que son necesarios para la rigidez. En este caso de estudio, para que exista una tarea que impulse a los robots, proponemos la recolección de un conjunto de objetos aleatoriamente colocados sobre la misma región. El objetivo de recolección funciona como una fuerza repulsiva que tiende a hacer separar los vehículos y desconectar la red, lo cual debe ser impedido por el objetivo de mantenimiento de la rigidez. Para comandar al conjunto de robots en esta aplicación, se agrega a la función objetivo \mathcal{J} , un término que

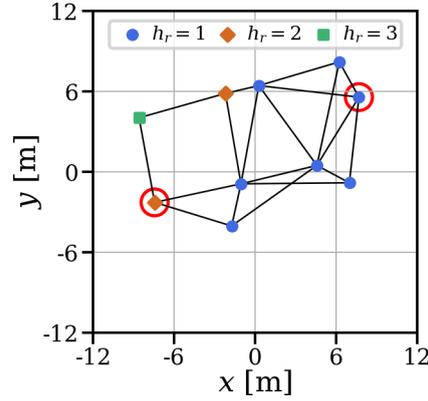


Figura 5.12: Estado inicial de la red de 10 agentes. Los agentes remarcados con circunferencias rojas son aquellos dotados con mediciones de posición absoluta.

actúa como un potencial de atracción hacia los objetos a recolectar,

$$\mathcal{J}_a(p) := \sum_{i \in \mathcal{V}} \|p_i - t_i\|^c, \quad c > 0, \quad (5.23)$$

donde t_i es la posición del objeto no recolectado más cercano al nodo i -ésimo, y el valor de $c \approx 1/2$ fue ajustado manualmente. Por simplicidad en la estrategia del control, y para visualizar mejor el efecto de los retardos temporales, tanto el término de la carga \mathcal{J}_l como el esquema de extensiones variables no fueron considerados en este esquema.

Realizamos 4 simulaciones con las mismas condiciones iniciales, pero con distintos retardos asociados a los tiempos de transmisión $\tau = 100, 200, 300$ y 400 ms. Dado que en este ejemplo $\mathcal{D}(h_r) = 6$, los retardos máximos (OWD) son $T_{\text{máx}} := \tau \mathcal{D}(h_r) / 2 = 0.3, 0.6, 0.9$ y 1.2 s. Esto significa que algunos agentes aplican su acción de control con retardos de ida y vuelta (*round-trip delay* (RTD)) de hasta 2.4 s. En Fig. 5.13 se observa el estado del conjunto de robots y los objetos a recolectar. Cada objeto es ilustrado mediante un punto cuadrado cuyo color, negro o gris, indica si el mismo fue recolectado o no, respectivamente. En las sucesivas figuras, se muestra el avance en la recolección de objetos para el caso sin retardo, y se omiten las correspondientes trayectorias en los casos con

$\tau > 0$, debido a que no existen diferencias apreciables en dichas ilustraciones. En donde sí se observa el efecto del retardo es en Fig. 5.14, donde se muestran los autovalores de rigidez de los subframeworks y del framework para las cuatro situaciones evaluadas. En todos los casos, los autovalores de rigidez se mantienen mayores a cero en todo instante, lo cual indica la robustez del controlador ante retardos temporales. Sin embargo, se puede ver como el controlador encuentra mayor dificultad en mantener estables dichos valores a medida que aumenta τ . Esto ocurre especialmente en la etapa inicial, donde el rápido movimiento de los agentes en la inicialización, junto con el retardo temporal, hace que la rigidez de los subframeworks se vea más comprometida, llegando a valores muy bajos, para $\tau = 300, 400\text{ms}$. Por otro lado, el efecto de la latencia se puede observar analizando las posiciones de los vehículos en función del tiempo, como indica Fig. 5.15. En la misma se puede observar la emergencia de oscilaciones características del efecto producido por retardos en el tiempo. Dichas oscilaciones crecen en amplitud y período con el aumento de τ , y aunque el controlador pueda mantener la rigidez del framework, degradan su desempeño.

El paso de tiempo elegido para implementar las simulaciones es igual a 25ms. Para mostrar que el mismo es suficiente para capturar tanto la dinámica de los vehículos como el efecto del retardo impuesto en las comunicaciones, comparamos tres casos con pasos de tiempo 1, 25 y 100ms para el retardo de $\tau = 300\text{ms}$. Para reducir las fuentes de aleatoriedad y facilitar la comparación, configuramos las funciones que generan los ruidos de medición, tanto de distancia como de posición absoluta, para que den las mismas muestras en las tres situaciones. En Fig. 5.16 se muestran los autovalores de rigidez en los primeros 25s de simulación para los tres casos. Se observa que el comportamiento obtenido es muy similar, no obstante, se puede apreciar que la caída del mínimo autovalor, que ocurre en el instante $t \approx 8\text{s}$, es más pronunciada en el caso con paso de tiempo 100ms. Es importante aclarar que los cambios discretos que aparecen en la evolución de los autovalores no se

deben a efectos de la simulación sino a eliminación de enlaces.

5.6. Resumen

En este capítulo se aplicó la teoría de rigidez por subframeworks desarrollada en el Capítulo 4 para el control coordinado del mantenimiento de la rigidez de un framework. En primer lugar, se analizaron los esquemas de posicionamiento y control de redes multi-robot existentes en la literatura, y se mencionaron sus capacidades y limitaciones. Luego, propusimos una adaptación de los algoritmos de posicionamiento utilizados en redes multi-robot, que prescinde de los algoritmos de consenso, mejorando su escalabilidad. Más adelante, propusimos un esquema de control de sistemas multirobot por optimización y desarrollamos las funciones de costo necesarias para cumplir los múltiples objetivos enunciados. Estos aportes fueron presentados en [72]. Luego, detallamos los cómputos y el intercambio de información necesario para poder aplicar dicha ley de control de forma descentralizada por los agentes de la red. El esquema propuesto es ventajoso respecto de protocolos anteriores debido a que no requiere el intercambio de información global, evitando los altos retardos y la gran carga comunicacional, especialmente en redes con gran diámetro. No obstante, debe monitorearse la carga de la red para que no sea demasiado elevada. Presentamos dos formas de limitar su valor: (i) mediante la inclusión de un costo en la función objetivo de control, (ii) mediante la reducción dinámica de las extensiones de los subframeworks. Para validar el esquema propuesto presentamos simulaciones que pretenden evaluar la escalabilidad y la robustez ante la latencia de la red, condiciones que comprometen esquema al control. Pudimos observar que el controlador es capaz de combinar los múltiples objetivos en ambas situaciones comprometedoras, lo cual es una propiedad fundamental en este tipo de aplicaciones. Estos resultados fueron presentados en [73].

Analizar el algoritmo de control propuesto bajo la óptica que distingue entre forma-

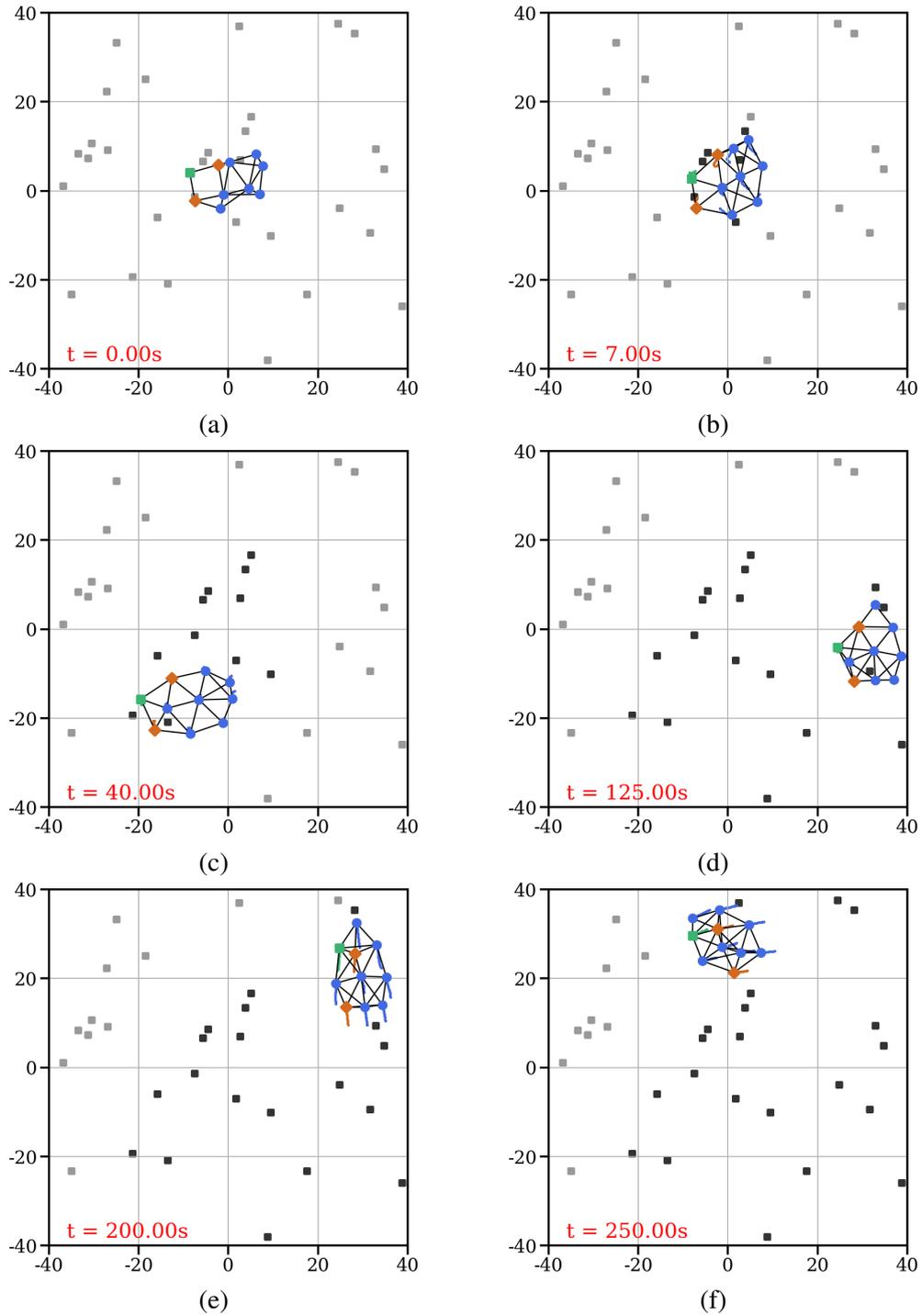


Figura 5.13: Evolución del estado de la red durante la misión.

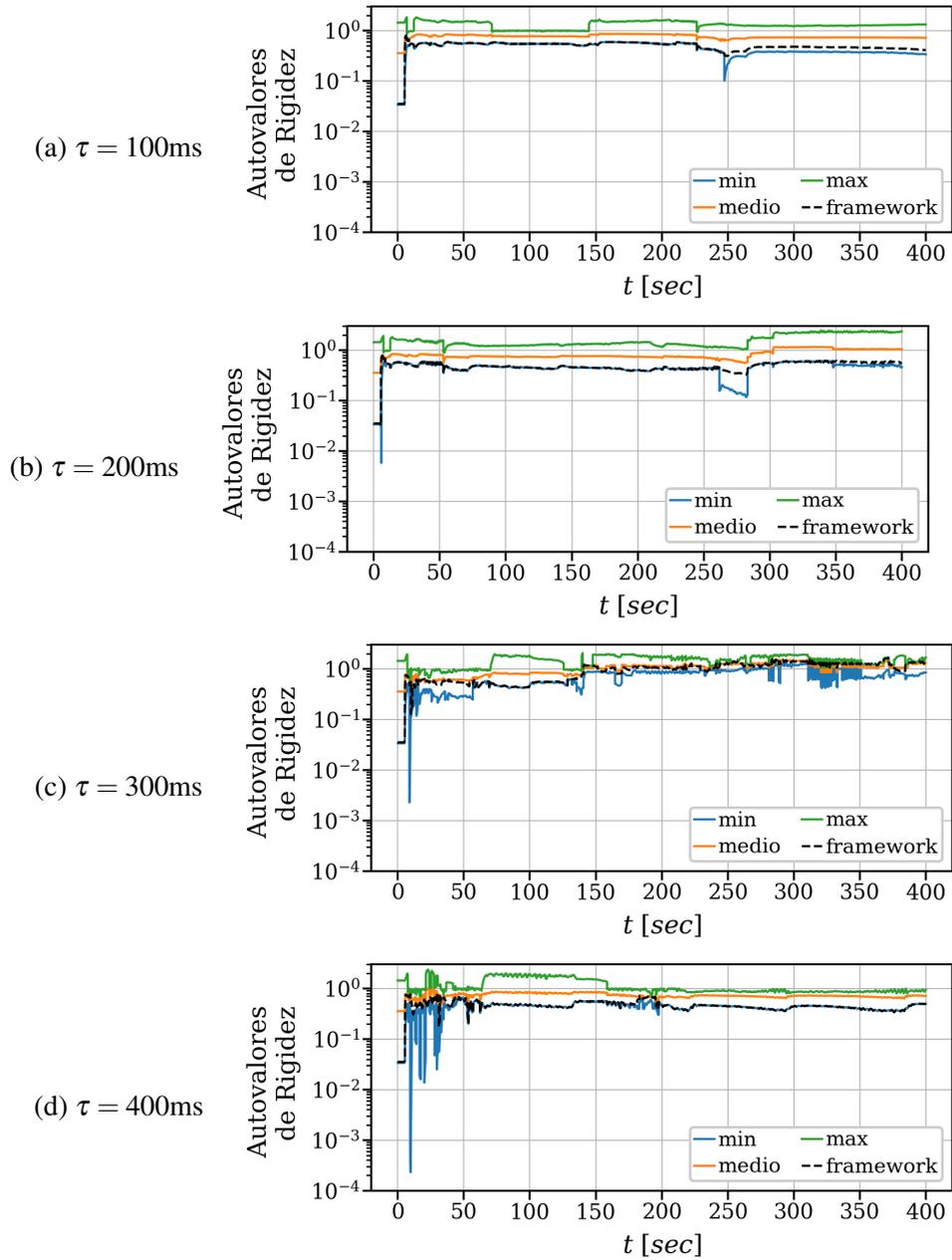


Figura 5.14: Evolución de los valores mínimo, promedio y máximo de los ν autovalores de rigidez.

ciones y *swarms* resulta interesante. Si bien el sistema de robots es comandado en función

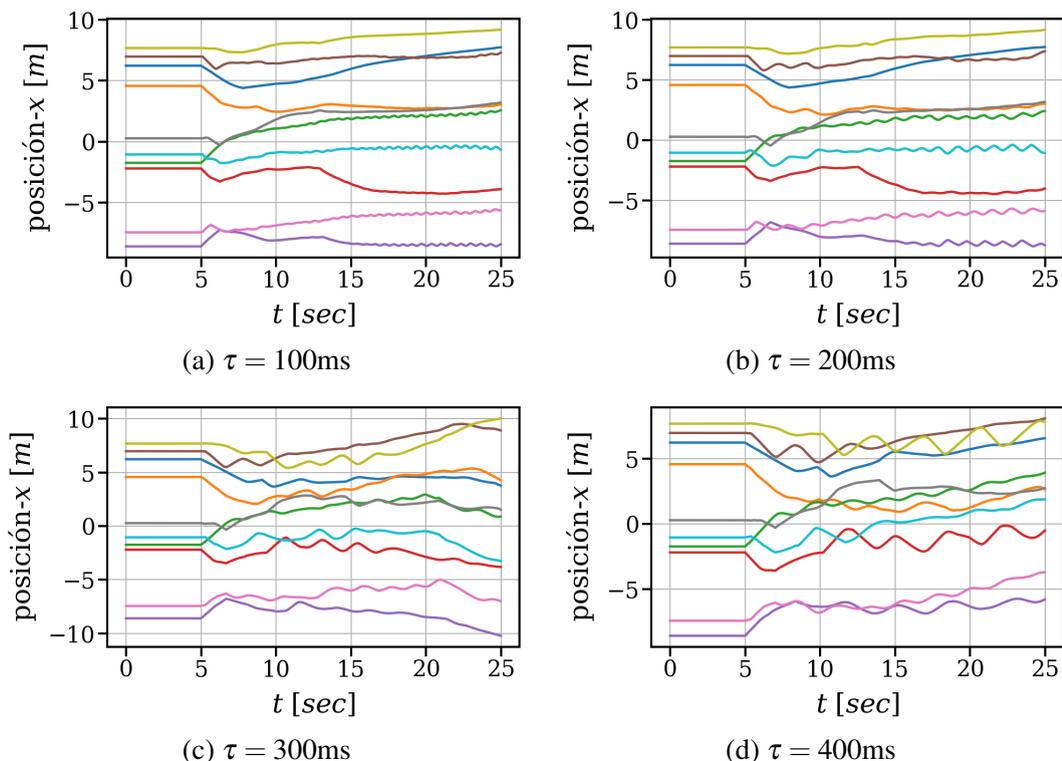
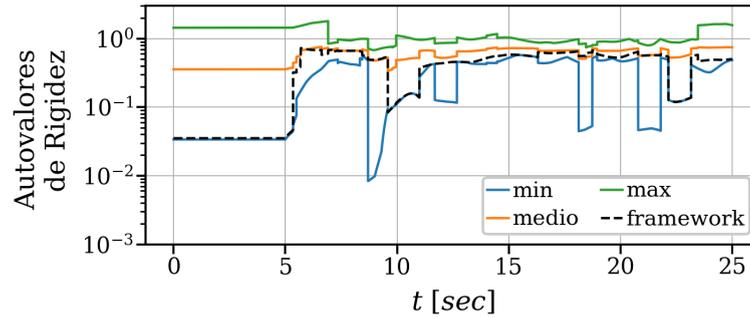


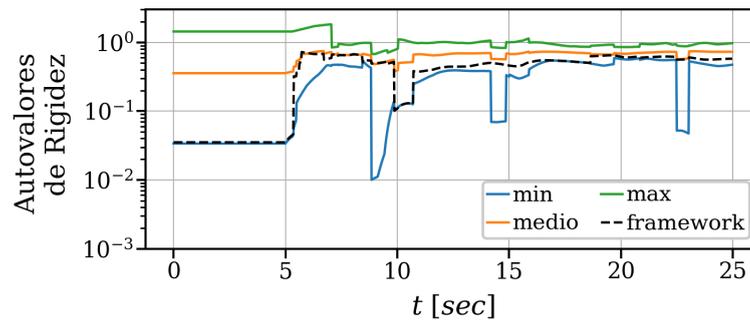
Figura 5.15: Oscilaciones de las trayectorias entre 0 y 25s producidas por el retardo temporal.

de reglas básicas dictadas por los objetivos de alto nivel, como el mantenimiento de la rigidez, a partir de las mismas emergen patrones regulares que tienden a preservar las distancias entre agentes. Esta estructura de formación no fueron directamente especificadas, por lo que el controlador puede desarmarla, en caso de ser necesario, para cumplir sus objetivos. Por lo tanto, se puede pensar a este como un esquema híbrido, en donde si bien no existen requerimientos específicos sobre el mantenimiento de las distancias relativas, una formación surge como comportamiento indirecto.

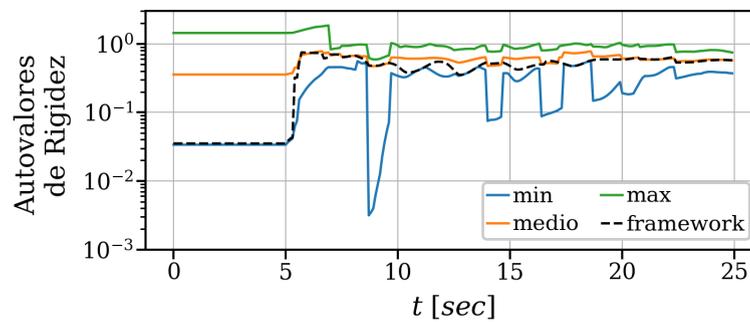
Es importante aclarar que, si bien como se mostró en el Capítulo 4, la utilización de subframeworks dispersos puede ser beneficiosa, ya que reduce la carga en las comunicaciones, en esta tesis asumimos una descomposición densa para validar el controlador.



(a) Paso de tiempo: 1ms



(b) Paso de tiempo: 25ms



(c) Paso de tiempo: 100ms

Figura 5.16: Comparación de tres pasos de tiempo de simulación.

Esto se hace por dos motivos: (i) en primer lugar, para probar el esquema en una situación crítica que maximice el estrés sobre la red, lo cual ayuda a enriquecer el análisis, (ii) el esquema de descomposición dispersa requiere además el control de la rigidez del framework vinculante, para lo cual es necesario diseñar un esquema de control específico, ya

que el grafo vinculante no se corresponde necesariamente con la red de comunicaciones. Sin duda que esto último resulta un caso interesante y será estudiado en trabajos futuros.

Capítulo 6

Cinematografía Aérea Multirobot

En el presente capítulo se propone una estrategia de coordinación y control para un sistema de múltiples vehículos aéreos no tripulados (VANTs) para la filmación aérea autónoma. Con este esquema, un sistema de múltiples VANTs con cámaras pan-tilt es concebido como una grúa robótica virtual, capaz de seguir trayectorias generadas durante el vuelo (*on-the-fly*) por un operador de cámara, mientras se cumplen objetivos específicos de la aplicación. El sistema garantiza de forma autónoma que los VANTs filmadores no van a aparecer en el campo de visión de los demás, y que las colisiones son evitadas, liberando al operador de estas tareas. Para validar el esquema propuesto se emplean herramientas avanzadas de simulación, las cuales permiten evaluar el desempeño del sistema con gran precisión.

6.1. Introducción

Cada día hay más aplicaciones en las que se usan vehículos aéreos no tripulados para hacer filmaciones (Fig. 6.1). Ejemplos de estos usos pueden encontrarse en la industria cinematográfica, cobertura de noticias, eventos deportivos, etc., donde los VANTs se uti-

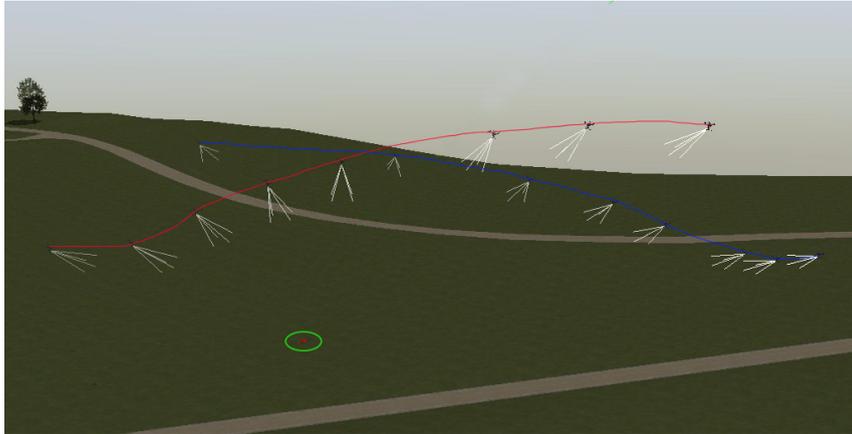


Figura 6.1: Cámaras montadas en dos VANTs haciendo un seguimiento de un POI (VANT rojo dentro de un círculo verde) mientras siguen una trayectoria teniendo en cuenta restricciones artísticas y de seguridad.

lizan para reemplazar sistemas de cámaras, como los helicópteros con cámaras (helicam), las cámaras suspendidas por cables, carros y grúas. El uso de VANTs reduce los costos, permite una rápida reconfiguración del sistema para diferentes tipos de tomas, a la vez que añade un nuevo universo de tomas que no es posible realizar con los sistemas de cámaras antes mencionados. Sin embargo, la utilización de los mismos en filmaciones profesionales requiere de pilotos expertos y calificados para su uso, además de sistemas de seguridad ante el evento de fallas en algún VANT. No obstante, en los últimos años, se ha avanzado en el diseño y construcción de sistemas de filmación autónomos que se utilizan sistemas de robots móviles para esta tarea.

Es evidente que la cinematografía autónoma se puede beneficiar ampliamente de los avances en los sistemas robóticos multi-agente. Las estrategias de control de formaciones, donde las restricciones espaciales se definen entre agentes, son una herramienta muy útil en misiones aéreas cooperativas, según lo estudiado en [120]. En este sentido, típicamente se da a los VANTs la tarea de seguir, lo más cerca posible, una trayectoria, con el fin de producir los planos deseados. La trayectoria define no solo la secuencia de puntos en el

espacio $[x_i, y_i, z_i]^T$ que deben ser seguidos por los vehículos, sino también el estado de las cámaras, típicamente del tipo *pan-tilt-zoom*. Para cada toma en particular se pueden definir distintos tipos de trayectorias y, aunque no es necesario, estas pueden ser especificadas en función de un objetivo móvil o estático a ser filmado, como puede ser un vehículo en movimiento o un paisaje. En general, este punto o zona de interés se denomina *Point of Interest* (POI). Adicionalmente, con la realimentación visual naturalmente disponible para esta tarea, el director de fotografía (*Director of Photography* (DOP)), podría querer ajustar la trayectoria durante el vuelo (maniobras *on-the-fly*), comandando directamente los vehículos. Debido a que se encuentran en uso simultáneo múltiples vehículos, las trayectorias resultantes deben evitar colisiones y, generalmente, resulta apropiado, por motivos artísticos, que los VANTs no aparezcan en el campo de visión (*field-of-view* (FOV)) de los demás. Esta última condición se denomina visibilidad mutua y es un problema central en la cinematografía aérea multirobot.

Para lograr los múltiples objetivos mencionados sin ayuda de un sistema de control, los operadores de cámara deberían coordinar en tiempo real sus comandos para evitar colisiones y visibilidad mutua, además de considerar el entorno, una situación que se vuelve aún más compleja en entornos dinámicos. Con el fin de reducir la complejidad de esta tarea para los operadores, un sistema de control debe garantizar automáticamente estos requerimientos.

Existen diversos trabajos en la literatura reciente cuyos aportes se centran en el desarrollo de sistemas cinematográficos autónomos utilizando vehículos aéreos. Para el seguimiento de trayectorias artísticas, muchos de estos trabajos tratan la planificación de trayectorias como un problema central [121, 122, 123, 14, 124, 125]. Llamamos trayectorias artísticas a aquellas que son definidas a partir de directrices cinematográficas, como se describe en [126]. El método adoptado es la generación *offline* (previamente al vuelo) de las mismas, tanto de secuencias de posiciones como de entradas de control, con las

que luego se alimenta un sistema de control *online* (durante el vuelo). Para cumplir esta estrategia de planificación se deben tener en cuenta múltiples entradas y restricciones, típicamente como un problema de optimización. No obstante, estas restricciones podrían no verificarse en tiempo de ejecución debido a la naturaleza de lazo abierto que poseen los enfoques de planificación y seguimiento de las trayectorias.

En los casos en los que se desea resolver el problema *online*, con computadoras a bordo de los vehículos, muchas veces se busca reducir la dimensionalidad del problema de optimización para disminuir la carga computacional del algoritmo. Una estrategia para lograrlo es suponer que hay disponible (o diseñar) un controlador independiente para la cámara [124, 127], que resuelve el problema de apuntamiento al POI. Otra de las estrategias más comunes es restringir grados de libertad (*degrees of freedom* (DOF)) de los vehículos o de las cámaras [124, 14, 13].

A diferencia de los trabajos mencionados, durante el desarrollo de esta tesis se diseñó un sistema donde se considera a la formación de VANTs como una grúa de brazos articulados. Esta grúa, donde se montan virtualmente múltiples cámaras, puede ser controlada por un operador a través de un control remoto, en una forma similar a como se realiza actualmente con una única cámara. Las variables de estado de la grúa articulada representan el estado de la formación, por lo tanto, resulta natural la formulación en espacio del cluster como la técnica de coordinación y control para el sistema. La misma permite: (i) definir el estado de la formación utilizando variables intuitivas y conocidas para el operador, (ii) contemplar el sistema articulado de cámaras como un todo, permitiendo que la optimización de la trayectoria se resuelva para todos los parámetros en simultáneo.

Como se desarrollará a lo largo de este capítulo, esta estrategia logra abstraer el control, permitiendo que un único piloto comande satisfactoriamente una formación de VANTs. Más aún, la formulación en espacio del cluster es versátil en el sentido de que permite cambiar la definición del estado dependiendo de la tarea, por ejemplo, del tipo de

toma deseada. Por lo tanto, una definición adecuada de los estados del cluster permite especificar trayectorias de formación típicas con una cantidad mínima de parámetros. Como ejemplo, el movimiento orbital de las cámaras (rotación respecto del POI y apuntamiento al mismo) se puede generar especificando una trayectoria en un parámetro angular del cluster. Además, tomando la distancia entre dos cámaras como una variable de estado del cluster, se requiere un único parámetro para acercarlas o apartarlas, independientemente de la pose de la grúa. En la Sección 6.2 se presentan dos definiciones de cluster para cinematografía aérea.

Para generar las señales de compensación que guíen el movimiento de la grúa virtual, se propone una estrategia de control a lazo cerrado basada en un problema de optimización. Debido a la naturaleza de la aplicación, este esquema es apropiado, ya que puede balancear los compromisos entre el seguimiento de una trayectoria, múltiples restricciones, y diferentes objetivos para la filmación. El problema de optimización propuesto, desarrollado en la sección 6.3, se beneficia de la técnica CSC, ya que se pueden elegir variables para el estado de la grúa virtual, que sean naturales al problema, lo que lleva a una formulación directa. Además, el problema se resuelve en simultáneo para la formación de VANTs y las orientaciones de las cámaras, lo que da una solución localmente óptima para todos los parámetros de la trayectoria, desde un punto de vista cinematográfico.

6.2. Variables del Cluster para Cinematografía

En esta tesis, se consideró una formación de VANTs 4-DOF estabilizados, que cuentan con un sistema de seguimiento de comandos de velocidad en x, y, z y en ángulo yaw (rotación en z). Cada vehículo cuenta, además, con una cámara montada sobre un *gimbal* que permite comandar ángulos *pan-tilt*, mientras que el ángulo *roll* (rotación en respecto de la normal a la cámara) se encuentra estabilizado en todo momento. Para el movimiento “panorámico” de la cámara, se controla directamente el ángulo *pan*, manteniendo constante

el ángulo *yaw* de cada VANT para evitar redundancia.

Distintas variables para representar el cluster pueden ser elegidas, dependiendo de la tarea requerida y del número de VANTs desplegados, lo que permite interpretar de diversas formas la estructura de la formación. Esta abstracción provee al director y a los operadores de cámara de una forma natural de comandar el grupo de robots como un mecanismo virtual articulado. Por ejemplo, para un pequeño número de VANTs, puede ser útil elegir polígonos para representar la formación, como por ejemplo un segmento para $v = 2$ o un triángulo para $v = 3$. Para equipos con mayor número de vehículos puede ser conveniente una descripción basada en otras estructuras más complejas como grúas, estructuras reticuladas, entre otras.

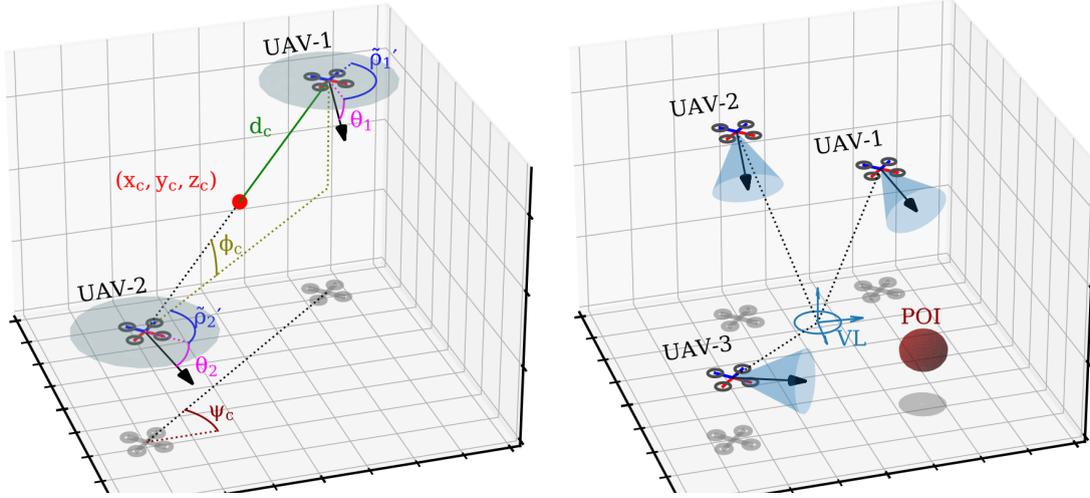
6.2.1. Formación con 2 VANTs

Sea x_i, y_i, z_i la posición del i -ésimo vehículo, y ρ_i, θ_i los ángulos de *pan* y *tilt* de su cámara, entonces el vector de estado de la formación en el espacio de robots $\mathbf{r} \in \mathbb{R}^{10}$ puede ser definido como

$$\mathbf{r} = [x_1 \ y_1 \ z_1 \ \rho_1 \ \theta_1 \ x_2 \ y_2 \ z_2 \ \rho_2 \ \theta_2]^T. \quad (6.1)$$

En este caso, se definen las variables de estado del cluster tales que

$$\mathbf{c} = [x_c \ y_c \ z_c \ \psi_c \ \phi_c \ d_c \ \tilde{\rho}_1 \ \theta_1 \ \tilde{\rho}_2 \ \theta_2]^T, \quad (6.2)$$



(a) Variables de estado en espacio del cluster para $v = 2$. Por simplicidad se ilustra $\tilde{\rho}'_i = \frac{\pi}{2} - \tilde{\rho}_i$. para v vehículos.

Figura 6.2: Vista esquemática de la formación y un objetivo a filmar.

donde el centroide de la formación x_c, y_c, z_c , su orientación ψ_c, ϕ_c , distancia del centroide a cada vehículo d_c , y la orientación de cada cámara $\tilde{\rho}_1, \theta_1, \tilde{\rho}_2, \theta_2$, resultan

$$\begin{aligned}
 \text{(centroide)} \quad & x_c = \frac{x_1 + x_2}{2}, \quad y_c = \frac{y_1 + y_2}{2}, \quad z_c = \frac{z_1 + z_2}{2}, \\
 \text{(ángulo yaw)} \quad & \psi_c = \arctan\left(\frac{x_2 - x_1}{y_1 - y_2}\right), \\
 \text{(ángulo roll)} \quad & \phi_c = \arctan\left(\frac{z_1 - z_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}}\right), \\
 \text{(media distancia)} \quad & d_c = \frac{1}{2}\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}, \\
 \text{(orientación)} \quad & \tilde{\rho}_1 = \rho_1 - \psi_c, \quad \theta_1 = \theta_1, \\
 \text{(orientación)} \quad & \tilde{\rho}_2 = \rho_2 - \psi_c, \quad \theta_2 = \theta_2.
 \end{aligned} \tag{6.3}$$

En la Fig. 6.2a se muestra un esquema del sistema junto con la representación de los parámetros del cluster propuestos. Si $\mathbf{c} = f(\mathbf{r})$ está dada por (6.3) y $\mathbf{J}^{-1}(\mathbf{c})$ es el jacobiano de f^{-1} entonces la acción de control en el espacio del cluster es transformada al espacio

robot mediante

$$\mathbf{u}_r = \mathbf{J}^{-1}(c)\mathbf{u}_c. \quad (6.4)$$

El modelo de la dinámica de la formación es $\dot{\mathbf{c}} = F(\mathbf{c}, \mathbf{u}_c) = \mathbf{u}_c$ dado que la misma está formada por VANTs estabilizados, los cuales se pueden modelar con un integrador simple.

6.2.2. Formación con v VANTs

Para la filmación autónoma con v vehículos, la formación puede ser interpretada como una grúa articulada multi-cámara con sus brazos vinculados en un punto *pivot* que llamamos líder virtual (VL). Dada esta descripción, las variables del cluster naturales son la posición y orientación del VL y las posiciones de los VANTs respecto del mismo en coordenadas esféricas, como se ilustra en Fig. 6.2b. Esto permite trasladar, rotar y manipular los brazos del mecanismo articulado de forma sencilla e intuitiva.

En este caso, los vectores de estados en el espacio robot y cluster $\mathbf{r}, \mathbf{c} \in \mathbb{R}^{6v}$ son la concatenación de v vectores $\mathbf{r}[i], \mathbf{c}[i] \in \mathbb{R}^6$ correspondientes a cada VANT. Sean $p_L, \xi_L \in \mathbb{R}^3$ la posición y orientación del VL con respecto a un sistema de referencia fijo a la tierra. Si el VL es un punto abstracto comandado por un operador para roto-trasladar la grúa, entonces su estado es conocido. Si, en cambio, el VL representa un objeto físico a ser seguido, se asume que su posición y orientación se pueden estimar con sensores. En ambos casos, tanto p_L, ξ_L se asumen conocidas con cierto error acotado. El estado de cada VANT es especificado en el espacio robot con el vector $\mathbf{r}[i] = [p_i^T \ \xi_i^T]^T$ que contiene su posición $p_i = [x_i \ y_i \ z_i]^T$ y el estado de su cámara $\xi_i = [\zeta_i \ \theta_i \ \rho_i]^T$ determinado por *zoom-tilt-pan*, respectivamente. El estado de los VANTs en espacio del cluster se define como $\mathbf{c}[i] = [q_i^T \ \eta_i^T]^T$ donde $q_i = [d_i \ \varepsilon_i \ \alpha_i]^T$ es la posición de los mismos en coordenadas esféricas (*distance-elevation-azimuth*) respecto del VL (Fig. 6.2b); $\eta_i \in \mathbb{R}^3$ representa el vector de apuntamiento de la i -ésima cámara, expresado en el sistema de referencia del

VL. La relación entre los vectores de estado en ambos espacios está dada por

$$\begin{aligned} \mathbf{c}[i] &= \begin{bmatrix} q_i \\ \eta_i \end{bmatrix} = f_i(r) = \begin{bmatrix} \phi(R_L^T(p_i - p_L)) \\ R_L^T \phi^{-1}(\xi_i) \end{bmatrix}, \\ \mathbf{r}[i] &= \begin{bmatrix} p_i \\ \xi_i \end{bmatrix} = f_i^{-1}(c) = \begin{bmatrix} R_L \phi^{-1}(q_i) + p_L \\ \phi(R_L \eta_i) \end{bmatrix}, \end{aligned} \quad (6.5)$$

donde ϕ es la transformación de coordenadas cartesianas a esféricas, y $R_L \in \mathbb{SO}(3)$ es la matriz que representa la orientación del VL, determinada por ξ_L .

Las ecuaciones de velocidad son obtenidas tomando las derivadas de (6.5). Denotando J_ϕ a la matriz jacobiana de 3×3 de ϕ , y $S(\cdot)$ la matriz de producto vectorial, entonces

$$\dot{\mathbf{c}}[i] = \begin{bmatrix} \dot{q}_i \\ \dot{\eta}_i \end{bmatrix} = \begin{bmatrix} J_\phi R_L^T [(\dot{p}_i - \dot{p}_L) - S(\dot{\xi}_L)(p_i - p_L)] \\ R_L^T [J_\phi^{-1} \dot{\xi}_i - S(\dot{\xi}_L) \phi^{-1}(\xi_i)] \end{bmatrix}. \quad (6.6)$$

Para obtener un modelo de la dinámica de la formación en el espacio del cluster de la forma $\dot{\mathbf{c}} = F(\mathbf{c}, \mathbf{u}_c)$ donde \mathbf{u}_c es la acción en el espacio del cluster, necesitamos considerar la dinámica del vector $\mathbf{r}[i]$. Para VANTs estabilizados, la misma se puede modelar con un integrador simple, por lo tanto, $\mathbf{u}_r[i] = [\dot{p}_i^T \ \dot{\xi}_i^T]^T$ representa la entrada de control en el espacio robot del i -ésimo agente. La relación entre $\mathbf{u}_c[i]$ y $\mathbf{u}_r[i]$ se obtiene a través de la matriz jacobiana de f_i

$$\mathbf{u}_c[i] = \begin{bmatrix} J_\phi R_L^T & 0 \\ 0 & R_L^T J_\phi^{-1} \end{bmatrix} \mathbf{u}_r[i] \quad (6.7)$$

Usando (6.5) y (6.7), se puede reescribir (6.6) como

$$\dot{\mathbf{c}}[i] = \mathbf{u}_c[i] - \begin{bmatrix} J_\phi R_L^T v_L + J_\phi S(R_L^T \omega_L) \phi^{-1}(q_i) \\ S(R_L^T \omega_L) \eta_i \end{bmatrix} \quad (6.8)$$

donde v_L, ω_L representan la velocidad de traslación y rotación del VL, respectivamente. Por último, la dinámica de la formación en espacio del cluster se puede escribir, a partir de (6.8), como $\dot{\mathbf{c}} = F(\mathbf{c}, \mathbf{u}_c, v_L, \omega_L)$.

6.3. Optimización para Cinematografía basada en CSC

Considere una formación de VANTs y sea $\mathbf{c}(t), \dot{\mathbf{c}}(t) \in \mathbb{R}^{dv}$ el vector de estados en espacio del cluster y su velocidad en el instante t . Se define una trayectoria artística a partir de las funciones del tiempo $\mathbf{a}(t), \dot{\mathbf{a}}(t) \in \mathbb{R}^{dv}$ especificadas en el espacio del cluster. La misma indica lineamientos artísticos, como las posiciones y los ángulos de observación de las cámaras, así como la suavidad en el movimiento de las cámaras requerida en la toma. La trayectoria artística típicamente es generada *offline*, y es adaptada por medio de maniobras *on-the-fly* realizadas por el operador. El controlador debe guiar la formación para seguir la misma lo más cerca posible, considerando restricciones como evasión de colisiones y de visibilidad mutua. Adicionalmente, el sistema incluye la posición y velocidad del POI $s(t), \dot{s}(t) \in \mathbb{R}^3$ para realizar el seguimiento del mismo con las cámaras.

A continuación presentamos una estrategia de control que genere acciones $\mathbf{u}_c(t) \in \mathbb{R}^{dv}$ para guiar al mecanismo virtual articulado, teniendo en cuenta los requerimientos artísticos y de seguridad mencionados. El esquema propuesto está basado en el problema de optimización (6.9) donde la acción de control es aquella que minimiza una función definida a partir de tres costos: (i) \mathcal{J}_a , que penaliza desviaciones de la trayectoria artística, (ii) \mathcal{J}_s pesa el error en el apuntamiento al POI, y (iii) \mathcal{J}_u que limita tanto la magnitud

como la variación de la acción de control. Funciones de restricción adicionales se utilizan para obtener una acción de control que respete la dinámica de la formación, y para evitar colisiones entre VANTs, entre los mismos y el POI, y la visibilidad mutua entre cámaras.

$$\begin{aligned}
& \underset{\mathbf{u}_c: [t_0, t_f] \rightarrow \mathbb{R}^{d_v}}{\text{minimizar}} && \int_{t_0}^{t_f} [\mathcal{J}_a(\mathbf{c}(t), \dot{\mathbf{c}}(t), \mathbf{a}(t), \dot{\mathbf{a}}(t)) + \mathcal{J}_s(\mathbf{c}(t), s(t)) + \mathcal{J}_u(\mathbf{u}_c(t), \dot{\mathbf{u}}_c(t))] dt \\
& \text{tal que} && \dot{\mathbf{c}}(t) = F(\mathbf{c}(t), \mathbf{u}_c(t)) && \text{(dinámica del cluster)} \\
& && \mathcal{C}(\mathbf{c}(t), s(t)) > 0 && \text{(evasión de colisiones)} \\
& && \mathcal{V}(\mathbf{c}(t)) > 0 && \text{(evasión de visibilidad mutua)}
\end{aligned} \tag{6.9}$$

La solución de este problema de optimización $\mathbf{u}_c^*(t)$ es la señal que opera los distintos grados de libertad del mecanismo articulado, y a través de las transformaciones cinemáticas del CSC es traducida a acciones de control para los VANTs y sus *gimbals*. El ciclo de control se cierra al alimentar al controlador en cada instante con el estado del cluster y su velocidad. Si bien (6.9) sirve para plantear conceptualmente los objetivos de control, no es apropiado resolver el mismo de forma *offline*, dado que generalmente las trayectorias artísticas y del POI no son conocidas *a priori*. En la Sección 6.3.4 se detalla el método por el cual se obtiene, de forma aproximada, la acción de control en tiempo real, mediante un abordaje de optimización por predicción de modelos. La Fig. 6.3 muestra un diagrama en bloques del sistema propuesto.

En las siguientes secciones se da un análisis detallado de las funciones de costos y restricciones.

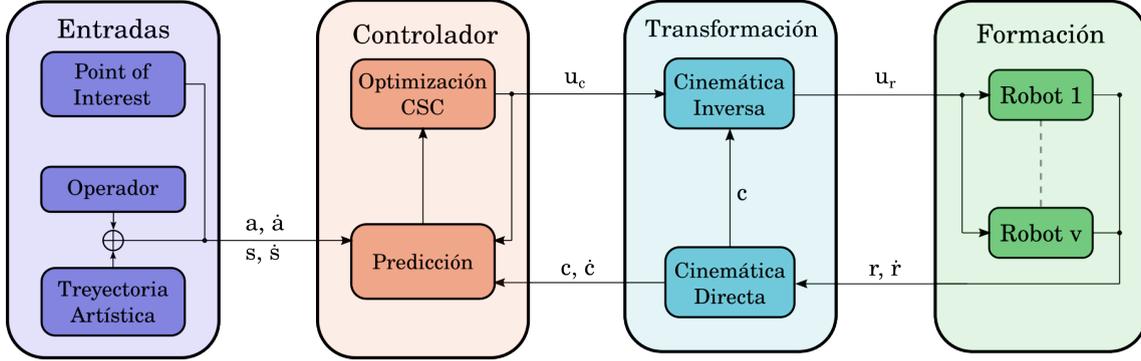


Figura 6.3: Arquitectura de control propuesta.

6.3.1. Funciones de Costo

Seguimiento de Trayectoria Artística

Para penalizar desviaciones de la trayectoria artística deseada, por parte de la formación, empleamos una función de costo

$$\mathcal{J}_a(\mathbf{c}, \dot{\mathbf{c}}, \mathbf{a}, \dot{\mathbf{a}}) := \|\mathbf{c} - \mathbf{a}\|_{Q_p}^2 + \|\dot{\mathbf{c}} - \dot{\mathbf{a}}\|_{Q_v}^2 \quad (6.10)$$

donde las normas son inducidas por matrices $Q_p, Q_v \geq 0$. Estas matrices, típicamente diagonales, pueden ser utilizadas para buscar ciertos comportamientos, como por ejemplo penalizar más fuertemente el error de seguimiento en ciertas variables, o incluso eliminar el seguimiento de la trayectoria para algunas de ellas. Por ejemplo, en el caso de la formación con $v = 2$, una definición posible para estas matrices esta dada por

$$\begin{array}{cccccccccc} x_c & y_c & z_c & \psi_c & \phi_c & d_c & \tilde{\rho}_1 & \theta_1 & \tilde{\rho}_2 & \theta_2 \\ Q_p = \text{diag}(& 1, & 1, & 3, & 1, & 1, & 0.5, & 0, & 0, & 0 &), \\ Q_v = \text{diag}(& 1, & 1, & 1, & 1, & 1, & 0, & 0, & 0, & 0 &), \end{array}$$

donde se penaliza en mayor medida el error de posición en z , y no se contempla error en las trayectorias de los ángulos *pan*, *tilt*, dado que el apuntamiento de las cámaras puede ser logrado mediante el costo de apuntamiento al POI.

Apuntamiento al POI

El $\mathcal{J}_s(\mathbf{c}, s)$, es utilizado para cumplir con objetivos de encuadre de la toma. Sea $\check{r}_{is} \in \mathbb{R}^3$ el vector unitario en que apunta de la i -ésima cámara al POI. Se desea alinear este vector con $\check{n}_i \in \mathbb{R}^3$, el vector unitario normal al plano de la cámara i , como indica Fig. 6.4. La desviación del apuntamiento deseado se puede penalizar de forma simultánea para todas las cámaras mediante el costo

$$\mathcal{J}_s(\mathbf{c}, s) := \sum_{i=1}^v q_i \|\check{r}_{is} - \check{n}_i\|^2, \quad (6.11)$$

el cual es positivo, excepto cuando todas cámaras apuntan de forma exacta al objetivo. El parámetro $q_i > 0$ puede ser elegido para ajustar la importancia relativa del apuntamiento de cada vehículo, e.g., $q_1 = 1, q_2 = 2$ le da prioridad al VANT-2 respecto del 1. La expresión de los dos vectores unitarios involucrados en este costo depende de la definición adoptada de las variables del cluster. A continuación se detalla la obtención de las mismas en los dos casos presentados en las secciones 6.2.1 y 6.2.2.

Para la formación con 2 agentes, se considera primero la posición del centroide del cluster $p_c := [x_c \ y_c \ z_c]^T$ como se muestra en la Fig. 6.2a, y los vectores unitarios \check{r}_{12} y \check{r}_{21} que apuntan de un vehículo a otro (Fig. 6.4). Estos últimos se obtienen a partir de

$$\check{r}_{12} = -R_z(\psi_c)R_x(\phi_c) \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \cos(\phi_c) \sin(\psi_c) \\ -\cos(\phi_c) \cos(\psi_c) \\ -\sin(\phi_c) \end{bmatrix}, \quad (6.12)$$

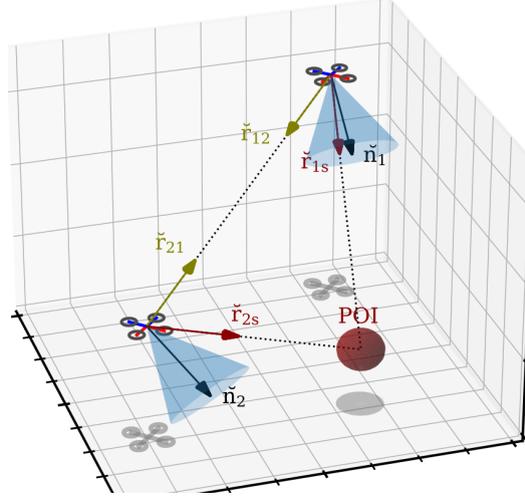


Figura 6.4: Vista esquemática del apuntamiento mutuo y al objetivo.

con $\check{r}_{21} = -\check{r}_{12}$. Con estas definiciones, es posible obtener los vectores que apuntan de cada cámara al objetivo del siguiente modo

$$r_{is} = s - (p_c + d_c \check{r}_{ji}), \quad \check{r}_{is} = r_{is} / \|r_{is}\|, \quad i, j \in \{1, 2\}. \quad (6.13)$$

Luego, el vector unitario normal al plano imagen de la cámara i se define, en términos del estado del cluster c , como

$$\check{n}_i = R_z(\psi_c + \tilde{\rho}_i) R_y(\theta_i) \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \cos(\theta_i) \cos(\psi_c + \tilde{\rho}_i) \\ \cos(\theta_i) \sin(\psi_c + \tilde{\rho}_i) \\ -\sin(\theta_i) \end{bmatrix}. \quad (6.14)$$

Para la formación de ν agentes, dado que las variables del cluster se especifican en función del líder virtual (VL), es conveniente expresar los vectores en cuestión en dicho sistema de referencia:

$$r_{is}^{VL} = s^{VL} - d_i \begin{bmatrix} \cos(\varepsilon_i) \cos(\alpha_i) \\ \cos(\varepsilon_i) \sin(\alpha_i) \\ \sin(\varepsilon_i) \end{bmatrix}, \quad \tilde{r}_{is}^{VL} = r_{is}^{VL} / \|r_{is}^{VL}\| \quad (6.15)$$

$$\tilde{n}_i^{VL} = \eta_i / \|\eta_i\| \quad (6.16)$$

Notar que si el VL se posa sobre el POI, entonces $s^{VL} = 0$, $\|r_{is}^{VL}\| = d_i$, y en ese caso

$$\tilde{r}_{is}^{VL} = - \begin{bmatrix} \cos(\varepsilon_i) \cos(\alpha_i) \\ \cos(\varepsilon_i) \sin(\alpha_i) \\ \sin(\varepsilon_i) \end{bmatrix}. \quad (6.17)$$

Función de Costo de Control

Para evitar la saturación de los actuadores de los vehículos y generar trayectorias suaves, debemos limitar la acción de control. Para ello empleamos una función de costo que penaliza tanto la norma de la acción de control, como la de su derivada:

$$\mathcal{J}_u(\mathbf{u}) := \|\mathbf{u}_c\|_{R_v}^2 + \|\dot{\mathbf{u}}_c\|_{R_a}^2 \quad (6.18)$$

donde $R_v, R_a \geq 0$. Por ejemplo, en el caso de la formación con $v = 2$, una definición posible para estas matrices esta dada por

$$\begin{array}{cccccccccccc} x_c & y_c & z_c & \psi_c & \phi_c & d_c & \tilde{\rho}_1 & \theta_1 & \tilde{\rho}_2 & \theta_2 \\ R_v = \text{diag}(& 5, & 5, & 5, & 2, & 1, & 1, & 20, & 20, & 20, & 20 &), \\ R_a = \text{diag}(& 1, & 1, & 1, & 1, & 1, & 1, & 10, & 10, & 10, & 10 &), \end{array}$$

donde se limita en mayor medida la velocidad en el ángulo *yaw* de la formación que en *roll*, y se penaliza fuertemente el movimiento de la cámara, con el objetivo de obtener una filmación suave.

6.3.2. Evasión de colisiones

Al considerar la evasión de colisiones, se deben tener en cuenta dos aspectos. El controlador debe garantizar que los vehículos en la formación no chocan entre sí, y además que ninguno de ellos se sitúe demasiado cerca del objetivo.

Evasión de colisiones entre vehículos

Para evitar la colisión entre los agentes, es necesario y suficiente que se cumpla que

$$\mathcal{C}_{ij}(\mathbf{c}) := \|r_{ij}\| - 2R_s > 0 \quad (6.19)$$

para todo par de VANTs, es decir, que la distancia sea mayor a un radio de seguridad R_s . En el caso de una formación de 2 vehículos, se sigue que $\|r_{ij}\| = 2d_c$, parámetro del cluster. Para la formación con v vehículos, $\|r_{ij}\|$ se obtiene de (6.15) como ya que $\|r_{ij}\| = \|r_{js}^{VL} - r_{is}^{VL}\|$.

Evasión de colisiones con el objetivo

En el caso de $v = 2$ vehículos, la distancia entre la cámara y el punto a filmar se obtiene tomando la norma del vector dado en (6.13). En el caso de v agentes, la distancia en cuestión coincide con la norma de r_{is}^{VL} . En ambos casos, se define la función de evasión

$$\mathcal{C}_{is}(\mathbf{c}, s) := \|r_{is}\| - 2R_s > 0. \quad (6.20)$$

Luego, se define la función de evasión de colisiones como $\mathcal{C}(\mathbf{c}, s)$ que es mayor a cero si y solo si $\mathcal{C}_{ij} > 0$ y $\mathcal{C}_{is} > 0$ para todo $i, j = 1, \dots, v$.

6.3.3. Evasión de visibilidad mutua

El controlador debe garantizar de forma autónoma que ninguno de los vehículos de la formación ingrese al campo de visión (FOV) de otro vehículo. Esto se logra definiendo el ángulo γ como la mitad del FOV, y el ángulo α_{ij} comprendido entre \check{n}_i y \check{r}_{ij} . La Fig. 6.5a muestra el esquema de una vista superior del VANT i con una cámara y su FOV resaltado, y el vehículo j cerca del FOV. Nótese que el campo de visión se encuentra representado como un cono recto circular con vértice en el centro del plano imagen y su generatriz alineada a la normal de este plano (\check{n}_i). La Fig. 6.5b representa lo que se captura desde una cámara, donde se marcó el ángulo γ como una línea del centro de la imagen al límite del FOV.

Para evitar que el vehículo j sea capturado por la cámara de i , debe garantizarse que $\alpha_{ij} > \gamma$, lo cual es equivalente a que

$$\mathcal{V}_{ij}(\mathbf{c}) := \cos(\gamma) - \check{n}_i^T \check{r}_{ij} > 0. \quad (6.21)$$

En caso de ser necesario, es posible considerar un radio de seguridad alrededor de cada

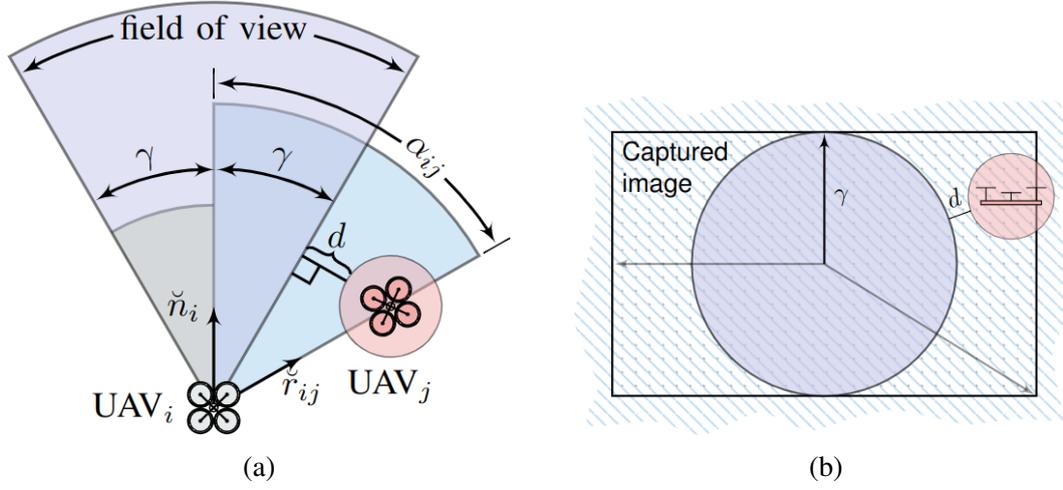


Figura 6.5: Representación esquemática del campo de visión de un VANT y la distancia de otros a éste.

VANT, R_s , resultando en un volumen esférico que no debe ser capturado por las cámaras. Esto se incluye en el modelo extendiendo el ángulo γ , mediante el agregado del término

$$\gamma_s := \arcsin\left(\frac{R_s}{2\|r_{ij}\|}\right). \quad (6.22)$$

La razón para incluir utilizar el FOV efectivo $\gamma + \gamma_s$ es que permite tener en cuenta incertezas en el estado de la formación, como puede ser en la posición de los VANTs, en las orientaciones de las cámaras, etc.

Luego, se define la función de evasión de visibilidad mutua como $\mathcal{V}(\mathbf{c})$ que es positiva si y solo si $\mathcal{V}_{ij} > 0$ para todo $i, j = 1, \dots, v$.

Es interesante notar que, en la definición de cluster para 2 vehículos (ecuaciones (6.12) y (6.14)), se puede ver mediante identidades trigonométricas, que el producto $\tilde{n}_i^T \tilde{r}_{ij}$ (6.22) depende únicamente de los ángulos $\phi_c, \tilde{\rho}_1, \tilde{\rho}_2, \theta_1, \theta_2$. Es decir, la restricción de visibilidad mutua depende del ángulo *roll* del cluster (6.2), y de los ángulos *pan-tilt* de ambas cámaras, y no es función de la posición de la formación ni del ángulo *yaw* de la misma. Esto es posible gracias a una adecuada selección de variables de estado en espacio del cluster.

En la definición para v agentes sucede algo similar, donde el producto $\check{n}_i^T \check{r}_{ij}$ depende de las variables geométricas $d_i, d_j, \varepsilon_i, \varepsilon_j, \alpha_i, \alpha_j, \rho_i, \theta_i$ y no depende de la posición del VL ni de su orientación.

6.3.4. Cómputo de la Acción de Control

Dado que generalmente las trayectorias artísticas y del POI no son conocidas *a priori*, la acción de control debe hallarse de forma iterativa. Para ello, se parte el intervalo de operación $[t_0, t_f]$ en N instantes con paso Δt , y se resuelve (6.9) en el instante t_k , de forma numérica para un horizonte de tiempo discreto $\hat{t}_{k+1}, \dots, \hat{t}_{k+H}$ con paso Δh . Para ello, se alimenta el controlador con las variables $\mathbf{c}, \dot{\mathbf{c}}, \mathbf{a}, \dot{\mathbf{a}}, s, \dot{s}$ en obtenidas en t_k , y se realizan predicciones de estas para $h = 1, \dots, H$ mediante:

$$\hat{\mathbf{c}}(\hat{t}_{k+h}) = F(\hat{\mathbf{c}}(\hat{t}_{k+h-1}), \mathbf{u}_c(t_k)), \quad \dot{\hat{\mathbf{c}}}(\hat{t}_k) = \dot{\mathbf{c}}(t_k) \quad (6.23)$$

$$\hat{\mathbf{c}}(\hat{t}_{k+h}) = \hat{\mathbf{c}}(\hat{t}_{k+h-1}) + \dot{\hat{\mathbf{c}}}(\hat{t}_{k+h-1}) \Delta h, \quad \hat{\mathbf{c}}(\hat{t}_k) = \mathbf{c}(t_k) \quad (6.24)$$

$$\hat{\mathbf{a}}(\hat{t}_{k+h}) = \hat{\mathbf{a}}(\hat{t}_{k+h-1}) + \dot{\hat{\mathbf{a}}}(t_k) \Delta h, \quad \hat{\mathbf{a}}(\hat{t}_k) = \mathbf{a}(t_k) \quad (6.25)$$

$$\hat{s}(\hat{t}_{k+h}) = \hat{s}(\hat{t}_{k+h-1}) + \dot{\hat{s}}(t_k) \Delta h, \quad \hat{s}(\hat{t}_k) = s(t_k) \quad (6.26)$$

donde la segunda columna de las expresiones anteriores representan las condiciones iniciales a partir de las cuales se predicen los estados en la ventana de tiempo. Se obtiene

entonces, numéricamente, la acción $\mathbf{u}_c^*(t_k)$ que, basándose en estas predicciones, resuelve:

$$\begin{aligned} \underset{\mathbf{u}_c(t_k)}{\text{minimizar}} \quad & \sum_{h=1}^H [\mathcal{J}_a(\hat{\mathbf{c}}(\hat{t}_{k+h}), \hat{\mathbf{c}}(\hat{t}_{k+h}), \hat{\mathbf{a}}(\hat{t}_{k+h}), \hat{\mathbf{a}}(t_k)) + \mathcal{J}_s(\hat{\mathbf{c}}(\hat{t}_{k+h}), \hat{s}(\hat{t}_{k+h})) \\ & + \mathcal{J}_u(\mathbf{u}_c(t_k), \hat{\mathbf{u}}_c(t_k))] \end{aligned} \quad (6.27)$$

tal que $\forall h \in \{1, \dots, H\}$:

$$\mathcal{C}(\mathbf{c}(\hat{t}_{k+h}), s(\hat{t}_{k+h})) > 0 \quad (\text{evasión de colisiones})$$

$$\mathcal{V}(\mathbf{c}(\hat{t}_{k+h})) > 0 \quad (\text{evasión de visibilidad mutua})$$

donde $\hat{\mathbf{u}}_c(t_k) := (\mathbf{u}_c(t_k) - \mathbf{u}_c^*(t_{k-1}))/\Delta t$. Este proceso es repetido en cada paso de control t_k para obtener la acción de control $\mathbf{u}_c^*(t_k)$ correspondiente. Notar que si la trayectoria artística $\mathbf{a}(t_k)$, y/o la trayectoria del POI $s(t_k)$, son definidas *offline*, es posible reemplazar sus predicciones por los verdaderos valores.

6.4. Validación

Para validar el método propuesto, se presentan resultados de simulación con diferentes escenas y tipos de tomas. En [128] se definen tomas típicas para sistemas de uno o más VANTs equipados con cámaras. Siguiendo estas definiciones, se presentan los resultados obtenidos realizando las tomas de *Fly-over* y *Orbital*, primero con 2 vehículos, y luego para 4, empleando las dos definiciones de cluster presentadas.

El banco de pruebas utilizado durante estas simulaciones consiste de hexacópteros *Typhoon h480* equipados con cámaras pan-tilt como vehículos filmadores, y un cuadrotor *Iris* como el punto de interés (POI) a ser filmado, ver Fig. 6.6. Se debe destacar que el FOV de las cámaras que poseen los *Typhoon* es configurado en las simulaciones para tener 120° , pero en el controlador se configura para tener 60° . Esto se hace porque resulta más

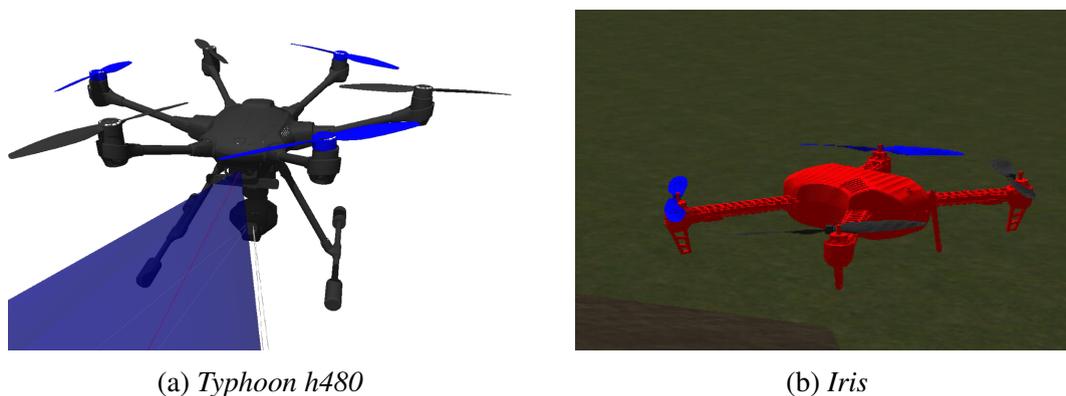


Figura 6.6: Modelos de los vehículos empleados para la validación.

sencillo visualizar los resultados en imágenes *first-person-view* (FPV) donde los vehículos filmadores son captados por el FOV-120°, pero rechazados por el FOV-60°, lo que permite visualizar en la imagen FPV la evasión de la visibilidad mutua. Esto se observa, por ejemplo, en Fig. 6.9.

Para el controlador estabilizante de los vehículos, es decir, aquel de bajo nivel que realiza un seguimiento a lazo cerrado de comandos de velocidad, se utilizó una versión del *stack* de PX4 (v1.11.0), estándar en este tipo de aplicaciones. El mismo corre en un entorno de ROS *Kinetic* [129], sistema operativo de robots de uso extendido, y es simulado en Gazebo 7, por medio del paquete *sitl_gazebo* de PX4. El controlador propuesto en la Sección 6.3.4, se implementó utilizando el paquete *optimize* de SciPy en PYTHON, mediante la técnica *Sequential Least Squares Programming* (SLSQP). El mismo corre en una estación de cómputo terrestre, la cual se comunica con los vehículos utilizando una interfaz Mavlink, provista en el paquete *mavros*. El sistema corre dentro de un contenedor en la plataforma de simulación OpenUAV [130]. Los resultados obtenidos en las distintas simulaciones pueden encontrarse en https://youtu.be/_OZ_zi9vtF8/ y en <https://youtu.be/DzzwkLm002Q>.

6.4.1. Ensayos con 2 VANTs

Toma *Orbital*

En la toma *orbital*, la formación rota alrededor de un punto en el espacio —en esta prueba en particular, ese punto es (0m, 0m, 10m)— siguiendo con las cámaras al Iris, el VANT objetivo. Este realiza un movimiento ascendente desde la posición (0m, 0m, 0m) a (0m, 0m, 20m). Entonces, mientras las cámaras siguen la trayectoria del Iris, cuando este se acerca a la altura $z = 10\text{m}$, los VANTs filmadores se ven forzados a modificar sus trayectorias y el apuntamiento de forma tal de evitar que ambas cámaras se apunten mutuamente, mientras intentan mantener enmarcado al objetivo. Cabe mencionar que esta toma estresa al sistema en una de las situaciones más complejas en cuanto a la optimización, ya que la trayectoria artística implica que ambos VANTs pasan por el centro de la imagen del otro VANT en forma simultánea.

Como fue discutido en la Sección 6.3.3, para evitar la visibilidad mutua, el controlador puede actuar sobre los estados $\phi_c, d_c, \tilde{p}_1, \tilde{p}_2, \theta_1, \theta_2$ que, como se observa en la Fig. 6.7, son los que más varían durante la prueba.

Para mantener al objetivo enmarcado, el controlador desvía a la formación de la trayectoria de referencia para satisfacer las restricciones de visibilidad. Este comportamiento puede observarse en la Fig. 6.7, en el intervalo de tiempo de $t \approx 300\text{s}$ a $t \approx 325\text{s}$. En esta figura puede observarse, también, que el despegue de los vehículos se realiza en $t \approx 222\text{s}$. Antes de ese tiempo, el controlador no puede actuar sobre todos los estados, por lo que se producen los transitorios que se observan hasta $t \approx 230\text{s}$.

Para esta toma, los pesos de las funciones de costo fueron ajustados para seguir la rotación definida en la trayectoria, penalizando en mayor medida el ángulo *yaw* de la formación ψ_c , permitiendo mayores desviaciones en la posición del centroide x_c, y_c, z_c , la distancia entre VANTs d_c , y el ángulo roll ϕ_c , para cumplir con las restricciones. El control

independiente de cada peso permite realizar un ajuste fino en la respuesta del controlador según la toma que se quiera realizar.

La Fig. 6.8 muestra los ángulos entre las normales del plano de cámara y el vector de apuntamiento mutuo $\alpha_{12} := \arccos(\check{n}_1^T \check{r}_{12})$ y $\alpha_{21} := \arccos(\check{n}_2^T \check{r}_{21})$. También se muestran, en la misma figura, los ángulos entre \check{n}_i y la línea que une a la cámara i con el objetivo \check{r}_{is} , es decir, $\alpha_{1s} := \arccos(\check{n}_1^T \check{r}_{1s})$ y $\alpha_{2s} := \arccos(\check{n}_2^T \check{r}_{2s})$. Con una línea punteada, se muestra el ángulo γ de la cámara. Esta figura permite verificar si una cámara ingresa en el campo de visión de la otra. Para que se evite la visibilidad mutua, se debe cumplir que α_{12} y α_{21} sean mayores a γ , mientras que α_{1s} y α_{2s} es deseable que se mantengan cercanos a cero, lo que indica que el objetivo está en el centro de la imagen.

En la Fig. 6.9 se muestra una imagen de la simulación en el entorno Gazebo sobre la que se resaltó, para cada vehículo, parte de la trayectoria que corresponde al intervalo de tiempo para t entre 300s y 325s. Además, en la misma figura, se muestra la imagen FPV de cada VANT para instantes de tiempo seleccionados. Puede verse que por un momento el vehículo de la izquierda pierde de vista al POI, lo que se debe a que la trayectoria es compleja, desde el punto de vista de las restricciones, y el seguimiento del objetivo tiene un costo, pero no se exige que, en todo momento, esté en el interior del FOV. Así, un vehículo que tiene un menor peso en el seguimiento del objetivo, lo pierde de vista para cumplir con las restricciones impuestas. Sin embargo, esto se esperaba, ya que puede verse a partir del gráfico de la Fig. 6.8, en $t \approx 312$ s, que α_{2s} cruza momentáneamente la línea del campo de visión, γ , indicando exactamente lo que se ve en la imagen.

Toma *Fly-Over*

En esta escena, la formación es comandada para realizar un vuelo sobre un objetivo fijo, ubicado en $(15\text{ m}, 0\text{ m}, 10\text{ m})$, mientras la formación comienza en $p_c = (0\text{ m}, 10\text{ m}, 15\text{ m})$ con una distancia de cada vehículo al centro de la formación $d_c = 10\text{ m}$, y $\psi_c = \phi_c = 0\text{ rad}$.

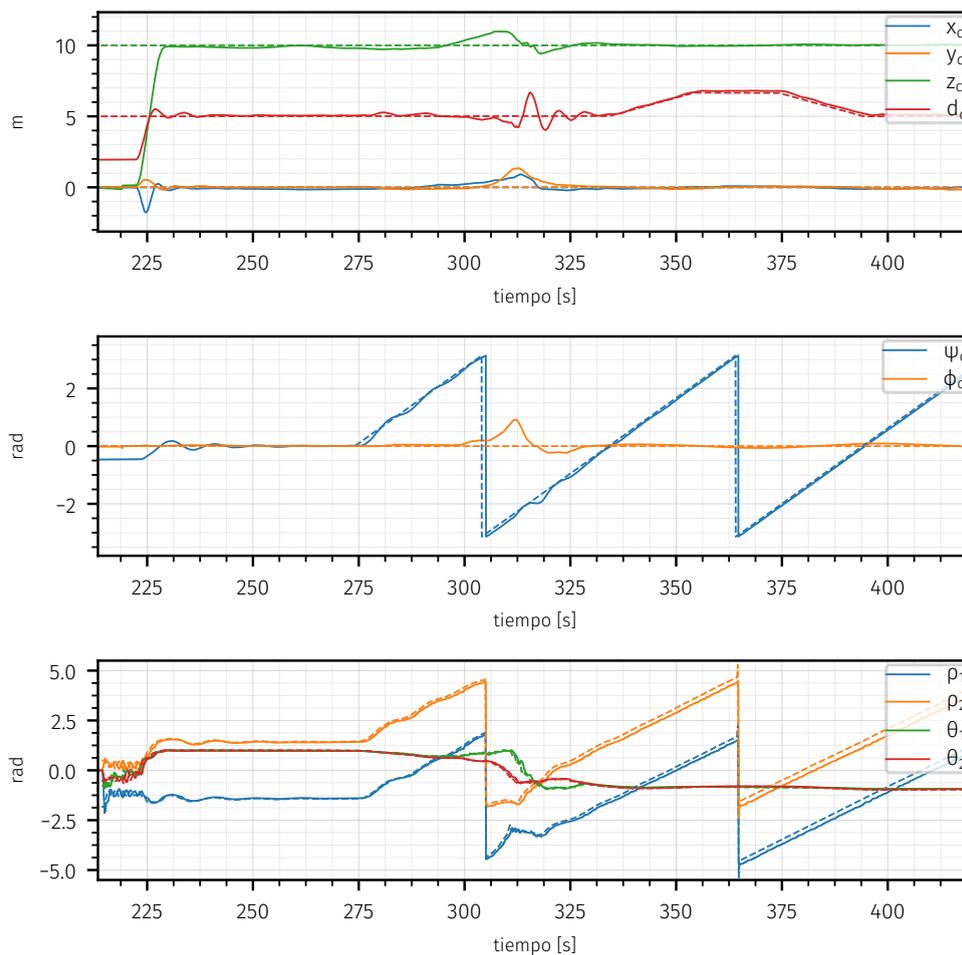


Figura 6.7: Evolución de los estados del cluster mientras se realiza la toma orbital. En líneas punteadas se muestran las referencias.

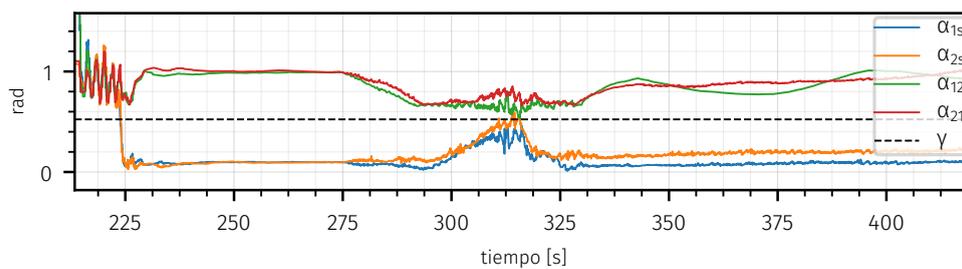


Figura 6.8: Ángulos formados entre la normal de cada cámara y la línea que las une con los demás actores de la escena (el otro VANT y el objetivo) durante la toma orbital.

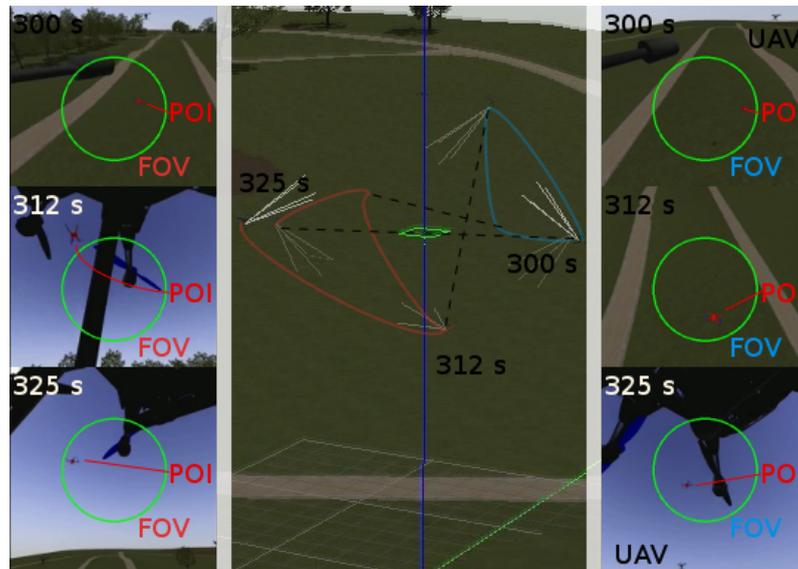


Figura 6.9: Trayectoria orbital aproximada y FPV para t entre 300 s y 325 s.

En $t \approx 200$ s, se comanda al sistema para volar hacia el objetivo a una velocidad en x_c de 0.5 m s^{-1} , usando la señal del operador incluido en el controlador. Con esta configuración, mientras los vehículos sobrevuelan al objetivo, el controlador debe rotar las cámaras para mantener al objetivo correctamente encuadrado. Al realizar esta maniobra, el VANT que se encuentra más próximo al POI sería capturado en la imagen del vehículo que se mueve por el costado. En Fig. 6.10 se muestra, de forma esquemática, la trayectoria resultante. Como se ve en la Fig. 6.11, el controlador genera desviaciones de la referencia en los ángulos de yaw y roll por aproximadamente -1 rad y -0.5 rad, respectivamente, y pequeños cambios en d_c , y así logra mantener a ambos vehículos fuera de los FOVs (Fig. 6.12) a la vez que sigue la trayectoria artística. Más aún, una vez que el POI es correctamente encuadrado por ambas cámaras (en $t \approx 175$ s), el encuadre se mantiene, en ambas cámaras, a lo largo de toda la toma, con apartamientos del centro de la imagen menores a 0.1 rad (en promedio) como se muestra en la Fig. 6.12.

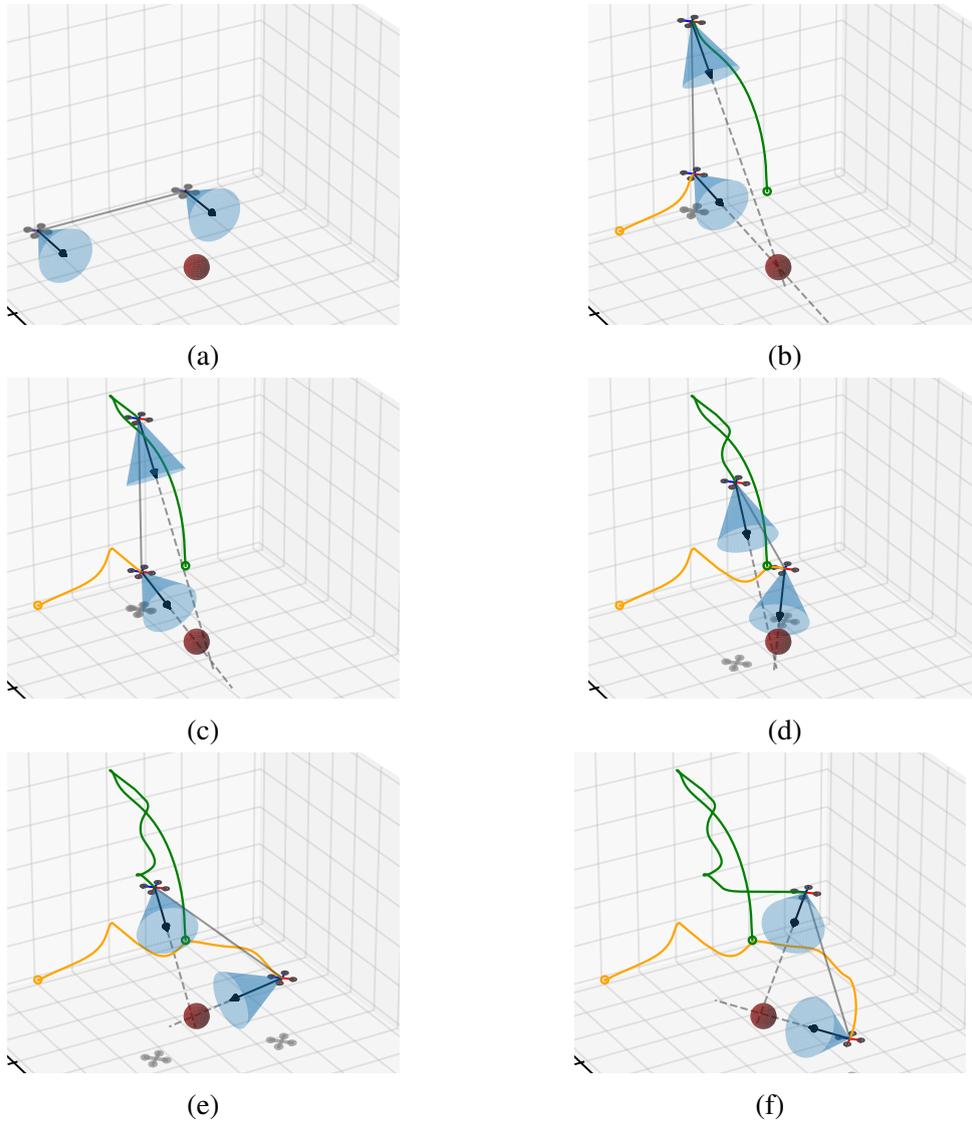


Figura 6.10: Representación esquemática de la trayectoria *flyover*.

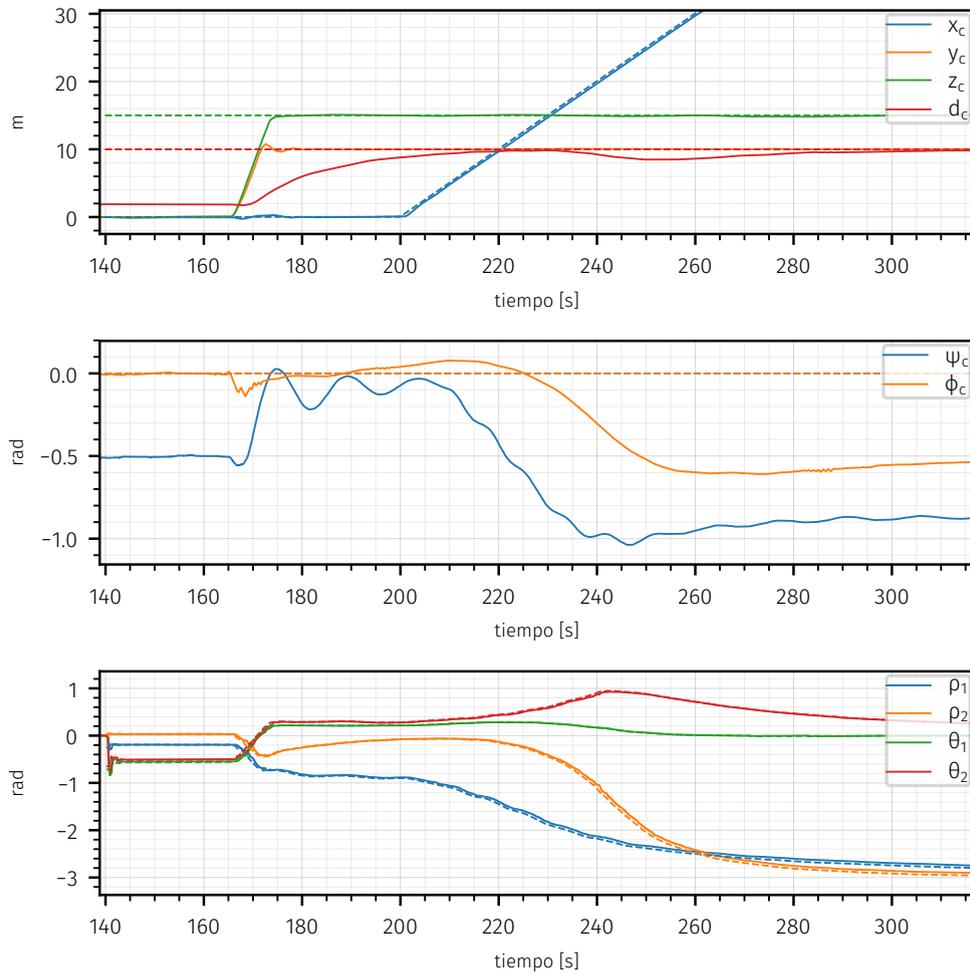


Figura 6.11: Evolución de los estados del cluster mientras se realiza la toma fly-over. En líneas punteadas se muestran las referencias.

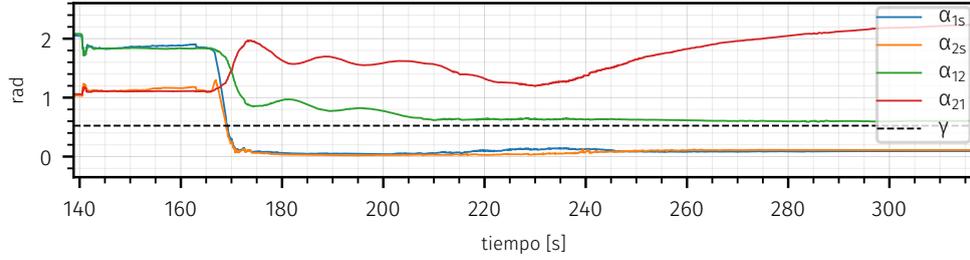


Figura 6.12: Ángulos formados entre la normal de cada cámara y la línea que las une con los demás actores de la escena (el otro VANT y el objetivo) durante la toma fly-over.

6.4.2. Ensayos con v VANTs

De forma similar a cómo se obtuvieron los resultados anteriores, con la definición de variables de cluster para v VANTs, realizamos dos tomas orbitales con $v = 4$. Las mismas consisten en la formación describiendo un círculo de radio 14.14 m, mientras las cámaras apuntan a un *Iris* estático ubicado a 30 m del *pivot* o VL de la grúa articulada virtual, lo cual hace que a medida que la grúa rota, la visibilidad mutua se vea comprometida. La trayectoria artística consiste únicamente en una velocidad constante de rotación respecto del eje z del sistema de referencia solidario al VL. A pesar de la simpleza de la trayectoria artística, resulta en una situación que compromete los objetivos del control, dado que los VANTs deben repetida y alternadamente desviarse de la misma para evitar la visibilidad mutua. Para evaluar el comportamiento del sistema propuesto, realizamos dos tomas correspondientes a diferentes velocidades de rotación, que inducen en el primer caso velocidades tangenciales de los VANTs de $\|\dot{p}_i\| = 2.2 \text{ m s}^{-1}$, y en el segundo caso de $\|\dot{p}_i\| = 4.4 \text{ m s}^{-1}$. La evolución de los ángulos α_{ij} y α_{is} para $i, j = 1, 2, 3, 4$ definidos de igual forma que en el caso con 2 vehículos, se muestran en Fig. 6.13. Se observa como, luego de los segundos iniciales, a partir de $t \approx 15 \text{ s}$, los ángulos de apuntamiento mutuo α_{ij} siguen trayectorias oscilantes, producto de la rotación de la formación respecto del VL. Se puede ver cómo, el controlador logra mantener ángulos de apuntamiento mutuo por encima del ángulo $\tilde{\gamma} := \gamma + \gamma_s$ en todo momento, excepto por algunos breves períodos. Es-

to puede deberse a las incertezas en las posiciones de los VANTs producto de los errores en las mediciones, y de perturbaciones externas, como puede ser el viento, que estresan al controlador. Por otro lado, los ángulos de apuntamiento al POI, se mantienen cercanos a cero, indicando do que el objetivo se encuentra bien encuadrado durante la mayor parte del tiempo de operación.

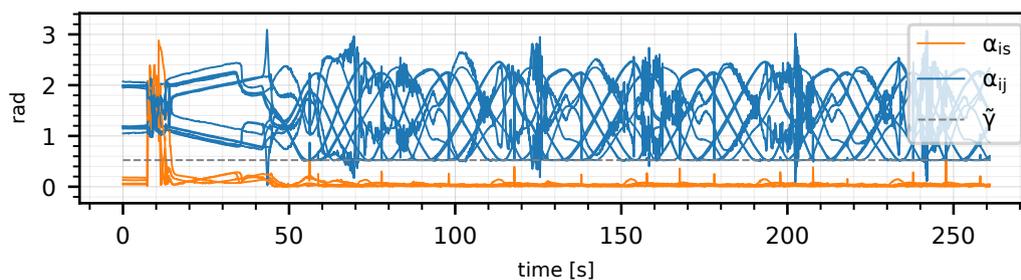
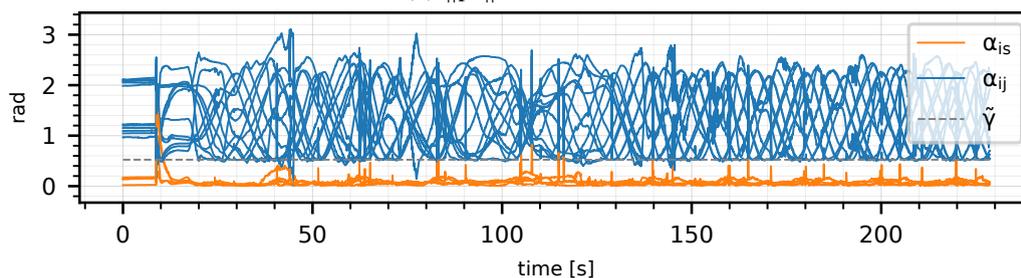
(a) $\|\dot{p}_i\| = 2.2 \text{ m s}^{-1}$ (b) $\|\dot{p}_i\| = 4.4 \text{ m s}^{-1}$

Figura 6.13: Ángulos formados entre la normal de cada cámara y la línea que las une con los demás actores de la escena (el otro VANT y el objetivo).

6.5. Resumen

Se presentó en este capítulo una configuración para un sistema de filmaciones aéreas autónomo, basado en la formulación de Control en Espacio del Cluster (CSC). Con este esquema, un sistema de múltiples VANTs con cámaras pan-tilt es concebido como una grúa robótica virtual, capaz de seguir trayectorias generadas *offline* a la vez que maniobras

comandadas por un operador de cámara en tiempo real, mientras se mantiene al punto de interés encuadrado. Como las variables de estado del cluster son definidas basándose en la aplicación, los parámetros controlados son intuitivos —y son los mismos que un operador terreno cambiaría para hacer modificaciones a la trayectoria en tiempo de ejecución. Además, el sistema garantiza que los VANTs filmadores no van a aparecer en el campo de visión de los demás, y que las colisiones son evitadas, con el costo de desviarse de la trayectoria predefinida, liberando al operador de estas tareas.

Dado que el sistema de grúa virtual está definido en términos de la formulación en espacio del cluster, se puede extender sin dificultades a sistemas con un mayor número de cámaras. Además, como la mayor parte de las restricciones son geométricas, y los parámetros del estado del cluster se definen a través de relaciones geométricas, su integración en el algoritmo de optimización es simple. Por esta misma razón, podemos identificar el esquema de coordinación propuesto como una formación.

El sistema propuesto fue evaluado primero empleando una implementación puramente en PYTHON, luego extendida a ROS, Gazebo y las herramientas de PX4. En nuestra experiencia con esquemas similares, la validación mediante herramientas de simulación de uso extendido, como las mencionadas, tiene una gran correlación con la validación experimental. Los temas desarrollados en este Capítulo fueron presentados en [74].

Capítulo 7

Conclusiones

7.1. Comentarios Finales

El abordaje formal propuesto para estudiar las formaciones de vehículos no tripulados (VNT) como de frameworks y frameworks generalizados ha mostrado poseer gran versatilidad. Los mismos permiten la incorporación, a los esquemas de control, de la red de relaciones existente en la formación, ya sea asociada a enlaces de comunicación entre agentes, a relaciones de sensado mutuo, o a requerimientos operativos. Esto se ha hecho con suficiente generalidad, sin especificar un dominio de operación en particular, de forma que los resultados obtenidos son aplicables tanto a vehículos terrestres, como aéreos o acuáticos.

Dos problemas distintos, pero ambos de interés en la comunidad de robótica, fueron tomados como plataformas para detectar desafío importantes en el control de sistemas multirobot. El primero, referido al control descentralizado de una red de VNTs con acceso parcial a los sistemas de posicionamiento externo, permitió estudiar en profundidad la influencia de la red de conexiones y mediciones en el comportamiento del sistema, y, por lo tanto, sus implicaciones en las estrategias de control. El segundo, consistente en el con-

trol de formaciones de VANTs para cinematografía aérea, permitió validar la técnica de Control en el Espacio del Cluster para simplificar el diseño y la operación de formaciones en aplicaciones con múltiples objetivos y dinámicas complejas.

Resultó clarificador, para comparar las estrategias de control de redes multirobot, analizar el problema del manejo de la información. La óptica adoptada de analizar, por un lado, el tipo de información a procesar (global o local), y por otro la forma en que la misma es procesada (centralizada o descentralizada) permitió entender por qué los algoritmos que deben procesar información global presentan limitaciones en cuanto a su escalabilidad. Estas ideas fueron profundizadas en el Capítulo 3 donde se presentaron nuevos resultados teóricos que permiten comprender mejor la relación entre la rigidez de los frameworks con la conectividad y el diámetro de los grafos subyacentes. Esto ayudó a visualizar cómo evoluciona la rigidez con el tamaño de una formación de robots móviles, lo que evidencia las ventajas de presentar una nueva forma de describir la misma en función de subframeworks. En el Capítulo 4 se discute en profundidad esta noción, que supone un aporte al estudio de la rigidez al describir la misma como una propiedad distribuida a lo largo de la red. Para ello, definimos formalmente el concepto de subframework y presentamos los teoremas que permiten expresar la rigidez en función de los mismos. En el Capítulo 5 se aplicó la teoría de subframeworks para diseñar el control coordinado del mantenimiento de la rigidez de un framework. Mediante validaciones mostramos que el mismo cumple con los objetivos planteados, y además presenta un comportamiento emergente asociado a la formación de un patrón triangular, el cual no es especificado en las primitivas de control. El esquema propuesto resulta ventajoso respecto de protocolos anteriores debido a que no requiere el intercambio de información global, evitando los problemas asociados en relación con la escalabilidad.

Por último, en el Capítulo 6 se presentó el caso de aplicación para filmaciones aéreas autónomas, basado en la formulación de Control en Espacio del Cluster (CSC). En

el mismo, los objetivos planteados fueron tanto operativos (evasión de colisiones) como artísticos (suavidad, visibilidad mutua), lo que muestra la versatilidad de esta técnica. Este caso sirve para visualizar cómo al modelar a la formación mediante un estado donde las variables son definidas basándose en la aplicación, se simplifica tanto el diseño del controlador, como la especificación y el comando por parte de un operador. Además, sirvió para validar la factibilidad de utilizar controlador basado en problemas de optimización para contemplar funciones de costo complejas que contemplen las interrelaciones entre los agentes, como es el caso de la evasión de visibilidad mutua.

Si bien los aportes presentados en esta tesis son valiosos, quedan problemas por resolver, algunos de los cuales fueron mencionados aquí. Resulta atractivo atacar estos problemas en el futuro, valiéndonos de la experiencia adquirida durante el período de esta tesis.

7.2. Trabajo Futuro

Entre las posibles líneas de investigación que surgen a partir del trabajo desarrollado en esta tesis destacamos las siguientes.

En primer lugar, es deseable continuar el estudio teórico de la rigidez, en particular, para comprender mejor la conectividad algebraica generalizada de frameworks [94, 104, 105]. Esto permitiría aprovechar unos cimientos teóricos más sólidos para mejorar los algoritmos de control.

En segundo lugar, resulta sumamente atractivo profundizar el estudio de las descomposiciones en subframeworks dispersas. En primer lugar, queremos diseñar un algoritmo descentralizado para resolver el Algoritmo 1, que permite obtener una descomposición dispersa subóptima. Además, se desea estudiar métodos para implementar un esquema de control de rigidez del framework vinculante. Esto podría lograrse mediante la inclusión al grupo de robots de un vehículo supervisor, por ejemplo un VANT, que puede asistir a un

subconjunto de robots terrestres mediante comunicaciones punto a punto con información que permita mantener la rigidez de este framework virtual. Además, en este contexto, resulta interesante validar el esquema control por subframeworks en entornos más realistas, por medio del uso de simuladores específicos para redes de robots, como por ejemplo [130, 131]. Para ello, es atractivo dotar de mayor realismo a los modelos empleados, como por ejemplo, contemplar el estado de la batería de los robots, la pérdida de paquetes durante la comunicación, la colisión y oclusión con objetos físicos, entre otras.

En tercer lugar, resulta muy atractivo extender los estudios realizados sobre frameworks de distancia a otras funciones más generales, que den lugar a frameworks dirigidos e incluso a frameworks generalizados. Estudiar las matrices laplacianas que surjan de ellos y sus propiedades resulta sumamente deseable. Aplicar los conceptos relacionados con la descomposición en subframeworks de este tipo de esquemas tiene un gran potencial debido a la gran versatilidad de los frameworks generalizados para modelar prácticamente cualquier tipo de formación de robots. Un punto de partida sería el estudio de la rigidez de frameworks de dirección relativa, el cual ha atraído el interés de los investigadores de robótica, dado que posee propiedades interesantes desde el punto de vista del control descentralizado [132, 67, 68]. Por ejemplo, es posible diseñar un algoritmo de posicionamiento de robots a partir de mediciones de dirección relativa cuya dinámica sea lineal, lo cual permite obtener garantías de convergencia global, y facilita su implementación. Sin embargo, la implementación descentralizada de algoritmos de control del mantenimiento de la rigidez asociada a las mediciones de dirección, sigue siendo un problema abierto. Creemos que es posible emplear los conceptos de subframeworks presentados aquí para tratar ese problema, lo cual supondría un gran avance en esa materia. En este mismo sentido, resulta interesante profundizar el estudio y la utilización de la técnica de Control en el Espacio del Cluster en términos de frameworks generalizados. Si bien esta estrategia es muy versátil dado que permite contemplar una vasta cantidad de problemas relacionados

con formaciones de robots, una implementación descentralizada de la misma sigue siendo un desafío. Creemos que los conocimientos adquiridos respecto de los frameworks generalizados permitirán abordar este problema con esa óptica y de esa manera ayudar a lograr este objetivo.

Bibliografía

- [1] Francisco Javier Pérez-Grau et al. «Semi-Autonomous Teleoperation of UAVs in Search and Rescue Scenarios». En: *2017 International Conference on Unmanned Aircraft Systems (ICUAS)* (2017), págs. 1066-1074.
- [2] Matthew David Dunbabin y Lino Marques. «Robots for Environmental Monitoring: Significant Advancements and Applications». En: *IEEE Robotics & Automation Magazine* 19 (2012), págs. 24-39.
- [3] Robin Ritz et al. «Cooperative Quadcopter Ball Throwing and Catching». En: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2012), págs. 4972-4978.
- [4] Bruno Gabrich et al. «A Flying Gripper Based on Cuboid Modular Robots». En: *2018 IEEE International Conference on Robotics and Automation (ICRA)* (2018), págs. 7024-7030.
- [5] Francisco Guido Rossomando et al. «Aerial Load Transportation with Multiple Quadrotors Based on a Kinematic Controller and a Neural SMC Dynamic Compensation». En: *Journal of Intelligent & Robotic Systems* (2020), págs. 1-12.
- [6] Rogerio Bonatti et al. «Autonomous Aerial Cinematography in Unstructured Environments with Learned Artistic Decision-Making». En: *Journal of Field Robotics* (ene. de 2020). ISSN: 1556-4967. DOI: [10.1002/rob.21931](https://doi.org/10.1002/rob.21931).

- [7] Yongcan Cao et al. «An Overview of Recent Progress in the Study of Distributed Multi-Agent Coordination». En: *IEEE Transactions on Industrial Informatics* 9.1 (2013), págs. 427-438. DOI: [10.1109/TII.2012.2219061](https://doi.org/10.1109/TII.2012.2219061).
- [8] Brian D. O. Anderson et al. «UAV Formation Control: Theory and Application». En: *Recent Advances in Learning and Control*. Ed. por Vincent D. Blondel, Stephen P. Boyd e Hidenori Kimura. London: Springer London, 2008, págs. 15-33. ISBN: 978-1-84800-155-8.
- [9] Florian Shkurti et al. «Multi-Domain Monitoring of Marine Environments Using a Heterogeneous Robot Team». En: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012, Vilamoura, Algarve, Portugal, October 7-12, 2012*. IEEE, 2012, págs. 1747-1753. ISBN: 978-1-4673-1737-5. DOI: [10.1109/IROS.2012.6385685](https://doi.org/10.1109/IROS.2012.6385685). URL: <http://dx.doi.org/10.1109/IROS.2012.6385685>.
- [10] Shannon Hood et al. «Bird's Eye View: Cooperative Exploration by UGV and UAV». En: *2017 International Conference on Unmanned Aircraft Systems (ICUAS)* (2017), págs. 247-255.
- [11] Khaled A. Ghamry, Mohamed Anouar El Kamel y Youmin Zhang. «Multiple UAVs in Forest Fire Fighting Mission Using Particle Swarm Optimization». En: *2017 International Conference on Unmanned Aircraft Systems (ICUAS)* (2017), págs. 1404-1409.
- [12] H. X. Pham et al. «A Distributed Control Framework for Multiple Unmanned Aerial Vehicles for Dynamic Wildfire Tracking». En: *IEEE Transactions on Systems, Man and Cybernetics: Systems, April 2018* April.5 (2018).

- [13] Tobias Nageli et al. «Real-time Planning for Automated Multi-View Drone Cinematography». En: *ACM Transactions on Graphics* 36.4 (2017), 132:1-132:10. ISSN: 0730-0301. DOI: [10.1145/3072959.3073712](https://doi.org/10.1145/3072959.3073712).
- [14] Tobias Nageli et al. «Real-Time Motion Planning for Aerial Videography With Dynamic Obstacle Avoidance and Viewpoint Optimization». En: *IEEE Robotics and Automation Letters* 2.3 (jul. de 2017), págs. 1696-1703. ISSN: 2377-3774. DOI: [10.1109/LRA.2017.2665693](https://doi.org/10.1109/LRA.2017.2665693).
- [15] Jonatan Olofsson, Edmund Førland Brekke y Tor Arne Johansen. «Cooperative Remote Sensing of Ice Using a Spatially Indexed Labeled Multi-Bernoulli Filter». En: *2017 International Conference on Unmanned Aircraft Systems (ICUAS)* (2017), págs. 591-600.
- [16] Ezequiel Pecker Marcosig, Juan I. Giribet y Rodrigo Castro. «Hybrid Adaptive Control for UAV Data Collection: A Simulation-Based Design to Trade-Off Resources Between Stability and Communication». En: *2017 Winter Simulation Conference (WSC)*. 2017, págs. 1704-1715.
- [17] Sayyed Majid Esmailifar y Fariborz Saghafi. «Cooperative Localization of Marine Targets by UAVs». En: *Mechanical Systems and Signal Processing* 87 (2017), págs. 23-42.
- [18] Christopher A. Kitts et al. «Experiments in the Control and Application of Automated Surface Vessel Fleets». En: *OCEANS'11 MTS/IEEE KONA* (2011), págs. 1-7.
- [19] Julian Barreiro-Gomez et al. «Distributed Formation Control of Multiple Unmanned Aerial Vehicles Over Time-Varying Graphs Using Population Games». En: *2016 IEEE 55th Conference on Decision and Control (CDC)* (2016), págs. 5245-5250.

- [20] Vijay Kumar, Naomi Leonard y A. Stephen Morse. «Cooperative Control: A Post-Workshop Volume». En: *2003 Block Island workshop on cooperative control*. 2005.
- [21] Hung La, Robert Lim y Weihua Sheng. «Multirobot Cooperative Learning for Predator Avoidance». En: *Control Systems Technology, IEEE Transactions on* 23 (ene. de 2015), págs. 52-63. DOI: [10.1109/TCST.2014.2312392](https://doi.org/10.1109/TCST.2014.2312392).
- [22] Chih-han Yu y Radhika Nagpal. «Biologically-Inspired Control for Multi-Agent Self-Adaptive Tasks». En: *Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI)*. 2010. URL: [files/ssr/files/aaai10-yu.pdf](https://files.ssr/files/aaai10-yu.pdf).
- [23] Ignacio Mas. «Cluster Space Framework For Multi-Robot Formation Control». Tesis doct. Santa Clara University, School of Engineering, ago. de 2011. URL: <https://sculib.scu.edu/record=b2563579>.
- [24] Ignacio Mas et al. «Formation Control for Multi-Domain Autonomous Vehicles Based on Dual Quaternions». En: *International Conference on Unmanned Aircraft Systems - ICUAS*. Jun. de 2017, págs. 723-730. DOI: [10.1109/ICUAS.2017.7991315](https://doi.org/10.1109/ICUAS.2017.7991315).
- [25] Pedro G. Artaxo et al. «Autonomous Cooperative Flight Control for Airship Swarms». En: *ArXiv abs/2004.07665* (2020).
- [26] Naomi Ehrich Leonard y Edward Fiorelli. «Virtual Leaders, Artificial Potentials and Coordinated Control of Groups». En: *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No.01CH37228)* 3 (2001), 2968-2973 vol.3.
- [27] Peng Song y V. Kumar. «A Potential Field Based Approach to Multi-Robot Manipulation». En: *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*. Vol. 2. 2002, 1217-1222 vol.2. DOI: [10.1109/ROBOT.2002.1014709](https://doi.org/10.1109/ROBOT.2002.1014709).

- [28] Jonathan R. Fink, M. A. Hsieh y Vijay R. Kumar. «Multi-Robot Nanipulation via Caging in Environments with Obstacles». En: *2008 IEEE International Conference on Robotics and Automation* (2008), págs. 1471-1476.
- [29] Tucker R. Balch y Ronald C. Arkin. «Behavior-Based Formation Control for Multirobot Teams». En: *IEEE Trans. Robotics Autom.* 14 (1998), págs. 926-939.
- [30] Christopher A. Kitts e Ignacio Mas. «Cluster Space Specification and Control of Mobile Multirobot Systems». En: *IEEE/ASME Transactions on Mechatronics* 14 (2009), págs. 207-218.
- [31] Calin A. Belta y Vijay R. Kumar. «Abstraction and control for Groups of robots». En: *IEEE Transactions on Robotics* 20 (2004), págs. 865-875.
- [32] Martin Saska et al. «System for Deployment of Groups of Unmanned Micro Aerial Vehicles in GPS-Denied Environments Using Onboard Visual Relative Localization». En: *Autonomous Robots* 41 (2017), págs. 919-944.
- [33] Zhiyun Lin et al. «Leader-Follower Formation via Complex Laplacian». En: *Autom.* 49 (2013), págs. 1900-1906.
- [34] Zhao Yunyun, Wang Xiangke y Zhu Hua-yong. «Multiple UAVs Configuration Formation Control via the Dual Quaternion Method». En: *2015 34th Chinese Control Conference (CCC)* (2015), págs. 7207-7211.
- [35] Ivan Maza et al. «Classification of Multi-UAV Architectures». En: *Handbook of Unmanned Aerial Vehicles*. Ed. por Kimon P. Valavanis y George J. Vachtsevanos. Dordrecht: Springer Netherlands, 2015, págs. 953-975. DOI: [10.1007/978-90-481-9707-1_119](https://doi.org/10.1007/978-90-481-9707-1_119). URL: https://doi.org/10.1007/978-90-481-9707-1_119.
- [36] Soon-Jo Chung et al. «A Survey on Aerial Swarm Robotics». En: *IEEE Transactions on Robotics* 34 (2018), págs. 837-855.

- [37] Georgy Skorobogatov, Cristina Barrado y Esther Salamí. «Multiple UAV Systems: A Survey». En: *Unmanned Systems* 08.02 (2020), págs. 149-169. DOI: [10.1142/S2301385020500090](https://doi.org/10.1142/S2301385020500090). eprint: <https://doi.org/10.1142/S2301385020500090>. URL: <https://doi.org/10.1142/S2301385020500090>.
- [38] J.P. Desai, J. Ostrowski y V. Kumar. «Controlling Formations of Multiple Mobile Robots». En: *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*. Vol. 4. 1998, 2864-2869 vol.4. DOI: [10.1109/ROBOT.1998.680621](https://doi.org/10.1109/ROBOT.1998.680621).
- [39] Herbert Tanner y Amit Mainwal. «Formation Stabilization of Multiple Agents Using Decentralized Navigation Functions». En: jun. de 2005, págs. 49-56. DOI: [10.15607/RSS.2005.I.007](https://doi.org/10.15607/RSS.2005.I.007).
- [40] Yun Choi y Doik Kim. «Distance-Based Formation Control With Goal Assignment for Global Asymptotic Stability of Multi-Robot Systems». En: *IEEE Robotics and Automation Letters* PP (feb. de 2021), págs. 1-1. DOI: [10.1109/LRA.2021.3061071](https://doi.org/10.1109/LRA.2021.3061071).
- [41] Maja J. Mataric. «Designing Emergent Behaviors: From Local Interactions to Collective Intelligence». En: *Proceedings of the Second International Conference on From Animals to Animats 2: Simulation of Adaptive Behavior: Simulation of Adaptive Behavior*. Honolulu, Hawaii, USA: MIT Press, 1993, págs. 432-441. ISBN: 0262631490.
- [42] Amanda J. C. Sharkey. «Robots, Insects and Swarm Intelligence». En: *Artificial Intelligence Review* 26 (2006), págs. 255-268.
- [43] J.A. Fax y R.M. Murray. «Information Flow and Cooperative Control of Vehicle Formations». En: *IEEE Transactions on Automatic Control* 49.9 (2004), págs. 1465-1476. DOI: [10.1109/TAC.2004.834433](https://doi.org/10.1109/TAC.2004.834433).

- [44] Shaoming He et al. «Distributed Estimation Over a Low-Cost Sensor Network: A Review of State-of-the-Art». En: *Inf. Fusion* 54 (2020), págs. 21-43.
- [45] Jiantao Shi. «Finite Horizon Cooperative Formation Control for Multi-agent Systems». En: *2020 39th Chinese Control Conference (CCC)*. 2020, págs. 4849-4852. DOI: [10.23919/CCC50068.2020.9189521](https://doi.org/10.23919/CCC50068.2020.9189521).
- [46] Wei Ren, Randal W. Beard y Ella M. Atkins. «Information Consensus in Multi-vehicle Cooperative Control». En: *IEEE Control Systems Magazine* 27.2 (2007), págs. 71-82. DOI: [10.1109/MCS.2007.338264](https://doi.org/10.1109/MCS.2007.338264).
- [47] Sonia Martínez, Jorge Cortés y Francesco Bullo. «Motion Coordination with Distributed Information». En: *IEEE Control Systems* 27 (2007), págs. 75-88.
- [48] G. Lafferriere et al. «Decentralized Control of Vehicle Formations». En: *Systems and Control Letters* 54.9 (2005), págs. 899-910. ISSN: 0167-6911. DOI: <https://doi.org/10.1016/j.sysconle.2005.02.004>. URL: <https://www.sciencedirect.com/science/article/pii/S0167691105000204>.
- [49] Samira Hayat, Evsen Yanmaz y Raheeb Muzaffar. «Survey on Unmanned Aerial Vehicle Networks for Civil Applications: A Communications Viewpoint». En: *IEEE Communications Surveys & Tutorials* 18 (2016), págs. 2624-2661.
- [50] Torsten Andre et al. «Application-Driven Design of Aerial Communication Networks». En: *IEEE Communications Magazine* 52.5 (2014), págs. 129-137. DOI: [10.1109/MCOM.2014.6815903](https://doi.org/10.1109/MCOM.2014.6815903).
- [51] José Alvarez-Hamelin, Aline Viana y Marcelo Dias de Amorim. «DHT-Based Functionalities Using Hypercubes». En: *Ad-Hoc Networking*. Vol. 212. Springer, ene. de 1970, págs. 157-176. ISBN: 978-0-387-34635-9. DOI: [10.1007/978-0-387-34738-7_12](https://doi.org/10.1007/978-0-387-34738-7_12).

- [52] John T. Shepard y Christopher A. Kitts. «A Multirobot Control Architecture for Collaborative Missions Comprised of Tightly Coupled, Interconnected Tasks». En: *IEEE Systems Journal* 12.2 (2018), págs. 1435-1446. DOI: [10.1109/JSYST.2016.2590430](https://doi.org/10.1109/JSYST.2016.2590430).
- [53] Wangyan Li et al. «A Survey on Multisensor Fusion and Consensus Filtering for Sensor Networks». En: *Discrete Dynamics in Nature and Society* 2015 (2015), págs. 1-12.
- [54] Robert E. Mahony, Vijay R. Kumar y Peter Corke. «Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor». En: *IEEE Robotics & Automation Magazine* 19 (2012), págs. 20-32.
- [55] Angelo Maria Sabatini. «Kalman-Filter-Based Orientation Determination Using Inertial/Magnetic Sensors: Observability Analysis and Performance Evaluation». En: *Sensors* 11.10 (2011), págs. 9182-9206. ISSN: 1424-8220. DOI: [10.3390/s111009182](https://doi.org/10.3390/s111009182). URL: <https://www.mdpi.com/1424-8220/11/10/9182>.
- [56] Dieter Fox et al. «A Probabilistic Approach to Collaborative Multi-Robot Localization». En: *Autonomous Robots* 8 (jun. de 2000), págs. 325-344. DOI: [10.1023/A:1008937911390](https://doi.org/10.1023/A:1008937911390).
- [57] Marco Cagnetti et al. «Cooperative Control of a Heterogeneous Multi-Robot System Based on Relative Localization». En: *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2014, págs. 350-356. DOI: [10.1109/IRoS.2014.6942583](https://doi.org/10.1109/IRoS.2014.6942583).
- [58] Salil Goel et al. «Cooperative Localization of Unmanned Aerial Vehicles Using GNSS, MEMS Inertial, and UWB Sensors». En: *Journal of Surveying Engineering-asce* 143 (2017), pág. 04017007.

- [59] Ian F. Akyildiz et al. «Wireless Sensor Networks: A Survey». En: *Comput. Networks* 38 (2002), págs. 393-422.
- [60] Guangjie Han et al. «Localization Algorithms of Wireless Sensor Networks: A Survey». En: *Telecommun. Syst.* 52.4 (abr. de 2013), págs. 2419-2436. ISSN: 1018-4864. DOI: [10.1007/s11235-011-9564-7](https://doi.org/10.1007/s11235-011-9564-7). URL: <https://doi.org/10.1007/s11235-011-9564-7>.
- [61] Steven Lanzisera, David T. Lin y Kristofer S. J. Pister. «RF Time of Flight Ranging for Wireless Sensor Network Localization». En: *2006 International Workshop on Intelligent Solutions in Embedded Systems*. 2006, págs. 1-12. DOI: [10.1109/WISES.2006.329127](https://doi.org/10.1109/WISES.2006.329127).
- [62] Guoqiang Mao, Baris Fidan y Brian D.O. Anderson. «Wireless Sensor Network Localization Techniques». En: *Computer Networks* 51.10 (2007), págs. 2529-2553. ISSN: 1389-1286. DOI: <https://doi.org/10.1016/j.comnet.2006.11.018>. URL: <https://www.sciencedirect.com/science/article/pii/S1389128606003227>.
- [63] Guangwei Zhu y Jianghai Hu. «A Distributed Continuous-Time Algorithm for Network Localization Using Angle-of-Arrival Information». En: *Automatica* 50.1 (ene. de 2014), págs. 53-63. ISSN: 0005-1098. DOI: [10.1016/j.automatica.2013.09.033](https://doi.org/10.1016/j.automatica.2013.09.033). URL: <https://doi.org/10.1016/j.automatica.2013.09.033>.
- [64] Youngchul Bae. «Robust Localization for Robot and IoT Using RSSI». En: *Energies* 12.11 (2019). ISSN: 1996-1073. DOI: [10.3390/en12112212](https://doi.org/10.3390/en12112212). URL: <https://www.mdpi.com/1996-1073/12/11/2212>.
- [65] Hao Xu et al. «Decentralized Visual-Inertial-UWB Fusion for Relative State Estimation of Aerial Swarm». En: *2020 IEEE International Conference on Robotics*

- and Automation (ICRA)*. 2020, págs. 8776-8782. DOI: [10.1109/ICRA40945.2020.9196944](https://doi.org/10.1109/ICRA40945.2020.9196944).
- [66] Thomas Ziegler et al. «Distributed Formation Estimation Via Pairwise Distance Measurements». En: *IEEE Robotics and Automation Letters* 6.2 (2021), págs. 3017-3024. DOI: [10.1109/LRA.2021.3062347](https://doi.org/10.1109/LRA.2021.3062347).
- [67] Fabrizio Schiano et al. «A Rigidity-Based Decentralized Bearing Formation Controller for Groups of Quadrotor UAVs». En: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2016, págs. 5099-5106. DOI: [10.1109/IROS.2016.7759748](https://doi.org/10.1109/IROS.2016.7759748).
- [68] Shiyu Zhao y Daniel Zelazo. «Bearing Rigidity Theory and Its Applications for Control and Estimation of Network Systems: Life Beyond Distance Rigidity». En: *IEEE Control Systems Magazine* 39.2 (2019), págs. 66-83. DOI: [10.1109/MCS.2018.2888681](https://doi.org/10.1109/MCS.2018.2888681).
- [69] Juan F. Presenza et al. «A New Upper Bound for the d-dimensional Algebraic Connectivity of Arbitrary Graphs». En: *arXiv preprint arXiv:2209.14893* (2022). arXiv: [2209.14893 \[math.CO\]](https://arxiv.org/abs/2209.14893).
- [70] Zhiyong Sun, Changbin Yu y Brian D. O. Anderson. «Distributed Optimization on Proximity Network Rigidity via Robotic Movements». En: *2015 34th Chinese Control Conference (CCC)*. 2015, págs. 6954-6960. DOI: [10.1109/ChiCC.2015.7260739](https://doi.org/10.1109/ChiCC.2015.7260739).
- [71] Daniel Zelazo et al. «Decentralized Rigidity Maintenance Control with Range Measurements for Multi-Robot Systems». En: *The International Journal of Robotics Research* 34.1 (2015), págs. 105-128. DOI: [10.1177/0278364914546173](https://doi.org/10.1177/0278364914546173).

- [72] Juan F. Presenza et al. «Subframework-Based Rigidity Control in Multirobot Networks». En: *2022 American Control Conference (ACC)*. 2022, págs. 3431-3436. DOI: [10.23919/ACC53348.2022.9867693](https://doi.org/10.23919/ACC53348.2022.9867693).
- [73] Juan F. Presenza et al. «Control de Rigidez por Subframeworks en Sistemas Multirobot». En: *XI Jornadas Argentinas de Robótica, Bariloche, Argentina*. 2022.
- [74] Patricio Moreno et al. «Aerial Multi-Camera Robotic Jib Crane». En: *IEEE Robotics and Automation Letters* 6.2 (2021), págs. 4103-4108. DOI: [10.1109/LRA.2021.3065299](https://doi.org/10.1109/LRA.2021.3065299).
- [75] Stéphane Petti y Thierry Fraichard. «Safe Motion Planning in Dynamic Environments». En: *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (2005)*, págs. 2210-2215.
- [76] Alireza Ahmadzadeh et al. «Multi-Vehicle Path Planning in Dynamically Changing Environments». En: *2009 IEEE International Conference on Robotics and Automation (2009)*, págs. 2449-2454.
- [77] Helen Oleynikova et al. «Voxblox: Incremental 3D Euclidean Signed Distance Fields for On-Board MAV Planning». En: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2017, págs. 1366-1373.
- [78] Bahare Sabetghadam et al. «Optimal Trajectory Planning for Autonomous Drone Cinematography». En: *2019 European Conference on Mobile Robots (ECMR)* (2019), págs. 1-7.
- [79] Demetri P Spanos, Reza Olfati-Saber y Richard M Murray. «Dynamic Consensus on Mobile Networks». En: *IFAC world congress*. Citeseer. 2005, págs. 1-6.
- [80] R. Olfati-Saber, J. A. Fax y R. M. Murray. «Consensus and Cooperation in Networked Multi-Agent Systems». En: *Proceedings of the IEEE* 95.1 (2007), págs. 215-233. DOI: [10.1109/JPROC.2006.887293](https://doi.org/10.1109/JPROC.2006.887293).

- [81] Julien M. Hendrickx et al. «Graph Diameter, Eigenvalues, and Minimum-Time Consensus». En: *Automatica* 50.2 (2014), págs. 635-640. ISSN: 0005-1098.
- [82] Mirka Miller y Jozef Sirán. «Moore Graphs and Beyond: A survey of the Degree/Diameter Problem». En: *Electronic Journal of Combinatorics* 1000 (2013).
- [83] Miroslav Fiedler. «Algebraic Connectivity of Graphs». En: *Czechoslovak Mathematical Journal* 23.2 (1973), págs. 298-305. URL: <http://eudml.org/doc/12723>.
- [84] Bojan Mohar. «The Laplacian Spectrum of Graphs». En: *Graph Theory, Combinatorics, and Applications*. Wiley, 1991, págs. 871-898.
- [85] Michael M. Zavlanos, Magnus B. Egerstedt y George J. Pappas. «Graph-Theoretic Connectivity Control of Mobile Robot Networks». En: *Proceedings of the IEEE* 99.9 (2011), págs. 1525-1540. DOI: [10.1109/JPROC.2011.2157884](https://doi.org/10.1109/JPROC.2011.2157884).
- [86] Chris Godsil y Gordon Royle. *Algebraic Graph Theory*. Vol. 207. Springer New York, NY, ene. de 2001. ISBN: 978-0-387-95220-8. DOI: [10.1007/978-1-4613-0163-9](https://doi.org/10.1007/978-1-4613-0163-9).
- [87] Reinhard Diestel. *Graph Theory*. 5th. Springer Publishing Company, Incorporated, 2017. ISBN: 3662536218.
- [88] Fan R. K. Chung. *Spectral Graph Theory*. Providence, RI: American Mathematical Society, 1997. ISBN: 0821803158.
- [89] Andries Brouwer y Willem Haemers. *Spectra of Graphs*. Springer New York, NY, ene. de 2012. ISBN: 9781461419389. DOI: [10.1007/978-1-4614-1939-6](https://doi.org/10.1007/978-1-4614-1939-6).
- [90] L. Asimow y B. Roth. «The Rigidity of Graphs». En: *Transactions of the American Mathematical Society* 245 (1978), págs. 279-289. ISSN: 00029947. URL: <http://www.jstor.org/stable/1998867> (visitado 23-02-2023).

- [91] Leonard Asimow y B. Roth. «The Rigidity of Graphs, II». En: *Journal of Mathematical Analysis and Applications* 68 (1979), págs. 171-190.
- [92] B. Roth. «Rigid and Flexible Frameworks». En: *The American Mathematical Monthly* 88.1 (1981), págs. 6-21. ISSN: 00029890, 19300972. URL: <http://www.jstor.org/stable/2320705> (visitado 12-09-2022).
- [93] R. Connelly. «Generic Global Rigidity». En: *Discrete & Computational Geometry* 33 (abr. de 2005), págs. 549-563. DOI: [10.1007/s00454-004-1124-4](https://doi.org/10.1007/s00454-004-1124-4).
- [94] Tibor Jordán y Shin-ichi Tanigawa. «Rigidity of Random Subgraphs and Eigenvalues of Stiffness Matrices». En: *SIAM Journal on Discrete Mathematics* 36 (sep. de 2022), págs. 2367-2392. DOI: [10.1137/20M1349849](https://doi.org/10.1137/20M1349849).
- [95] Ileana Streinu y Louis Theran. «Combinatorial Genericity and Minimal Rigidity». En: *SCG '08*. 2008.
- [96] Guangwei Zhu y Jianghai Hu. «Stiffness Matrix and Quantitative Measure of Formation Rigidity». En: *Proceedings of the 48th IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*. 2009, págs. 3057-3062. DOI: [10.1109/CDC.2009.5400387](https://doi.org/10.1109/CDC.2009.5400387).
- [97] Daniel Zelazo et al. «Rigidity Maintenance Control for Multi-Robot Systems». En: *Robotics: Science and Systems VIII*. Jul. de 2012, págs. 473-480. DOI: [10.15607/RSS.2012.VIII.060](https://doi.org/10.15607/RSS.2012.VIII.060).
- [98] Gordon D. Laman. «On Graphs and Rigidity of Plane Skeletal Structures». En: *Journal of Engineering Mathematics* 4 (1970), págs. 331-340.
- [99] Ignacio Mas y Christopher A. Kitts. «Object Manipulation Using Cooperative Mobile Multi-Robot Systems». En: *Lecture Notes in Engineering and Computer Science: Proc. of The World Congress on Engineering and Computer Science* (oct. de 2012), págs. 324-329. ISSN: 2078-0958.

- [100] Paul Mahacek, Christopher A. Kitts e Ignacio Mas. «Dynamic Guarding of Marine Assets Through Cluster Control of Automated Surface Vessel Fleets». En: *IEEE/ASME Transactions on Mechatronics* 17 (2012), págs. 65-75.
- [101] Ignacio Mas y Christopher Kitts. «Cooperative Tasks Using Teams of Mobile Robots». En: *IAENG Transactions on Engineering Technologies*. Ed. por Haeng Kon Kim et al. Vol. 247. Lecture Notes in Electrical Engineering. Springer Netherlands, 2014, págs. 83-99. ISBN: 978-94-007-6817-8. DOI: [10.1007/978-94-007-6818-5_7](https://doi.org/10.1007/978-94-007-6818-5_7).
- [102] Thomas Adamek, Christopher Kitts e Ignacio Mas. «Gradient-Based Cluster Space Navigation for Autonomous Surface Vessels». En: *IEEE/ASME Transactions on Mechatronics* 20.2 (abr. de 2015), págs. 506-518. ISSN: 1083-4435. DOI: [10.1109/TMECH.2013.2297152](https://doi.org/10.1109/TMECH.2013.2297152).
- [103] Guangwei Zhu. «Quantitative Analysis of Multi-agent Formations: Theory and Applications». Tesis doct. Purdue University, 2013.
- [104] Alan Lew et al. «On the d -Dimensional Algebraic Connectivity of Graphs». En: *arXiv preprint arXiv:2205.05530* (2022). DOI: [10.48550/ARXIV.2205.05530](https://doi.org/10.48550/ARXIV.2205.05530). URL: <https://arxiv.org/abs/2205.05530>.
- [105] Alan Lew et al. «Rigidity Expander Graphs». En: *arXiv preprint arXiv:2304.01306* (2023). arXiv: [2304.01306](https://arxiv.org/abs/2304.01306) [math.CO].
- [106] Nair Maria Maia de Abreu. «Old and New Results on Algebraic Connectivity of Graphs». En: *Linear Algebra and its Applications* 423.1 (2007). Special Issue devoted to papers presented at the Aveiro Workshop on Graph Spectra, págs. 53-73. ISSN: 0024-3795. DOI: <https://doi.org/10.1016/j.laa.2006.08.017>. URL: <https://www.sciencedirect.com/science/article/pii/S0024379506003971>.

- [107] Tibor Jordán. «Extremal Problems and Results in Combinatorial Rigidity». En: *10th Japanese-Hungarian Symposium on Discrete Mathematics and its Applications*. Mayo de 2017, págs. 2367-2392.
- [108] Ali Moradi Amani et al. «Distributed Rigidity Recovery in Distance-Based Formations Using Configuration Lattice». En: *IEEE Transactions on Control of Network Systems* 7.3 (2020), págs. 1547-1558. DOI: [10.1109/TCNS.2020.2984683](https://doi.org/10.1109/TCNS.2020.2984683).
- [109] Tolga Eren et al. «Merging Globally Rigid Formations of Mobile Autonomous Agents». En: *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, 2004. AAMAS 2004*. (2004), págs. 1260-1261.
- [110] Laura Krick, Mireille E. Broucke y Bruce A. Francis. «Stabilisation of Infinitesimally Rigid Formations of Multi-Robot Networks». En: *International Journal of Control* 82.3 (2009), págs. 423-439.
- [111] J. Aspnes et al. «A Theory of Network Localization». En: *IEEE Transactions on Mobile Computing* 5.12 (dic. de 2006), págs. 1663-1678.
- [112] Zheng Yang y Yunhao Liu. «Understanding Node Localizability of Wireless Ad Hoc and Sensor Networks». En: *IEEE Transactions on Mobile Computing* 11.8 (ago. de 2012), págs. 1249-1260. ISSN: 1536-1233. DOI: [10.1109/TMC.2011.122](https://doi.org/10.1109/TMC.2011.122). URL: <https://doi.org/10.1109/TMC.2011.122>.
- [113] Pavel Pascacio et al. «Collaborative Indoor Positioning Systems: A Systematic Review». En: *Sensors (Basel, Switzerland)* 21 (2021).
- [114] Huthaifa Obeidat et al. «A Review of Indoor Localization Techniques and Wireless Technologies». En: *Wireless Personal Communications* 119 (jul. de 2021). DOI: [10.1007/s11277-021-08209-5](https://doi.org/10.1007/s11277-021-08209-5).

- [115] Giuseppe C. Calafiore, Luca Carlone y Mingzhu Wei. «A Distributed Gradient Method for Localization of Formations Using Relative Range Measurements». En: *2010 IEEE International Symposium on Computer-Aided Control System Design*. 2010, págs. 1146-1151. DOI: [10.1109/CACSD.2010.5612764](https://doi.org/10.1109/CACSD.2010.5612764).
- [116] Ryan K. Williams et al. «Distributed Combinatorial Rigidity Control in Multi-Agent Networks». En: *52nd IEEE Conference on Decision and Control*. 2013, págs. 6061-6066. DOI: [10.1109/CDC.2013.6760847](https://doi.org/10.1109/CDC.2013.6760847).
- [117] SAXE J. B. «Embeddability of Weighted Graphs in k-Space is Strongly NP-Hard». En: *Proc. of 17th Allerton Conference in Communications, Control and Computing, Monticello, IL* (1979), págs. 480-489. URL: <https://cir.nii.ac.jp/crid/1570854174860444416>.
- [118] S. Bouabdallah y R. Siegwart. «Backstepping and Sliding-mode Techniques Applied to an Indoor Micro Quadrotor». En: *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. 2005, págs. 2247-2252. DOI: [10.1109/ROBOT.2005.1570447](https://doi.org/10.1109/ROBOT.2005.1570447).
- [119] Minh-Duc Hua et al. «A Control Approach for Thrust-Propelled Underactuated Vehicles and its Application to VTOL Drones». En: *IEEE Transactions on Automatic Control* 54 (2009), págs. 1837-1853.
- [120] Kwang-Kyo Oh, Myoung-Chul Park y Hyo-Sung Ahn. «A Survey of Multi-Agent Formation Control». En: *Automatica* 53 (2015), págs. 424-440. ISSN: 0005-1098. DOI: [10.1016/j.automatica.2014.10.022](https://doi.org/10.1016/j.automatica.2014.10.022).
- [121] Christophe Lino y Marc Christie. «Intuitive and Efficient Camera Control with the Toric Space». En: *ACM Trans. Graphics* 34.4 (2015), págs. 1-12. ISSN: 0730-0301, 1557-7368. DOI: [10.1145/2766965](https://doi.org/10.1145/2766965).

- [122] Quentin Galvane et al. «Automated Cinematography with Unmanned Aerial Vehicles». En: *Workshop on Intell. Cinematography and Editing*. Eurographics Association, 2016. ISBN: 978-3-03868-005-5. DOI: [10.2312/wiced.20161097](https://doi.org/10.2312/wiced.20161097).
- [123] Niels Joubert et al. «Towards a Drone Cinematographer: Guiding Quadrotor Cameras using Visual Composition Principles». En: *arXiv:1610.01691 [cs]* (oct. de 2016). arXiv:1610.01691.
- [124] Rogerio Bonatti et al. «Autonomous Drone Cinematographer: Using Artistic Principles to Create Smooth, Safe, Occlusion-Free Trajectories for Aerial Filming». En: *arXiv:1808.09563 [cs]* (ago. de 2018). arXiv: 1808.09563.
- [125] Chong Huang et al. «ACT: An Autonomous Drone Cinematography System for Action Scenes». En: *IEEE Int. Conf. Robotics and Automation*. Mayo de 2018, págs. 7039-7046. ISBN: 978-1-5386-3081-5. DOI: [10.1109/ICRA.2018.8460703](https://doi.org/10.1109/ICRA.2018.8460703).
- [126] Ioannis Mademlis et al. «Autonomous UAV Cinematography: A Tutorial and a Formalized Shot-Type Taxonomy». En: *ACM Comput. Surveys* 525 (sep. de 2019), 1-33. ISSN: 0360-0300. DOI: [10.1145/3347713](https://doi.org/10.1145/3347713).
- [127] Marta Marques et al. «Trajectory Planning and Control for Drone Replacement for Multidrone Cinematography». En: *IFAC-PapersOnLine* 52.12 (2019). 21st IFAC Symposium on Automatic Control in Aerospace ACA 2019, págs. 334-339. ISSN: 2405-8963. DOI: [10.1016/j.ifacol.2019.11.265](https://doi.org/10.1016/j.ifacol.2019.11.265).
- [128] A. Torres-González et al. «A Multidrone Approach for Autonomous Cinematography Planning». En: *Advances in Intelligent Systems and Computing*. Ene. de 2018, págs. 337-349. ISBN: 978-3-319-70832-4. DOI: [10.1007/978-3-319-70833-1_28](https://doi.org/10.1007/978-3-319-70833-1_28).

- [129] Alexander Tiderko, Frank Hoeller y Timo Rohling. «The ROS Multimaster Extension for Simplified Deployment of Multi-Robot Systems». En: *Robot Operating System (ROS): The Complete Reference (Volume 1)*. Ed. por Anis Koubaa. Cham: Springer International Publishing, 2016, págs. 629-650. ISBN: 978-3-319-26054-9. DOI: [10.1007/978-3-319-26054-9_24](https://doi.org/10.1007/978-3-319-26054-9_24). URL: https://doi.org/10.1007/978-3-319-26054-9_24.
- [130] Harish Anand, Zhiang Chen y Jnaneshwar Das. «The OpenUAV Swarm Simulation Testbed: a Collaborative Design Studio for Field Robotics». En: *ArXiv abs/1910.00739* (2019).
- [131] Andrea Testa, Andrea Camisa y Giuseppe Notarstefano. «ChoiRbot: A ROS 2 Toolbox for Cooperative Robotics». En: *IEEE Robotics and Automation Letters* 6.2 (2021), págs. 2714-2720. DOI: [10.1109/LRA.2021.3061366](https://doi.org/10.1109/LRA.2021.3061366).
- [132] J. Vander Hook, P. Tokekar y V. Isler. «Algorithms for Cooperative Active Localization of Static Targets With Mobile Bearing Sensors Under Communication Constraints». En: *IEEE Transactions on Robotics* 31.4 (2015), págs. 864-876.