



**HAL**  
open science

# Bearing-based localization and control for multiple quadrotor UAVs

Fabrizio Schiano

► **To cite this version:**

Fabrizio Schiano. Bearing-based localization and control for multiple quadrotor UAVs. Robotics [cs.RO]. Université de Rennes, 2018. English. NNT : 2018REN1S009 . tel-01780065

**HAL Id: tel-01780065**

**<https://theses.hal.science/tel-01780065>**

Submitted on 27 Apr 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



**THÈSE / UNIVERSITÉ DE RENNES 1**

*sous le sceau de l'Université Européenne Bretagne Loire*

pour le grade de

**DOCTEUR DE L'UNIVERSITÉ DE RENNES 1**

*Mention : Automatique, Productique et Robotique*

**École doctorale MathSTIC**

présentée par

**Fabrizio Schiano**

préparée à l'unité de recherche IRISA – UMR6074  
Institut de Recherche en Informatique et Systeme Aléatoires  
et au centre INRIA - Rennes Bretagne Atlantique

---

# **Bearing-based Localization and Control for Multiple Quadrotor UAVs**

**Thèse soutenue à Rennes  
le 11/01/2018**

devant le jury composé de :

**Alcherio Martinoli** / *Rapporteur*

Associate Professor, Ecole Polytechnique Fédérale de  
Lausanne (EPFL), Switzerland

**Nicolas Marchand** / *Rapporteur*

Directeur de Recherche CNRS - HDR, GIPSA-lab Gre-  
noble, Control Systems Department, France

**Bruno Siciliano** / *Examineur*

Full professor, Università degli studi di Napoli Federico II,  
Italy

**Sophie Pinchinat** / *Examineur*

Full professor, ISTIC (Department of Electrical Enginee-  
ring and Computer Science) University of Rennes 1,  
France

**Pascal Morin** / *Examineur*

Full Professor, Institut des Systèmes Intelligents et de Ro-  
botique, Paris, France

**Paolo Robuffo Giordano** / *Directeur de thèse*

Directeur de Recherche CNRS, IRISA-INRIA, Rennes,  
France



*To my family and to Ida*

*Always improve yourself*

---

## Résumé en français

Dans la littérature, l'une des premières apparitions des automates fut recensée dans un texte Daoïste, intitulé Lie Zi (ou Liezi), dans lequel était décrite une rencontre entre le Roi Mu de Zhou (1023-957 B.C.) et Yan Shi, un ingénieur en mécanique [1]. On dit que l'ingénieur montra au roi sa construction mécanique, d'aspect et de taille comparables à celles d'un humain. Cet exemple témoigne de l'intérêt immémorial de l'Homme pour les créations mimant le comportement humain, primant même sur son désir de reproduire le comportement des animaux ou des insectes. Selon moi, l'exemple le plus fascinant est celui des animaux volants et des insectes. Depuis ses débuts, l'Homo Sapiens ne cesse d'observer des *corps volants* et de s'imaginer voler. Il lui fallut beaucoup de temps pour réaliser ce rêve, mais, désormais, nous vivons dans un monde où se déplacer en avion est devenu monnaie courante et où nous ne sommes qu'à un pas de faire voler nos voitures. Dans cette optique, il semble que la robotique soit entrée dans une nouvelle ère, dans laquelle les robots sont pleinement intégrés à nos sociétés. Le lecteur de ce manuscrit pourra d'ailleurs faire le parallèle avec la révolution numérique du 21<sup>ème</sup> siècle. Cela fait d'ailleurs plusieurs décennies que les robots sont utilisés dans des applications civiles et militaires. Aujourd'hui, ils sont principalement employés par l'industrie, l'armée et l'agriculture, mais ils sont aussi présents en médecine, sous la forme de nanorobots, dans nos foyers etc. La liste ne cesse de s'allonger, et, malgré cela, chaque année, de nouvelles manières d'utiliser les robots, insoupçonnées encore il y a vingt ans, émergent. C'est le cas, par exemple, des robots cuisiniers<sup>1</sup>, serveurs, ou bien de ceux qui seront utilisés pour administrer des médicaments plus efficacement à des patients [2, 3].

Imiter le comportement d'un seul animal ou insecte est encore le centre d'attention de grands sujets de recherche [4–6]. Cependant, depuis des siècles, les chercheurs se sont aussi intéressés au comportement collectif des animaux, des insectes et des bactéries [7]. Pour cette raison, l'un des objectifs de la communauté robotique

---

<sup>1</sup><http://www.moley.com/>

est de produire des robots capables d’imiter des comportements présents dans la nature, tels que l’essaimage (en anglais *swarming*), le rassemblement (*flocking*), l’attroupelement (*herding*), et le rassemblement en bancs (*shoaling/schooling*) [8–15]. Même en dehors de la robotique, de nombreux chercheurs étudient ces comportements pour pouvoir les répliquer. Un aspect fascinant de ces derniers est que les agents de tels groupes semblent souvent agir en suivant seulement leur propre intention, alors qu’une ruche d’abeilles, une colonie de fourmis ou une migration d’oiseaux apparaissent particulièrement organisées, comme si leurs membres agissaient dans un but commun. Les chercheurs comprirent que dans ces situations, les agents suivent quelques règles simples, le plus souvent sans avoir besoin d’un chef. En suivant celles-ci, le comportement du groupe semble alors harmonieux. Dans la Fig. 1.1 sont présentés deux exemples de coopération chez les fourmis de feu qui surgissent grâce aux principes mentionnés. Dans ces situations, et dans bien d’autres également, les



Figure 1 – Exemples de comportements collectifs exposés par des *fourmis de feu*. Fig. 1(a): un tas de 500 *fourmis de feu*, composées d’une couche partiellement mouillée de fourmis sur le fond et les fourmis sèches sur sommet, [16]. Fig. 1.1(b): flottabilité et élasticité du radeau de fourmis, comme indiqué par immersion essayée par une brindille, [16].

agents (animaux ou insectes) ne disposent que des informations qu’ils ressentent, et ne peuvent agir que localement (c’est-à-dire par rapport à eux-mêmes). Il n’existe pas d’information partagée par tous les agents, et aucune forme d’encadrement.

Durant la dernière décennie, un intérêt croissant fut porté aux applications multi-robots. En particulier, la recherche sur la coordination multi-véhicules débuta vers la fin des années 1980, dans le domaine de la robotique mobile (se référer à [17, 18] pour un état de l’art dans ce domaine, valable jusqu’à 2006). La plupart des scénarios multi-robots sont similaires aux comportements de groupe adoptés par certains animaux et insectes, abordés précédemment. D’ailleurs, l’un des éléments communs à toute formation organisée est l’objectif principal (par exemple, la construction d’une ruche, le transport d’un morceau volumineux de nourriture d’un endroit à un autre, la migration de tout un groupe d’oiseaux etc) à atteindre de manière décentralisée (chaque agent ne connaît qu’en partie l’état de ses voisins). Pour atteindre cet

objectif, un élément que l'on retrouve dans chaque formation organisée est la faculté de chaque agent à interagir seulement avec une sous-partie du groupe. Ce sous-partie du groupe est d'habitude appelé *les voisins* de l'agent considéré. Certains agents sont même capables de se repérer par rapport aux autres et de prendre des décisions, en se basant uniquement sur des informations locales.

En résumé, il arrive souvent que des agents de la formation doivent effectuer des actions nécessitant des mesures relatives (relatives aux autres agents de la formation, et non à un référentiel central au groupe). Parmi ces mesures relatives on trouve les distances relatives, les *bearings*<sup>2</sup> relatives, les positions relatives et, bien sûr, des combinaisons des trois. En robotique, ce type de mesure peut être obtenu à partir de capteurs embarqués, comme, par exemple, des capteurs ultrasonores, des scanners lasers, des caméras ou des émetteurs/récepteurs radiofréquences. Leur utilisation, pour extraire des mesures relatives, permet à la structure de contrôle de s'affranchir de la présence de systèmes de localisation centralisés, comme le dispositif de capture de mouvement proposé par Vicon, ou le GNSS et l'utilisation de, par exemple, algorithmes SLAM [19, 20]. Ces concepts nous amènent à un des motifs principaux du développement des systèmes de multi-agent qui est la possibilité de *décentralisation*.

Les solutions décentralisées jouent un rôle significatif dans des applications de multi-agent puisqu'elles permettent d'utiliser des algorithmes indépendamment de la taille de groupe (*scalable algorithms*). Cependant, même si l'importance et les avantages de décentralisation sont évidentes à la communauté de multi-robot, il est rare de trouver des solutions purement décentralisées pour des systèmes de multi-robot. En effet, c'est le cas de toutes ces applications qui comptent lourdement sur des systèmes de localisation centralisés comme le système de capture de mouvement de Vicon ( Fig. 1.2(a) ) ou le GNSS ( Fig. 1.2(b) ). Ces applications sont très intéressantes, mais nécessitent des ajustements substantiels afin d'être déployées dans un scénario réaliste, pour lequel les systèmes centralisés ne sont pas disponibles (par exemple, dans un bâtiment détruit par un séisme, sous l'eau, sous terre, dans l'espace, ou dans un endroit où le signal GPS est faible).

En robotique, les robots mobiles terrestres sont encore largement employés dans les applications multi-robots [19, 21], (voir Fig. 1.2(c) ). Cela est principalement dû à leur coût relativement faible, leur facilité d'utilisation, mais aussi leur sécurité (d'habitude) intrinsèque. Pour toutes ces raisons, à leurs débuts, les applications multi-robots permirent de mettre en avant les atouts des systèmes multi-agents, ainsi que les verrous technologiques inhérents à leur élaboration. Le principal inconvénient

---

<sup>2</sup>Dans cette Thèse le terme anglais *bearing* se réfère à une mesure non métrique, obtenue en temps-réel via l'utilisation d'une caméra montée sur le drone et visualisant les autres drones de la formation. Cette mesure est alors représentée par le vecteur unitaire pointant vers le drone visualisé.

d'un robot terrestre est que son champ d'action est, le plus souvent, limité à deux dimensions. Ceci, combiné à l'intérêt croissant des chercheurs pour les drones, a suscité un fort intérêt pour l'application des concepts des formations multi-robots terrestres à leurs homologues aériens et, en particulier, ceux capables de décoller et d'atterrir verticalement (en anglais *VTOL UAVs*) [22].

Dans cette thèse, nous nous intéressons plus particulièrement aux *quadrotors*, des drones de taille réduite, et à l'utilisation d'informations visuelles pour les coordonner. Cette technologie a d'ailleurs été appliquée avec succès à la fouille de sites dévastés par des catastrophes naturelles, telles que les séismes de Chirstchurch en Nouvelle-Zélande (2011), Emilia-Romagna (2012) et Amatrice (2016), ou bien après le désastre nucléaire de Fukushima Daiichi, au Japon (2011). De plus, les drones sont fréquemment utilisés pour filmer des événements sportifs comme les Jeux Olympiques d'hiver de Sotchi, en Russie, ou la coupe d'Europe 2016, ayant eu lieu en France. Un autre exemple est celui du spectacle de Lady Gaga, organisé pendant la mi-temps du Super Bowl (2017), et pour lequel 300 drones *Shooting Star* de chez Intel furent déployés. Il s'agissait de l'une des premières fois que les drones étaient utilisés pour un événement télévisuel. Selon l'institut de recherche Gartner<sup>3</sup>, l'industrie des drones personnels ou commerciaux devrait générer des revenus de l'ordre de 5 milliards d'euros en 2017 et plus de 9,4 milliards d'euros en 2020 [23].

Malgré cette croissance exponentielle du domaine de la robotique, l'utilisation d'un dispositif entièrement autonome dans un environnement inconnu et déstructuré constitue toujours, aujourd'hui, un sujet actif de recherche. Il n'est donc pas surprenant que trois ambitieux projets aient été financés à hauteur de 22,2 millions d'euros ces six dernières années pour approfondir la thématique de la coordination multi-robots [24–26]. De la même manière, les compétitions de robotique dédiées aux applications multi-robots sont de plus en plus fréquentes et attirent toujours plus d'investissements<sup>4</sup>.

## Contributions de la Thèse

L'objectif de la thèse est de proposer des innovations pour résoudre les problèmes mentionnés auparavant (en particulier, le contrôle de la formation et la localisation par coopération), dans le cas d'un groupe de drones de type *quadrotor*, équipés de caméras monoculaires. Un intérêt est porté sur la mise en place d'un contrôle de la formation entièrement décentralisé et de techniques d'estimation reposant sur (i) des mesures relatives, obtenues à partir de caméras embarquées, et (ii) une communication locale utilisant le standard Wi-Fi. Pour que les résultats obtenus

---

<sup>3</sup>[www.gartner.com](http://www.gartner.com)

<sup>4</sup>[www.mbzirc.com](http://www.mbzirc.com), [www.robocup.org](http://www.robocup.org), [www.eurathlon.eu](http://www.eurathlon.eu)



(a)



(b)



(c)



(d)

FIGURE 2 – 1.2(a) : Transport coopératif d’un objet rigide avec quatre UAVs. Copyright d’Université de la Pennsylvanie (USA). 1.2(b) : un drapeau américain fait de 300 drones *Shooting Star* par Intel au spectacle de mi-temps du Super Bowl (2017). Copyright d’Intel. 1.2(c) : un groupe de robots terrestres naviguants et déplacent des objets autour d’un entrepôt. Copyright de systèmes Kiva. 1.2(d) : structure de filament extensible construite par des robots. Copyright de l’*Institute for Computational Design and Construction* à l’Université de Stuttgart.

soient les plus réalistes possibles, certaines limitations classiques des capteurs sont prises en considération. Une approche alternative permettant de résoudre le problème de localisation dans un environnement inconnu, à l’aide de capteurs embarqués, est également envisageable avec des techniques de mapping/SLAM [19]. Cependant, cela nécessiterait soit une carte détaillée de l’environnement, soit la possibilité d’exécuter des algorithmes complexes de type SLAM en temps réel, directement sur les robots. Par ailleurs, nous pensons que la résolution du problème de localisation par coopération, uniquement à partir *d’informations locales* (capteurs et communication) aux drones, offrirait au système plus de souplesse, lui permettrait d’évoluer dans des environnements inconnus et n’impliquerait pas qu’il faille embarquer de nombreux capteurs ni une grande puissance de calcul sur chaque drone.

Pour contrôler une formation de drones, il est important d’avoir une quantité *métrique* de quel récupérer les distances entre les robots. Cependant, cette information ne peut être obtenue uniquement à partir d’informations visuelles (*non-métriques*) d’entrée. Ainsi, dans ce manuscrit, nous présentons également une méthode permettant d’extraire en temps réel l’échelle de la formation (et donc les distances entre les robots) à l’aide de mesures de *bearing* et en supposant que les robots ont connais-

sance des vitesses linéaires et angulaires exprimées dans leur repère (*body-frame*). La dernière hypothèse est, selon nous, assez faible puisque chaque drone doit avoir connaissance de ces grandeurs, afin de contrôler son vol.

Les résultats théoriques présentés dans ce manuscrit ont été validés de manière approfondie en simulation et durant des expérimentations. Ces dernières ont été réalisées avec un groupe de drones de type *quadrotors* (Annexe B).

## Structure de la thèse

Le présent manuscrit est divisé en trois grandes parties. La première (I) contient une courte introduction sur la robotique qui détaille plus particulièrement les applications multi-robots, en insistant sur celles ayant recours à des drones. Une synthèse de ces thématiques est proposée, et plusieurs états de l'art issus de la littérature sont cités pour appréhender le contexte dans lequel s'inscrivent les travaux. La deuxième partie (II), quant à elle, constitue le cœur de la thèse. Elle en présente les contributions principales. Les résultats étayés dans cette seconde partie ont fait l'objet des publications suivantes : [27–29]. La troisième (III), et dernière, partie conclut le manuscrit, expose les futurs travaux à mener et s'achève sur deux annexes, permettant au lecteur d'approfondir certains aspects techniques. Un sommaire est détaillé ci-dessous.

### Aperçu de la Partie I

La première partie contient un état de l'art exhaustif sur le contrôle et la localisation des formations multi-robots, et introduit des fondamentaux sur la *théorie algébrique des graphes* et de la rigidité.

**Le Chapitre 2** donne un aperçu des concepts portant sur les formations multi-robots, et explicite ceux appliqués aux drones. Il inclut notamment une brève description des problèmes majeurs inhérents au contrôle des formations multi-agents, en se focalisant sur les problèmes de *consensus* et de contrôle de la formation. Dans sa conclusion, ce chapitre décrit le problème de localisation par coopération à partir de mesures relatives.

**Le Chapitre 3** propose une introduction sur la théorie algébrique des graphes. Celle-ci est fondamentale pour, ensuite, exposer les principes de la théorie de la rigidité. La dernière partie de ce chapitre approfondit la théorie de la rigidité, appliquée aux cas de la distance et des contraintes de *bearing*. Cela permettra d'aboutir, par la suite, au concept clé de cette thèse, à savoir la *matrice de rigidité bearing* (BRM).

**Le Chapitre 4** décrit les équations modélisant un seul drone et fournit quelques détails sur la rigidité bearing en  $\mathbb{R}^3 \times \mathbb{S}^1$ , qui seront utilisés dans les chapitres suivants.

## Aperçu de la Partie II

La seconde partie de ce manuscrit décrit les contributions apportées par les travaux conduits dans le cadre de cette Thèse. Il s’agit de techniques permettant de contrôler et de localiser, de manière décentralisée, une formation de drones de type *quadrotors* dans  $\mathbb{R}^3 \times \mathbb{S}^1$ . Ensuite, cette méthode est combinée avec un algorithme assurant la rigidité de la formation, une propriété essentielle pour la convergence de cette dernière et pour les modèles de localisation. La dernière contribution repose sur une analyse non-linéaire de l’observabilité d’un système multi-agents composé de plusieurs quad-rotors. Cette étude montre qu’il est possible d’estimer l’échelle de la formation uniquement à partir de mesures de *bearing* et des vitesses linéaires et angulaires des robots, exprimées dans leur repère.

**Le Chapitre 5** aborde la conception d’un algorithme permettant de contrôler et de localiser, de manière décentralisée, une formation. Il est inspiré de [30, 31]. Cet algorithme de localisation est capable d’estimer les positions et le lacet des agents. Cependant, il est important de garder en mémoire que ce dernier n’a pas connaissance des poses absolues des robots. Par conséquent, il existera systématiquement un décalage lié à une rototranslation globale du système. Il subsiste également une ambiguïté d’échelle, liée à l’absence de données de distance (métriques). Celle-ci est levée par l’introduction d’une unique mesure de distance. Des expérimentations avec cinq quad-rotors sont présentées à la fin du chapitre.

**Le Chapitre 6** expose le problème de maintenance de la *rigidité bearing* d’une formation de quadrotors. Cette propriété est d’une importance capitale pour résoudre les problèmes de contrôle et de localisation de la formation, présentés dans les chapitres précédents. La stratégie de maintenance mise en place est robuste à des limitations des capteurs telles que le champ de vision réduit des caméras, leur faible portée, ainsi que d’éventuelles occlusions entre les agents lors du mouvement de la formation.

**Le Chapitre 7** détaille l’analyse de l’observabilité du système non-linéaire constitué par plusieurs drones. Celle-ci est préliminaire à la réalisation d’un filtre de Kalman étendu, qui est implémenté directement dans  $SE(3)$ , ceci lui permettant d’estimer les positions et les orientations des agents de la formation. Il est important de noter que, par rapport à l’algorithme de localisation présenté au Chapitre 5, les positions sont toujours estimées sans avoir connaissance de la *rototranslation globale* du système, mais avec la bonne échelle. Ce résultat est obtenu sous l’hypothèse (réaliste) que les valeurs d’entrée transmises à chaque agent, dans le repère qui leur est propre (*body-frames*), sont connues.

## Aperçu de la Partie III

**Le Chapitre 8** présente les conclusions des travaux de Thèse et en résume les

contributions. De plus, des pistes de recherche pertinentes à explorer sont proposées et discutées. Celles-ci sont d'ailleurs actuellement étudiées par l'auteur de ce manuscrit.

**L'Annexe A** détaille certains concepts mathématiques présentés au Chapitre 6.

**L'Annexe B** décrit brièvement l'architecture (hardware et software) utilisée durant les expérimentations présentées dans les Chapitres 5,6,7.

---

# Abstract

Since humans exist, they have been witnessing the great power of nature. Among the many fascinating behaviors we find in nature, one which has inspired the work of researchers from all over the world is the show offered every day by insects and vertebrates with their collective behaviors. Indeed, one aspect common to all previous categories is that they give birth to complex *cooperative behaviors* through really simple actions. If, to this concept, we add also the *environment* variable we are talking about a really specific social network mechanism which is called *stigmergy*. Stigmergy is the phenomenon of indirect communication mediated by modifications of the environment. As an example, it is intriguing to think about ants. Each ant is mainly able to do two simple actions: *(i)* leave traces of different pheromone *perfumes* in the environment and *(ii)* follow these traces. Relying only on these *local* capabilities ants are capable to give birth to exceptionally complex behaviors such as the construction of anthills, transportation of food (alone or in groups), sticking together to form a whole and float in water or resist to an external force.

Since many years researchers from all over the world are trying to understand the very small details of these behaviors in order to replicate them. Human beings try to replicate nature because they are aware that nature often presents them with efficient solutions to very specific problems (e.g., flying between two places, flocking or moving an object from one point to another). This is made possible by the fact that nature is constantly solving an optimization problem. This sophisticated process is usually known as *evolution*. Among the many disciplines tackling these problems, we are interested in robotics and specifically in multi-robot applications.

In this broad context, the aim of this Thesis is to give contributions to the state of the art on the collective behavior of a group of flying robots, specifically quadrotor UAVs, which can only rely on their onboard capabilities and not on a centralized system (e.g., Vicon or GNSS) in order to safely navigate in the environment. We achieve this goal by giving a possible solution to the problems of formation control

and localization from onboard sensing and local communication. We tackle these problems exploiting mainly concepts from algebraic graph theory and the so-called *theory of rigidity*. This allows us to solve these problems in a decentralized fashion, and propose decentralized algorithms able to also take into account some typical sensory limitations. The onboard capabilities we referred to above are represented by an onboard monocular camera and an inertial measurement unit (IMU) in addition to the capability of each robot to communicate (through RF) with some of its neighbors. This is due to the fact that an IMU and a camera represent a possible minimal, lightweight and inexpensive configuration for the autonomous localization and navigation of a quadrotor UAV. Notice that sensor limitations are present both in robotics and in nature (e.g., ants have a limited sensing range of the pheromone traces, birds have a limited field of view and so on).

A first contribution of this Thesis is the design of a formation control technique that allows the robots to achieve a certain *shape* only through bearing measurements coming from onboard monocular cameras and, at least, one distance measurement (e.g., coming from a rangefinder). In addition to this, the bearing formation can also be steered in 3D space without changing the bearings between the robots. We also couple this control algorithm with an estimation of the relative poses between the robots of the formation which is able to converge also for non-stationary agents. A second contribution of this Thesis consists in a strategy able to maintain formation rigidity over time against sensing limitations (limited field of view of the camera, maximum/minimum range of the camera, and occluded visibility).

Finally, in order to cope with the missing scale information from pure camera measurements, a third contribution of this Thesis consists in a technique able to estimate the scale of a formation of quadrotor UAVs only through bearing measurements and known agent ego-motion (body-frame linear/angular velocity).

All the theoretical developments discussed in this Thesis are corroborated by simulations and experiments run by using a group of quadrotor UAVs.

The reported results show the effectiveness of proposed techniques in controlling the motion of multiple quadrotor UAVs only relying on (constrained) onboard sensing/communication capabilities.

**Keywords:** multi-robot, formation control, cooperative localization, aerial robotics, rigidity theory.

---

# Acknowledgements

Here I am, at the end of another journey of my life trying to thank all the people who have contributed to it. For sure it was the most tiring (my first white hairs are the proof) but also a fascinating journey of my life. It was the one which changed the most my way of working and approaching different problems both in my professional and personal life.

For this, first of all, I should thank my supervisor: Paolo. Before starting the Ph.D., or sometimes at its beginning, I read that the admiration for your supervisor is inversely proportional to time, but for me was exactly the opposite. The more I went through my Ph.D. the more I recognized how lucky I was to be advised, encouraged and shaken up by him. As any Ph.D., mine was full of highs and lows and mainly thanks to his advice I was able to get across the main obstacle and learn from every error. I truly appreciated his ability to always be there when I asked for his help and to give me more space when I wanted to count mainly on my forces.

Then, I would like to thank Roberto, who guided me for 6 months (and more) in Boston during my visiting period. When I arrived in Boston I thought it would have been really hard to end up again with an amazing advisor but it (luckily) happened again. We addressed difficult topics which I thought to be impossible for me to deal with, but thanks to his patience I was able to get really a lot from those six months. As Paolo, he was incredibly available when I needed him and this helped a lot in giving my Ph.D. a final important boost. The decision of spending six months working with him was among the best decisions I have made during my entire Ph.D.

Another big thank you goes to the head of the Lagadic team: Francois. He gave me the opportunity to be part of an amazing, talented team. Thanks for our occasional, always interesting, talks. With him too, I was truly impressed that whenever I knocked at his door he was available to listen and to help me.

Thanks to my parents I became who I am right now. For this, I will be always

grateful to you both. I could not ask for something better than you. During the years of my Ph.D., besides being the father that everyone would ask for, he was a priceless source of professional advice. I do not think I would be here without him conveying to me all the passion for math, engineering and so on. My mother is one (of the two) anchors for stability and balance in my life. She can understand my mood from the first ten words even if we are speaking over the phone, thousands of kilometers away from each other.

Thanks to my brother who has always believed in me and who, I know, was secretly my first supporter.

My grandmother would have been proud of me finishing a Ph.D. even without knowing what a Ph.D. is, thanks for your continuous admiration and love.

I am grateful that I have many friends in my life but thanking all of them one by one would be quite hard. Therefore I say already sorry if I forgot you. Thanks to Philip, one of my best friends. Thanks for listening to my highs, but mainly to my lows during these three years. For me, you are the proof that friendship does not need 10 years to be built. Thanks to Bruno, Arturo, Anna, Luca, Mirko for your support during these years. We are definitely sure that our friendship is immune to distance. Thanks to Riccardo for his invaluable advice regarding my Ph.D. but mainly for his friendship. I am really glad to have met you during my Ph.D. Thanks to Francesco for introducing me to the Ferrari oven and for his friendship which brought a lot of interesting things in my life and I am sure it will continue to do so. Thanks to Giovanni for having listened to several problems during my Ph.D. At some points knocking at the door of the robotics room was almost the normal consequence of my frustration during experiments. Thanks to Rosa, Fausta and Riccardo for our serene tranquille at home with lots of food and less of Ph.D.-related-stuff. Thanks to Carlotta, Luiz, Roberto, Antonella, Claudia and Melania for our amazing times in Rennes. Thanks to the most recent Italian-Lagadic-subgroup: Marco, Marco and Claudio. To all of you, I hope that our friendship, started thanks to my period in Rennes will not end even if our ways will separate us. Thanks to all my friends in Naples, even talking to you for just 5 minutes helped me a lot sometimes. Thanks a lot to my Bostonian friends: Claudia, Alessandra, Caterina, Morgane, Johnny, Fernando, Mirza and Jamie. Without you, my experience in the US would have been way harder. The same goes for the Ph.D. students in the robotics lab in Boston, in particular Zac and Bee. Thanks to Nicola for having helped me with a session of experiments which finished really early (in the morning). Thanks to Pedro for the lots of coffees shared which brought a lot of nice and interesting conversations. Thanks to Muhammad for his help in the last session of experiments of my Ph.D. and for the interesting chats. Thanks to all the members of the Lagadic team, Fabien, Aurelien, Nicolas, Pol, Bryan, Hadrien, Axel, Firas, Louise, Suman, Aly, Vishnu and

much more who were present in this amazing journey.

The last paragraph is to say GRAZIE to the most important person and the love of my life, Ida. Without you, I honestly think that my Ph.D. would have been ten times harder. You are the main source of balance/happiness/joy of my life and the person who every night listened to me in these years. You motivated me with your sincere and unconditional love. For this reason, I am and will be immensely grateful.

You all made one of my dreams come true. Thank you.

Fabrizio.

## **Acronyms and abbreviations**

**AGL** Above Ground Level

**BLOS** Beyond Line of Sight

**BRM** Bearing Rigidity Matrix

**DBOM** Dynamic Bearing Observability Matrix

**EKF** Extended Kalman Filter

**GNSS** Global Navigation Satellite System

**IMU** Inertial Measurement Unit

**LOS** Line of Sight

**MAS** Multi-Agent System

**MSL** Median Sea Level

**RF** Radio Frequency

**ROS** Robot Operating System

**SfM** Structure from Motion

**SLAM** Simultaneous Localization and Mapping

**UAV** Unmanned Aerial Vehicle

**VTOL** Vertical Take-off and Landing

---

# Contents

Contributions de la Thèse . . . . .	iv
<b>Notation</b>	<b>xix</b>
<b>Chapter 1 Introduction</b>	<b>1</b>
1.1 Thesis contributions . . . . .	4
1.2 Structure of the Thesis . . . . .	6
<b>Part I Preliminaries and state of the art</b>	<b>9</b>
<b>Chapter 2 Multi-robot systems</b>	<b>11</b>
2.1 Introduction . . . . .	11
2.2 Decentralization . . . . .	14
2.3 Multi-aerial vehicles . . . . .	15
2.4 Multi-agent control problems . . . . .	20
2.4.1 Consensus and rendezvous . . . . .	21
2.4.2 Formation control . . . . .	22
2.5 Cooperative localization algorithms . . . . .	24
<b>Chapter 3 Algebraic graph theory and graph rigidity</b>	<b>27</b>
3.1 Introduction . . . . .	27
3.2 Graph theory . . . . .	28
3.2.1 Graph theory definitions . . . . .	29
3.3 Algebraic graph theory . . . . .	31
3.3.1 Laplacian matrix and connectivity . . . . .	32
3.4 Rigidity theory . . . . .	33
3.4.1 The Laman's theorem and some definitions for the case of distance constraints . . . . .	36

3.4.2	The distance-rigidity matrix . . . . .	39
3.4.3	The case of bearing constraints . . . . .	44
3.4.4	Similarity between the rigidity matrix and the Jacobian of robotic manipulators . . . . .	45
<b>Part II Contributions</b>		<b>47</b>
<b>Chapter 4 Main modeling assumptions</b>		<b>49</b>
4.1	Introduction . . . . .	49
4.2	Agent model . . . . .	49
4.3	Directed bearing rigidity in $\mathbb{R}^3 \times \mathbb{S}^1$ . . . . .	52
<b>Chapter 5 Formation control and localization in <math>\mathbb{R}^3 \times \mathbb{S}^1</math></b>		<b>57</b>
5.1	Introduction . . . . .	58
5.1.1	Chapter overview . . . . .	60
5.2	Decentralized formation control . . . . .	60
5.2.1	Rigidity-based control of bearing frameworks in $\mathbb{R}^3 \times \mathbb{S}^1$ . . .	61
5.2.2	Rigidity-based localization of time-varying bearing frameworks in $\mathbb{R}^3 \times \mathbb{S}^1$ . . . . .	63
5.2.3	Coordinated motions in the null-space of the bearing rigidity matrix . . . . .	65
5.2.4	Discussion . . . . .	66
5.3	Simulation results . . . . .	68
5.4	Experimental results . . . . .	69
5.5	Conclusions . . . . .	72
<b>Chapter 6 Rigidity maintenance</b>		<b>75</b>
6.1	Introduction . . . . .	76
6.1.1	Chapter overview . . . . .	77
6.2	Cooperative localization from bearing measurements . . . . .	78
6.3	A bearing rigidity maintenance strategy . . . . .	78
6.3.1	Design of the inter-agent weights . . . . .	79
6.3.1.1	Minimum/maximum range . . . . .	79
6.3.1.2	Limited field of view . . . . .	80
6.3.1.3	Occluded visibility . . . . .	81
6.3.2	The bearing rigidity eigenvalue . . . . .	83
6.3.3	The bearing rigidity maintenance controller . . . . .	87
6.3.4	Discussion . . . . .	88
6.4	Experimental results . . . . .	90
6.5	Conclusions . . . . .	91

<b>Chapter 7</b>	<b>Nonlinear observability and estimation for multi-agent systems</b>	<b>93</b>
7.1	Introduction . . . . .	94
7.1.1	Prior work . . . . .	94
7.1.2	Main contributions . . . . .	96
7.1.3	Chapter overview . . . . .	97
7.2	Preliminaries . . . . .	97
7.2.1	General notation . . . . .	97
7.2.2	Formation, agent and measurement model . . . . .	98
7.2.3	Elements of Riemannian geometry . . . . .	99
7.2.4	Elements of local nonlinear observability . . . . .	101
7.3	<i>Dynamic</i> Bearing Observability Matrix . . . . .	102
7.3.1	Matrix $\tilde{\mathbf{R}}_A$ . . . . .	103
7.3.2	Matrix $\tilde{\mathbf{R}}_B$ . . . . .	105
7.3.3	Numerical verification of the ranks of $\tilde{\mathbf{R}}_A$ and $\tilde{\mathbf{R}}$ . . . . .	109
7.4	A multi-agent Extended Kalman Filter . . . . .	110
7.5	Experimental results . . . . .	112
7.6	Conclusions and future works . . . . .	113
<b>Part III</b>	<b>Conclusions and Future Work</b>	<b>117</b>
<b>Chapter 8</b>	<b>Conclusions and future work</b>	<b>119</b>
8.1	Summary and contributions . . . . .	119
8.2	Open issues and future perspectives . . . . .	121
<b>Appendix A</b>	<b>Additional technical details associated to Chapt. 6</b>	<b>127</b>
A.1	How to go from (6.15) to (6.16) . . . . .	127
A.2	Useful derivatives for the computation of the derivative of $\lambda_6^B$ . . . . .	128
A.3	Towards the derivatives of the weights . . . . .	129
<b>Appendix B</b>	<b>Simulation and experimental architecture</b>	<b>131</b>
B.1	Simulations architecture . . . . .	131
B.2	Real experiments architecture . . . . .	132
B.2.1	The quadrotor . . . . .	132
B.2.2	The Vicon motion capture system . . . . .	134
<b>Bibliography</b>		<b>139</b>
<b>Vita</b>		<b>159</b>



---

# Notation

## General notation conventions

Unless otherwise stated, the main conventions used in the notation of this Thesis are the following

- Scalar quantities are represented by lowercase symbols such as  $u, v$ , and so on.
- Elements of  $\mathbb{R}^n$  and similar sets are interpreted as column vectors and represented by bold lowercase symbols such as  $\mathbf{u}, \mathbf{v}$ , and so on.
- $\mathbf{e}_d$  represents the  $d$ -th element of the standard basis in  $\mathbb{R}^n$ .
- We use the notation  $(a, b, c)$  to indicate a vertical concatenation of elements (scalars, vectors or matrices) and  $[a \ b \ c]$  for horizontal concatenations.
- $\mathbf{I}_n$  is used to represent the identity matrix of dimension  $n \times n$ .
- $\mathbf{0}_{n \times m}$  is used to represent the  $n \times m$  matrix with all elements equal to zero. If  $m = 1$  we also use  $\mathbf{0}_n$ .
- $\mathbf{1}_n$  represents a vector of all ones of dimension  $n \times 1$ .
- Matrixes of real numbers, i.e. elements of  $\mathbb{R}^{n \times m}$ , are indicated with capital letters such as  $\mathbf{A}, \mathbf{B}$  and so on.
- $\mathbf{A}^T$  denotes the matrix transpose of  $\mathbf{A}$ .
- $\text{sym}(\mathbf{A})$  is an operator that extracts the symmetric component of a matrix  $\mathbf{A}$ , i.e.,  $\text{sym}(\mathbf{A}) = \frac{1}{2}(\mathbf{A} + \mathbf{A}^T)$ .
- $\text{skew}(\mathbf{A})$  is an operator that extracts the skew-symmetric component of a matrix  $\mathbf{A}$ , i.e.,  $\text{skew}(\mathbf{A}) = \frac{1}{2}(\mathbf{A} - \mathbf{A}^T)$ .

- $\mathbf{A}^\dagger$  indicates the pseudoinverse of  $\mathbf{A}$ .
- $\otimes$  denotes the Kronecker product.
- $\text{tr}(\mathbf{A})$  is the trace of a matrix  $\mathbf{A}$ .
- $\dot{f}(\cdot), \ddot{f}(\cdot)$  are the first and second derivative of a scalar function  $f$ .
- $\text{stack}(\cdot)$  is an operator which returns a matrix containing a vertical stacking of the arguments.
- $\mathbb{S}^n$  represents the  $n$ -dimensional sphere defined as  $\mathbb{S}^n = \{\mathbf{v} \in \mathbb{R}^{n+1} : \mathbf{v}^\top \mathbf{v} = 1\}$ .
- $\langle \cdot, \cdot \rangle$  represents the Riemannian metric.
- $\mathcal{M}$  represents a Riemannian manifold and  $T_x \mathcal{M}$  is the tangent space of a manifold  $\mathcal{M}$  at a point  $x \in \mathcal{M}$ .
- $\text{SO}(3)$  denotes the special orthogonal group of dimension three, i.e. the space of 3-D rotations  $\text{SO}(3) = \{\mathbf{R} \in \mathbb{R}^{3 \times 3} : \mathbf{R}^\top \mathbf{R} = \mathbf{I}, \det(\mathbf{R}) = 1\}$ .
- $\text{SE}(3)$  denotes the special Euclidean group of dimension three, i.e. the space of 3-D poses  $\text{SE}(3) = \{(\mathbf{p}, \mathbf{R}) : \mathbf{R} \in \text{SO}(3), \mathbf{p} \in \mathbb{R}^3\}$ . the space of 3-D poses  $\text{SE}(3) = \{(\mathbf{p}, \mathbf{R}) : \mathbf{R} \in \text{SO}(3), \mathbf{p} \in \mathbb{R}^3\}$
- $\mathcal{W}$  represents an absolute 3-D world reference frame.
- $\mathcal{A}_i$  represents a body reference frame attached to the  $i$ -th agent.
- $\mathbf{R}_i \in \text{SO}(3)$  represents a rotation matrix transforming from  $\mathcal{A}_i$  to  $\mathcal{W}$ .
- Let  $f : \mathbb{R}^n \mapsto \mathbb{R}, \mathbf{p} \mapsto f(\mathbf{p})$  be a generic scalar function of a vector argument. We indicate with

$$\nabla_{\mathbf{p}} f(\mathbf{p}) = \begin{bmatrix} \left. \frac{\partial f(\mathbf{p})}{\partial p_1} \right|_{\mathbf{p}} \\ \left. \frac{\partial f(\mathbf{p})}{\partial p_2} \right|_{\mathbf{p}} \\ \vdots \\ \left. \frac{\partial f(\mathbf{p})}{\partial p_n} \right|_{\mathbf{p}} \end{bmatrix}$$

the column vector built by stacking the partial derivatives of  $f$  with respect to the elements of  $\mathbf{p}$ . This vector is also called the gradient of  $f$  w.r.t  $\mathbf{p}$ .

## Graph theory

- $\mathcal{V}$  represents the vertex set which is a finite set of elements:  $\mathcal{V} = \{v_1, \dots, v_N\}$ .
- $\mathcal{E}$  represents the edge set which is a subset of ordered pairs of  $[\mathcal{V}^2]$ , the *2-element subsets* of  $\mathcal{V}$ :  $[\mathcal{V}^2] = \{(v_i, v_j), i = 1, \dots, N, j = 1, \dots, N, i \neq j\}$ .  $\mathcal{E} \subseteq [\mathcal{V}^2]$  and  $\mathcal{G}$  is said to be *undirected* if  $(v_i, v_j) \in \mathcal{E} \implies (v_j, v_i) \in \mathcal{E}$ , while it is said to be *directed* if  $(v_i, v_j) \in \mathcal{E} \not\implies (v_j, v_i) \in \mathcal{E}$ .
- $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  represents a graph  $\mathcal{G}$  with vertex set  $\mathcal{V}$  and edge set  $\mathcal{E}$ .
- $e_k$  represents the  $k$ -th edge associated to the graph  $\mathcal{G}$ .
- $\mathcal{N}_i = \{j \in \mathcal{V} \mid (i, j) \in \mathcal{E}\} \subset \mathcal{V}$  represents the set of neighbors of an agent  $i$  with respect to a graph  $\mathcal{G}$ .
- $\mathcal{O}_i = \{j \in \mathcal{V} \mid (j, i) \in \mathcal{E}\}$  represents the set of agents  $j$  for which  $i$  is a neighbor (this distinction is important for directed graphs).
- $\mathbf{p}_i$  represents the position of agent  $i$  in a world frame (usually in  $\mathbb{R}^2$  or  $\mathbb{R}^3$ )
- $\varphi_i, \vartheta_i, \psi_i$  represent the respectively the roll, pitch and yaw associated to  $\mathbf{R}_i$
- $\mathbf{q}_i$  represents the configuration of the agent  $i$  which can be either the pair  $(\mathbf{p}_i, \psi_i)$  or  $(\mathbf{p}_i, \mathbf{R}_i)$  depending on the context
- $\mathbf{p}_{ij}$  represents the vector  $(\mathbf{p}_j - \mathbf{p}_i) \in \mathbb{R}^3$
- $d_{ij}$  represents the actual distance between agents  $i$  and  $j$  computed as  $\|\mathbf{p}_j - \mathbf{p}_i\|$
- $\beta_{ij}$  represents the bearing unit-norm vector between the agent  $i$  and  $j$  expressed in the body frame of the agent  $i$ , and it is defined as

$$\beta_{ij} = \mathbf{R}_i^T \frac{\mathbf{p}_j - \mathbf{p}_i}{\|\mathbf{p}_j - \mathbf{p}_i\|} = \mathbf{R}_i^T \frac{\mathbf{p}_{ij}}{d_{ij}} \in \mathbb{S}^2$$



---

# Introduction

## Contents

---

1.1 Thesis contributions . . . . .	4
1.2 Structure of the Thesis . . . . .	6

---

ONE of the first appearances of automata is reported in a Daoist text called *Lie Zi* (or *Liezi*) in which it is described a rendezvous between King Mu of Zhou (1023-957 B.C.) and Yan Shi, a mechanical engineer. It seems that the engineer showed the king a life-size, human-shaped figure of his mechanical creation [1]. This is an example to show that humans have been interested in replicating human behaviors for a long time and, maybe, even before they tried to replicate animal/insect behaviors. Regarding the latter, the best and most fascinating example is, in my opinion, the one involving flying animals and insects. Homo Sapiens has seen *flying bodies* since the beginning of his time and since then he dreamed of flying. It took some time for Homo Sapiens to figure out how to fly but right now we live in a world in which airplanes are a completely normal thing and flying cars are likely to come next. It seems indeed that robotics is now entering a new era in which robots are more commonly accepted as an integral part of our society. It can be considered similar to what happened to the digital revolution at the beginning of the 21st century. Robots are being employed both in civil and military applications since decades now. At the current time, some of their main applications are industrial robots, military robots, agricultural robots, household robots, nanorobots, medical robots and so on. This list seems to expand year after year and robots are starting to be used in scenarios which we did not even imagine of two decades ago. Examples of this trend are represented by the robots that now are used as waiters, cooks<sup>1</sup> or the ones which will be used to deliver drugs in human bodies efficiently [2,3].

---

<sup>1</sup><http://www.moley.com/>

Replicating a *single* animal/insect behavior is still the focus of great research [4–6] but, since decades now, researchers are also being interested in replicating the collective behaviors of *multiple* animals, insects and bacteria [7]. For this reason, one of the goals of the robotic community is to make robots able to mimic behaviors present in nature such as swarming, flocking, herding and shoaling (schooling) [8–15]. Several researchers, not only in robotics, are interested in studying, understanding and replicating these behaviors. One of the fascinating aspects of these behaviors is that the *agents* of these groups seem often to act *only* by following their own plan but yet it seems that a beehive, an ant colony or a bird migration are incredibly organized as if the agents are pursuing a master plan. Researchers understood that in these cases it happens that all the agents of the group are following some *easy* rules without, usually, the need of a supervisor. By following these rules, they are capable of giving birth to some specific group behaviors. In Fig. 1.1 there are some examples of cooperative behaviors exhibited by fire ants which arise thanks to the mentioned principles. In the case highlighted in Fig. 1.1, and in many others, the



Figure 1.1 – Examples of collective behaviors exhibited by fire ants. Fig. 1.1(a): a raft of 500 fire ants, composed of a partially wetted layer of ants on the bottom and dry ants on top, from [16]. Fig. 1.1(b): buoyancy and elasticity of the ant raft, as shown by attempted submersion by a twig, from [16].

agents (animals/insects) can use only what they sense and act locally (with respect to themselves). There is neither a global information shared between the agents nor some sort of supervision happening.

The last decades have witnessed a growing interest in multi-robot applications, in particular, research about multi-vehicle coordination started around the end of the 1980s in the field of mobile robotics (refer to [17, 18] for the state of the art up to 2006). Between the multi-robot scenarios, many of them are dealing with cases that are similar to the formation-type behaviors mentioned above for animals and insects. We can say that one common aspect of *formation-type behaviors* is that they have a particular centralized goal (i.e., build the anthill/beehive, transport a big piece of food from one point to another, migration of a whole group from one place to another

and so on) to achieve in a decentralized manner (each agent knows only part of the state of its neighbors). In order to do that, one aspect that is common to all these groups is the ability, for each agent, of sensing and communication with a limited subset of the whole group. This subgroup is usually referred as the *neighbors* of the considered agent. Some agents are even able to localize themselves with respect to other agents of the formation and make decisions based only on *local information*.

To summarize, often it happens that the agents of the formation have to perform actions which involve relative measurements (relative to the different agents of the formation and not to a central unit/reference frame). Examples of relative measurements are relative distances, relative bearings<sup>2</sup> or positions and of course combinations of the previous are possible. In robotics, this kind of measurements can be retrieved directly from onboard sensors; some examples are ultrasound sensors, laser scanners, cameras and radio-frequency transmitters/emitters. Using this type of sensors to extract relative measurements allows freeing the whole control structure from the presence of centralized localization systems as Vicon/GNSS and the use of, for example, SLAM algorithms [19, 20]. These concepts bring us to one of the main reasons for the development of multi-agent systems which is the possibility of *decentralization*.

Decentralized solutions play a significant role in multi-agent applications since they allow for *scalable* algorithms (in the sense of computational and communicational loads) with respect to the group size. However, even if the importance and the advantages of decentralization are quite clear to the multi-robot community it seldom happens to come across purely decentralized solutions for multi-robot systems. Indeed, this is the case of all those applications which are heavily relying on centralized localization systems such as the Vicon motion capture system (Fig. 1.2(a)) or the GNSS (Fig. 1.2(b)). These applications have a high value, but they would need substantial adjustments to be deployed in a real-world scenario where centralized systems are not available (e.g., inside a collapsed building after an earthquake, underwater, underground, in weak-GNSS locations or even in deep space).

In robotics, ground (mobile) robots continue to be widely used for multi-robot applications [19, 21], (see Fig. 1.2(c)). The use of ground robots is mostly due to their relatively low cost, their ease of use and control and they are (usually) intrinsic safeness. For all these reasons, at the beginning of multi-robot applications, they helped understand the main strengths of multi-agent systems along with the challenges present in their design. The main limitation of a ground robot is its pervasiveness limited mainly to 2D scenarios. This, and the fast-paced growth of

---

<sup>2</sup>In this Thesis the term *bearing* refers to the non-metric information that can be instantaneously recovered from an onboard camera looking at other UAVs in the scene (that is, the unit bearing vector pointing towards a UAV).

the UAV technology, induced a strong interest in applying the multi-robot concepts apprehended through ground robot applications, to aerial robots and in particular to Vertical Take-Off and Landing (VTOL) UAVs [22].

In this Thesis, we are particularly interested in small-size UAVs, specifically quadrotors and in their coordination using mainly visual information. These objects have been successfully used in different scenarios like the investigation of sites after catastrophic events like the earthquakes of Christchurch (2011) in New Zealand, Emilia-Romagna (2012) and Amatrice (2016) in Italy or after the Fukushima Daiichi (2011) nuclear disaster in Japan. They are massively used to film sportive events like the winter Olympics in Sochi, Krasnodar Krai, Russia and the UEFA Euro 2016 held in France. A recent example which involved 300 Shooting Star drones by Intel and that has been literally celebrated in the news was the one during Lady Gaga's performance for the halftime show at the Super Bowl (2017). This event marked the first time in which drones have been used in a televised event. According to the research firm Gartner<sup>3</sup>, globally the market revenue coming from the production of commercial and personal drones should hit 5 billion euros in 2017 and grow more than 9.4 billion euros in 2020 [23].

Despite the exponential progress in the robotics field, the use of a fully autonomous system in a real-world unknown and unstructured environment is still a subject of active research. For this reason, it is not surprising that three very ambitious research projects [24–26] about multi-robot coordination have been funded by the European Union for a total of about 22.2 millions of Euros in the last six years. Similarly, robotics competitions focused on multi-robot application are receiving exponential interest, and funding, all over the world<sup>4</sup>.

## 1.1 Thesis contributions

Keeping these considerations in mind, this Thesis aims at advancing the state-of-the-art in overcoming the problems mentioned above (in particular formation control and cooperative localization) for the case of a group of quadrotor UAVs equipped with monocular cameras. We address the design of fully-decentralized formation control and estimation techniques based on (i) relative measurements retrievable through onboard cameras and (ii) local communication through Wi-Fi. To make the results of this work even closer to a real-world deploying, we also deal with some typical sensor limitations. An alternative approach to tackle the problem of localization in unknown environments through onboard sensors is an absolute localization through mapping/SLAM techniques [19]. This would require though

---

<sup>3</sup>[www.gartner.com](http://www.gartner.com)

<sup>4</sup>[www.mbzirc.com](http://www.mbzirc.com), [www.robocup.org](http://www.robocup.org), [www.eurathlon.eu](http://www.eurathlon.eu)

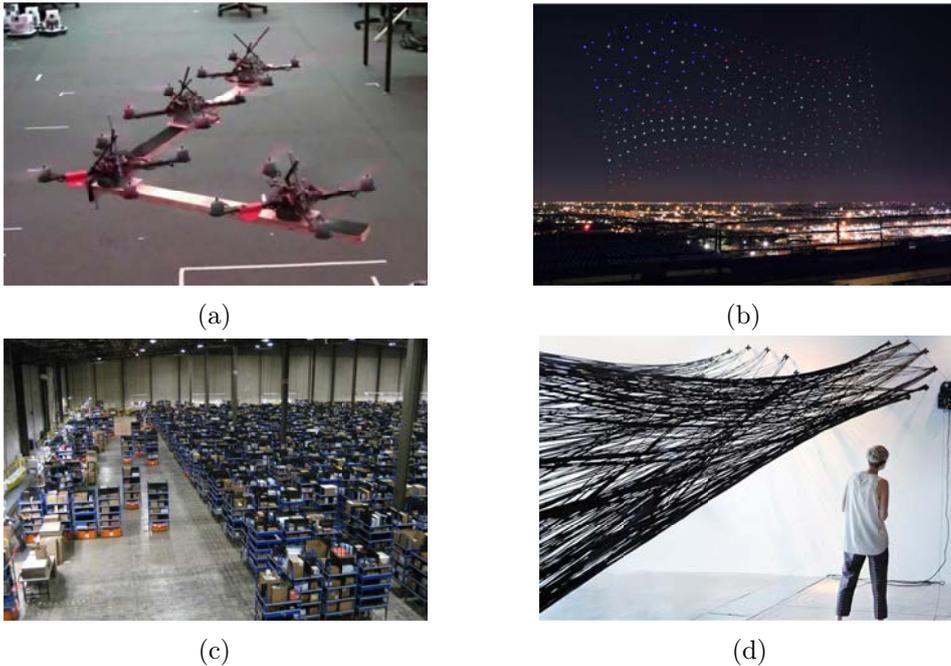


Figure 1.2 – **1.2(a)**: cooperative transportation of a rigid object with four UAVs. Copyright of University of Pennsylvania (USA). **1.2(b)**: an American flag made of 300 Shooting Star drones by Intel at the halftime show of the Super Bowl (2017). Copyright of Intel. **1.2(c)**: a group of ground robots navigating and moving objects around a warehouse. Copyright of Kiva systems. **1.2(d)**: tensile filament structure built by multiple robots. Copyright of the Institute for Computational Design and Construction at the University of Stuttgart.

either a detailed map of the environment or the possibility of running complex SLAM algorithms in real-time onboard the robots. On the other hand, we believe that solving the cooperative localization problem relying only on the *local skills* (local sensing and local communication) of the quadrotor UAVs provides our system with a better flexibility allowing it to work in entirely unknown environments without having to embed on each robot significant sensing/computational capabilities.

In order to solve the formation control for a group of robots, it is important to have a *metric* quantity from which to retrieve the distances between the robots. As well-known, this is not retrievable solely from the *non-metric* visual inputs. Therefore, in this Thesis, we also present a method for retrieving online the *scale* of the formation (and therefore all the distances between the robots) through bearing measurements and assuming that the robots know their body-frame linear/angular velocities. The last assumption is, in our opinion, quite mild since any UAV needs knowledge of these body-frame quantities in order to control its flight.

All the theoretical results presented throughout the Thesis have been validated through extensive sets of simulations and experiments. The experiments were performed using as a platform a group of quadrotor UAVs (Appendix B).

## 1.2 Structure of the Thesis

This Thesis is divided into three main parts. The first part (Part I) contains a short introduction about robotics going more in details into multi-robot applications and especially the ones involving unmanned aerial vehicles (UAVs). A state of the art for these topics is provided, and several survey papers are referenced to help the reader in building up a well-defined context. The second part (Part II) represents the core of the Thesis and its main contributions. The results illustrated in this part correspond to the following author's publications: [27–29]. The third and last part (Part III) of the Thesis contains its conclusions, future work and two appendices which will help the reader in the understanding of some details of the Thesis. Below there is a summary of the content of each part of the Thesis.

### Outline of part Part I

This part contains an extensive state-of-the-art for the topics of multi-robot formation control and localization and the fundamentals of algebraic graph theory and the theory of rigidity.

**Chapt. 2** provides an overview of the multi-robot concepts and specifically of the ones applied to UAVs. There is a brief description of the main multi-agent control problems focusing more on the consensus and the formation control problems. This chapter ends with the description of the cooperative localization problem from relative measurements.

**Chapt. 3** gives an introduction on algebraic graph theory. This is fundamental for the explanation of the concepts of rigidity theory. The last part of this chapter is devoted to explaining more the theory of rigidity both in the case of distance and bearing constraints. This will allow us to arrive at the essential concept, for this Thesis, of bearing rigidity matrix (BRM).

**Chapt. 4** describes the equations regarding the model of the single UAV and we give some details about bearing rigidity in  $\mathbb{R}^3 \times \mathbb{S}^1$  which are instrumental for the following chapters.

### Outline of Part II

The second part of this Thesis presents the contributions of this work related to a technique used for the decentralized formation control and localization of quadrotor UAVs in  $\mathbb{R}^3 \times \mathbb{S}^1$ . Then, we couple this method with an algorithm able to maintain rigidity of the formation, an essential property for the convergence of the formation control and localization schemes. The last contribution consists in a nonlinear observability analysis of a multi-agent system composed of multiple quadrotor UAVs. This study shows that it is possible to estimate the scale of a formation only from bearing measurements and linear/angular velocities of the robots in their body

frames.

**In Chapt. 5** is addressed a decentralized formation control and localization algorithm which takes inspiration from [30,31]. Note that the localization algorithm can estimate the positions and the yaw angle of the agents up to a *global roto-translation* of the whole framework. There is also an ambiguity related to the scale due to the absence of a metric information<sup>5</sup>. This ambiguity is removed with the introduction of a single distance measurement. Experiments with five quadrotors are presented at the end of the chapter.

**In Chapt. 6** we deal with the problem of maintaining *bearing rigidity* of a formation of quadrotor UAVs. This property is of paramount importance to solve the problems of formation control and localization which we addressed in the previous chapter. The maintenance strategy can cope with some sensor limitations as limited field of view of the cameras, their limited range and the possibility of occlusions between the agents occurring during motion.

**In Chapt. 7** is presented an observability analysis of the nonlinear system made of different UAVs. This analysis is preliminary to the design of an extended Kalman filter (EKF), implemented directly on SE(3) which is able to estimate the positions and orientations of the agents of the formation. It is important to note that, with respect to the localization algorithm proposed in Chapt. 5, the positions are always estimated up to a *rototranslation* of the whole framework but with the right scale without, for the latter, depending on the availability of a distance measurement. This is obtained by taking advantage of the (realistic) assumption of knowing the inputs given to each agent in their own body frame.

### Outline of Part III

**In Chapt. 8** are reported the conclusions of the Thesis and the main contributions brought to the state-of-the-art are summarized. Moreover, some open issues are listed and we discuss future directions which would be worth exploring. Note that some of the proposed future directions are the subject of the author's current research.

**In Appendix A** there are some mathematical details relative to Chapt. 6.

**In Appendix B** it is briefly described the hardware and software architecture used to carry out the experiments described in Chapt. 5 to 7.

---

<sup>5</sup>Visual inputs, and therefore bearing measurements are *non-metric* measurements.



## Part I

# Preliminaries and state of the art



---

# Multi-robot systems

## Contents

---

2.1	Introduction . . . . .	11
2.2	Decentralization . . . . .	14
2.3	Multi-aerial vehicles . . . . .	15
2.4	Multi-agent control problems . . . . .	20
2.4.1	Consensus and rendezvous . . . . .	21
2.4.2	Formation control . . . . .	22
2.5	Cooperative localization algorithms . . . . .	24

---

## 2.1 Introduction

As pointed out in the introduction of this Thesis (Chapt. 1), research on multi-agent systems, and in particular multi-robot systems (e.g., quadrotors), has flourished over the last decades with a number of theoretical and experimental results. The improvements in this field were also made possible by the constant technological advancements in onboard sensing, communication and computing power [32–36]. These systems have many potential advantages with respect to single-agent systems (e.g., decentralized processing, resilience against failures of individual agents, faster task completion times and so on).

In a lot of multi-robot scenarios robots are acting either by themselves (with respect to other robots) or just with a limited awareness of the presence of other robots in their environment (Fig. 2.1). Since many years the robotics community has been attracted by the so called multi-agent systems (MASs) [18]. A MAS usually denotes a group of several agents which are able to communicate and sense the whole group or part of it. Some multi-agent systems can also give birth to collective behaviors in order to fulfil different tasks and/or interact with the environment. Several reasons



Figure 2.1 – An example of four robots working together, in an automotive factory, on the same car. The robots are fulfilling different tasks and they have a really limited awareness of other robots and of their environment (from which they are separated through a cage).

pushed researchers in developing such systems in lieu of single agent systems. One of them is that a MAS could be helpful in solving complex tasks in a robust (to single robot failures) and highly flexible way. The tasks addressed by multi-robot groups are usually the ones which would be hard or even impossible for a single agent and where the use of multiple robots translates in a drastic reduction of time for the completion of the task. An example is the PATH project at the University of Berkeley<sup>1</sup> which demonstrated multiple cars driving together in platoons. This project served as inspiration for many projects which followed. as the one undertaken by the VISLAB<sup>2</sup> which, between many achievements, completed the Vislab Intercontinental Autonomous Challenge (VIAC), see Fig. 2.2. More recent examples of multi-robot applications which have a big impact on the current society are the ones given by the mobile robots used to move shelves around warehouses at Amazon and Alibaba, see Fig. 2.3 while refer to [37] for an early description of the Kiva Systems which is behind the robots at Amazon warehouses. Within the scope of robotics, the following are between the most promising multi-robot fields of application: autonomous search and rescue, precision agriculture<sup>3</sup>, military systems, mobile sensor networks, transportation systems (Fig. 2.2), firefighting, cooperative localization, surveillance [22, 38]<sup>4</sup>, medical robotics, exploration, intervention and mapping in dangerous or inaccessible/unknown areas [19, 32, 39, 40], target tracking [35, 41], large antenna synthetization with multiple robots, cooperative building of 3D structures [42, 43], complex cooperative manipulation and transportation [33, 44–46].

<sup>1</sup>California Partners for Advanced Transit and Highways, [www.path.berkeley.edu](http://www.path.berkeley.edu)

<sup>2</sup>[www.vislab.it](http://www.vislab.it)

<sup>3</sup><http://www.precisionhawk.com/>

<sup>4</sup><https://www.aeryon.com/>

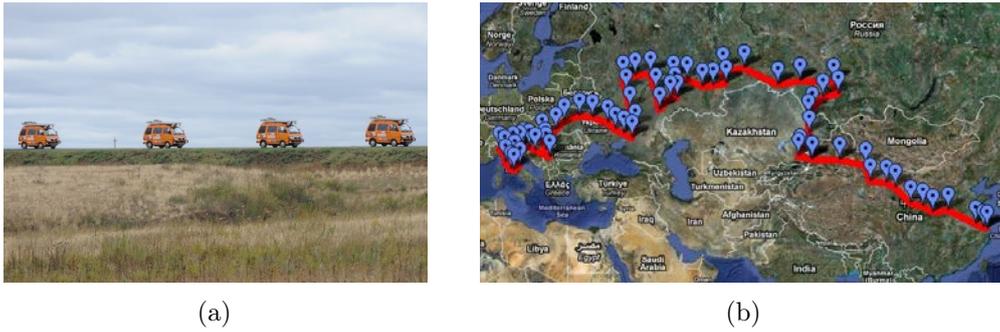


Figure 2.2 – Fig. 2.2(a): convoy of vehicles driving for the Vislab Intercontinental Autonomous Challenge (VIAC) which one of the first examples of autonomous driving challenges. Fig. 2.2(b): the route covered during the challenge which lasted 100 days in which 15.926 km were covered, from Parma (Italy) to Shanghai (China).

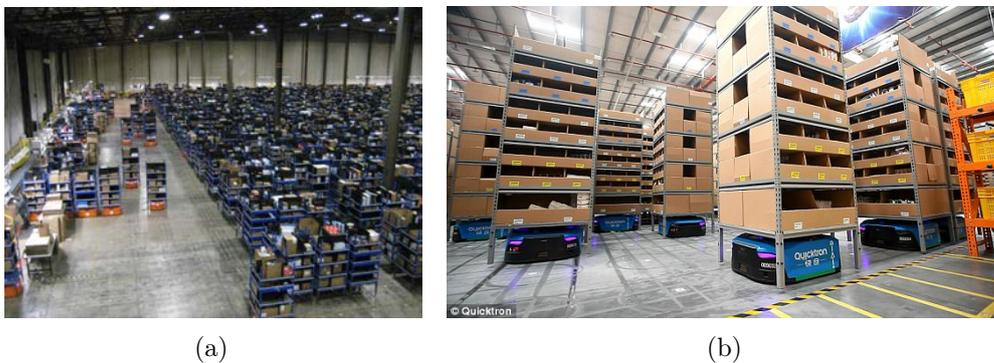


Figure 2.3 – Multiple robots moving shelves around a warehouse at Amazon (Fig. 2.3(a)) and Alibaba<sup>5</sup>(Fig. 2.3(b)).

Anywhere on page where float appears Another reason for the development of MASs is the understanding that it would be easier to control ten agents with limited capabilities than embed one agent with the capabilities of the aggregate ten-agents-system<sup>6</sup>. When designing a control algorithm for a group of agents it is important to make them rely on their onboard capabilities and not on a central unit otherwise a single failure could compromise the functioning of the whole system. In brief, the goal of research about MASs is to design control techniques which allow these systems to exploit all the strengths mentioned above<sup>7</sup>.

Among the multi-robot community, at the end of 1990s and beginning of 2000s,

<sup>5</sup>[www.flashhold.com](http://www.flashhold.com)

<sup>6</sup>It is interesting to notice that interest in MASs really blossomed at the beginning of the 2000s. This was also the time in which it happened the switch, in microprocessor manufacturing, from single-core CPUs to multi-core CPUs. The reason of going from single-core to multi-core CPUs is the same which drove the development of MASs: researchers realized that it was easier to build a processor with multiple cores than one core with the same capabilities of a multi-core processor.

<sup>7</sup>In this work the term multi-agent system is often used as a synonym of multi-robot system. If one wants to be more precise a multi-robot system can be seen in general as a subset of multi-agent systems.

research about the coordination of multiple aircraft and especially UAVs started to become really active [47]. As already said, a central part of this work is about the decentralized formation control and estimation for a group of quadrotor UAVs. For this reason the following sections describe more into details the state-of-the-art of these topics, exploring first the concept of decentralization and giving a classification of the different UAVs, pointing out the differences between fixed-wing and rotary-wing UAVs.

However, just before diving into these notions, we will shortly describe a concept which is instrumental for the rest of this work: decentralization.

## 2.2 Decentralization

Beside multi-agent systems and robotics, decentralization plays a fundamental role in many domains such as economics, finance, politics and society. One basic and practical example of decentralization is the one of a group of people transporting a heavy load which would be impossible to even lift for one person (see Fig. 2.4(a)). Another example would be the one in which ants cooperate to transport a piece of food [44, 46], (see Fig. 2.4(b)). In these cases usually there is no central unit and there are communication capabilities but usually communication is not used for the whole task but only in its critical parts (e.g., people are not talking all the time while transporting a heavy load). In the same way there are sensing capabilities but usually they are not exploited for the whole duration of the task and/or they are changing over time depending on the relative configuration of the agents (e.g. people are not looking all the time at the other members of the group while transporting a heavy load). The key feature in all these cases is decentralization.

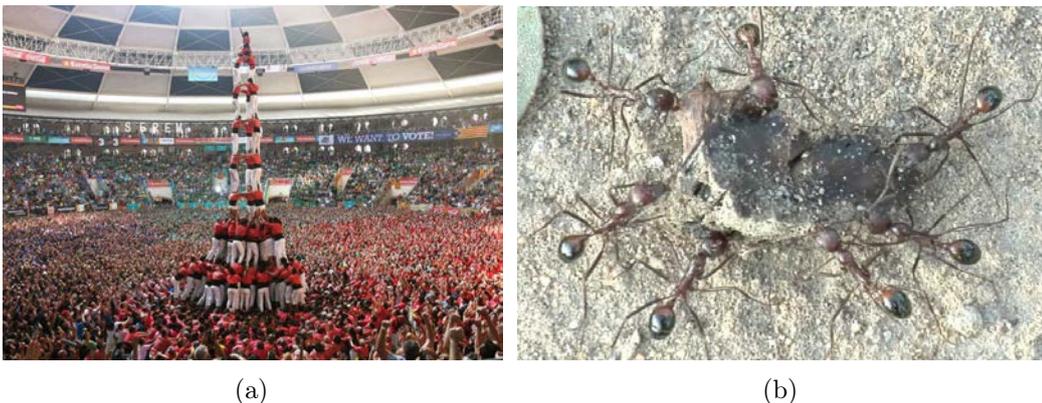


Figure 2.4 – Examples of decentralized behaviors in humans and ants. Fig. 2.4(a): a group of people performing in the 25th castells competition at Tarraco Arena ring in Tarragona, Spain. Fig. 2.4(b): *cockerelli* ants retrieving a piece of fig, from [44]

On the other hand, according to [48], a centralized system is such when at least one agent needs to sense a global information or communicate with all the other agents at once. In control theory terms a centralized system is one in which the controller is able to communicate and control with all the agents of the formation [49–51]. This kind of structure (see Fig. 2.5(a) for an example) is usually easier to implement than a decentralized structure but it carries with it different problems:

- The complexity of the algorithm scales with the number of agents of the formation
- The control structure is not robust, in the sense that the presence of a central control unit identifies also a single point of failure.
- The agents of the formation need to deal with a quantity of information that grows with the size of the agent group.

A decentralized system (see Fig. 2.5(b) for an example) instead scales well against the number of agents, it is more resilient to external threats and allows the agents to deal only with a limited quantity of information which, in general, does not grow with the group size<sup>8</sup>.

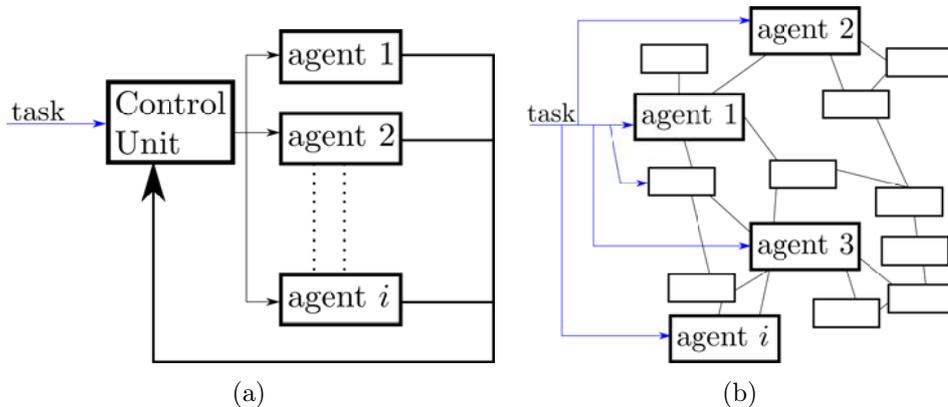


Figure 2.5 – Examples of schemes of a centralized (Fig. 2.5(a)) and decentralized (Fig. 2.5(a)) system

## 2.3 Multi-aerial vehicles

When we refer to an Unmanned Aerial Vehicle (UAV) we indicate a flying vehicle which does not have a human pilot onboard<sup>9</sup>. UAVs can be classified according to

<sup>8</sup>Note that decentralized is often used as a synonym of distributed.

<sup>9</sup>International Civil Aviation Organization (ICAO). Unmanned Aircraft Systems (UAS); ICAO: Montreal, QC, Canada, 2011.

Class & Weight, w (kg)	Category & Weight, w (kg)	Normal Employment	Normal Operating Altitude, h (ft)	Normal Mission Radius (km)	Example Platform
Class I w<150	Small w>20	Tactical Unit (employs launch system)	$h \leq 5000$ AGL	50 (LOS)	Luna, Hermes 90
	Mini $2 \leq w \leq 20$	Tactical Unit (manual launch)	$h \leq 3000$ AGL	25 (LOS)	ScanEagle, Skylark, Raven, DH3, Aladin, Strix
	Micro w<2	Tactical Patrol/-section, Individual (single operator)	$h \leq 200$ AGL	5 (LOS)	Black Widow
Class II $150 \leq w \leq 600$	Tactical	Tactical Formation	$h \leq 10000$ AGL	200 (LOS)	Sperwer, Iview 250, Hermes 450, Aerostar, Ranger
Class III w>600	Strike/Combat	Strategic/National	$h \leq 65000$	Unlimited (BLOS)	
	HALE	Strategic/National	$h \leq 65000$	Unlimited (BLOS)	Global Hawk
	MALE	Operational/Theater	$h \leq 45000$ MSL	Unlimited (BLOS)	Predator A, Predator B, Heron, Heron TP, Hermes 900

Table 2.1 – NATO Unmanned Aerial Systems UASs Classification Guide. September 2009 JCGUAV meeting

many different features [52]. An example of classification upon weight is reported in table 2.1. However, classification of UAVs is outside of the scope of this work and therefore we only deal with the difference between UAVs which are defined fixed-wing (e.g., Fig. 2.6(a)) and rotary-wing (e.g., Fig. 2.6(b)) and we do it from an high level perspective.

Each of the two categories has its own benefits and downsides. The advantages and disadvantages of fixed-wing UAVs, with respect to rotary-wing UAVs, are



Figure 2.6 – Examples of fixed and rotary wing UAVs. Fig. 2.6(a): the MQ-9 Reaper, from General Atomics. Fig. 2.6(b): the Mavic Pro, from DJI.

reported in table 2.2.

Advantages	Disadvantages
Simpler structure and therefore less maintenance required	Usual need of a runway or a launcher to take-off and landing
More efficient aerodynamics which translates in longer flight times at higher speeds	No hovering capabilities
Capability of carrying greater payloads for longer distances	Normally not suited for indoor applications due to their low degree of manoeuvrability when compared to rotary wing UAVs

Table 2.2 – Some of the advantages and disadvantages of fixed-wing UAVs

Regarding rotary-wing UAVs, their advantages and disadvantages, with respect to fixed-wing UAVs, are reported in table 2.3.

Advantages	Disadvantages
Ability of vertical take-off and landing (VTOL)	Limited flight time
Hovering capability	More complex mechanics and electronics and therefore more maintenance needed
Higher manoeuvrability, especially indoor	Lower cruising speed

Table 2.3 – Some of the advantages and disadvantages of rotary-wing UAVs

In both categories there is ongoing research to overcome the listed disadvantages. Examples include fully-actuated multi-rotor UAVs [53] as also VTOL fixed-wing

UAVs [54]<sup>10</sup>. There are works, as [38], which envision to use swarms of both fixed-wing and rotary-wing UAVs.

However, even if UAVs are being applied in many fields the regulations related to these objects are still at an embryological state, refer to [55] for a survey and to Fig. 2.7 for a global overview of the UAV regulations up to October 2016. In addition to Fig. 2.7 it is also interesting to have a look at the global distribution of first releases of UAV regulations in the world in Fig. 2.8 to understand the pioneering countries of the world in the field of UAV regulations.

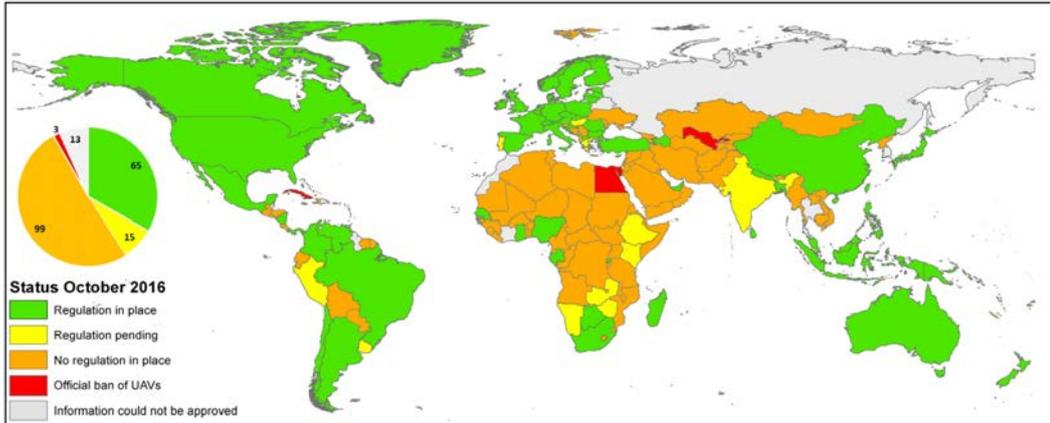


Figure 2.7 – Global overview of current status of UAV regulations on a country-level resolution (status: October 2016), from [55]

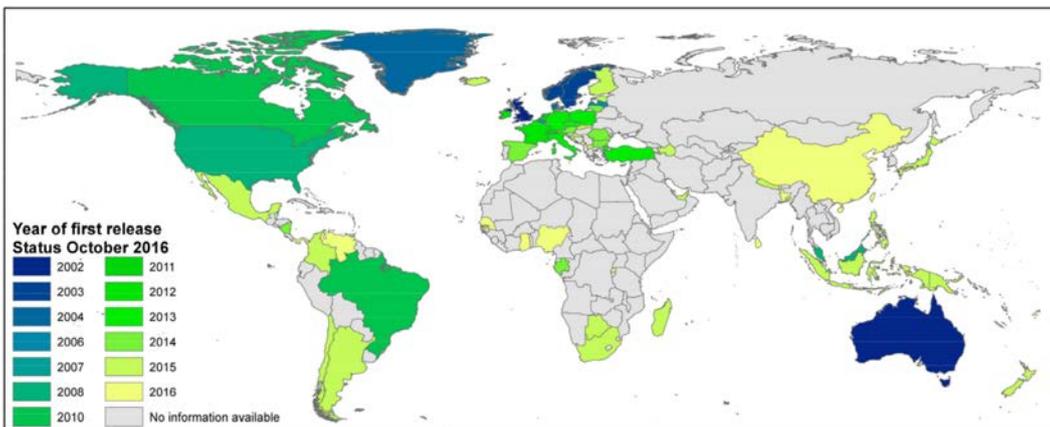


Figure 2.8 – Global distribution of first releases of UAV regulations on country level resolution (status: October 2016), from [55]

Based on the advantages of rotary-wing UAVs listed in table 2.3 and especially because of their relatively low-cost and high pervasiveness in the 3D space we chose to use quadrotor UAVs for our research. The spectrum of application of quadrotor UAVs

<sup>10</sup>When a UAV is a hybrid between a fixed-wing and rotary-wing UAV is usually addressed as *compound*.

is really broad and, according to [55], beside many military applications, between the quadrotor UAVs civil application there are: high-resolution surface reconstruction in the geosciences, documentation of cultural heritage and archaeological sites, precision agriculture and forest change detection, support for disaster management, surveying and mapping, land administration and wildlife observation. Beside these we can add package delivery, events filming (e.g., concerts, sport events) and the film industry.

The fields of application of single UAVs can be seen as a subset of the ones of group of UAVs. Indeed, multiple UAVs can fulfil the tasks assigned to a single UAV in a fraction of the time a single UAV would take [56]. Beside these tasks there are others which are only suitable for multiple UAVs and that are therefore impossible for a single UAVs. These are all the tasks in which cooperation is not only better but fundamental such as aerial mobile manipulation [57], cooperative load-carrying [58] and cooperative construction [42, 59, 60] or really specific cooperative tasks [61].

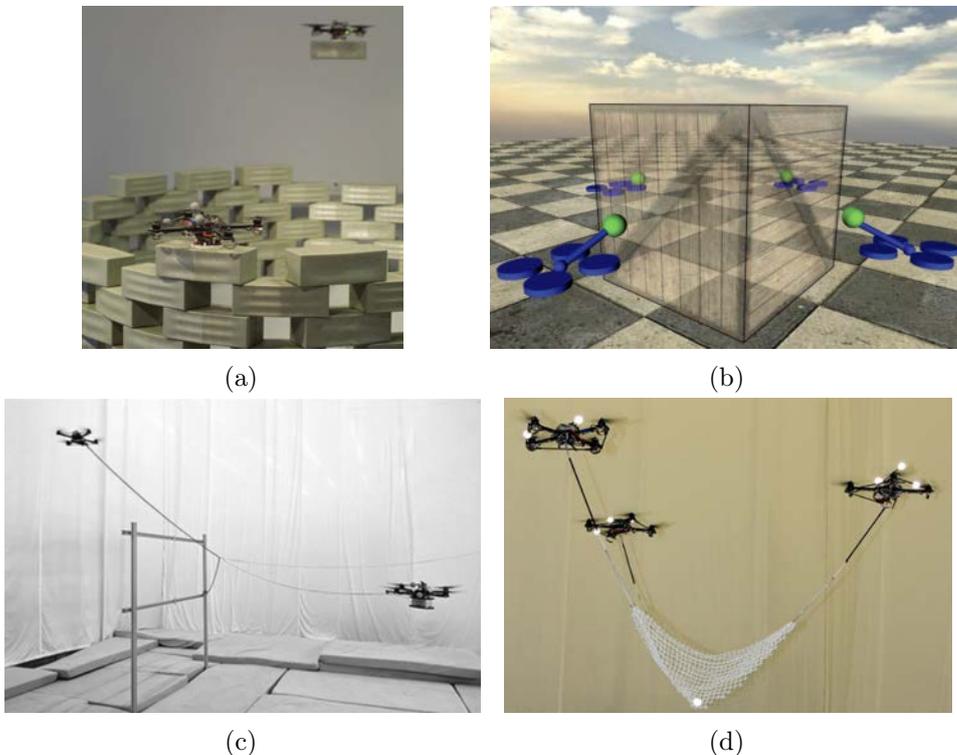


Figure 2.9 – 2.9(a): two quadrotors assembling a structure, from [56]. 2.9(b): concept of *flying-hand* for transportation of a rigid object with four UAVs commanded through human hand movements, from [57]. 2.9(c): two quadrotors tying a knot to build a 7.4 m long rope bridge one can walk on, from [60]. 2.9(d): three quadrotors attached to a net used to juggle and catch a ball through, from [61]

All the works referenced in Fig. 2.9, together with others such as [34, 62, 63], achieve quite impressive results but they are all relying on a centralized system to retrieve the poses of the involved agents. Again, the aim of this Thesis is to

find decentralized algorithms for (i) the formation control and (ii) cooperative localization of a group of quadrotor UAVs without the need of a centralized system and therefore relying only on the local skills of the robots. Some results in this sense are starting to appear in the literature [64, 65]. Hence, in the following sections a comprehensive state-of-the-art of these two topics, specifically for UAVs, is given.

Different classifications have been proposed over the years to put some order within the realm of multi-agent systems (e.g., [66] focuses on the sensing capabilities and the interaction topology of the agents while [67]). For the next sections we adopted a terminology (and a structure) similar to [48].

## 2.4 Multi-agent control problems

A good starting point to have an idea about some of the most common multi-agent control problems in the literature is Fig. 2.10.

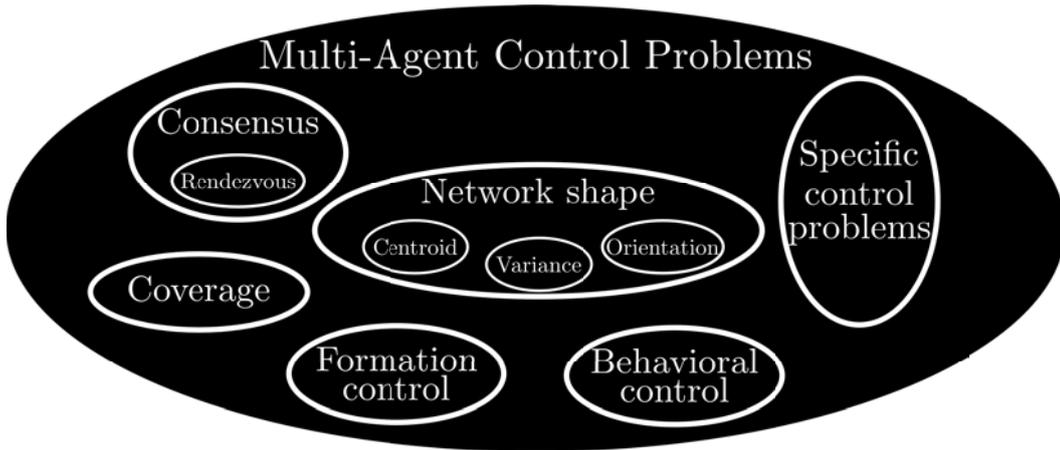


Figure 2.10 – Main multi-agent control problems

Following the structure of [48], we go through the same problems but focusing more on the problems of formation control and localization which are instrumental for this work.

Note that, when dealing with the control of multi-agent systems, due to the already intrinsic complexity of the problem, it is common to start the design of the control algorithm by considering the nodes/agents/robots as having simple dynamics (e.g., simple/double integrators). Therefore, usually the dynamics of the agents is not taken into account. This does not prevent the addition of the dynamics of the agent in a second stage of the design of the algorithm.

### 2.4.1 Consensus and rendezvous

Consensus-based algorithms have their roots in computer science [68] but, since many years now, they are playing a central role in the multi-robot literature [67, 69, 70], refer to [71] for a survey. The consensus problem, according to [69], is the process of reaching an agreement concerning a certain quantity of interest depending on the state of all the agents. Without going too much into details we can think about the consensus protocol as a solution to the following problem. We have a set of  $N$  agents with an internal state  $\mathbf{x}_i \in \mathbb{R}^m$ . As we said above, the internal dynamics for the state evolution of each agent is considered to be the one of a single integrator and therefore:  $\dot{\mathbf{x}}_i = \mathbf{u}_i$  (with  $\mathbf{u}_i$  the input to agent  $i$ ). Let us also associate an interaction graph  $\mathcal{G}$  to the formation, where the agents are the vertexes of  $\mathcal{G}$ . The problem is to design the control inputs  $\mathbf{u}_i$  so that all the states  $\mathbf{x}_i$  converge to the same common value  $\bar{\mathbf{x}}$ :

$$\lim_{t \rightarrow \infty} \mathbf{x}_i(t) = \bar{\mathbf{x}}, \forall i \quad (2.1)$$

The main characteristic of the consensus protocol is that it achieves (2.1) by making use only of relative information with respect to the state of the neighbors, therefore basing everything on relative sensing and on a decentralized structure of the control algorithm. A basic solution to this problem is achieved by choosing the inputs  $\mathbf{u}_i$  such that:

$$\mathbf{u}_i = \sum_{j \in \mathcal{N}_i} (\mathbf{x}_j(t) - \mathbf{x}_i(t)) \quad (2.2)$$

where  $\mathcal{N}_i$  is the set of neighbors of agent  $i$ . It is also possible to express (2.2) in a compact form for all the agents as:

$$\dot{\mathbf{x}}(t) = -(\mathbf{L} \otimes \mathbf{I}_m)\mathbf{x}(t) \quad (2.3)$$

where  $\mathbf{x} \in \mathbb{R}^{Nm}$  is the stack of all the  $\mathbf{x}_i$ , and  $\mathbf{L} \in \mathbb{R}^{N \times N}$  is the Laplacian associated to the graph  $\mathcal{G}$ . More details about the Laplacian matrix can be found in Sect. 3.2.

Another key feature of the consensus protocol is that the convergence is assured if the graph  $\mathcal{G}$  is connected (i.e., the second smallest eigenvalue  $\lambda_2$  of the Laplacian matrix  $\mathbf{L}$  is non-null, see Sect. 3.3).

The consensus protocol has several applications and depending on them it gets different names. For example:

- **Rendezvous:** a consensus protocol to meet at a common point. Its goal is to uniform the positions of the agents.
- **Alignment:** a consensus protocol to point in the same direction. Its goal is to uniform the angles of the agents.

- Distributed estimation: a consensus protocol to agree on the estimation of some distributed quantity (e.g., average humidity/temperature in a room)
- Synchronization: a consensus protocol to agree on the same time regardless of phase shifts or different rates in the clocks.

The consensus protocol is the model on which are based many decentralized laws for control and/or estimation. A more generic form of the (2.2), neglecting the time dependence, is:

$$\dot{\mathbf{x}}_i = \mathbf{u}_i = \sum_{j \in \mathcal{N}_i} k_i f(\mathbf{x}_j - \mathbf{x}_i) \quad (2.4)$$

where  $f$  is a local action function of the relative state between the agent  $i$  and its neighbors. For instance, assume that  $\mathbf{p}_i \in \mathbb{R}^m$  is the agent position in some  $m$ -dimensional space and let  $\mathbf{p}^* = (\mathbf{p}_1^*, \dots, \mathbf{p}_N^*)$  a set of desired positions. The dynamics of the agent  $i$  is such that  $\dot{\mathbf{p}}_i = \mathbf{u}_i$ . Then the following law converges to the *desired shape* up to, of course, a global translation:

$$\mathbf{u}_i = k_i \sum_{j \in \mathcal{N}_i} (\mathbf{p}_i - \mathbf{p}_j) - (\mathbf{p}_i^* - \mathbf{p}_j^*) \quad (2.5)$$

It is important to note that in order to implement the (2.5) there is the need for the agent  $i$  to know its relative position with respect to its neighbors in a common reference frame.

### 2.4.2 Formation control

Another possibility of multi-agent control is the so called *formation control*, which is of big interest for this work. In general, formation control is the problem of controlling a group of agents by imposing geometric constraints on their relative poses. Depending on the states of the robots and on their sensing capabilities one can impose certain constraints which result in different behaviors (e.g., through interdistances [72, 73] or angles [74–77] between the agents). For example, in [77] it is presented a formation controller which is able to keep a formation defined in terms of bearing constraints and which requires only bearing measurements. The controller converges almost globally and maintains limited inter-agent distances even without metric informations. Regarding specifically UAVs, one of the first works about formation flight control is represented by [78].

Formation control can be also seen as a specialized version of (2.2) and (neglecting the time-dependence) can be expressed with the following, which is a sort of generalization of the (2.5)

$$\mathbf{u}_i = \sum_{j \in \mathcal{N}_i} (\mathbf{o}_{ij}^i - \mathbf{o}_{ij}^{id}) \quad (2.6)$$

where  $\mathbf{o}_{ij}^i \in \mathbb{R}^m$  is the current relative measurement between  $i$  and  $j$  and  $\mathbf{o}_{ij}^{id} \in \mathbb{R}^m$  is an offset which encloses the desired formation information. Note that this approach can be applied easily to both bearing and distance cases. In the first case  $\mathbf{o}_{ij}^i, \mathbf{o}_{ij}^{id}$  would be a bearing while in the second case a distance. Of course other options are possible depending on the application.

Formation control is still a really active field of research, for example in [79] the desired formation is defined in terms of both inter-agent distances and angles and there is no need of a global or common reference frame.

Formation control can be implemented through different architectures depending on the requirements. Some renowned implementations of formation control are the leader-follower and flocking paradigms.

Leader-follower, as its name suggests, consists of a type of formation control in which there is one agent which acts as a leader and the remaining agents are the followers, see Fig. 2.11(a). In the literature there is plenty of applications to this paradigm (e.g., target tracking, environment patrolling) but one of its main disadvantages is that it creates a system with a single point of failure (the leader).

Flocking takes inspiration from nature and it refers to a certain behavior exhibited by birds. A similar behaviour is showed in groups of fishes and it is referred as *schooling* (if the group is swimming in the same direction in a coordinated manner) or *shoaling* (if the group of fishes stay together for social reasons). The difference with respect to the leader-follower behavior is that in this case there is no leader. On the other side, the thing in common with the leader-follower architecture is that the agents cannot see all the other agents of the formation. In the case of birds the formation changes over time as the birds in front get tired they leave their place to birds which are behind them optimizing the efficiency of the migration or other goals. [80] represents a recent and complete work specifically on flocking and rendezvous behaviors.

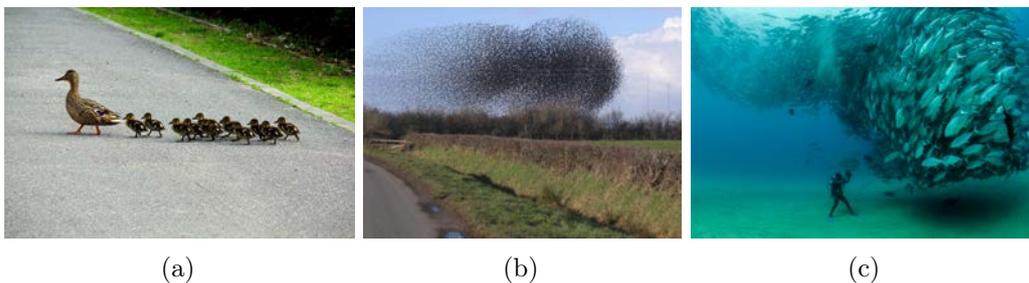


Figure 2.11 – Example of leader-follower, flocking and shoaling. Fig. 2.11(a): leader-follower happening between a mother duck and her ducklings, Fig. 2.11(b): example of flocking of birds, Fig. 2.11(c): example of shoaling of fishes <sup>11</sup>.

<sup>11</sup>[www.octavioaburto.com/cabo-pulmo](http://www.octavioaburto.com/cabo-pulmo)

## 2.5 Cooperative localization algorithms

As the problem of formation control, *cooperative localization* is another fundamental problem in the multi-robot applications. In this context, with cooperative localization we refer to the process of estimating the configuration<sup>12</sup> of the agents in a common reference frame from a sparse set of relative measurements between the robots. As in the rest of the Thesis, the aim is to solve the localization problem without relying on a centralized localization system as a GNSS or the Vicon motion capture system.

The topic of cooperative localization was firstly investigated in [81]. Many more followed based on different approaches for the localization, such as probabilistic approaches [82, 83], based on Kalman filters [84–86] or particle filters [87]. Note that cooperative localization plays a central role not only in multi-robot systems but also in many others multi-agent systems as sensor networks [88–91], Structure from Motion (SfM) [92, 93] in computer vision, and graph drawings [94]<sup>13</sup>. This problem is of paramount importance in all the scenarios highlighted in Sect. 2.1 such as the ones in which the robots need to navigate in an unknown environment and therefore they need to know their relative configuration. In this cases usually the absolute configurations (with respect to a centralized system) of the robots is not needed but the relative one (with respect to a reference frame attached to the formation) would be enough. The solution to the localization problem can follow different approaches, which strongly depend on the sensors carried onboard by the different agents used for the relative measurements. Some work has been done in position-based localization [99], bearing-based localization [100, 101] and a mix of the two [102, 103]. Our interest lies in the second category because, as we said already, we assume the robots equipped with monocular cameras.

There is one thing though which is seldom taken into account in the literature mentioned above. Cooperative localization algorithms need, together with relative sparse measurements, also the identity associated to these measurements. This means that each robot when it acquires a measurement it needs also to be capable to associate the measurement to a specific robot. This operation is often referred as *tagging* [104]. The identities of the robots need to be coherent between the robots of the formation. Here we want to stress out that this operation, according to the robot sensing capabilities, could be not trivial but it is not in the scope of our work. Therefore, all the algorithms described in the rest of the Thesis assume that tagging is an intrinsic capability of the robots along with their sensing capabilities, and

---

<sup>12</sup>The meaning of the term *configuration* depends on the application. It can refer to positions (in  $\mathbb{R}^2$  or  $\mathbb{R}^3$ ), positions and orientations and so on.

<sup>13</sup>Note that some works [95, 96] use the specific term *relative mutual localization* (RML) to refer the cooperative localization in scenarios in which the measurements are expressed with respect to the body frames of the agents [97, 98], like the one of this Thesis.

consequently we do not deal explicitly with the *measurement-anonymosity* problem. Freeing the proposed algorithms from the need of identities is left as a possible interesting future work. About this topic, addressed as *anonymous localization*, research has been done both for the case of position measurements [95, 105] (in 2D) and for the one of bearing measurements [96, 106, 107] (respectively in 2D and 3D).

In this chapter we gave some details about multi-agent systems, focusing on multi-robot systems (e.g., UAVs) and going through the most relevant problems to this field of research. Deliberately, we gave more emphasis to two of these problems: formation control and cooperative localization. The reason for this choice is simple: we believe that the theory of rigidity together with algebraic graph theory represents the perfect tool to tackle both problems. Therefore, in the following Chapt. 3 we give some details about these topics which represents the theoretical core on which this work is based.



---

# Algebraic graph theory and graph rigidity

## Contents

---

3.1	Introduction . . . . .	27
3.2	Graph theory . . . . .	28
3.2.1	Graph theory definitions . . . . .	29
3.3	Algebraic graph theory . . . . .	31
3.3.1	Laplacian matrix and connectivity . . . . .	32
3.4	Rigidity theory . . . . .	33
3.4.1	The Laman's theorem and some definitions for the case of distance constraints . . . . .	36
3.4.2	The distance-rigidity matrix . . . . .	39
3.4.3	The case of bearing constraints . . . . .	44
3.4.4	Similarity between the rigidity matrix and the Jacobian of robotic manipulators . . . . .	45

---

## 3.1 Introduction

IN the previous Chapt. 2 we gave a brief overview about multi-robot systems and we understood that an ideal scenario for multi-robot applications would be one which is embedded with the following characteristics. There should be availability of relative poses with respect to any other robot in the group in a common reference frame. There should be the possibility to communicate with any other robot in the group with no delays. Each agent should have unlimited memory and processing power (onboard/offboard). In real-world scenarios this is seldom the

case. Instead, in many realistic applications, these systems are subject to several limitations, requirements and constraints such as:

- limited sensing: partial measurement of the other robot states (e.g., distance, bearing), lack of a common shared frame, occlusions, limited field of view, limited range
- limited communication: occlusions, limited range, delays, maximum data rate
- limited memory and processing power.

*Graphs* represent a powerful mathematical structure to encode, in a compact manner, these limitations. Indeed, they are built to describe and model pairwise relationship between specific objects. For this reason graph theory plays a central role in this Thesis and lays the groundwork for the so-called *rigidity theory* which helps us solving the problems of formation control and cooperative localization. Hence, the following sections describe more in details graph theory and the theory of rigidity, two topics which are broadly used throughout the rest of the Thesis.

## 3.2 Graph theory

According to [108], the first appearance of graph theory can be found in a paper by Leonhard Euler (Fig. 3.1) on the *Seven Bridges of Königsberg* and published in 1736. His work was then followed by Augustin-Louis Cauchy [109] and Gergonne L'Huilier [110] and gave birth to the mathematical field of *topology*. Since then,



Figure 3.1 – Leonhard Euler, Swiss mathematician (1707-1783)

graph theory has been massively applied to many distinct areas such as sociology, computer science, mathematics, biology, chemistry and physics (refer to [111] for a thorough analysis of graph theoretic methods and their application in the analysis and design of multi-agent systems).

First of all, it is important to understand that several graphs can be associated to a single group of robots to encode different capabilities/properties of the group [112] and the information/action flow among the robots, such as:

- Sensing graph: it encodes what agents can be sensed by each agent
- Communication graph: it encodes to what agents each agent can communicate some sort of information
- Action graph: for each control action, encodes what robots are (locally) affected

More graphs can be defined depending on the application. In the following section we go through some definitions of graph theory which are of help for the next developments.

### 3.2.1 Graph theory definitions

Let us define:

- $\mathcal{V}$  as the vertex set which is a finite set of elements:  $\mathcal{V} = \{v_1, \dots, v_N\}$
- $[\mathcal{V}^2]$  as the 2-element subsets of  $\mathcal{V}$  defined as

$$[\mathcal{V}^2] = \{(v_i, v_j), i = 1, \dots, N, j = 1, \dots, N, i \neq j\}$$

- $\mathcal{E}$  as the edge set which is a subset of ordered pairs of  $[\mathcal{V}^2]$  such as  $\mathcal{E} \subseteq [\mathcal{V}^2]$

Then we indicate with  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  a graph  $\mathcal{G}$  with vertex set  $\mathcal{V}$  and edge set  $\mathcal{E}$ . In general a graph  $\mathcal{G}$  can be *directed*<sup>1</sup> or *undirected* Fig. 3.2.  $\mathcal{G}$  is said to be *undirected* if

$$(v_i, v_j) \in \mathcal{E} \implies (v_j, v_i) \in \mathcal{E},$$

while it is said to be *directed* if

$$(v_i, v_j) \in \mathcal{E} \not\Rightarrow (v_j, v_i) \in \mathcal{E}$$

An undirected graph can be thought of as a special case of a directed graph where if an edge  $e_{ij} \subseteq \mathcal{E}$  then  $e_{ji} \subseteq \mathcal{E}$ . An undirected graph is said to be *complete* if for every pair of vertexes there is an edge connecting them. If the graph is directed, it is complete if each pair of vertexes is connected by a unique edge, one in each direction. In this Thesis we indicate with  $\mathcal{K}$  the complete graph associated to a set of nodes.

---

<sup>1</sup>Directed graphs are often referred as *digraphs*.

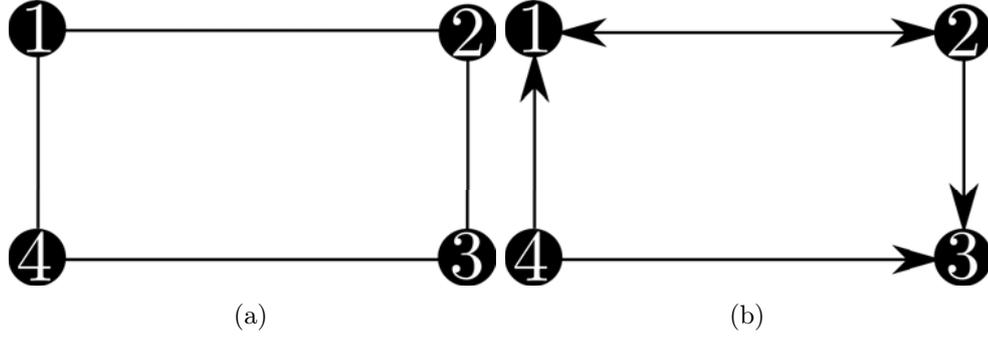


Figure 3.2 – Examples of undirected (Fig. 3.2(a)) and directed (Fig. 3.2(b)) graphs

A node  $v_j$  is said *adjacent* (*neighbor*) of  $v_i$  if  $(v_i, v_j) \in \mathcal{E}$  and therefore, given a node  $v_i$ , the set  $\mathcal{N}_i$  defined as

$$\mathcal{N}_i = \{j \in \mathcal{V} \mid (i, j) \in \mathcal{E}\} \subset \mathcal{V} \quad (3.1)$$

represents the set of neighbors<sup>2</sup> of an agent  $i$  with respect to a graph  $\mathcal{G}$ . With  $\mathcal{O}_i$  we represent the set of agents  $j$  for which  $i$  is a neighbor defined as

$$\mathcal{O}_i = \{j \in \mathcal{V} \mid (j, i) \in \mathcal{E}\}, \quad (3.2)$$

we note that the distinction between  $\mathcal{N}_i$  and  $\mathcal{O}_i$  is only relevant in case of directed graphs.

We can also define the *degree* of a node  $v_i$ , indicated as  $deg(v_i)$ , as the number of edges to which  $v_i$  is connected, therefore it holds, for a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , the so-called *handshaking lemma*<sup>3</sup>:

$$\sum_{v \in \mathcal{V}} deg(v) = 2|\mathcal{E}|.$$

For a directed graph there is the need, with respect to a generic vertex  $v_i$ , to define the *in-degree* (defined as  $deg^-(v_i)$ ) and the *out-degree* (defined as  $deg^+(v_i)$ ) respectively as the number of edges *entering* and *exiting*  $v_i$ . Therefore it holds the so-called *degree sum formula*:

$$\sum_{v \in \mathcal{V}} deg^-(v) = \sum_{v \in \mathcal{V}} deg^+(v) = |\mathcal{E}|.$$

If, for every vertex  $v \in \mathcal{V}$ ,  $deg^-(v) = deg^+(v)$  the graph is called a *balanced directed graph*.

We can also define a *path* as a finite or infinite sequence of edges which connect a sequence of distinct vertices. If a path starts and ends in the same vertex the path

<sup>2</sup>More formal definitions of neighbors, in different senses, can be found in [113].

<sup>3</sup>This formula was first introduced by L. Euler in *Seven Bridges of Königsberg*.

is called *cycle*. An undirected graph is said *connected* if there exists a path joining any two vertexes in  $\mathcal{V}$ . On the other side a directed graph is said *strongly connected* if there exists a directed path joining any two vertexes, while it is said to be *weakly connected* if there exists an undirected graph connecting any two vertexes in  $\mathcal{V}$ .

An extension to the concept of graph explained so far can be the one of *weighted graph*. Usually the weight is associated to each edge of the graph. The easiest way to think about weighted graphs is with respect to sensing graphs and a weight associated to an edge in order to encode, e.g., how reliable is the corresponding measurement. Usually to each edge is associated a weight between 0 and 1 (where 0 indicates a loss of measurement and 1 maximum reliability) but other choices are completely legit. The concepts of weighted graphs are heavily exploited in this Thesis in Chapt. 6.

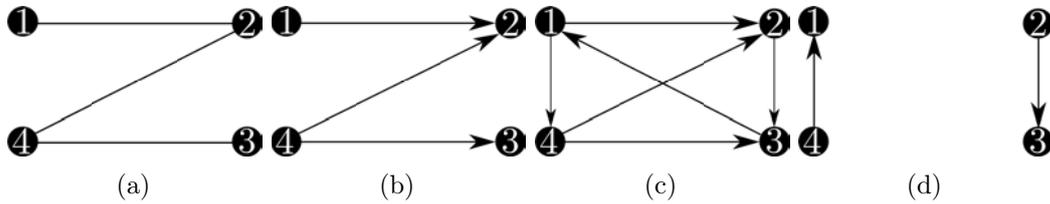


Figure 3.3 – Fig. 3.3(a): connected graph, Fig. 3.3(b): weakly connected graph, Fig. 3.3(c): strongly connected graph, Fig. 3.3(d): disconnected graph

### 3.3 Algebraic graph theory

Graphs are great tools to encode, in a compact way, different information but it is possible to boost their power if we associate to them algebraic methods. This, gives birth to a branch of mathematics called *algebraic graph theory*. This is probably the feature which made graphs so widespread in so many different fields of engineering. One of the most investigated branches of algebraic graph theory is the one which links graphs to linear algebra. Regarding this, it is important to introduce three fundamental matrixes: adjacency matrix, degree matrix, incidence matrix.

Given a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  the *adjacency matrix*  $\mathbf{A} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$  (where  $|\mathcal{V}|$  is the number of nodes) has elements  $\mathbf{A}_{ij}$  defined as:

$$\mathbf{A}_{ij} = \begin{cases} 1 & \text{if the edge } e_{ij} \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases} \quad (3.3)$$

The *degree matrix*  $\mathbf{D} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$  has elements  $\mathbf{D}_{ij}$  defined as:

$$\mathbf{D}_{ij} = \begin{cases} \text{deg}(v_i) & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases} \quad (3.4)$$

Note that to a directed graph are associated an *in-degree* and an *out-degree* matrix.

The *incidence matrix*  $\mathbf{E}$ , for directed graphs, has elements  $\mathbf{E}_{ij} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{E}|}$  defined as<sup>4</sup>

$$\mathbf{E}_{ij} = \begin{cases} -1 & \text{if the edge } e_{ij} \text{ exits from vertex } v_i \\ +1 & \text{if the edge } e_{ij} \text{ enters into vertex } v_i \\ 0 & \text{otherwise.} \end{cases} \quad (3.5)$$

Another fundamental matrix in algebraic graph theory is the so-called *Laplacian matrix*  $\mathbf{L} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$  (refer to [114, 115]) which can be built in two different ways starting from the matrixes defined above

$$\mathbf{L} = \mathbf{D} - \mathbf{A} \quad (3.6)$$

$$\mathbf{L} = \mathbf{E}\mathbf{E}^T. \quad (3.7)$$

The two definitions are equivalent and the latter is not dependent on the particular labelling or orientation chosen for the graph. For an undirected graph  $\mathcal{G}$  the matrix  $\mathbf{L}$  results symmetric and positive semi-definite. This means that all its  $|\mathcal{V}|$  eigenvalues  $\lambda_i$  are real and non-negative. If we order them as

$$0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_{|\mathcal{V}|} \quad (3.8)$$

it holds that the graph  $\mathcal{G}$  is connected if and only if  $\lambda_2 > 0$  [114, 115] (e.g., refer to Fig. 3.4 for some examples).

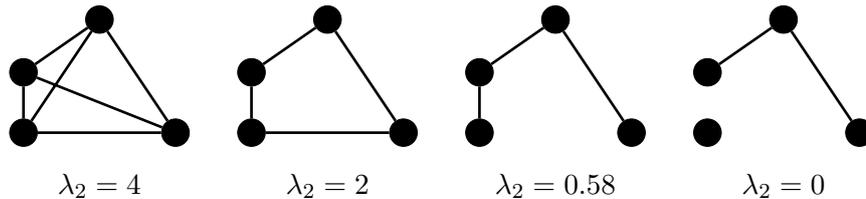
The definitions given above are the most common ones in the literature. There are different definitions of these matrices but, in this work, when we refer to the adjacency, degree, incidence and Laplacian matrices we are referring to the definitions in (3.3) to (3.7)

### 3.3.1 Laplacian matrix and connectivity

The Laplacian matrix defined in (3.6–3.7) represents therefore a link between combinatorial properties of a graph (existence of a path among two agents) and spectral properties of a matrix associated to a graph (condition on  $\lambda_2$ ). This means that there is therefore an equivalence between the combinatorial and spectral properties of graphs which plays a fundamental role for control and estimation purposes. The power of the Laplacian matrix is that its second smallest eigenvalue  $\lambda_2$  offers a measure of the degree of connectivity of the graph to which the matrix is associated. For this reason  $\lambda_2$  is often referred as the *connectivity eigenvalue* (or algebraic connectivity). It is important to highlight that  $\lambda_2$  is a global property of the graph which means that it is influenced by all the nodes (agents) of the graph.

---

<sup>4</sup>Some authors use the opposite notation.

Figure 3.4 – Graphs with different connectivity eigenvalues  $\lambda_2$ 

For example, the Laplacian matrix can be used to study the consensus protocol (refer to Sect. 2.4.1), one of the major problems in multi-agent applications. It can be shown, by analyzing (2.3), that the convergence to an arbitrary equilibrium is related to the spectral properties of the matrix  $L$ . In particular  $L$  represents the state-transition matrix in the closed-loop dynamics of the system. Basically the consensus converges faster as the  $\lambda_2$  increases and vice versa.

Previously, at the beginning of Sect. 3.2, we mentioned that to a specific group of robots we can associate different graphs. From this we can understand that the Laplacian matrix can be useful to model the connectivity of different properties of the group such as sensing, communication or action. For example, if the underlying communication graph associated to the group of robots is connected it means that there can be a flow of information between each pair of robots.

Analogously to connectivity and taking inspiration from structural mechanics one can define a stronger property which allows to study different properties of a group of agents and it involves not only the graph associated to the agents but also their relative position in space (2D or 3D). This property is called *rigidity* and it is discussed in the following section.

### 3.4 Rigidity theory

Rigidity has its roots in mechanics, this theory was born as a combinatorial theory for characterizing the *stiffness* or *flexibility* of structures made of rigid bodies connected by flexible linkages or hinges. One of the first works considered related to the theory of rigidity and graphs is the so-called *Euler-conjecture about polyhedra*.

Rigidity theory has found a large number of applications in the context of *formation control* (see Sect. 2.4.2) based on, e.g., relative distance measurements [72, 116–119] and relative bearing measurements [76, 112, 120–123], mainly differing in the assumptions and/or simplifications taken at the design stage. For instance, regarding the bearing case, the authors of [76, 112] have considered the formation control of planar kinematic agents by assuming a *common reference frame* and an *undirected topology* for the measurement graph (i.e., all measurements are assumed

to be reciprocal), and [122] has extended these results to arbitrary dimensions. The work [121] has instead dropped the assumption of a common reference frame while, however, retaining that of an undirected measurement topology. Another example is [124] in which they apply distance-based rigidity to beacon localization.

As mentioned several times, rigidity theory plays a pivotal role also in the dual problem of *cooperative localization* (see Sect. 2.5) from local relative measurements [99, 100, 119, 125, 126]. Indeed, the rigidity of the formation is a necessary requirement for recovering, from the available relative measurements, a consistent solution of the localization problem in a common shared frame.

It is interesting to recognize how (mechanical) rigidity played some role in the design of the first working aircraft of history, the one flown by Wright's brothers at Kill Devil Hills, North Carolina on December 17, 1903. Indeed, at the Smithsonian National Air and Space Museum, a small demo is offered to the visitors for understanding how, without the rigidity of some parts of the plane the airplane would have never taken off, see Fig. 3.5. The Wrights were aware of earlier biplane designs and especially the 1896 Chanute-Herring glider which had a special bracing. Steel wires crisscrossed between vertical wooden struts that supported the upper and lower wings, creating a simple, *rigid* structure. This structure was tested by the Wrights for the first time on a kite they built in 1899.

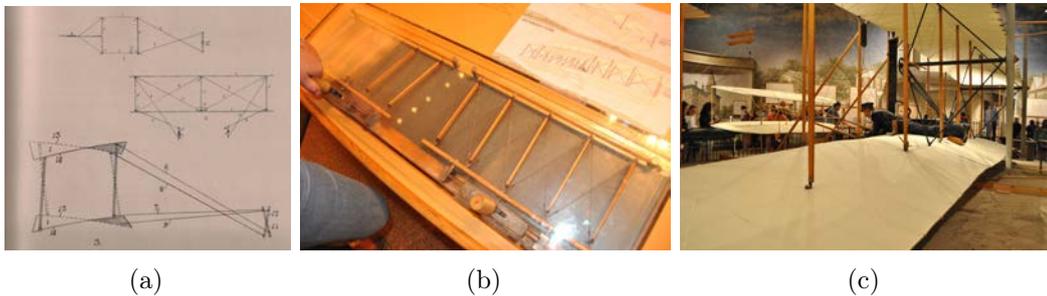


Figure 3.5 – Wright brothers and rigidity. Fig. 3.5(a): shows some drafts by Wilbur Wright about a kite to test warping for roll control. Fig. 3.5(b): shows a demo available at the Smithsonian National Air and Space Museum to stress out the difference that a rigid structure made in the design of the the 1903 Wright Flyer, the craft that ushered in the age of flight. Fig. 3.5(c): at the same museum, shows the 1903 Wright Flyer, the craft that ushered in the age of flight.

Rigidity can also play a combining role between two usually separated communities, the multi-robot and the parallel robots ones. This has been a topic of discussion in the workshop *Rigidity Theory for Multi-agent Systems Meets Parallel Robots* held during the 2017 IFAC congress in Toulouse (France)<sup>5</sup>. Intuitively each robot of a

<sup>5</sup>For more details refer to the website of the workshop: <https://parrigidwrkshp.sciencesconf.org/>.

multi-agent system could be seen as a passive joint of a virtual mechanical parallel robot and each measurement between them as a rigid connection.

As aforementioned, the theory of rigidity can be formulated in case of any kind of geometrical constraints on the edges, distances and bearings are two possible cases. In this section, we focus on the case of distance constraints because, for the sake of illustration, they are more intuitive to deal with. Therefore, each edge in the following graphs represents the distance between the interested nodes. From here we gradually evolve towards the case of bearing constraints in 3D space, which is the case considered in all the scenarios of this Thesis and is detailed in Sect. 4.3. Note that the main results, theorems, and definitions described below for the case of distance constraints usually apply flawlessly to the case of bearing constraints.

In order to define and understand rigidity we need to introduce the concept of *framework* (or *formation*). A *framework* is identified, in the case of distance-constraints in 2D, as a pair  $(\mathcal{G}, \mathbf{p})$ , where  $\mathbf{p} = \text{stack}(\mathbf{p}_1, \dots, \mathbf{p}_N)$ ,  $\mathbf{p}_i \in \mathbb{R}^2$  is the position of the  $i$ -th robot and  $N$  is the total number of robots of the formation. In the previous Sect. 3.3.1 we gave some details about connectivity. Rigidity is a stronger property and it is possible to prove that if a framework is rigid, the underlying graph is also connected [127], while the vice versa is not true. Roughly speaking, a framework  $(\mathcal{G}, \mathbf{p})$  is rigid if satisfying the distance constraints associated with the edges of  $\mathcal{G}$  univocally determines the shape (spatial arrangement of the positions  $\mathbf{p}$ ) of all the agents of the formation up to a roto-translation. An example to understand better this concept is reported in Fig. 3.6.

Another intuitive way to define rigidity is the following. A framework  $(\mathcal{G}, \mathbf{p})$  is rigid if the only motions satisfying all the constraints over the edges in  $\mathcal{E}$  are those which would be allowed in case  $\mathcal{G}$  was the complete graph  $\mathcal{K}$ .<sup>6</sup> This leads us to a great advantage of rigid graphs which is related to the complexity of a hypothetical formation control algorithm associated with a group of agents with an underlying rigid topology. Shortly, one way of controlling  $N$  agents is by regulating all the distances among  $N$  agents to some desired values. This would yield to a complete graph  $\mathcal{K}$  which has  $N(N - 1)$  edges. This translates into a complexity of the formation control algorithm of  $O(N^2)$ . Instead, if a graph is rigid in two dimensions, only  $2N - 3$  edges<sup>7</sup> are needed to control all the distances between the agents and therefore the complexity decreases to  $O(N)$ , i.e., linear versus quadratic.

As stated in [72], if a graph is rigid for a certain set of positions, it will be rigid for *almost* all other positions. The adverb *almost* refers to some nongeneric sets of positions as the ones corresponding to collinear agents<sup>8</sup>. This is also why in order to

<sup>6</sup>Note that these concepts are formalized in (3.10–3.11).

<sup>7</sup>This precise number of edges comes from the Laman's theorem, defined in Sect. 3.4.1.

<sup>8</sup>In the literature, these cases are sometimes referred as *degenerate cases*.

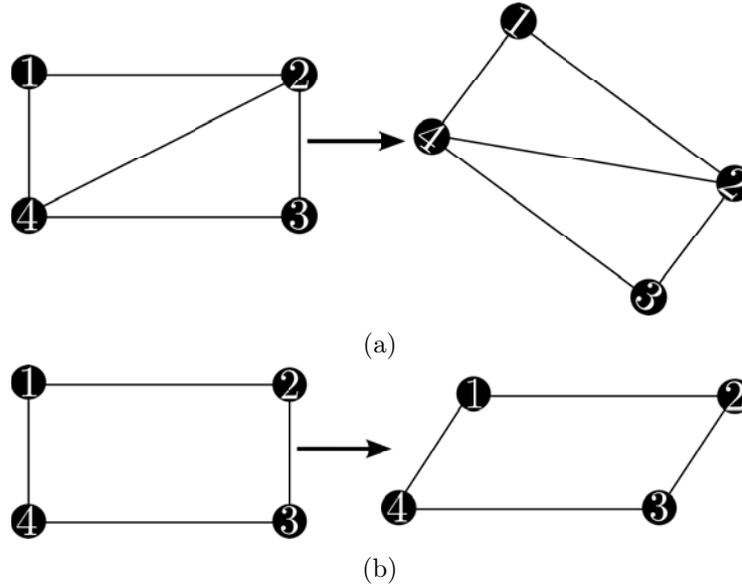


Figure 3.6 – Examples of rigid and nonrigid frameworks. The framework in Fig. 3.6(a) is rigid because it cannot be deformed while satisfying the distance constraints over the edges. The framework in Fig. 3.6(b) is not rigid because it can be deformed by a smooth motion without violating the distance constraints over the edges.

study rigidity we can employ two different approaches:

- Combinatorial approach: through *Laman's theorem* [128]. This approach does not take into account the positions of the agents and is only function of the topology of  $\mathcal{G}$ . Indeed, it relies on the previous concept whose idea is that if a formation is rigid in a set of positions, it will be rigid *almost* anywhere else.
- Algebraic approach: through linear algebra and the concept of *rigidity matrix* [129, 130]. This approach takes also into account the positions of the agents.

The theory of rigidity brings with it several definitions and theorems and therefore in the following part of this section, we briefly go through some of them, refer to [72] and its bibliography for more details.

### 3.4.1 The Laman's theorem and some definitions for the case of distance constraints

In order to introduce the Laman's theorem we need the concept of *induced subgraph* of a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  [72]. Let  $\mathcal{V}'$  be a subset of  $\mathcal{V}$ , then the subgraph of  $\mathcal{G}$  induced by  $\mathcal{V}'$  is the graph  $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$  where  $\mathcal{E}'$  includes all the edges of  $\mathcal{E}$  that are incident on a vertex pair in  $\mathcal{V}'$ .

The *Laman's theorem* states the following: a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  modeling a formation in two dimensions with  $|\mathcal{V}|$  vertexes and  $|\mathcal{E}|$  edges is rigid if and only if there exists a subgraph  $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$  with  $2|\mathcal{V}| - 3$  edges such that for every subgraph  $\mathcal{V}''$  of  $\mathcal{V}$ , the induced subgraph  $\mathcal{G}'' = (\mathcal{V}'', \mathcal{E}'')$  of  $\mathcal{G}'$  has a number of edges  $|\mathcal{E}''|$  which is:  $|\mathcal{E}''| \leq 2|\mathcal{V}''| - 3$ .

Laman's theorem provides a test, applicable to graphs in two dimensions and for the case of distance constraints, to check whether a graph is rigid. This is why rigid graphs with a number of edges equal to  $2|\mathcal{V}| - 3$  are sometimes called *Laman graphs*.<sup>9</sup> A theorem for graphs in three dimensions does not exist but a partial extension of the Laman's theorem to three dimensions is discussed in [72] and it requires the graph  $\mathcal{G}$  of the considered framework  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  to have  $3|\mathcal{V}| - 6$  edges.

Another important concept in the theory of rigidity is the one of *minimal rigidity* [129]. Intuitively, a framework  $(\mathcal{G}, \mathbf{p})$  is minimally rigid if removing any of its edges will cause the loss of rigidity, see Fig. 3.7. Furthermore, we can refer to rigid graphs which are rigid but not minimally rigid as *redundantly rigid graphs* [131], and an edge is called a *redundant edge* if the graph remains rigid after its removal [132].

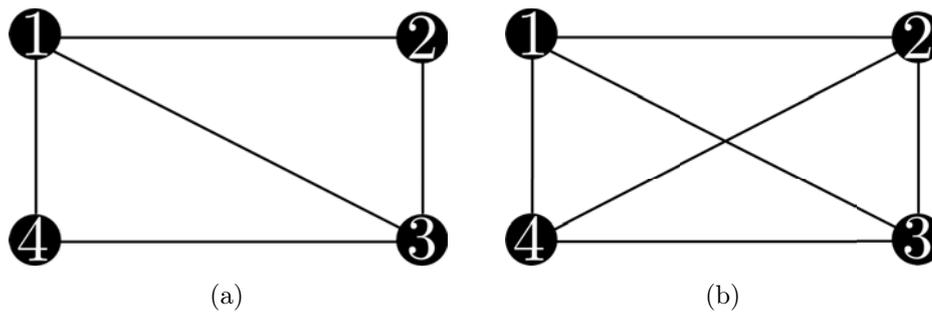


Figure 3.7 – Examples of minimally rigid (Fig. 3.7(a)) and redundantly rigid (Fig. 3.7(b)) graphs.

Minimally rigid graphs have been described in details in several works such as [116]. A procedure to build minimally rigid graphs in two dimensions is represented by the so-called *Henneberg construction* [129, 133, 134]. As stated in [72], for graphs in three dimensions there exist some operations that are analogous to the Henneberg construction, but they are in the form of conjectures [129] and it is therefore not sure that they represent necessary and sufficient conditions to build and deconstruct all minimally rigid graphs.

There are several and equivalent formal definitions of rigidity, for example in [116, 135] the following definition is given. Let us define a *rigidity function*

<sup>9</sup>Sometimes they are also called *plane isostatic graphs*.

associated with the framework  $(\mathcal{G}, \mathbf{p})$  as the function  $\mathbf{g}_{\mathcal{G}} : \mathbb{R}^{2|\mathcal{V}|} \rightarrow \mathbb{R}^{|\mathcal{E}|}$  given by

$$\mathbf{g}_{\mathcal{G}}(\mathbf{p}) := \text{stack}(\dots, \|\mathbf{p}_k - \mathbf{p}_j\|^2, \dots). \quad (3.9)$$

The  $i$ -th component of  $\mathbf{g}_{\mathcal{G}}(\mathbf{p})$  corresponds to the value of the geometric constraint (distance in this case) over the edge  $e_i \in \mathcal{E}$ . After defining the rigidity function it is possible to give the following definition. A framework  $(\mathcal{G}, \mathbf{p})$  is *rigid* if there exists a neighborhood  $\mathcal{U} \subset \mathbb{R}^{2|\mathcal{V}|}$  of  $\mathbf{p}$  such that

$$\mathbf{g}_{\mathcal{G}}^{-1}(\mathbf{g}_{\mathcal{G}}(\mathbf{p})) \cap \mathcal{U} = \mathbf{g}_{\mathcal{K}}^{-1}(\mathbf{g}_{\mathcal{K}}(\mathbf{p})) \cap \mathcal{U} \quad (3.10)$$

where  $\mathcal{K}$  is the complete graph with the same vertices as  $\mathcal{G}$ . The definition (3.10) asserts that, in a neighborhood  $\mathcal{U}$  of  $\mathbf{p}$ , the possible *shapes* corresponding to the values of the constraints over the edges  $\mathcal{E}$  in  $\mathcal{G}$  are the same that one would obtain by considering the edges of the complete graph  $\mathcal{K}$ . Moreover, if this is true for all positions  $\mathbf{p}$ , and not only for a neighborhood of  $\mathbf{p}$ , the framework is globally rigid and therefore the following definition.

A framework  $(\mathcal{G}, \mathbf{p})$  is *globally rigid* if

$$\mathbf{g}_{\mathcal{G}}^{-1}(\mathbf{g}_{\mathcal{G}}(\mathbf{p})) = \mathbf{g}_{\mathcal{K}}^{-1}(\mathbf{g}_{\mathcal{K}}(\mathbf{p})) \quad (3.11)$$

for all possible values of  $\mathbf{p}$ . Global rigidity, as it comes out of the last definition, is a property which is stronger than rigidity. Moreover, in [132] it is shown that *redundant rigidity* is a necessary (but not sufficient) condition for global rigidity. A typical example to understand the difference between a rigid and a globally rigid graph is reported in Fig. 3.8.<sup>10</sup> Strictly speaking, it is important to highlight that rigidity (minimal or not) does not uniquely define the shape of a formation but it ensures that if the formation assumes a specific shape, it will not be able to deform from that shape smoothly. On the other hand, global rigidity ensures that, if the formation satisfies a set of distance constraints, the shape is univocally (globally) defined.

In order to have a better understanding of the definition of rigidity in (3.10) we can refer to Fig. 3.9 which depicts how the rigidity function works in the case of a rigid graph. Let us assume that we have a certain set  $\bar{\mathbf{p}}$  of positions of the agents. If we apply the function  $\mathbf{g}_{\mathcal{G}}$  to  $\bar{\mathbf{p}}$ , from the set of  $\mathbb{R}^{2|\mathcal{V}|}$  we will go to the set of  $\mathbb{R}^{|\mathcal{E}|}$ , specifically we will end up in  $\mathbf{g}_{\mathcal{G}}(\bar{\mathbf{p}})$ , that is, the value of the distances over the edges for the particular configuration  $\bar{\mathbf{p}}$ . Due to the definition (3.9) of the rigidity function this point consists of a stack of  $\|\bar{\mathbf{p}}_i - \bar{\mathbf{p}}_j\|^2$ . The definition (3.10) expresses the

<sup>10</sup>In accordance to the definition of global rigidity given above sometimes rigidity is referred to as *local rigidity* in order to highlight that it is a property *local* to the current positions of the agents. In this thesis we refer to it simply as *rigidity*.

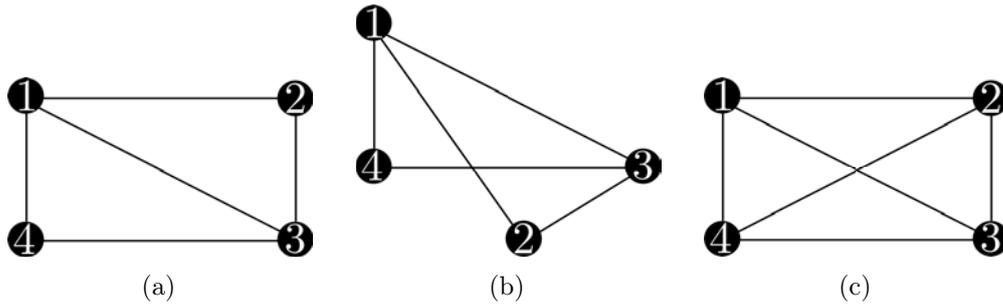


Figure 3.8 – Global rigidity: Fig. 3.8(a) is rigid but not globally rigid because the same distance constraints can be satisfied, e.g., also by the graph in Fig. 3.8(b). On the contrary, the graph in Fig. 3.8(c) is globally rigid. Note also that Fig. 3.8(a) cannot be continuously deformed into Fig. 3.8(b). Therefore, locally, the graphs in Figs. 3.8(a) and 3.8(b) are rigid graphs, although not in a global sense.

following concept. For a rigid graph, the set that one gets from the counter-image  $g_{\mathcal{G}}^{-1}(\bar{\mathbf{p}})$  (the shaded area) is the same set that one would get by replacing  $\mathcal{G}$  with  $\mathcal{K}$ , at least in a neighborhood  $U_{\bar{\mathbf{p}}}$  of  $\bar{\mathbf{p}}$ . On the other hand, in the case of global rigidity this holds for all  $\mathbf{p}$  (not just in a neighborhood of  $\bar{\mathbf{p}}$ ).

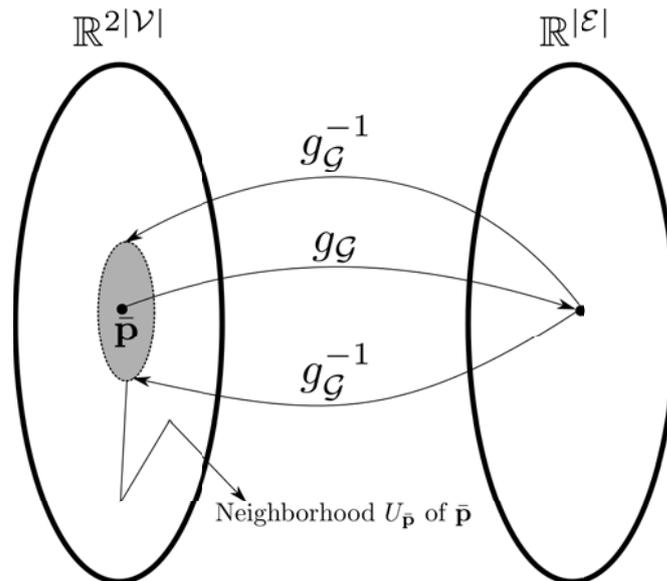


Figure 3.9 – The rigidity function  $g_{\mathcal{G}}$  in the case of a rigid framework

### 3.4.2 The distance-rigidity matrix

As pointed out above, an alternative approach to the combinatorial one provided by the Laman's theorem for the study the rigidity of a formation is the algebraic one based on the *Rigidity Matrix*.

Again, let us consider a framework  $(\mathcal{G}, \mathbf{p})$  where  $\mathbf{p}$  is the stack of all the positions

$\mathbf{p}_i \in \mathbb{R}^2$  of the agents. We can define a rigidity function, which is slightly different from (3.9)

$$\mathbf{g}_{\mathcal{G}}(\mathbf{p}) := \frac{1}{2} \text{stack}(\dots, \|\mathbf{p}_k - \mathbf{p}_j\|^2, \dots). \quad (3.12)$$

The rigidity matrix is defined as the Jacobian  $\mathbf{J}_{\mathbf{g}_{\mathcal{G}}}$  of the rigidity function with respect to the positions of the agents. This matrix is useful to study some concepts of graph rigidity which would be impossible to study only through combinatorial approaches (refer to [72, 116] and related works for more details). Thanks to the introduction of the rigidity matrix, and therefore within the algebraic approach, one can study the so-called infinitesimal rigidity. The idea of *infinitesimal rigidity* plays a central role in the design of decentralized formation controllers and localization algorithms, both in the case of distance constraints and in the one of bearing ones. Infinitesimal rigidity has the goal of studying the flexibility of a framework under instantaneous (infinitesimal) motions of the agents. It is discussed, with respect to distance constraints, in [116].

The idea of infinitesimal rigidity is to move infinitesimally the agents while keeping the rigidity function (3.12) constant up to the first order. Let  $\delta\mathbf{p}$  be an infinitesimal motion of the framework  $(\mathcal{G}, \mathbf{p})$ . Then the Taylor expansion of  $\mathbf{g}_{\mathcal{G}}$  about  $\mathbf{p}$  is

$$\mathbf{g}_{\mathcal{G}}(\mathbf{p} + \delta\mathbf{p}) = \mathbf{g}_{\mathcal{G}}(\mathbf{p}) + \mathbf{J}_{\mathbf{g}_{\mathcal{G}}} \delta\mathbf{p} + h.o.t.,$$

therefore the rigidity matrix can be also seen as the linear term in the Taylor expansion of the rigidity function. The rigidity function stays constant up to first order when  $\mathbf{J}_{\mathbf{g}_{\mathcal{G}}} \delta\mathbf{p}$  is null, which is when  $\delta\mathbf{p}$  is in the null-space of the rigidity matrix.

In [135] it is stated: a framework with  $N$  agents  $(\mathcal{G}, \mathbf{p})$  is *infinitesimally rigid* in the plane if  $\dim(\text{Ker}(\mathbf{J}_{\mathbf{g}_{\mathcal{G}}})) = 3$ , or equivalently

$$\text{rank}(\mathbf{J}_{\mathbf{g}_{\mathcal{G}}}) = 2N - 3 \quad (3.13)$$

Note that infinitesimal rigidity implies rigidity but the vice versa is not true. In order to understand better these concepts an example of infinitesimally and not infinitesimally rigid graph is reported in Fig. 3.10. Note that the two graphs are both rigid and globally rigid. In particular the second graph (Fig. 3.10(b)) is rigid because of the Laman's theorem. Indeed the graph has three edges which is equal to  $2|\mathcal{V}| - 3$ . The same graph is not rigid because if one would build the rigidity matrix, in the specific configuration of Fig. 3.10(b), it would be possible to check that the rank of the matrix is equal to two (instead of three) and therefore (3.13) would not be satisfied. Geometrically, this (point-wise) loss of rank is due to the fact that, at first order, agents 1 and 3 can infinitesimally move orthogonally to the edge (1, 3) without changing their distance, thus (again infinitesimally) deforming the shape of the formation.

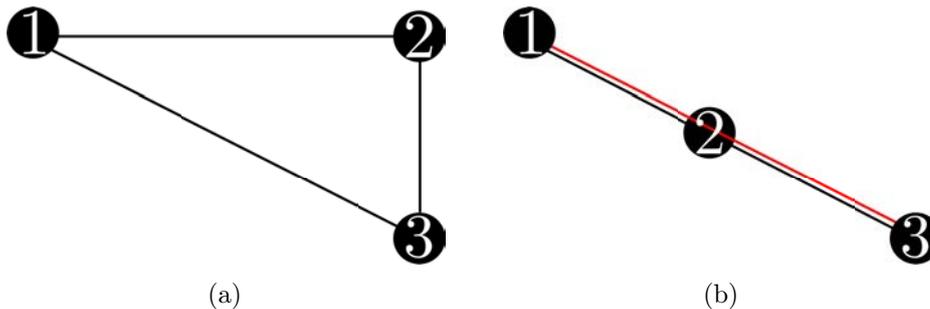


Figure 3.10 – Examples of infinitesimally rigid (Fig. 3.10(a)) and not infinitesimally rigid (Fig. 3.10(b)) graphs. Note that the red color is used to highlight that there is still an edge between the agents 1 and 3.

Finally, bringing together the definitions of infinitesimal and minimal rigidity one can define a *minimally infinitesimally rigid* framework as one which is both infinitesimally and minimally rigid.

In the case of distance constraints on the plane, if the rank of the rigidity matrix is  $2|\mathcal{V}| - 3$  (or equivalently  $2N - 3$ ) its null-space will be of dimension 3. This, in two dimensions for distance constraints, makes sense because it means that in the null-space of the rigidity matrix there are only three vectors which correspond to agent velocities which do not change the distance constraints. These vectors correspond to the planar translation (two vectors) and planar rotation (one vector) of the whole framework in two dimensions.

In [136] it is also specified that the result in (3.13) extends easily to graphs in three dimensions. In this case, it is possible to prove that a framework is rigid if and only if the rigidity matrix has a rank of  $3|\mathcal{V}| - 6$ . This means that the null-space has dimension 6 and its vectors correspond to three translations and three rotations of the whole framework in three dimensions.

We said several times that rigidity plays an essential role in solving the problem of formation control. Indeed, in [116], after defining the rigidity matrix they design a gradient-based control law based on this matrix. Thanks to the gradient-like structure of the controller and on the intrinsic decentralized structure of the rigidity matrix (each column corresponds to an agent and contains information which is relative only to the neighbor of the considered agent) the result is a controller which has a decentralized structure (i.e., the controller for each agent relies just on quantities dependent on itself or its neighbors). Furthermore, in [116] they prove that infinitesimal rigidity is a sufficient condition for local asymptotic stability of a gradient-based controller, for multi-vehicle systems, which uses only local information.<sup>11</sup>

<sup>11</sup>In [116], the definition of infinitesimal rigidity is the one coming from [135].

Note that the choice of a controller based on the rigidity matrix and which is gradient-based is not casual. Indeed, if instead of a gradient-based method, it would have been chosen, e.g., a Gauss-Newton approach the controller would end up being dependent not simply on the rigidity matrix  $\mathbf{J}_{g_G}$  (or more precisely on its transpose) but on the so-called pseudoinverse of the rigidity matrix which can be defined as  $\mathbf{J}_{g_G}^\dagger = \left(\mathbf{J}_{g_G}^\top \mathbf{J}_{g_G}\right)^{-1} \mathbf{J}_{g_G}^\top$ . Without going too much into the details, the inverse operation of a matrix in some way *mixes* the elements of the matrix to which is applied. Then, because the pseudoinverse uses the inverse operation, designing a controller built on this matrix would yield a loss of its decentralized structure. However, gradient-based methods have also some drawbacks. One of them is the speed of convergence which is one of the worse among the optimization methods. The discussion about this topic is not within the scope of this Thesis but interesting details can be found in [99], where they try to speed up a gradient-based method by choosing a better, decentralized, stepsize.

Thanks to the introduction of the rigidity matrix we can understand a difference between connectivity and infinitesimal rigidity. Connectivity and rigidity are both global properties of the whole group of agents to which the considered graph is associated. However, connectivity is associated to a graph while infinitesimal rigidity takes into account also the particular configuration of the robots. Note that our algorithms rely on the concept of infinitesimal rigidity (for bearing constraints).<sup>12</sup> This property becomes evident by looking at two rigidity matrices corresponding to a group of agents in two dimensions, always in the case of distance constraints. The matrix in (3.14) corresponds to a graph like the one in Fig. 3.11(a) which is not rigid and indeed it has a rank which is less than  $2|\mathcal{V}| - 3$ .

$$\begin{bmatrix} x_1 - x_2 & y_1 - y_2 & x_2 - x_1 & y_2 - y_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & x_2 - x_3 & y_2 - y_3 & x_3 - x_2 & y_3 - y_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & x_3 - x_4 & y_3 - y_4 & x_4 - x_3 & y_4 - y_3 \\ x_1 - x_4 & y_1 - y_4 & 0 & 0 & 0 & 0 & x_4 - x_1 & y_4 - y_1 \end{bmatrix}$$

(3.14)

---

<sup>12</sup>This can be summarized by saying that infinitesimal rigidity is not a *purely topological property* as connectivity.

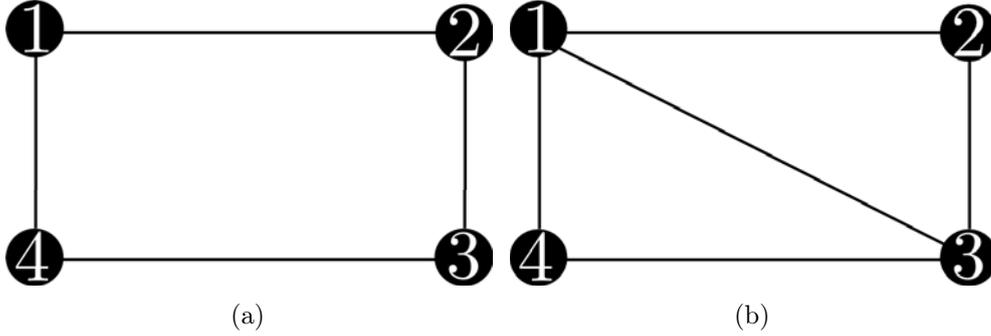


Figure 3.11 – Example of not-rigid (Fig. 3.11(a)) and rigid (Fig. 3.11(b)) graphs in the case of distance constraints in two dimensions for four agents.

Instead, the following matrix corresponds to a rigid graph like the one in Fig. 3.11(b)

$$\begin{bmatrix}
 x_1 - x_2 & y_1 - y_2 & x_2 - x_1 & y_2 - y_1 & 0 & 0 & 0 & 0 \\
 0 & 0 & x_2 - x_3 & y_2 - y_3 & x_3 - x_2 & y_3 - y_2 & 0 & 0 \\
 0 & 0 & 0 & 0 & x_3 - x_4 & y_3 - y_4 & x_4 - x_3 & y_4 - y_3 \\
 x_1 - x_4 & y_1 - y_4 & 0 & 0 & 0 & 0 & x_4 - x_1 & y_4 - y_1 \\
 x_1 - x_3 & y_1 - y_3 & 0 & 0 & x_3 - x_1 & y_3 - y_1 & 0 & 0 \\
 0 & 0 & x_2 - x_4 & y_2 - y_4 & 0 & 0 & x_4 - x_2 & y_4 - y_2
 \end{bmatrix}$$

(3.15)

The matrix in (3.15) seems to have a rank always greater or equal than  $2|\mathcal{V}| - 3$ . But, if we look closer it is possible to understand that some terms of (3.15) can become equal to zero even if the underlying graph contains the edge corresponding to that element. This happens, for example, when agent 1 has the same  $x$ -coordinate (or  $y$ -coordinate) of agent 2 and, in general, when two or more agents are collinear or coincident with each other. In this case the graph would be rigid but not infinitesimally rigid, as for the example in Fig. 3.10(b). This leads to the conclusion that infinitesimal rigidity is a stronger property than rigidity. Indeed, global rigidity can be seen as a sort of rigidity which ensures that the positions of the robot are nondegenerate (e.g., not all aligned).

In this section, we went through the primary definitions and results about the theory of rigidity focusing our attention on the case of distance constraints. We concentrated on distance constraints for two main reasons. The first one is a historical reason: distance constraints, because of their *analogy* with rigidity in mechanical systems, were the first ones to be tackled and investigated by the

multi-robot community. The second reason is that we simply believe that this case is more intuitive than the bearing one.

### 3.4.3 The case of bearing constraints

However, since some years there has been an interest also in applying the theory of rigidity to scenarios with bearing constraints [76, 100, 112, 121, 122, 137, 138]. This happened mainly because bearing measurements are often retrievable from cheaper and more accessible sensors, such as vision-based sensors (e.g., cameras) and angle-of-arrival sensor (e.g., wireless network devices<sup>13</sup>). Moreover, a formation based on angles can be scaled up easier than one based only on distances. A difference between distance-rigidity and bearing-rigidity is that the rigid body motions in the second case also include a contraction/expansion of the whole frame along with the usual roto-translation. Indeed, only with bearing measurements there is no control over the scale of the formation and therefore the size of the desired shape cannot be controlled. This is one of the reasons why Chapt. 7 deals with the scale estimation through bearing measurements and known agent ego-motion (body-frame linear/angular velocity).

Contrary to distance measurements (which are scalar quantities), bearing measurements are vector quantities. Therefore, one has to consider in which frame these measurements are expressed.

In the literature, two main cases have been considered: all bearing measurements expressed in a common (world) frame [76, 112], and all bearing measurements expressed in the local body frames of the agents [30, 31, 77, 140]. This Thesis, as stated several times, lies in the second group because this is the kind of measurement retrievable with a monocular camera. A third category could be represented by the works which analyze both cases, such as [122] in which there is an extension of a bearing-based formation control from two to arbitrary dimensions. In [122] there is also the study of some of the connections between distance and bearing rigidity showing that a framework in  $\mathbb{R}^2$  is infinitesimally bearing-rigid if and only if is also infinitesimally distance-rigid<sup>14</sup>.

In the case of a bearing measurement  $\beta_{ij}$  (between the agents  $i$  and  $j$ ) expressed with respect to a common reference frame, the vector  $\beta_{ij}$  is the 3D unit-norm vector expressed as

$$\beta_{ij} = \frac{\mathbf{p}_j - \mathbf{p}_i}{\|\mathbf{p}_j - \mathbf{p}_i\|} = \frac{\mathbf{p}_{ij}}{d_{ij}} \in \mathbb{S}^2, \quad (3.16)$$

<sup>13</sup>Indeed, the direction of arrival of a radio signal can be estimated through different kinds of antennas and used for radio source localization [139].

<sup>14</sup>This result cannot be generalized to  $\mathbb{R}^3$  or higher dimensions.

while in the second case of measurements expressed in the body-frames, the bearing is expressed as

$$\beta_{ij} = \mathbf{R}_i^T \frac{\mathbf{p}_j - \mathbf{p}_i}{\|\mathbf{p}_j - \mathbf{p}_i\|} = \mathbf{R}_i^T \frac{\mathbf{p}_{ij}}{d_{ij}} \in \mathbb{S}^2, \quad (3.17)$$

where  $\mathbf{p}_{ij} = \mathbf{p}_j - \mathbf{p}_i$ ,  $d_{ij} = \|\mathbf{p}_j - \mathbf{p}_i\|$  and  $\mathbf{R}_i$  is a rotation matrix that rotates a vector from the world frame to the body frame. Regarding the second case, more details are given in Sect. 4.2 while in this section we will continue to focus on the first case where the considered framework for bearing rigidity is still the pair  $(\mathcal{G}, \mathbf{p})$ .

Note that bearing rigidity is often addressed as *parallel* rigidity. This is due to an alternative but equivalent definition of rigidity given by introducing the parallelism between vectors (as done in [141]). Let us define the following orthogonal projector matrix for any non-null vector  $\mathbf{x} \in \mathbb{R}^d$  ( $d \geq 2$ ) as

$$\mathbf{P}_x := \mathbf{I}_d - \frac{\mathbf{x} \mathbf{x}^T}{\|\mathbf{x}\| \|\mathbf{x}\|}. \quad (3.18)$$

$\mathbf{P}_x$  projects the vector  $\mathbf{x}$  onto the orthogonal complement of  $\mathbf{x}$ . Thanks to the definition of this projector matrix we can define two vectors  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$  as *parallel* to each other if and only if  $\mathbf{P}_x \mathbf{y} = \mathbf{0}$  or  $\mathbf{P}_y \mathbf{x} = \mathbf{0}$ .

Two frameworks  $(\mathcal{G}, \mathbf{p})$  and  $(\mathcal{G}, \mathbf{p}')$  are *bearing equivalent* if

$$\mathbf{P}_{(\mathbf{p}_i - \mathbf{p}_j)} (\mathbf{p}'_i - \mathbf{p}'_j) = \mathbf{0} \quad \forall e_i \in \mathcal{E}.$$

Two frameworks  $(\mathcal{G}, \mathbf{p})$  and  $(\mathcal{G}, \mathbf{p}')$  are *bearing congruent* if

$$\mathbf{P}_{(\mathbf{p}_i - \mathbf{p}_j)} (\mathbf{p}'_i - \mathbf{p}'_j) = \mathbf{0} \quad \forall e_i \in \mathcal{K}.$$

Therefore, it is possible to give an alternative definition, to the ones given in Sect. 3.4.1, of rigidity and global rigidity.

A framework  $(\mathcal{G}, \mathbf{p})$  is *bearing/parallel rigid* if there exists a constant  $\epsilon > 0$  such that any framework  $(\mathcal{G}, \mathbf{p}')$  that is bearing equivalent to  $(\mathcal{G}, \mathbf{p})$ , and satisfies  $\|\mathbf{p} - \mathbf{p}'\| < \epsilon$  is also bearing congruent to  $(\mathcal{G}, \mathbf{p})$ .

A framework  $(\mathcal{G}, \mathbf{p})$  is *globally bearing/parallel rigid* if every framework which is bearing equivalent to  $(\mathcal{G}, \mathbf{p})$  is also bearing congruent to  $(\mathcal{G}, \mathbf{p})$ .

### 3.4.4 Similarity between the rigidity matrix and the Jacobian of robotic manipulators

To make the understanding of the rigidity matrix easier, in this section we want to point out to the reader a useful similarity between the rigidity matrix and the *well-known* Jacobian in the case of robotic manipulators. A typical task in robotics consists in reaching a certain desired position  $\mathbf{p}_d$  of the end effector. Therefore

the classical task could be written as  $\mathbf{p}(\mathbf{q}) \rightarrow \mathbf{p}_d$ . Where  $\mathbf{p}$  is the position of the end-effector of the considered robot and  $\mathbf{q}$  is a vector which contains all the joints variables. In this case, one would differentiate the  $\mathbf{p}$  and have the *well-known* equation

$$\dot{\mathbf{p}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \quad (3.19)$$

where the matrix  $\mathbf{J}$  is the *Jacobian* associated to the manipulator and it contains the dynamics of the robot.

In our case, the task is to bring the robots from a certain configuration, with certain bearings  $\beta_{ij}^*$ , to a desired one which will have  $\beta_{ij}^d$ . In the same way, in the case of distances, the task would be to steer the agents from a configuration with some distances  $d_{ij}^*$  to a desired configuration with  $d_{ij}^d$ . The dual equation of (3.19) in our case is, using the same notation used so far:

$$\mathbf{g}_{\mathcal{G}} = \mathbf{J}_{\mathbf{g}_{\mathcal{G}}}\dot{\mathbf{p}} \quad (3.20)$$

where  $\mathbf{J}_{\mathbf{g}_{\mathcal{G}}}$  is the Jacobian of the function  $\mathbf{g}_{\mathcal{G}}$  and therefore the rigidity matrix.

In this chapter, the theory of rigidity has been addressed to give the reader an introduction to this vast topic. In the next Chapt. 4 we will specialize the theory of rigidity for the case of bearing rigidity in  $\mathbb{R}^3 \times \mathbb{S}^1$  (which is the case considered in this Thesis) also giving an explicit expression of the bearing rigidity matrix which has a fundamental role in the whole Thesis.

**Part II**

**Contributions**



---

# Main modeling assumptions

## Contents

---

4.1	Introduction	49
4.2	Agent model	49
4.3	Directed bearing rigidity in $\mathbb{R}^3 \times \mathbb{S}^1$	52

---

## 4.1 Introduction

IN this chapter we bring together the main modeling assumptions and definitions which are common to the chapters of this part of the Thesis. More specifically, Chapt. 5 and 6 share the same modeling assumptions listed in this chapter while in Chapt. 7 there are some differences which are specifically addressed and explained in details.

## 4.2 Agent model

As we said several times, in this Thesis we want to be able to control a group of  $N$  quadrotor UAVs in an unstructured environment only relying on their onboard capabilities. To fulfill this task, the robots are required to be able to localize themselves with respect to other agents of the formation. The literature shows how this is possible relying on different sensing capabilities and on the capability of communication among the agents (e.g., by exchanging data over a radio communication channel). From a theoretical point of view, the most important aspect of these different sensing modalities is the quantity of information they provide. For instance, the sensors could provide estimates of distance (e.g., from wireless signal strength), bearing directions (e.g., with monocular cameras), translations (e.g., with stereo or depth cameras), or both rotations and translations (e.g., using cameras and two-view

Structure from Motion [142, 143]). In this Thesis, we assume that each agent can measure the direction (but not the distance) of a subset of neighboring agents in its reference frame<sup>1</sup>. This setup is arguably the most practical with today's most popular hardware, which, due to weight, cost, and power consumption considerations, is usually limited to an Inertial Measurement Unit (IMU) and a monocular camera [144, 145].

Let  $\mathcal{W} : \{\mathbf{O}_{\mathcal{W}}, \mathbf{X}_{\mathcal{W}}, \mathbf{Y}_{\mathcal{W}}, \mathbf{Z}_{\mathcal{W}}\}$  represent a world frame with  $\mathbf{Z}_{\mathcal{W}}$  aligned with the vertical (gravity) direction and  $\mathcal{A}_i : \{\mathbf{O}_{\mathcal{A}_i}, \mathbf{X}_{\mathcal{A}_i}, \mathbf{Y}_{\mathcal{A}_i}, \mathbf{Z}_{\mathcal{A}_i}\}$  the  $i$ -th body frame attached to each quadrotor UAV. Following the modeling assumptions of [27, 30, 31, 77, 146], we consider a group of  $N$  'velocity-controlled' quadrotor UAVs with agent dynamics

$$\begin{pmatrix} \dot{\mathbf{p}}_i \\ \dot{\psi}_i \end{pmatrix} = \begin{pmatrix} \mathbf{R}_i & 0 \\ \mathbf{0} & 1 \end{pmatrix} \begin{pmatrix} \mathbf{u}_i \\ w_i \end{pmatrix} \quad (4.1)$$

where  $\mathbf{p}_i \in \mathbb{R}^3$  is the quadrotor 3D position in  $\mathcal{W}$ ,  $\psi_i \in \mathbb{S}^1$  the yaw angle about  $\mathbf{Z}_{\mathcal{W}}$ , and  $\mathbf{R}_i = \mathbf{R}_z(\psi_i) \in \text{SO}(3)$  is the canonical rotation around the world  $\mathbf{z}$ -axis. The quantities  $\mathbf{u}_i \in \mathbb{R}^3$  and  $w_i \in \mathbb{R}$  represent the *body-frame* linear velocity and yaw rate which are assumed to be known and controllable. The configuration of the  $i$ -th quadrotor is then denoted with  $\mathbf{q}_i = (\mathbf{p}_i, \psi_i) \in \mathbb{R}^3 \times \mathbb{S}^1$ , while  $\mathbf{q} = (\mathbf{p}, \boldsymbol{\psi}) \in (\mathbb{R}^3 \times \mathbb{S}^1)^N$  is the configuration of the whole formation, with  $\mathbf{p} = (\mathbf{p}_1, \dots, \mathbf{p}_N) \in \mathbb{R}^{3N}$  and  $\boldsymbol{\psi} = (\psi_1, \dots, \psi_N) \in \mathbb{S}^N$ . We stress that, as in [27, 30, 31, 77, 146], the (absolute) yaw angle  $\psi_i$  is not considered as an available quantity to the  $i$ -th quadrotor. Indeed, additional sensors, such as compasses, would be needed for obtaining a consistent yaw measurement for all the UAVs of the group with, however, a typically limited reliability (e.g., compasses would fail to operate soundly indoor or close to strong magnetic fields). As a consequence, the  $N$  quadrotors are *not* assumed to share, as a group, a common (global) reference frame where to express local measurements and control inputs. The reader is referred to Figs. 4.1 and 4.2 for a better understanding of the introduced quantities. In the first one, for simplicity, the origin of the body frame of the quadrotor  $\mathbf{O}_{\mathcal{A}_i}$  coincides with the origin of the world frame  $\mathbf{O}_{\mathcal{W}}$  while in the second one a group of three quadrotor UAVs is illustrated.

Each quadrotor is also assumed equipped with an onboard calibrated camera that allows an agent  $i$  to measure the *relative bearing vector* with respect to an agent  $j$  in visibility, i.e., the 3D unit-norm vector

$$\boldsymbol{\beta}_{ij} = \mathbf{R}_i^T \frac{\mathbf{p}_j - \mathbf{p}_i}{\|\mathbf{p}_j - \mathbf{p}_i\|} = \mathbf{R}_i^T \frac{\mathbf{p}_{ij}}{d_{ij}} \in \mathbb{S}^2, \quad (4.2)$$

where  $\mathbf{p}_{ij} = \mathbf{p}_j - \mathbf{p}_i$ ,  $d_{ij} = \|\mathbf{p}_j - \mathbf{p}_i\|$  and  $\mathbb{S}^2$  is the *unit sphere*, i.e. the space of 3-D

<sup>1</sup>We refer to this reference frame as the *body-frame of the robot*

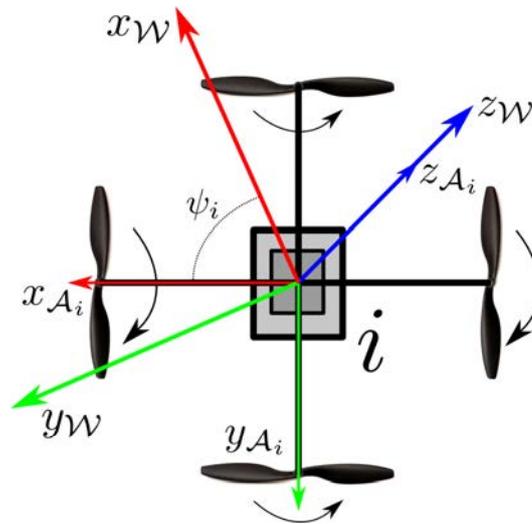


Figure 4.1 – A quadrotor UAV with the origin of its body frame  $\mathcal{A}_i$  which is coincident with the origin of the world reference frame  $\mathcal{W}$

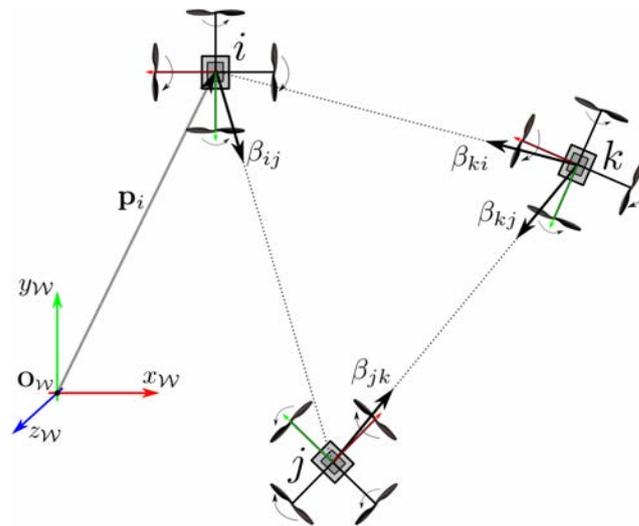


Figure 4.2 – Group of three quadrotor UAVs with the corresponding body reference frames  $\mathcal{A}_i, \mathcal{A}_j, \mathcal{A}_k$  and the world frame  $\mathcal{W}$

unit-norm vectors defined as

$$\mathbb{S}^2 = \{\mathbf{v} : \mathbf{v} \in \mathbb{R}^3, \|\mathbf{v}\| = 1\} \quad (4.3)$$

This agent relative bearing  $\beta_{ij}$  can be retrieved by ‘derotating’ the actual bearing measurement among quadrotors  $i$  and  $j$  (e.g., from an onboard camera) by the roll and pitch angles which can be typically measured by exploiting the built-in IMU, see also [77]. Indeed, by exploiting the known roll/pitch angles  $(\phi_{\mathcal{A}_i}, \theta_{\mathcal{A}_i})$  the unit vector  $\beta_{ij}$  can be obtained from the measured  $\beta_{\mathcal{A}_i, \mathcal{A}_j}$  as  $\beta_{ij} = \mathbf{R}_y(\theta_{\mathcal{A}_i})\mathbf{R}_x(\phi_{\mathcal{A}_i})\beta_{\mathcal{A}_i, \mathcal{A}_j}$  and can, thus, be treated as a quantity available to agent  $i$ .

The control strategies discussed in the next Chapt. 5 and 6 are based on model (4.1) and assume that each agent  $i$  can measure the (body-frame) bearing vector  $\beta_{ij}$  with respect to other neighboring agents together with its own (body-frame) velocity commands  $(\mathbf{u}_i, w_i)$  [22, 147]. Instead, Chapt. 7 considers a more general model which takes into account a rotation matrix which is a function of the full orientation of the agent and not only of the yaw angle  $\psi_i$  described above. Moreover, with special focus on the quadrotor platform, we assume that the UAV is able (through a low-level controller such as, for instance, the one proposed in [148]) to track a smooth reference trajectory  $(\mathbf{p}_i(t), w_i(t))$  in the four-dimensional space  $\mathbb{R}^3 \times \mathbb{S}^1$ .

### 4.3 Directed bearing rigidity in $\mathbb{R}^3 \times \mathbb{S}^1$

In this section there is a specialization of the theory of rigidity, explained in Sect. 3.4, for the case of bearing constraints. We also give relevant definitions and properties of directed bearing formations and bearing rigidity in  $\mathbb{R}^3 \times \mathbb{S}^1$  which are useful in the next chapters. Many of the introduced concepts are an extension of the SE(2) case treated in [27, 30, 31] to which the interested reader is referred for full details.

Let  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  be a *directed* graph, where  $\mathcal{V} = \{1 \dots N\}$  is the vertex set and  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$  the edge set [149]. Presence of an edge  $e_k = (i, j)$  in  $\mathcal{E}$  represents the possibility for agent  $i$  to measure the relative bearing  $\beta_{ij}$  (4.2) to agent  $j$ . Graph  $\mathcal{G}$  is designed as *directed* as we do not require, in general, reciprocity of the relative bearing measurements. (i.e., agent  $i$  may measure agent  $j$  but *not* be measured by agent  $j$ ). This way, the typical visibility constraints of onboard cameras due to, e.g., limited field of view or occluded line-of-sight, can be directly accommodated at the modeling stage. We assume, however, that agents  $i$  and  $j$  can communicate if either  $(i, j) \in \mathcal{E}$  or  $(j, i) \in \mathcal{E}$  (i.e., the communication graph is taken as the *undirected* counterpart of the directed sensing graph  $\mathcal{G}$ ).

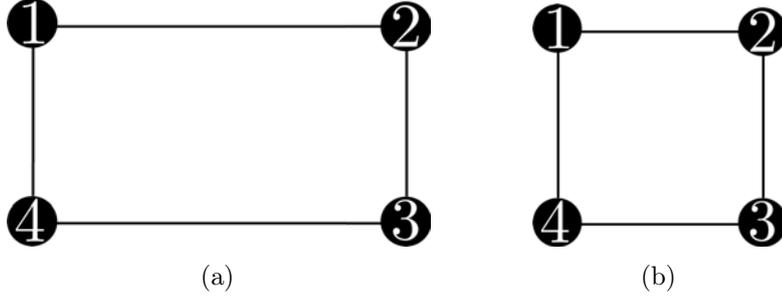


Figure 4.3 – Examples of two frameworks which are bearing equivalent but not bearing congruent

We let  $\mathcal{N}_i = \{j \in \mathcal{V} \mid (i, j) \in \mathcal{E}\}$  represent the usual set of neighbors of an agent  $i$ , and  $\mathcal{O}_i = \{j \in \mathcal{V} \mid (j, i) \in \mathcal{E}\}$  the set of agents  $j$  for which  $i$  is a neighbor<sup>2</sup>. We also let  $\mathbf{1}_N$  and  $\mathbf{I}_N$  represent a vector of all ones and the identity matrix of dimension  $N$ , respectively.

A slightly different way of defining bearing rigidity, with respect to the definitions given in Sect. 3.4.3, is the one which involves the concepts of framework congruency and equivalence. In this case, the difference is that a framework (or also formation) [30, 31, 150] is the pair  $(\mathcal{G}, \mathbf{q})$  where  $\mathbf{q}$  follows the definition given in Sect. 4.2.

Two frameworks  $(\mathcal{G}, \mathbf{q})$  and  $(\mathcal{G}, \mathbf{q}')$  are denoted *bearing equivalent* if  $\beta_{ij}(\mathbf{q}) = \beta_{ij}(\mathbf{q}') \forall (i, j) \in \mathcal{E}$  and *bearing congruent* if  $\beta_{ij}(\mathbf{q}) = \beta_{ij}(\mathbf{q}') \forall i, j \in \mathcal{V}, i \neq j$ . A framework  $(\mathcal{G}, \mathbf{q})$  is defined *bearing rigid* (or simply *rigid* in the following) if there exists a neighborhood  $\mathcal{U}$  of  $\mathbf{q}$  such that any framework  $(\mathcal{G}, \mathbf{q}'), \mathbf{q}' \in \mathcal{U}$ , that is *bearing equivalent* to  $(\mathcal{G}, \mathbf{q})$  is also *bearing congruent* to  $(\mathcal{G}, \mathbf{q})$ . Also in this case, logically, bearing congruency implies bearing equivalence, but the opposite is not true (see Fig. 4.3). By extension, a framework is instead defined *globally bearing rigid* if  $\mathcal{U} = (\mathbb{R}^3 \times \mathbb{S}^1)^N$ , i.e., if *all* frameworks which are bearing equivalent to  $(\mathcal{G}, \mathbf{q})$  are also bearing congruent to  $(\mathcal{G}, \mathbf{q})$ . A non-rigid framework is also termed *roto-flexible*, and a framework  $(\mathcal{G}, \mathbf{q})$  is said to be *minimally rigid* if  $(\mathcal{G}, \mathbf{q})$  is rigid and the removal of any edge yields a roto-flexible framework.

As for the rigidity in the case of distance constraints, the notion of bearing rigidity can also be characterized from an infinitesimal perspective by introducing the so-called *directed bearing function* and corresponding *directed bearing rigidity matrix*. The directed bearing function (bearing function from now on) associated to a framework  $(\mathcal{G}, \mathbf{q})$  is the map  $\beta_{\mathcal{G}}(\mathbf{q}) : (\mathbb{R}^3 \times \mathbb{S}^1)^N \rightarrow (\mathbb{S}^2)^{|\mathcal{E}|}$

$$\beta_{\mathcal{G}}(\mathbf{q}) = \text{stack} \left( \beta_{e_1}, \dots, \beta_{e_{|\mathcal{E}|}} \right) \quad (4.4)$$

<sup>2</sup>This distinction between sets  $\mathcal{N}_i$  and  $\mathcal{O}_i$ , needed because of the *directed* nature of graph  $\mathcal{G}$ , will play an important role in Sects. 6.3.2 and 6.3.3.

where the notation  $e_i \in \mathcal{E}$  is used to represent a directed edge in the graph  $\mathcal{G}$  according to any chosen labeling. The world-frame (directed) bearing rigidity matrix (BRM) is the Jacobian of the bearing function with respect to the agent configuration  $\mathbf{q}$ , that is, the matrix

$$\mathbf{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q}) = \frac{\partial \beta_{\mathcal{G}}(\mathbf{q})}{\partial \mathbf{q}} \in \mathbb{R}^{3|\mathcal{E}| \times 4N}. \quad (4.5)$$

Let  $\mathcal{N}(\cdot)$  represent the span of the null-space of a matrix. A framework  $(\mathcal{G}, \mathbf{q})$  is said to be *infinitesimally bearing rigid* at some point  $\mathbf{q}$  if  $\mathcal{N}(\mathbf{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q})) = \mathcal{N}(\mathbf{B}_{\mathcal{K}_N}^{\mathcal{W}}(\mathbf{q}))$ , with  $\mathcal{K}_N$  being the complete directed graph. Otherwise a framework is said to be *infinitesimally roto-flexible*. Since it can be shown that  $\dim \mathcal{N}(\mathbf{B}_{\mathcal{K}_N}^{\mathcal{W}}(\mathbf{q})) = 5$  (see, e.g., [27, 77]), it follows that a framework  $(\mathcal{G}, \mathbf{q})$  in  $\mathbb{R}^3 \times \mathbb{S}^1$  is infinitesimally rigid if and only if the following is satisfied

$$\text{rank}(\mathbf{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q})) = 4N - 5. \quad (4.6)$$

For infinitesimally rigid frameworks in  $\mathbb{R}^3 \times \mathbb{S}^1$ , the 5-dimensional null-space of the bearing rigidity matrix (BRM) is also well-understood: it corresponds to the three rigid-body translations, a dilation relative to a reference point, and a coordinated rotation about a vertical axis passing through a reference point [30, 31, 77]. Equivalently, we can say that the 5-dimensional null-space of matrix  $\mathbf{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q})$  is spanned by those velocities  $\dot{\mathbf{q}}$  that do not change the bearing function  $\beta_{\mathcal{G}}(\mathbf{q})$  (i.e., the value of the measured bearing vectors).

The notion of bearing infinitesimally rigidity and, in particular, the properties of the bearing rigidity matrix and its null-space play a central role in the design of *decentralized* bearing formation controller and localization schemes. A role analogous to the (more common case of) distance-constrained infinitesimal rigidity and distance-constrained rigidity matrix [99, 116, 118, 119, 151]. Note that in [122] there is a theorem that expresses a relationship between the three kinds of bearing rigidity which is summarized in Fig. 4.4.

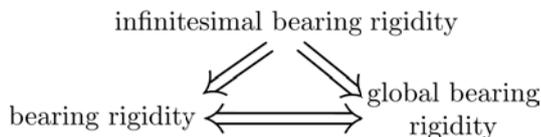


Figure 4.4 – Relationship between the three kinds of bearing rigidity introduced in this section, from [122]

Extending the results of [31] to the case of frameworks in  $\mathbb{R}^3 \times \mathbb{S}^1$ , the  $k$ -th row block of the world-frame bearing rigidity matrix  $\mathbf{B}_{\mathcal{G}}^{\mathcal{W}}$  in (4.5) associated to edge

$e_k = (i, j)$  has expression

$$\frac{\partial \beta_{ij}}{\partial \mathbf{q}} = \begin{bmatrix} -\mathbf{0} & -\underbrace{\frac{\mathbf{P}_{ij} \mathbf{R}_i^T}{d_{ij}}}_i & -\mathbf{0} & -\underbrace{\frac{\mathbf{P}_{ij} \mathbf{R}_i^T}{d_{ij}}}_j & -\mathbf{0} & \dots & \underbrace{-\mathbf{S} \beta_{ij}}_{3N+i} & -\mathbf{0} \end{bmatrix} \in \mathbb{R}^{3 \times 4N}. \quad (4.7)$$

Here,  $\mathbf{P}_{ij} = \mathbf{I}_3 - \beta_{ij} \beta_{ij}^T$  is the orthogonal projector onto the orthogonal complement of  $\beta_{ij}$ , and  $\mathbf{S} = [[0 \ 0 \ 1]^T]_{\times}$  where  $[\cdot]_{\times}$  indicates the usual skew-symmetric matrix operator.

We then note that the bearing rigidity matrix  $\mathcal{B}_{\mathcal{G}}^{\mathcal{W}}$  is a function of interdistances, relative bearings, and *absolute* yaw rotations. Indeed, the bearing rigidity matrix relates changes in the bearing function  $\beta_{\mathcal{G}}$  to the *world-frame* velocities  $\dot{\mathbf{q}} = (\dot{\mathbf{p}}, \dot{\boldsymbol{\psi}})$  of the framework

$$\dot{\beta}_{\mathcal{G}} = \mathcal{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q}) \begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\boldsymbol{\psi}} \end{bmatrix}. \quad (4.8)$$

The world-frame rigidity matrix relates variations in the bearing function to the *world-frame* agent velocities  $(\dot{\mathbf{p}}, \dot{\boldsymbol{\psi}})$ . Exploiting (4.1), one can also define a *body-frame* bearing rigidity matrix  $\mathcal{B}_{\mathcal{G}}(\mathbf{q})$  as

$$\dot{\beta}_{\mathcal{G}} = \mathcal{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q}) \begin{bmatrix} \text{diag}(\mathbf{R}_i) & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_N \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{w} \end{bmatrix} = \mathcal{B}_{\mathcal{G}}^{\mathcal{W}} \mathbf{T}(\boldsymbol{\psi}) \begin{bmatrix} \mathbf{u} \\ \mathbf{w} \end{bmatrix} = \mathcal{B}_{\mathcal{G}}(\mathbf{q}) \begin{bmatrix} \mathbf{u} \\ \mathbf{w} \end{bmatrix}, \quad (4.9)$$

which explicits the dependance on the body-frame velocity inputs  $\mathbf{u} = \text{stack}(\dots \mathbf{u}_i \dots) \in \mathbb{R}^{3N}$  and  $\mathbf{w} = \text{stack}(\dots w_i \dots) \in \mathbb{R}^N$ , respectively the linear and angular velocities. The  $(3 \times 4N)$   $k$ -th row block of  $\mathcal{B}_{\mathcal{G}}(\mathbf{q})$  associated to an edge  $e_k = (i, j)$  has expression

$$\begin{bmatrix} -\mathbf{0} & -\underbrace{\frac{\mathbf{P}_{ij}}{d_{ij}}}_i & -\mathbf{0} & -\underbrace{\frac{\mathbf{P}_{ij} {}^i \mathbf{R}_j}{d_{ij}}}_j & -\mathbf{0} & \dots & \underbrace{-\mathbf{S} \beta_{ij}}_{3N+i} & -\mathbf{0} \end{bmatrix}, \quad (4.10)$$

where  ${}^i \mathbf{R}_j = \mathbf{R}_z(\psi_j - \psi_i) = \mathbf{R}_z(\psi_{ij})$ .

It is worth noting that the body-frame bearing rigidity (4.10) matrix is, again, a function of measured bearings  $\beta_{ij}$  and interdistances  $d_{ij}$ . However, contrary to the previous case, it is not a function of absolute yaw rotations  $\psi_i$ , but, instead, of relative orientations  $\psi_{ij} = \psi_j - \psi_i$  among neighboring agents. This fact will be important for the next developments.

Note that the same considerations we did for the world-frame bearing rigidity matrix about infinitesimal rigidity extend to the *body-frame* bearing rigidity matrix since  $\text{rank}(\mathcal{B}_{\mathcal{G}}(\mathbf{q})) = \text{rank}(\mathcal{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q}))$ , being  $\mathbf{T}(\boldsymbol{\psi})$  in (4.9) a square non-singular matrix.

Now we are finally ready to present to the reader the main contributions of this Thesis which are divided into three chapters:

- Chapt. 5 presents a formation control and localization algorithm for a group of  $N$  quadrotor UAVs based on the theory of rigidity.
- Chapt. 6 presents a solution to the problem of *maintaining rigidity* in spite of sensing limitations like (i) limited field of view of the onboard cameras, (ii) limited range of the onboard cameras, (iii) possibility of occlusions between the agents of the formation during motion.
- Chapt. 7 presents a nonlinear observability analysis of the system composed of  $N$  quadrotor UAVs. The observability analysis is the first step towards the design of an Extended Kalman Filter which can also estimate the scale of the formation (and therefore the real distances between the robots) using only bearing measurements and the linear/angular body-frame velocities of the robots.

# Formation control and localization

## in $\mathbb{R}^3 \times \mathbb{S}^1$

### Contents

5.1	Introduction . . . . .	58
5.1.1	Chapter overview . . . . .	60
5.2	Decentralized formation control . . . . .	60
5.2.1	Rigidity-based control of bearing frameworks in $\mathbb{R}^3 \times \mathbb{S}^1$ . . . . .	61
5.2.2	Rigidity-based localization of time-varying bearing frameworks in $\mathbb{R}^3 \times \mathbb{S}^1$ . . . . .	63
5.2.3	Coordinated motions in the null-space of the bearing rigidity matrix . . . . .	65
5.2.4	Discussion . . . . .	66
5.3	Simulation results . . . . .	68
5.4	Experimental results . . . . .	69
5.5	Conclusions . . . . .	72

IN this chapter we consider the problem of controlling a formation of quadrotor UAVs equipped with onboard cameras able to measure relative bearings in their local body frames with respect to neighboring UAVs. The control goal of this work is twofold: (i) steering the agent group towards a formation defined in terms of desired bearings, and (ii) actuating the group motions in the ‘null-space’ of the current bearing formation. The proposed control strategy relies on an extension of the rigidity theory to the case of *directed* bearing frameworks in  $\mathbb{R}^3 \times \mathbb{S}^1$  (see Sect. 4.3). This extension allows to devise a *decentralized* bearing controller which does not need presence of a common reference frame or of reciprocal bearing measurements for the agents.

## 5.1 Introduction

As stated in the previous chapters, the problem of formation control of multiple mobile robots has been extensively studied over the last decade in the robotics and control communities. The basic goal of most formation controllers is to coordinate a robot team in order to achieve some desired spatial arrangement. Most formation control schemes differ in their assumptions about, for example:

- the robot mobility: planar robots, aerial vehicles, possible presence of non-holonomic constraints or underactuation
- the employed sensing technology: local measurements from onboard sensors such as cameras, lidars, sonars, radio transmitters/receivers, or absolute/global measurement from centralized facilities as motion capture systems (mainly for indoor applications) and GNSS (for outdoor applications).
- the overall architecture: centralized versus decentralized coordination schemes.

A challenging scenario that still motivates considerable research efforts is that of *decentralized* formation control of mobile robots based only on local sensing, in which the robots are assumed to only be able to obtain *relative measurements* with respect to other robots in the group. Typical examples of relative sensing include range sensors (such as lidars and ultrasound) for retrieving inter-robot distances, or bearing sensors (such as cameras) for obtaining the bearing angle to other robots within visibility. The use of local sensing also entails the possible lack of a *common reference frame* to which all the individually collected measurements and control actions can be expressed, a limitation that poses additional challenges for solving the formation control problem [67, 152, 153]. As we already mentioned, these scenarios are motivated by the goal of deploying highly autonomous multi-robot teams in ‘non-trivial’ environments (e.g., inside buildings, underwater, underground, or even in deep space). Usually, in these scenarios centralized sensing facilities, such as GNSS, are not available, and the robots can only rely on their ‘local skills’ (local sensing and computing power, and local communication with neighboring agents).

In Sect. 3.4 we explained that, in all these cases, the correct theoretical framework for analyzing and controlling the geometrical properties of robot formations defined in terms of relative measurements has proven to be the theory of *formation rigidity* [72] described already in Chapt. 3.

Cameras are a widespread sensing technology which is often found onboard quadrotor UAVs mainly because of their low weight and limited power consumption [145]. It is then interesting to study how to design suitable vision-based

cooperative control strategies to be exploited in more complex applications such as exploration, coverage, surveillance or target tracking.

Nevertheless, when addressing visual-based formation control in GNSS-denied environments, the assumptions of a *common reference frame* and of an *undirected topology* for the measurement graph can be hard to realize. For instance, the requirement of keeping constant mutual visibility among *all* robot pairs can easily become unfeasible because of the limited camera field of view, and likewise for the possible presence of a common reference frame shared by a group of robots that has only access to local relative measurements. It is then essential to investigate possible bearing formation control strategies that can relax as much as possible these assumptions.

In this respect, to the best of our knowledge, the only decentralized bearing formation controller that does not require the presence of a common reference frame *and* of an undirected sensing topology has been proposed in [77] by exploiting the (body-frame) measured bearings and a *single* distance measurement among an arbitrary pair of robots. However, the machinery presented in [77] requires a very special structure for the chosen measurement graph (the one in Fig. 5.1(a)) which, among others, must contain two special agents able to measure and to be measured by any other agent in the group. While instrumental, this choice is also unnecessarily overconstraining since *any* bearing rigid topology (with, in general, far less constraining requirements than the ones exploited in [77]), for example the graphs in Figs. 5.1(b) and 5.1(c) would still allow solving a bearing formation control problem.

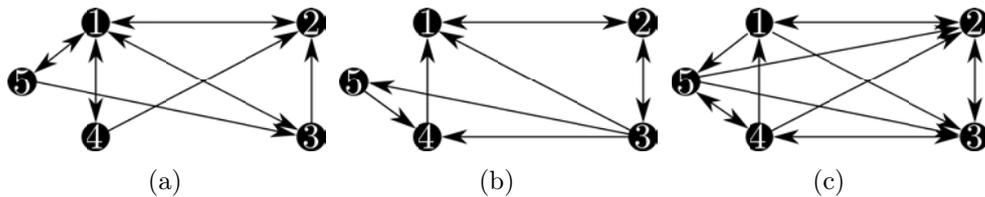


Figure 5.1 – Three examples of rigid graphs with 5 agents. Fig. 5.1(a) is the *overconstrained* graph used in [77]. Fig. 5.1(b) is another example of rigid graph which does not require all the constraints of the previous one and Fig. 5.1(c) is yet another rigid graph chosen randomly.

Our contribution in this context has been the generalization of the ideas of [77]. Indeed, we proposed a *fully decentralized* bearing formation controller that only requires the presence of a generic (directed) bearing rigid topology. Furthermore, as in [77], the proposed control strategy is also complemented with the possibility of steering the quadrotor group along all the bearing-preserving motion directions: these can be shown to consist of a collective translation, an expansion with respect to a reference point, and a coordinated rotation relative to a vertical reference axis.

This possibility is particularly useful when, for instance, we would need to collectively steer the quadrotor group for navigation or exploration purposes while maintaining the desired bearing formation optimized for the task at hand.

The two goals discussed above of bearing formation stabilization and group collective steering are here achieved under a minimal number of assumptions compared to the existing literature. In particular:

- the quadrotors are only assumed able to collect bearing measurements (through onboard cameras) and to impose motion commands in their (local) body-frames (which do not need to be aligned or ‘coordinated’ in some special ways),
- the bearing measurements are *not* necessarily required to be reciprocal, and the resulting (directed) sensing topology has no special constraints (apart from the ‘necessary condition’ of yielding a rigid bearing formation),
- a *single* (but arbitrary) quadrotor pair is additionally assumed able to measure its inter-distance (needed to retrieve the correct scale of formation which would be, otherwise, unobservable without employing estimation strategies such as those described in Chapt. 7 or in [146]). Note that, besides the intrinsic importance of knowing the scale of the formation, this quantity is then needed for correctly implementing the coordinated rotation in the null-space of the bearing rigidity matrix (BRM).

Finally, we also provide an experimental validation of the overall approach employing a group of four quadrotor UAVs and simulations with the help of MATLAB/SIMULINK and the V-REP simulator.

### 5.1.1 Chapter overview

The rest of the chapter is organized as follows: Sect. 5.2 illustrates the proposed bearing formation control strategy for robots evolving in  $\mathbb{R}^3 \times \mathbb{S}^1$ , which can be representative of the quadrotor case. Subsequently, Sect. 5.3 reports the simulation results obtained by considering a group of quadrotor UAVs while Sect. 5.4 presents the experiments with real robots. Finally, Sect. 6.5 concludes the chapter and discusses some future directions.

## 5.2 Decentralized formation control

Consider a *bearing rigid* framework  $(\mathcal{G}, \mathbf{q})$  in  $\mathbb{R}^3 \times \mathbb{S}^1$  consisting of  $N$  agents with dynamics (4.1). Let  $\mathbf{q}_d$  be a *desired* configuration such that  $(\mathcal{G}, \mathbf{q}_d)$  is bearing rigid, and let  $\mathbf{b}_{\mathcal{G}}^d = \boldsymbol{\beta}_{\mathcal{G}}(\mathbf{q}_d) = (\boldsymbol{\beta}_{e_1}^d \dots \boldsymbol{\beta}_{e_l}^d)$  be the corresponding desired value for the

bearing function. Our goal is to design a *decentralized* bearing formation controller able to accomplish two distinct objectives.

A first objective is the bearing formation stabilization: by acting on the control inputs  $(\mathbf{u}_i, w_i)$ , the controller should steer the  $N$  agents towards a configuration  $\mathbf{q}^*$  *equivalent* to  $\mathbf{q}_d$ , i.e., such that  $\beta_{\mathcal{G}}(\mathbf{q}^*) = \mathbf{b}_{\mathcal{G}}^d$ . Because of the framework rigidity, equivalence will also imply *congruency* with  $\mathbf{q}_d$ . Hence, the fulfilment of this first objective will ensure that  $\mathbf{q}(t)$  reaches the (correct) desired shape modulo a possible translation, vertical rotation, and scaling (i.e., the motions spanning the null-space of the bearing rigidity matrix).

A second independent objective, ‘orthogonal’ to the first one, is the possibility to steer the agent group along the motion directions that do not affect the bearing rigidity function (for not interfering with the previous formation control goal), i.e., such that  $\beta_{\mathcal{G}}(\mathbf{q}(t)) = \text{const}$ . These are spanned by the null space of the rigidity matrix which, for a rigid framework in  $\mathbb{R}^3 \times \mathbb{S}^1$ , consists of the aforementioned collective translation, expansion and coordinated rotation.

Fulfillment of this second objective will then allow to, for instance, collectively steer the agent group *while* maintaining a desired bearing formation for navigation or exploration purposes.

Finally, the formation controller should be decentralized and only based on information locally available or communicated by 1-hop neighbors.

As explained above, a complete solution to this control problem has been already proposed in [77] by, however, exploiting the structure of a very special interaction graph  $\mathcal{G}$  and a *single* distance measurement  $d_{ij} = \|\mathbf{p}_i - \mathbf{p}_j\|$  among an arbitrary pair of robots. Indeed, measurement of this distance allows to fix (without ambiguities) the scale of the formation, which is then needed for correctly implementing the coordinated rotation in the null-space of the bearing rigidity matrix. Coping with this requirement may be quite challenging from a sensing/technological point of view because of, e.g., mutual occlusions (for large quadrotor groups or in cluttered environments) and/or limited camera field of view (with respect to the last two problems the reader is referred to Chapt. 6).

Therefore, the goal of the rest of the section is to generalize the machinery presented in [77] to the case of a generic bearing rigid framework. This will be achieved by suitably combining and extending the results of [30, 31] which have addressed, in a separated way, the problems of rigidity-based decentralized localization and control for bearing formations for frameworks in  $\text{SE}(2)$ .

### 5.2.1 Rigidity-based control of bearing frameworks in $\mathbb{R}^3 \times \mathbb{S}^1$

A first contribution of this work is the following.

**Proposition 5.1.** *The null-space of the bearing rigidity matrix can be explicitly characterized as*

$$\begin{aligned} \mathcal{N}(\mathcal{B}_G^W(\mathbf{q})) &= \text{span} \left\{ \begin{bmatrix} \mathbf{1}_{N_3} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{p} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{p}^\perp \\ \mathbf{1}_N \end{bmatrix} \right\} \\ &= \text{span} \{ \mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3 \} \end{aligned} \quad (5.1)$$

where  $\mathbf{1}_N$  is a vector of all ones of dimension  $N$ ,  $\mathbf{1}_{N_3} = \mathbf{1}_N \otimes \mathbf{I}_3$ ,  $\mathbf{p}^\perp = (\mathbf{I}_N \otimes \mathbf{S})\mathbf{p}$ , and  $\otimes$  denotes the matrix Kronecker product.

*Proof.* The proof of Prop. 5.1 that vectors  $[\mathbf{1}_{N_3}^T \ \mathbf{0}^T]^T$  and  $[\mathbf{p}^T \ \mathbf{0}^T]^T$  belong to  $\mathcal{N}(\mathcal{B}_G^W)$  can be found in [30, 31]. The explicit expression for the last null-space vector in (5.1) (not present in [30, 31]) can be shown as follows: consider the  $k$ -th element of  $\mathcal{B}_G^W \begin{bmatrix} \mathbf{p}^\perp \\ \mathbf{1}_N \end{bmatrix}$ , which has the following expression

$$\begin{aligned} \frac{P_{ij} \mathbf{R}_i^T \mathbf{S}(\mathbf{p}_j - \mathbf{p}_i)}{d_{ij}} - \mathbf{S}\beta_{ij} &= P_{ij} \mathbf{R}_i^T \mathbf{S} \mathbf{R}_i \frac{\mathbf{R}_i^T (\mathbf{p}_j - \mathbf{p}_i)}{d_{ij}} - \mathbf{S}\beta_{ij} = \\ &= P_{ij} \mathbf{S}\beta_{ij} - \mathbf{S}\beta_{ij} = \mathbf{S}\beta_{ij} - \mathbf{S}\beta_{ij} = \mathbf{0}, \end{aligned} \quad (5.2)$$

where the properties  $\mathbf{R}_i^T \mathbf{S} \mathbf{R}_i = \mathbf{S}$  and  $P_{ij} \mathbf{S}\beta_{ij} = \mathbf{S}\beta_{ij}$  were used (the last one exploits the fact that  $\mathbf{S}\beta_{ij} \perp \beta_{ij}$ ).  $\square$

Each null-space vector of this particular basis for  $\mathcal{N}(\mathcal{B}_G^W(\mathbf{q}))$  represents one of the coordinated motions discussed above, namely, three translations along the world axes, an expansion about  $\mathbf{O}_W$ , and a rotation about a vertical axis passing through  $\mathbf{O}_W$ .

Remember that, as pointed out in Sect. 4.3, it is also possible to define a *body-frame* bearing rigidity matrix  $\mathcal{B}_G(\mathbf{q})$ . Let now  $\mathbf{e}_F(\mathbf{q}) = \mathbf{b}_G^d - \beta_G(\mathbf{q})$  denote the bearing formation control error to be regulated to zero for solving the first objective (formation stabilization). As shown in [30], minimization of  $\|\mathbf{e}_F\|$  can be obtained by implementing the following *scale-free* controller based on the *body-frame* rigidity matrix,

$$\begin{bmatrix} \mathbf{u} \\ \mathbf{w} \end{bmatrix} = k_c \begin{bmatrix} \text{diag}(d_{ij}) & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_N \end{bmatrix} \mathcal{B}_G(\mathbf{q})^T \mathbf{b}_G^d, \quad k_c > 0 \quad (5.3)$$

that results in the  $i$ -th agent velocity command

$$\begin{cases} \mathbf{u}_i &= -k_c \sum_{(i,j) \in \mathcal{E}} P_{ij} \beta_{ij}^d + k_c \sum_{(j,i) \in \mathcal{E}} {}^i \mathbf{R}_j P_{ji} \beta_{ji}^d \\ \mathbf{w}_i &= k_c \sum_{(i,j) \in \mathcal{E}} \beta_{ij}^T \mathbf{S} \beta_{ij}^d \end{cases}. \quad (5.4)$$

The reader is referred to [31] for an almost global stability proof for frameworks in SE(2) that can be directly extended to the case under consideration. Furthermore, the centroid  $\bar{\mathbf{p}} = \mathbf{1}_{N_3}^T \mathbf{p} / N$  and ‘scale’  $\mathbf{p}^T \mathbf{p}$  of the formation can be shown to be invariant under the action of (5.4).

It is worth noting that controller (5.4) has a decentralized structure depending only on the interaction graph  $\mathcal{G}$  and on relative quantities. In particular, it does not require knowledge of any distance measurement (from which the term *scale-free*), nor knowledge of any common reference frame shared by the agent group. However, controller (5.4) requires *communication* among agents since, if there exists an edge  $(j, i) \in \mathcal{E}$  (i.e., an agent  $j$  is measuring agent  $i$ ), agent  $i$  needs to receive the bearing measurement  $\beta_{ji}$  and desired bearing  $\beta_{ji}^d$  from agent  $j$  (second term of  $\mathbf{u}_i$ ).

Furthermore, controller (5.4) also needs access to the relative orientation  ${}^i\mathbf{R}_j$  among neighboring pairs which is a quantity not available from direct measurements and here, again, bearing rigidity comes in handy. Indeed, if a framework is bearing rigid, all the relative orientations among agent pairs are univocally fixed by the existing inter-agent bearings constraints. Therefore, it is in principle conceivable to recover/estimate the relative rotations  ${}^i\mathbf{R}_j$  by processing the measured inter-agent bearings.

This insight has, indeed, been exploited in [77, 121]. In [121] the assumption of reciprocal measurements for *all* agent pairs (undirected sensing graph  $\mathcal{G}$ ) allows for an algebraic computation of all the needed relative orientations. An analogous solution is exploited in [77] which, instead, relies on a very special construction of the (directed) sensing graph  $\mathcal{G}$ . The procedure of [77] cannot be, however, generalized to *generic* bearing rigid frameworks such as those considered in this venue. In order to cope with this problem, we now detail an extension of the localization algorithm introduced in [30] for obtaining a (decentralized) estimation of the relative orientations  ${}^i\mathbf{R}_j$  in presence of a *generic* bearing rigid graph and of *non-stationary* agents.

### 5.2.2 Rigidity-based localization of time-varying bearing frameworks in $\mathbb{R}^3 \times \mathbb{S}^1$

The decentralized localization algorithm proposed in [30] allows the agent group to estimate their *unscaled* positions<sup>1</sup> and orientations with respect to a common reference frame by only exploiting the available relative bearing measurements. The localization algorithm exploits the world-frame bearing rigidity matrix  $\mathcal{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q})$ .

---

<sup>1</sup>Here, the term ‘unscaled’ means that the agent positions are estimated up to a common scale factor.

Let  $\hat{\mathbf{q}} = (\hat{\mathbf{p}}, \hat{\boldsymbol{\psi}})$  be an estimation of the true  $\mathbf{q}$  and define the bearing estimation error as  $\mathbf{e}_L(\mathbf{q}, \hat{\mathbf{q}}) = \boldsymbol{\beta}_G(\mathbf{q}) - \boldsymbol{\beta}_G(\hat{\mathbf{q}})$ . Assuming  $\boldsymbol{\beta}_G(\mathbf{q}(t)) = \text{const}$ , minimization of  $\|\mathbf{e}_L\|$  can be obtained by this gradient descent based on the bearing rigidity matrix,

$$\begin{bmatrix} \dot{\hat{\mathbf{p}}} \\ \dot{\hat{\boldsymbol{\psi}}} \end{bmatrix} = k_e \mathcal{B}_G^W(\hat{\mathbf{q}})^T \boldsymbol{\beta}_G(\mathbf{q}), \quad k_e > 0. \quad (5.5)$$

Under the action of (5.5), the estimation  $\hat{\mathbf{q}}(t)$  will converge towards a configuration equivalent to  $\mathbf{q}$ . Bearing rigidity of the framework  $(G, \mathbf{q})$  will also imply congruency with  $\mathbf{q}$ . Therefore, at convergence ( $\mathbf{e}_L = \mathbf{0}$ ), the estimated  $\hat{\mathbf{q}}$  will reach a configuration such that

$$\begin{cases} \hat{\mathbf{p}} &= s(\mathbf{I}_N \otimes \mathbf{R}_z(\bar{\psi}))\mathbf{p} + \mathbf{1}_N \otimes \mathbf{t} \\ \hat{\boldsymbol{\psi}} &= \boldsymbol{\psi} + \mathbf{1}_N \bar{\psi} \end{cases} \quad (5.6)$$

for an arbitrary translation  $\mathbf{t} \in \mathbb{R}^3$ , rotation angle  $\bar{\psi} \in \mathbb{S}^1$  and scaling factor  $s \in \mathbb{R}^+$ . Any neighboring pair can then replace the unknown  ${}^i\mathbf{R}_j$  with the estimated  ${}^i\hat{\mathbf{R}}_j = \mathbf{R}_z(\hat{\psi}_j - \hat{\psi}_i)$  by exchanging the two estimates  $(\hat{\psi}_i, \hat{\psi}_j)$  over local communication.

The estimator (5.5) is fully decentralized and only requires the bearings in  $\boldsymbol{\beta}_G(\mathbf{q})$  as measured quantities. However, the estimator (5.5) also assumes  $\boldsymbol{\beta}_G(\mathbf{q}(t)) = \text{const}$  while the inter-agent relative bearings will be in general time-varying under the action of controller (5.4)<sup>2</sup>. Presence of a time-varying  $\boldsymbol{\beta}_G(\mathbf{q}(t))$  can clearly prevent convergence of the estimation error. This issue can be, however, addressed by adding to (5.5) the following feedforward term for taking into account the agent motion,

$$\begin{bmatrix} \dot{\hat{\mathbf{p}}} \\ \dot{\hat{\boldsymbol{\psi}}} \end{bmatrix} = k_e \mathcal{B}_G^W(\hat{\mathbf{q}})^T \boldsymbol{\beta}_G(\mathbf{q}) + \begin{bmatrix} \text{diag}(\mathbf{R}_z(\hat{\psi}_i)) & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_N \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{w} \end{bmatrix}. \quad (5.7)$$

**Proposition 5.2.** *If the initial estimation error  $\|\mathbf{e}_L(t_0)\|$  is small enough and  $s = 1$  then (5.7) will guarantee  $\|\mathbf{e}_L(t)\| \rightarrow 0$  in case of time-varying bearings  $\boldsymbol{\beta}_G(\mathbf{q}(t)) \neq \text{const}$ .*

*Proof.* The closed-loop dynamics of the estimation error is

$$\begin{aligned} \dot{\mathbf{e}}_L &= \boldsymbol{\beta}_G(\mathbf{q}) \begin{bmatrix} \mathbf{u} \\ \mathbf{w} \end{bmatrix} - \mathcal{B}_G^W(\hat{\mathbf{q}}) \begin{bmatrix} \dot{\hat{\mathbf{p}}} \\ \dot{\hat{\boldsymbol{\psi}}} \end{bmatrix} = \\ &= -k_e \mathcal{B}_G^W(\hat{\mathbf{q}}) \mathcal{B}_G^W(\hat{\mathbf{q}})^T \boldsymbol{\beta}_G(\mathbf{q}) + (\boldsymbol{\beta}_G(\mathbf{q}) - \mathcal{B}_G^W(\hat{\mathbf{q}})) \begin{bmatrix} \mathbf{u} \\ \mathbf{w} \end{bmatrix}. \end{aligned} \quad (5.8)$$

The first term of (5.8) represents the (nominal) closed-loop dynamics of the constant bearing case, while the second term of (5.8) is a perturbation due to the agent motion.

<sup>2</sup>Indeed,  $\boldsymbol{\beta}_G(\mathbf{q}(t)) = \text{const}$  only for stationary agents or for agents moving along the directions (5.1).

Since the nominal closed-loop dynamics is asymptotically stable [30], one can resort to the theory of perturbed systems [154] for analyzing the stability of (5.8). In particular, if the perturbation term can be shown to be *vanishing* with respect to the estimation error  $e_L$ , one can conclude local stability of the overall system (5.8) under mild conditions. Consider the  $k$ -th row of the body-frame rigidity matrix (4.10): this depends on the quantities  $\beta_{ij}$ ,  $d_{ij}$  and  ${}^i\mathbf{R}_j$ . By inspection one can then verify that, when  $e_L = \mathbf{0}$  (i.e., when (5.6) holds),  ${}^i\mathbf{R}_j = {}^i\hat{\mathbf{R}}_j$ ,  $\beta_{ij} = \hat{\beta}_{ij}$  and  $d_{ij} = s\hat{d}_{ij}$ . Assuming  $s = 1$  in (5.6) then results in  $e_L \rightarrow \mathbf{0} \implies \mathcal{B}_G(\mathbf{q}) - \mathcal{B}_G(\hat{\mathbf{q}}) \rightarrow \mathbf{0}$  which concludes the proof.  $\square$

A correct scale ( $s = 1$ ) of the estimated formation  $\hat{\mathbf{q}}$  is then necessary in order to compensate for the effects of the agent motion in the estimation dynamics. It is well-known that the formation scale cannot be retrieved from only bearing measurements without introducing the concepts of nonlinear observability of Chapt. 7 already or the ones illustrated in [146]. Therefore, here we exploit the presence of the *single* pair of agents, indexed as  $\iota$  and  $\kappa$ , which is assumed able to also measure its relative distance  $d_{\iota\kappa}$ . One can then consider the following ‘augmented’ cost function,

$$\frac{1}{2}(k_e e_L^T e_L + k_d (\hat{\mathbf{p}}_{\iota\kappa}^T \hat{\mathbf{p}}_{\iota\kappa} - d_{\iota\kappa}^2)^2), \quad k_d > 0 \quad (5.9)$$

meant to enforce the constraint  $\|\hat{\mathbf{p}}_{\iota\kappa}\| = \|\hat{\mathbf{p}}_\iota - \hat{\mathbf{p}}_\kappa\| = d_{\iota\kappa}$  in the estimated  $\hat{\mathbf{q}}$ . As shown in [30], minimization of (5.9) is obtained by complementing the update law (5.7) with the additional (decentralized) terms  $\mp k_d (\hat{\mathbf{p}}_{\iota\kappa}^T \hat{\mathbf{p}}_{\iota\kappa} - d_{\iota\kappa}^2) \hat{\mathbf{p}}_{\iota\kappa}$  in the  $\iota$ -th and  $\kappa$ -th entries of  $\dot{\hat{\mathbf{p}}}$ , respectively.

If a distance measurement is not available, the estimator (5.7) will not be able to recover the correct scale factor  $s$ , but it will still track changes in the formation scale thanks to the feedforward term. If the initial mismatch between actual and estimated scale is small enough, this is often enough for allowing (5.7) to still provide a consistent estimation of  $\mathbf{q}$ . Sect. 5.3 will indeed show results in this sense, by comparing the control/estimation performance when including/not-including the single distance measurement  $d_{\iota\kappa}$ .

### 5.2.3 Coordinated motions in the null-space of the bearing rigidity matrix

As a final step, we address the fulfilment of the second control objective, that is, the implementation of the null-space motions spanned by (5.1).

This can be achieved by realizing the world-frame velocity  $\dot{\mathbf{q}}_s = \mathbf{n}_1 \boldsymbol{\nu} + \mathbf{n}_2 \lambda + \mathbf{n}_3 w$  which imposes to the framework a common linear velocity  $\boldsymbol{\nu} \in \mathbb{R}^3$ , an expansion rate  $\lambda \in \mathbb{R}$  about  $\mathbf{O}_W$ , and a coordinated rotation with angular speed  $w$  about a

vertical axis passing through  $\mathbf{O}_{\mathcal{W}}$ . The corresponding *body-frame* velocities  $(\mathbf{u}_s, \mathbf{w}_s)$  to be added to the formation control inputs  $(\mathbf{u}, \mathbf{w})$  in (5.3) are then

$$\begin{bmatrix} \mathbf{u}_s \\ \mathbf{w}_s \end{bmatrix} = \begin{bmatrix} \text{diag}(\mathbf{R}_i^T) & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_N \end{bmatrix} \dot{\mathbf{q}}_s. \quad (5.10)$$

While (5.10) realizes the second control objective, it is usually more interesting to implement an expansion rate and coordinated rotation about a specific point of interest *attached* to the formation itself, rather than about the (arbitrary) origin of the world frame  $\mathbf{O}_{\mathcal{W}}$ . For instance, an often convenient choice is to implement these motions relative to the formation centroid  $\bar{\mathbf{p}} = \mathbf{1}_{N_3}^T \mathbf{p} / N$ . This can be obtained by using as basis for  $\mathcal{N}(\mathcal{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q}))$  the set  $\{\mathbf{n}_1, \mathbf{n}_2 - \mathbf{n}_1 \bar{\mathbf{p}}, \mathbf{n}_3 - \mathbf{n}_1 \mathbf{S} \bar{\mathbf{p}}\}$  which eventually results in the  $i$ -th agent velocity commands

$$\begin{cases} \mathbf{u}_{s_i} &= \mathbf{R}_i^T (\boldsymbol{\nu} + \lambda(\mathbf{p}_i - \bar{\mathbf{p}}) + w \mathbf{S}(\mathbf{p}_i - \bar{\mathbf{p}})) \\ w_{s_i} &= w \end{cases}. \quad (5.11)$$

An actual implementation of (5.11) would require the (non-available) quantities  $(\mathbf{p}_i, \psi_i, \bar{\mathbf{p}})$ : exploiting the estimator (5.7), each agent can replace the true  $(\mathbf{p}_i, \psi_i)$  with the estimated  $(\hat{\mathbf{p}}_i, \hat{\psi}_i)$ . The corresponding average  $\hat{\bar{\mathbf{p}}} = \mathbf{1}_{N_3}^T \hat{\mathbf{p}} / N$  can then be obtained by resorting to any distributed averaging filtering technique such as the well-known PI *average consensus filter* (PI-ACE) [70]. Indeed, given a (time-varying) vector quantity  $\mathbf{x}(t) \in \mathbb{R}^N$  with each component  $x_i(t)$  locally available to agent  $i$ , the PI-ACE filter allows every agent to distributedly build an estimation converging to the average  $\bar{x}(t) = \sum_{i=1}^N x_i(t) / N$  with a tunable dynamics that can be made faster than the underlying dynamics of each agent in the system, see also [34, 119, 155] for some applications in the context of multi-robot distributed control.

It is worth noting that knowledge of the correct scale factor ( $s = 1$ ) in the estimated  $\hat{\mathbf{q}}$  is not required for implementing the null-space motions associated to vectors  $\mathbf{n}_1$  and  $\mathbf{n}_2$  (translation and expansion), but it is instead required for correctly implementing the coordinated rotation associated to vector  $\mathbf{n}_3$ . Indeed,  $\mathbf{n}_1$  does not depend on  $\mathbf{p}$  and  $\mathbf{n}_2$  is *homogeneous* in  $\mathbf{p}$ , with its direction thus unaffected by any scaling of the agent positions. This is, however, not the case for vector  $\mathbf{n}_3$  which is *not* homogeneous in  $\mathbf{p}$ . Similarly to the estimation case, if a distance measurement is not available for fixing the scale of  $\hat{\mathbf{q}}$ , the coordinated rotation will not be exactly implemented.

#### 5.2.4 Discussion

We conclude by emphasizing that, as stated at the beginning of the section, the proposed control/estimation scheme does *not* require a special topology for the

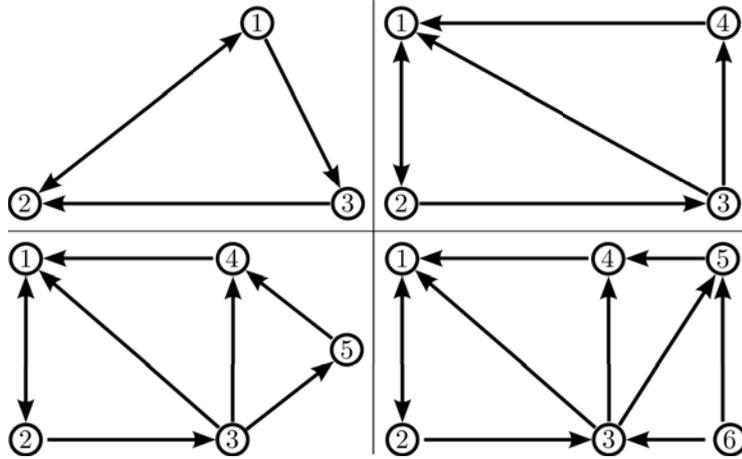


Figure 5.2 – Possible minimal bearing rigid topologies for  $N \in \{3, 4, 5, 6\}$  (note that some arrows are bi-directional)

interaction graph (besides being bearing-rigid); the bearing controller (5.4), the localization algorithm (5.7), and the null-space motions (5.11) have the same (decentralized) expression for all agents only as a function of the measured bearings and body-frame linear/angular velocities. The only exception is the inclusion of the distance measurement  $d_{l\kappa}$  which adds an additional control term to agents  $l$  and  $\kappa$ . In this sense, we believe that the work presented in this chapter represents a significant generalization of the strategy reported in [77] which, to the best of our knowledge, is the closest related work to our setting, and relied on a much more constrained design of the agent group.

We also note that the correct formation scale could be retrieved without assuming the presence of an (additional) distance measurement  $d_{l\kappa}$  (and, thus, presence of two ‘special agents’ in the group). Indeed, the unknown robot inter-distances could be estimated online by processing the measured inter-robot bearings and the (known) robot own motions similarly to what done in the context of scale estimation for point features [156, 157]. In this respect, a possible solution will be presented in Chapt. 7. A different approach has been instead followed by [146] which proposes an *active* scale estimation strategy for bearing formations of quadrotor UAVs.

We finally wish to briefly discuss the practical implications of requiring directed (bearing) rigidity for the robot formation (which, as often stated, is an underlying necessary condition of the proposed machinery). As well-known, minimal rigidity requires presence of a  $|\mathcal{E}| = O(N)$  number of edges (i.e., of inter-robot measurements/constraints) in the framework vs. the quadratic complexity of the complete (directed) graph  $\mathcal{K}_N$  (for which the complexity would be  $N(N-1) = O(N^2)$ ). As illustration, Fig. 5.2 shows some possible *minimal* bearing rigid topologies for the case of  $N \in \{3 \dots 6\}$  agents for which  $|\mathcal{E}| = 4, 6, 8, 10$ , respectively.

### 5.3 Simulation results

In this section we present some simulation results involving  $N = 6$  quadrotor UAVs. The simulation is run by considering the full dynamics of 6 quadrotor UAVs simulated via the 3D physical simulator V-REP [158], for more details about the software architecture the reader is referred to Appendix B. In this case, the robustness of the bearing controller is tested against the discrepancies between the nominal agent (4.1) and the actual quadrotor flight dynamics, as well as against noise and discretization in the measured bearings (which are sampled at 60 Hz for mimicking an actual onboard camera).

The initial configuration  $\mathbf{q}(t_0)$  and estimated  $\hat{\mathbf{q}}(t_0)$  were generated by adding to the desired  $\mathbf{q}_d$  a uniformly distributed random perturbation of amplitude 1 m for the positions and 120 deg for the orientation. A graph  $\mathcal{G}$  with  $|\mathcal{E}| = 20$  directed edges was then randomly generated under the constraint of guaranteeing bearing rigidity at  $\mathbf{q}(t_0)$ ,  $\hat{\mathbf{q}}(t_0)$  and  $\mathbf{q}_d$ , and the following gains were used:  $k_c = 1$  in (5.4),  $k_e = 5$  in (5.7) and  $k_d = 10$  in (5.9). Finally, in all simulations the null-space velocity commands (5.11) were activated, for convenience of illustration, only after convergence to the desired formation  $\mathbf{b}_{\mathcal{G}}^d$ : the five coordinated motions were first actuated one at the time and then all together.

Figs. 5.3 and 5.4 report the simulation results. In particular, the bottom Fig. 5.3(a) shows the behavior of  $\|\mathbf{e}_F(t)\|$  (the bearing control error, solid blue line) and of  $\|\mathbf{e}_L(t)\|$  (the estimation error, solid red line), while the top Fig. 5.3(a) depicts the five null-space velocity commands ( $\nu$ ,  $\lambda$ ,  $w$ ) in (5.11) given by a human operator through a joystick. It is then possible to verify how (i) both the control and estimation bearing errors converge to zero (despite their initial large value and despite the presence of time-varying bearings  $\beta(\mathbf{q}(t))$ ) and how (ii) the implementation of the coordinated motions (5.11) has no disturbing effect on the bearing errors (as expected).

Let  $\lambda_6(\mathbf{q}) \geq 0$  represent the sixth smallest eigenvalue of the following square matrix, which we refer to as *symmetric bearing rigidity matrix*<sup>3</sup>

$$\mathbf{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q}) = \mathcal{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q})^T \mathcal{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q}). \quad (5.12)$$

Since for an infinitesimal rigid framework in  $\mathbb{R}^3 \times \mathbb{S}^1$  it holds  $\text{rank}(\mathcal{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q})) = 4N - 5$ , the quantity  $\lambda_6(\mathbf{q})$  can be taken as a measure of the framework bearing rigidity. Indeed,  $\lambda_6(\mathbf{q}) > 0$  if and only if  $(\mathcal{G}, \mathbf{q})$  is infinitesimal rigid and  $\lambda_6(\mathbf{q}) = 0$  otherwise. Fig. 5.3(b) then reports the behavior of  $\lambda_6^C(t) = \lambda_6(\mathbf{q}(t))$  and  $\lambda_6^E(t) = \lambda_6(\hat{\mathbf{q}}(t))$ , that is, the rigidity measures for the ‘control’ framework  $(\mathcal{G}, \mathbf{q})$  and the ‘estimation’ framework  $(\mathcal{G}, \hat{\mathbf{q}})$ . One can then check how both frameworks remained

<sup>3</sup>Note the notation difference between  $\mathbf{B}$  and  $\mathcal{B}$

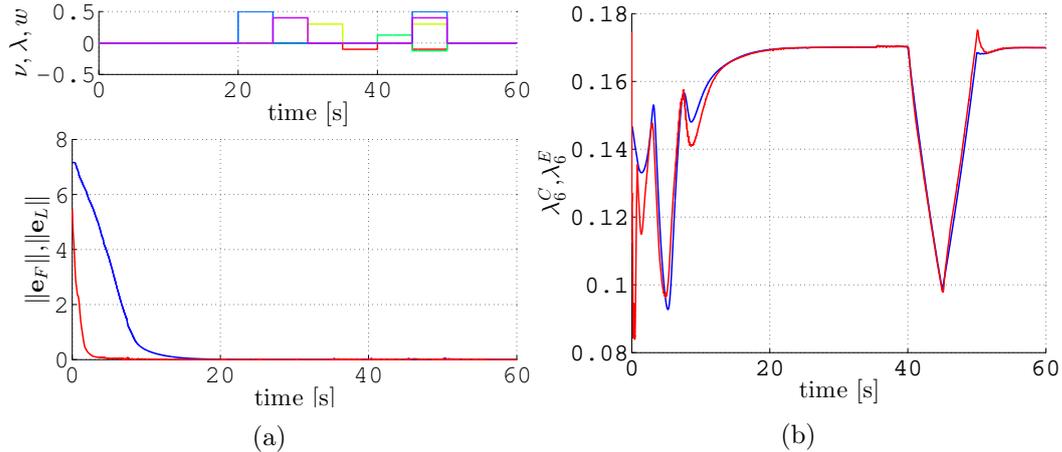


Figure 5.3 – Results of the simulation. **5.3(a)**-top: behavior of the five null-space motion commands  $\nu(t)$  (blue, purple, yellow)  $\lambda(t)$  (green) and  $w(t)$  (red). **5.3(a)**-bottom: behavior of the bearing control error  $\|e_F(t)\|$  and of the localization control error  $\|e_L(t)\|$ . **5.3(b)**: behavior of the rigidity eigenvalues  $\lambda_6^C(t)$  (control – blue) and  $\lambda_6^E(t)$  (estimation – red)

rigid throughout the motion of the agents, thus confirming congruency between  $\mathbf{q}$  and  $\mathbf{q}_d$  (correct agent formation), and between  $\mathbf{q}$  and  $\hat{\mathbf{q}}$  (correct agent localization).

As an additional measure of the localization performance, we considered the quantity  $\mathbf{e}_\psi = (\mathbf{I}_N - \mathbf{1}_N \mathbf{1}_N^T / N) (\boldsymbol{\psi} - \hat{\boldsymbol{\psi}})$ : this represents the *disagreement* between the orientation estimation error and its mean value, and it should vanish in presence of a correct localization<sup>4</sup> as, indeed, reported in Fig. 5.4(b). A converging  $\mathbf{e}_\psi(t)$  then allows to correctly compute the missing terms  ${}^i \mathbf{R}_j$  in the bearing controller (5.4). Finally, Fig. 5.4(b) depicts the behavior of the ‘formation scale error’ defined as  $e_s(t) = \|\mathbf{p}(t) - \mathbf{1}_N \otimes \bar{\mathbf{p}}(t)\| - \|\hat{\mathbf{p}}(t) - \mathbf{1}_N \otimes \hat{\bar{\mathbf{p}}}(t)\|$  which, again, converges to zero, as expected, thanks to the additional distance constraint in (5.9).

It is worth noting that the ‘distortions’ present in Fig. 5.4(b) are mainly due to the higher-order quadrotor dynamics neglected by model (4.1) which, roughly speaking, introduces an unmodeled lag between commanded and actual velocities. The proposed control strategy is nevertheless robust enough for coping with these model inaccuracies.

## 5.4 Experimental results

This section discusses the experiment conducted for validating the formation controller described in Sect. 5.2.

For the experiment illustrated in this section we used a group of four quadrotor of the ones described in Sect. B.2. The TeleKyb framework [159] was used for

<sup>4</sup>Indeed, a correct localization implies that  $\hat{\boldsymbol{\psi}}(t) \rightarrow \boldsymbol{\psi}(t) + \mathbf{1}_N \bar{\boldsymbol{\psi}}$  (see (5.6)).

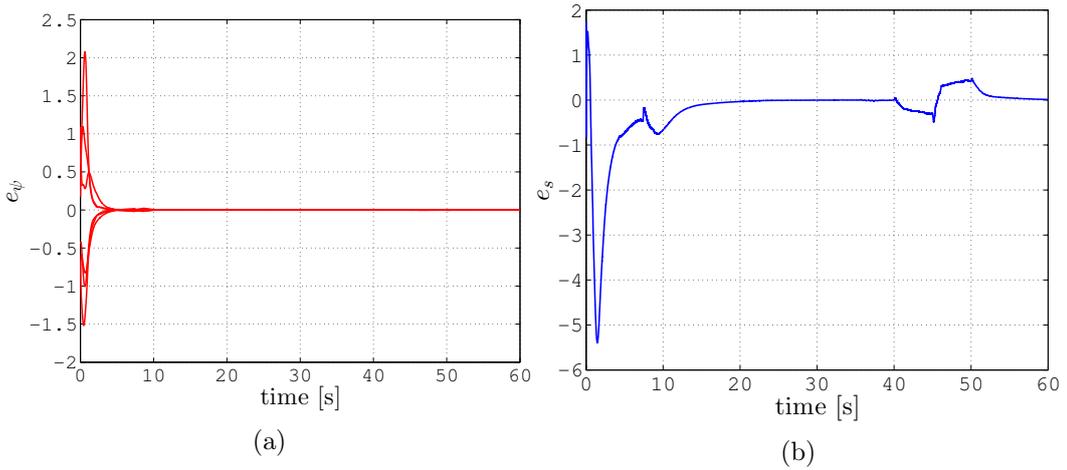


Figure 5.4 – Results of the first simulation. 5.4(a): behavior of the orientation estimation error  $e_\psi(t)$ . 5.4(b): behavior of the formation scale error  $e_s(t)$

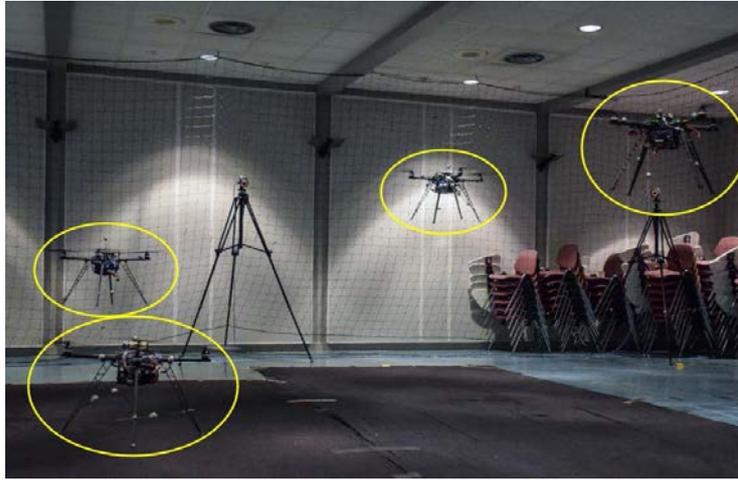


Figure 5.5 – Four quadrotors flying in the flying arena

interfacing our bearing control algorithm with the MK-Quadro low-level controller. The onboard low-level controller is the one described in [160] and it is in charge of letting the orientation of the quadrotor track a desired reference. The experiments were performed in our flying arena which is equipped with a Vicon motion capture system, employed for reconstructing the body-frame bearing measurements  $\beta_{ij}$  that would have been obtained by an onboard camera (refer to Sect. B.2.2 for more details).

The reported experiment followed a pattern similar to the simulation results described in the previous section: (i) regulation towards a desired bearing formation, (ii) actuation of the null-space motions (5.11), (iii) regulation towards a different desired bearing formation, (iv) actuation of the null-space motions (5.11). Additionally, we implemented, at every 6 seconds, a random switch among all the possible rigid

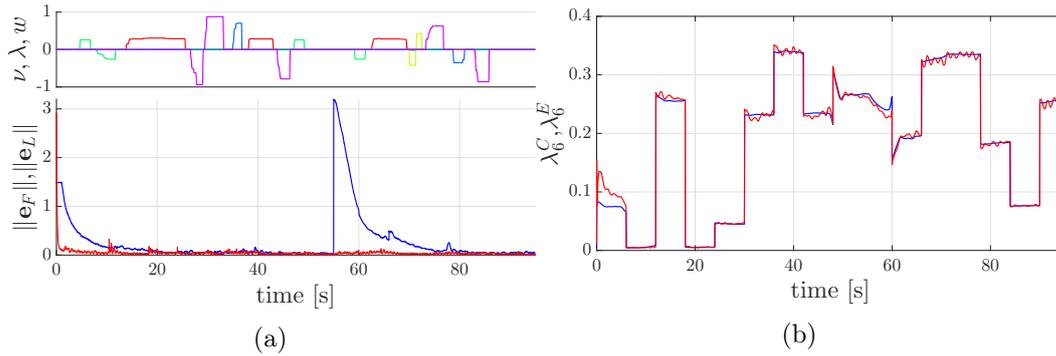


Figure 5.6 – Results of the experiment. Fig. 5.6(a)-top: behavior of the five null-space motion commands  $\nu(t)$  (blue, purple, yellow)  $\lambda(t)$  (green) and  $w(t)$  (red). Fig. 5.6(a)-bottom: behavior of the bearing control error  $\|e_F(t)\|$  and of the localization control error  $\|e_L(t)\|$ . Fig. 5.6(b): behavior of the rigidity eigenvalues  $\lambda_6^C(t)$  (control – blue) and  $\lambda_6^E(t)$  (estimation – red)

topologies for the sensing graph in order to show the robustness of our approach also against possible topology changes during motion.

Fig. 5.6–5.7 report the results of the experiment. The UAV formation starts far from the desired configuration but, after about 20 seconds, the norm of the formation control error  $\|e_F(t)\|$  drops below 4% of its initial value (Fig. 5.6(a)). On the other hand, convergence of the estimator error  $\|e_L(t)\|$  is quite fast even though the initial estimated  $\hat{\mathbf{q}}(t_0)$  was generated by adding to the real  $\mathbf{q}(t_0)$  a uniformly distributed random perturbation of amplitude 1.5 m for the positions  $\mathbf{p}(t_0)$  and 80 deg for the orientations  $\psi(t_0)$ . Convergence of the estimated  $\hat{\mathbf{q}}(t)$  towards a configuration congruent with  $\mathbf{q}(t)$  (and with the correct scale) can also be appreciated in Fig. 5.7 where the orientation estimation error  $e_\psi(t)$  and the formation scale error  $e_s(t)$  are shown. One can then verify, again, how a consistent estimation of the orientations  $\hat{\psi}$  and of the formation scale  $s$  could be obtained despite the (unavoidable) non-idealities present in any real implementation.

Finally, as stated before, the underlying graph  $\mathcal{G}$  switches randomly at every 6 seconds across all the possible rigid topologies for the  $N = 4$  quadrotors (in particular, we allowed switches among graphs with  $|\mathcal{E}| \in \{6, 7, 8, 9, 10, 11, 12\}$ ). Fig. 5.6(b) shows the behavior of the rigidity measures  $\lambda_6^C(t)$  and  $\lambda_6^E(t)$  which ‘jump’ at every 6 seconds (as expected) because of the topology switches. Nevertheless, these topology changes (and associated increases/decreases of the control/estimation framework rigidity) did not negatively affected the overall performance of the proposed bearing control strategy.

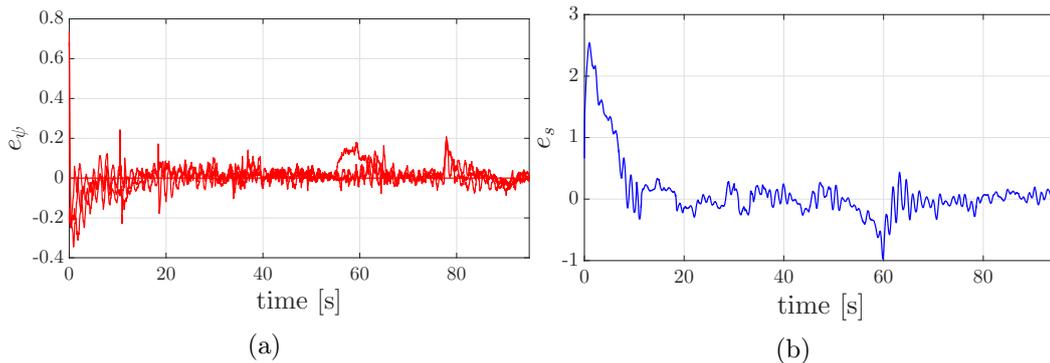


Figure 5.7 – Results of the experiment. Fig. 5.7(a): behavior of the orientation estimation error  $e_\psi(t)$ . Fig. 5.7(b): behavior of the formation scale error  $e_s(t)$

## 5.5 Conclusions

In this chapter we have considered the problem of devising a decentralized control strategy for controlling a group of quadrotor UAVs able to measure relative bearings in their own body frames. In particular, we considered two control objectives: (i) stabilization of the quadrotor formation towards a desired bearing configuration, and (ii) steering of the whole formation along the motion directions in the null-space of the bearing rigidity matrix. To this end, a suitable  $\mathbb{R}^3 \times \mathbb{S}^1$  extension and combination of the SE(2) directed bearing rigidity control/localization algorithms introduced in [30, 31] has been developed for the case of non-stationary agents, together with a full explicit characterization of the null-space of the bearing rigidity matrix. This allowed to devise a decentralized bearing controller able to meet the two control objectives without the need of a common reference frame for the agent group, nor the requirement of reciprocal bearing measurements (i.e., of an undirected measurement topology). Simulation and experimental results on real quadrotors have been proposed to illustrate the various features of the approach.

A fully-onboard implementation of the proposed ideas by equipping the UAVs with onboard cameras for retrieving relative bearings and estimating the quadrotor body-frame linear/angular velocities  $(\mathbf{u}_i, \mathbf{w}_i)$  represents a fundamental future work of this work and, in general, of this Thesis. By doing this we will be able to free ourselves from the need of an external motion capture system. Refer to Chapt. 8 for more details and a future direction regarding this issue. Another logical consequence of this work is the extension of the ideas presented in [119] for dealing with the issue of bearing rigidity *maintenance* in the presence of sensor constraints. Some of these constraints are the limited field of view and range of the onboard cameras and the possible occlusions generated by different UAVs. A possible solution to this problem is given in Chapt. 6 and it would then allow the quadrotor group to *flexibly* navigate in cluttered environment under the (controlled) possibility of

losing/gaining neighbors while ensuring a minimum level of bearing rigidity for the formation. Another interesting extension, as we pointed out several times, is the (decentralized) integration of Structure from Motion (SfM) schemes, such as [146], able to recover *online* the missing scale information by processing the measured bearings and known agent motion (and, thus, avoiding the requirement of a special agent pair able to additionally measure its inter-distance). An alternative solution to this problem is presented in Chapt. 7.



---

# Rigidity maintenance

## Contents

---

6.1	Introduction . . . . .	76
6.1.1	Chapter overview . . . . .	77
6.2	Cooperative localization from bearing measurements . . . . .	78
6.3	A bearing rigidity maintenance strategy . . . . .	78
6.3.1	Design of the inter-agent weights . . . . .	79
6.3.2	The bearing rigidity eigenvalue . . . . .	83
6.3.3	The bearing rigidity maintenance controller . . . . .	87
6.3.4	Discussion . . . . .	88
6.4	Experimental results . . . . .	90
6.5	Conclusions . . . . .	91

---

IN the previous chapters we considered the concept of rigidity and its properties with respect to the problems of formation control and estimation when applied to a group of quadrotor UAVs. In this chapter of the Thesis we are concerned with a distributed technique which is able to maintain rigidity while moving a formation of quadrotor UAVs. We asked ourselves the question: *what are the challenges to maintain bearing rigidity in a real-world scenario?* The answer to this question is strictly related to the aforementioned concepts of bearing rigidity and therefore to the sensors used to retrieve the bearing measurements (e.g., monocular cameras) and to their limitations. Nevertheless, in this chapter we present the details of a strategy we adopted to maintain *bearing rigidity* of a formation of quadrotor UAVs. The goal is to maintain the rigidity during motion despite the presence of several sensing constraints. This strategy can be coupled with a formation controller like the one presented in Chapt. 5. This would allow to have a distributed control strategy which is able to reach a desired formation and/or steering in the 3D space,

all this while coping with sensors limitations. The sensing constraints we explicitly took into account were:

- minimum and maximum range of the camera
- limited field of view of the camera
- possible occlusions caused by the agents of the formation

To this end, a decentralized gradient-based control action was developed, based on a suitable *degree of infinitesimal rigidity* linked to the spectral properties of the bearing rigidity matrix. The approach was first validated in a simulation environment and then through real experiments employing five quadrotor UAVs.

## 6.1 Introduction

The use of local sensing (such as onboard cameras or range sensors), however, entails many challenges. For instance, in the absence of centralized aids, each robot is typically only able to collect measurements and impose control actions in its *local* body-frame: when exchanging information over communication, the group then faces the need of, e.g., agreeing over some *common shared frame* where to express any quantity of interest. Furthermore, local (onboard) sensing also forces to cope with any sensing limitation (such as limited range/field of view, or occluded visibility) that can prevent retrieving the needed measurements with respect to neighboring robots.

While a number of rigidity-based formation control/localization schemes has been proposed over the years, to the best of our knowledge only a few previous works (e.g., [31]) have considered the issue of *maintaining/preserving* rigidity of the formation (which is a global property like graph connectivity [114, 161]) during the agent motion despite the possible presence of sensing constraints. Rigidity maintenance is a fundamental problem since, as explained, losing formation rigidity prevents convergence of any formation control/localization scheme run by the robot group.

In this respect, in this chapter we present a work which considers the problem of *bearing* rigidity maintenance for a robot group equipped with onboard monocular cameras able to measure *relative bearings* with respect to other robots in visibility. Indeed, cameras are a widespread sensor modality for mobile robots, and the problem of coordinating the motion/formation of a robot group from only camera (bearing) measurements has attracted large attention in the robotics and control community [77, 120–122, 125, 126, 162]. Cameras, however, also suffer from all the

shortcomings listed above: they only provide relative (and unscaled) measurements in the local *body-frame* of the measuring agent, and are affected by several sensing constraints such as limited field of view, limited range, and occluded visibility. Another big limitation of most of the commercial cameras is the ability to work *only* in environments with an appropriate luminosity. Some recent work is being conducted to solve these problems on quadrotor UAVs [163] by using event-based cameras (as [164]) but it is outside the scope of this work.

Taking inspiration from [119], in this work we then propose a control strategy able to ensure maintenance at all times of a minimum level of bearing rigidity for the robot formation during motion. The controller consists of a decentralized gradient descent action based on a suitable “degree of bearing rigidity” which is directly related to the spectral properties of the so-called bearing rigidity matrix already introduced in Chapt. 4. Furthermore, we assume a *directed* measurement topology (no need of reciprocal measurements among neighboring pairs), and explicitly consider three typical sensing constraints affecting onboard cameras:

(C1): minimum/maximum range,

(C2): limited field of view, and

(C3): possible occluded visibility because of the alignment of multiple robots in the camera field of view.

Finally, we experimentally validate the proposed machinery with five quadrotor UAVs navigating in an indoor flying arena. To the best of our knowledge, the problem of bearing rigidity maintenance *and* the use of the complex (but quite realistic) sensing model (C1)–(C3) for camera-based formation control purposes are two novel contributions of this work.

### 6.1.1 Chapter overview

The rest of this chapter is organized as follows: Sect. 6.2 quickly recaps the localization algorithm described in Sect. 5.2.2. Then Sect. 6.3 presents the various details of the proposed bearing rigidity maintenance strategy together with a discussion on the issue of decentralization and scalability. Subsequently, Sect. 5.4 reports the results of our experimental validation with five quadrotor UAVs, and Sect. 6.5 concludes the chapter and discusses some future directions.

## 6.2 Cooperative localization from bearing measurements

The modeling assumptions of this work are exactly the same of the previous Chapt. 5 and they are specified in Chapt. 4. Moreover, as suggested by (4.10), the implementation of bearing formation control algorithms for agents evolving in  $\mathbb{R}^3 \times \mathbb{S}^1$  requires the availability of the relative orientation  ${}^i\mathbf{R}_j$  and, possibly, distance  $d_{ij}$  among neighboring robots (see, e.g., [27, 31, 77]). These quantities, in our context, are not assumed to be directly measurable. A possible workaround is to then estimate these (needed) quantities via some bearing-based cooperative localization scheme.

In this respect, in this chapter we use the decentralized localization algorithm for rigid frameworks in  $\mathbb{R}^3 \times \mathbb{S}^1$  presented in Sect. 5.2.2. Therefore, letting  $\hat{\mathbf{q}}_i = (\hat{\mathbf{p}}_i, \hat{\psi}_i)$  represent the estimation of the true agent  $i$  configuration  $\mathbf{q}_i$ , we assume, from now on, that the agent group runs the localization algorithm (of Sect. 5.2.2), and treat  $\hat{\mathbf{q}}_i$  as a sufficiently good approximation of the true  $\mathbf{q}_i$  (always up to a roto-translation of the whole framework).

Note that the localization algorithm presented in Sect. 5.2.2 is entirely decentralized and can cope with any *time-varying* bearing function  $\beta_{\mathcal{G}}(\mathbf{q}(t))$  (i.e., the agents do not need to remain stationary or move in ‘special’ ways) as long as the formation is infinitesimally bearing rigid. It does, however, require the availability of at least one inter-agent distance  $d_{\iota\kappa}$  among an arbitrary agent pair  $(\iota, \kappa)$  for fixing the scale of the formation (which, otherwise, would be unobservable only from bearing measurements). This (possibly single) inter-agent distance can be either considered as an additional measurement among a special agent pair equipped with a distance sensor, or it can be estimated online by processing the measured bearings and the *known* agent body-frame velocities  $(\mathbf{u}_i, w_i)$  as extensively explained in Chapt. 7 and [146] (thus, without requiring, in this case, any additional sensing capability).

## 6.3 A bearing rigidity maintenance strategy

The goal of this section is to present an algorithm for bearing rigidity maintenance able to also take into account some typical limitations of the employed onboard sensors (cameras), that is, minimum/maximum range, limited field of view, and possible line-of-sight (visibility) occlusions due to the alignments of multiple robots in the camera field of view. Our strategy is inspired by the connectivity/(distance) rigidity maintenance controllers presented in [34, 119, 155] and can be summarized as follows: the sensing constraints affecting a pair of agents  $(i, j)$  are encoded in a suitable scalar

*weight*<sup>1</sup>  $w_{ij} = w_k$  associated to the edge  $e_k = (i, j)$ , with  $w_k = 0$  if the constraint is not satisfied, and  $0 < w_k \leq 1$  otherwise<sup>2</sup>. Let  $\mathbf{W} = \text{diag}(w_k \mathbf{I}_3) \in \mathbb{R}^{3|\mathcal{E}| \times 3|\mathcal{E}|}$  be a diagonal matrix collecting all the  $|\mathcal{E}|$  weights  $w_k = w_{ij}$ , and define the *weighted* body-frame rigidity matrix as  $\mathbf{WB}_{\mathcal{G}} = \mathbf{WB}_{\mathcal{G}}^{\mathcal{W}} \mathbf{T}$ . Any weight  $w_k \rightarrow 0$  will cause the corresponding  $k$ -th row block of  $\mathbf{WB}_{\mathcal{G}}$ , associated to edge  $e_k = (i, j)$ , to vanish, potentially diminishing the rank of  $\mathbf{WB}_{\mathcal{G}}$  (and, thus, possibly threatening bearing rigidity of the framework). The agents will then implement a gradient-based controller able to preserve infinitesimal bearing rigidity of the formation by ensuring fulfilment of the rank condition  $\text{rank}(\mathbf{WB}_{\mathcal{G}}^{\mathcal{W}}) = 4N - 5$  for the weighted bearing rigidity matrix. The next sections detail the various steps needed to implement this gradient control action.

### 6.3.1 Design of the inter-agent weights

We start addressing the design of the inter-agent weights  $w_{ij}$  associated to each edge  $e_k = (i, j)$  that will encode the sensing constraints among agent pairs. Since, as explained in Sect. 6.1, in this work we consider the three constraints (C1) minimum/maximum range, (C2) limited field of view and (C3) occluded visibility. The weights  $w_{ij}$  will then be designed as the product of three individual terms

$$w_{ij} = w_{R_{ij}} w_{F_{ij}} w_{V_{ij}} \quad (6.1)$$

accounting for (C1)–(C3), respectively.

#### 6.3.1.1 Minimum/maximum range

As first constraint, we consider that the employed cameras have a minimum/maximum range beyond which no detection of neighboring quadrotors is possible (refer to Fig. 6.1 for a real-world scenario associated to this particular constraint). In particular  $d_{min}, d_{max}$  represent respectively the minimum and maximum range of the camera and, obviously,  $d_{min} < d_{max}$ . The weight  $w_{R_{ij}}(d_{ij})$  is then designed as a function that smoothly vanishes (with vanishing derivative) when  $d_{ij} \rightarrow d_{min}$  and  $d_{ij} \rightarrow d_{max}$ , and has a maximum at the midpoint between  $d_{min}$  and  $d_{max}$ . To this end, we choose the following function

$$w_{R_{ij}}(d_{ij}) = \begin{cases} 0 & d_{ij} < d_{min} \\ \frac{1}{2} + \frac{1}{2} \cos\left(\pi \frac{2d_{ij} - (d_{min} + d_{max})}{d_{max} - d_{min}}\right) & d_{min} \leq d_{ij} \leq d_{max} \\ 0 & d_{ij} > d_{max} \end{cases} \quad (6.2)$$

for which a representative shape is shown in Fig. 6.2a.

<sup>1</sup>With a small abuse of notation, we use either  $w_k$  or  $w_{ij}$  to indicate the weight associated to an edge  $e_k = (i, j)$ .

<sup>2</sup>We also note that the introduction of time-varying weights  $w_{ij}(t)$  naturally induces a corresponding time-varying edge set  $\mathcal{E}(t)$  (the set of all edges  $e_k$  for which  $w_k(t) > 0$ ).



Figure 6.1 – Fig. 6.1(a) is a snapshot from the onboard camera mounted on a quadrotor. It is taken when the robot is close to lose a link (with respect to the robot highlighted in yellow) due to the *maximum range* constraint. Fig. 6.1(b) is the graph view corresponding to the case of Fig. 6.1(a).

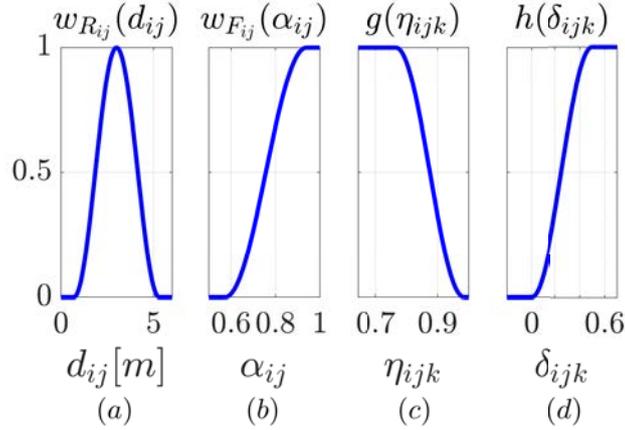


Figure 6.2 – Representative shapes for the weights  $w_{R_{ij}}(d_{ij})$  (a),  $w_{F_{ij}}(\alpha_{ij})$  (b),  $g(\eta_{ijk})$  (c), and  $h(\delta_{ijk})$  (d)

### 6.3.1.2 Limited field of view

A second constraint is related to the limited field of view of the onboard cameras (refer to Fig. 6.3 for a real-world scenario associated to this particular constraint). Let  $\mathbf{o}_C \in \mathbb{S}^2$  be the (constant and known) direction of the camera optical axis in the quadrotor body-frame and consider the scalar product  $\alpha_{ij} = \mathbf{o}_C^T \boldsymbol{\beta}_{ij}$ . Clearly,  $\alpha_{ij} = 1$  when  $\boldsymbol{\beta}_{ij}$  and  $\mathbf{o}_C$  are perfectly aligned, and  $\alpha_{ij} \rightarrow -1$  as the angle between  $\boldsymbol{\beta}_{ij}$  and  $\mathbf{o}_C$  increases. Letting  $-1 \leq \alpha_{min} < 1$  represent the camera field of view, we then design weight  $w_{F_{ij}}$  as the following function with a maximum for  $\alpha_{max} \leq \alpha_{ij} \leq 1$  and smoothly vanishing with vanishing derivative for  $\alpha_{ij} \rightarrow \alpha_{min}$

$$w_{F_{ij}}(\alpha_{ij}) = \begin{cases} 0 & \alpha_{ij} < \alpha_{min} \\ \frac{1}{2} - \frac{1}{2} \cos\left(\pi \frac{\alpha_{ij} - \alpha_{min}}{\alpha_{max} - \alpha_{min}}\right) & \alpha_{min} \leq \alpha_{ij} \leq \alpha_{max} \\ 1 & \alpha_{ij} > \alpha_{max} \end{cases} \quad (6.3)$$

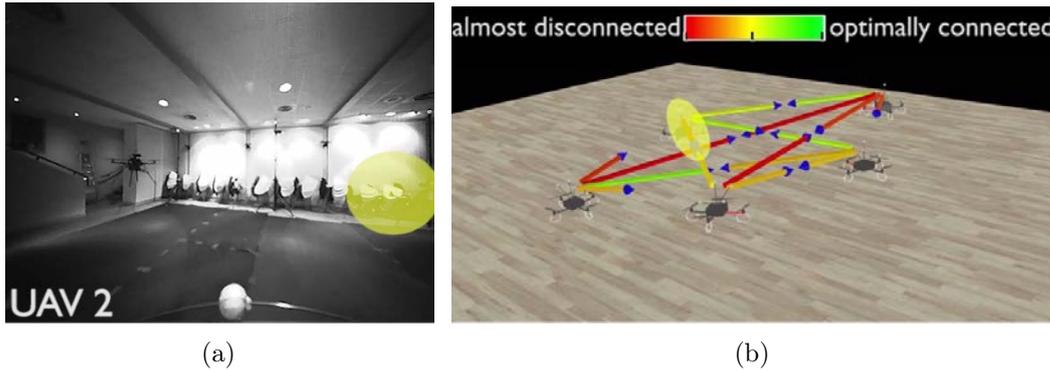


Figure 6.3 – Fig. 6.3(a) is a snapshot from the onboard camera mounted on a quadrotor. It is taken when the robot is close to lose a link (with respect to the robot highlighted in yellow) due to the *limited field of view* constraint. Fig. 6.3(b) is the graph view corresponding to the case of Fig. 6.3(a).



Figure 6.4 – Fig. 6.4(a) highlights why the *occluded visibility* constraint is needed. Indeed, it is possible to notice how the visibility of the robot in red, with respect to the *observing* robot, is being occluded by the robot in green. As usual, Fig. 6.4(b) is the graph view corresponding to the case of Fig. 6.4(a).

A representative shape for the (6.3) is shown in Fig. 6.2b.

### 6.3.1.3 Occluded visibility

A final constraint considered in this work is that of possible occluded visibility because of two (or more) quadrotors (close to be) aligned in front of a camera (with, thus, the closest quadrotor occluding the visibility of the farthest ones). The design of the weight  $w_{V_{ij}}$  meant to encode this constraint requires some care and, thus, we proceed by steps (Fig. 6.4(a) gives the reader a better understanding of this constraint through a real-world scenario).

Let us first focus on the situation depicted in Fig. 6.5 in which an agent  $i$  is measuring exactly two agents  $j$  and  $k$ . Our goal is to define a function  $w_{V_{ijk}}$  that captures the possible occlusions of agent  $k$  (third subscript) on the edge  $(i, j)$  (first two subscripts), and, analogously, a function  $w_{V_{ikj}}$  capturing the occlusions of  $j$  on

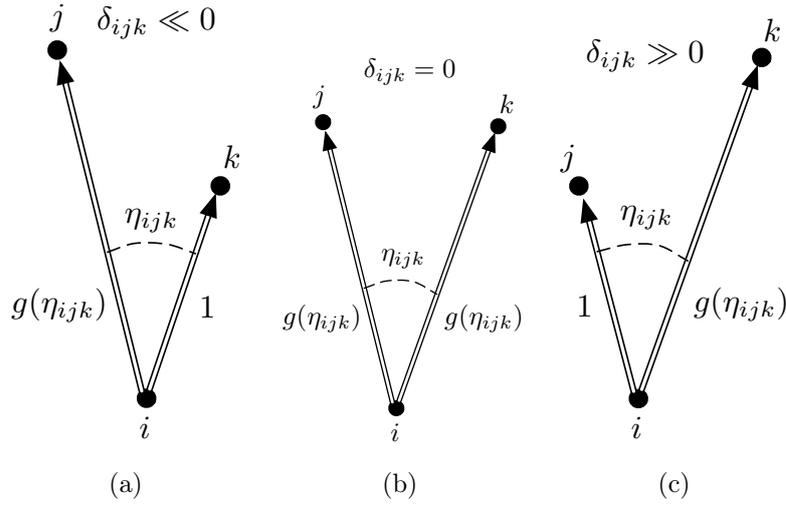


Figure 6.5 – Scheme which depicts the weights associated to the occluded visibility between a triple of agents.

the edge  $(i, k)$ . To this end, let  $\eta_{ijk} = \beta_{ij}^T \beta_{ik}$  represent the (cosine of the) angle between the two bearing measurements of  $j$  and  $k$  by agent  $i$ , and  $\delta_{ijk} = d_{ik} - d_{ij}$  the relative distance of  $k$  and  $j$  with respect to  $i$ . Note that  $\eta_{ijk} = \eta_{ikj}$  and  $\delta_{ijk} = -\delta_{ikj}$ .

Since, unlike the previous case, we now wish to penalize *small* angular displacements between  $j$  and  $k$  with respect to  $i$  (which could lead to a possible occlusion), we introduce the following penalty function

$$g(\eta_{ijk}) = \begin{cases} 0 & \eta_{ijk} \geq \eta_{max} \\ \frac{1}{2} - \frac{1}{2} \cos\left(\pi \frac{\eta_{ijk} - \eta_{max}}{\eta_{max} - \eta_{min}}\right) & \eta_{min} \leq \eta_{ijk} \leq \eta_{max} \\ 1 & \eta_{ijk} \leq \eta_{min} \end{cases} \quad (6.4)$$

where  $-1 \leq \eta_{min} < \eta_{max} \leq 1$  are suitable parameters representing the activation/deactivation of function  $g(\cdot)$ . Therefore,  $g(\eta_{ijk}) = 0$  if agents  $j$  and  $k$  are close to be aligned from the vantage point of  $i$  ( $\eta_{ijk} \geq \eta_{max}$ ), while  $g(\eta_{ijk}) = 1$  if agents  $j$  and  $k$  are far from being aligned ( $\eta_{ijk} \leq \eta_{min}$ ). Fig. 6.2c shows a representative plot of function  $g(\eta_{ijk})$ .

The weight  $w_{V_{ijk}}$ , associated to the edge  $(i, j)$ , should be clearly based on the ‘angular penalty’ function  $g(\eta_{ijk})$ . However, weight  $w_{V_{ijk}}$  should also take into account the fact that, regardless of the value of  $\eta_{ijk}$ , agent  $j$  can be occluded by agent  $k$  *only if* agent  $k$  is front of  $j$  (i.e., if  $\delta_{ijk} < 0$ ), and symmetrically for the weight  $w_{V_{ikj}}$  associated to the edge  $(k, j)$ . As depicted in Fig. 6.5, a possibility is to have:

- 1)  $w_{V_{ijk}} = g(\eta_{ijk})$  and  $w_{V_{ikj}} = 1$  if  $\delta_{ijk} < 0$ : agent  $k$  is in front of  $j$  and, thus, only the edge  $(i, j)$  is penalized by  $g(\eta_{ijk})$  (Fig. 6.5(a));

- 2)  $w_{V_{ijk}} = 1$  and  $w_{V_{ikj}} = g(\eta_{ijk})$  if  $\delta_{ijk} \gg 0$ : agent  $j$  is in front of  $k$  and, thus, only the edge  $(i, k)$  is penalized by  $g(\eta_{ijk})$  (Fig. 6.5(c));
- 3)  $w_{V_{ijk}} = w_{V_{ikj}} = g(\eta_{ijk})$  if  $\delta_{ijk} = 0$ : agents  $j$  and  $k$  are at the same distance from  $i$  and, therefore, the two edges  $(i, j)$  and  $(i, k)$  are equally penalized by  $g(\eta_{ijk})$  (Fig. 6.5(b))

In order to obtain this goal we then introduce a second penalty function

$$h(\delta_{ijk}) = \begin{cases} 0 & \delta_{ijk} \leq 0 \\ \frac{1}{2} + \frac{1}{2} \cos\left(\pi \frac{\delta_{ijk} - \delta_{max}}{\delta_{max}}\right) & 0 \leq \delta_{ijk} \leq \delta_{max} \\ 1 & \delta_{max} \leq \delta_{ijk} \end{cases} \quad (6.5)$$

where  $\delta_{max} > 0$  is an activation parameter. Fig. 6.2d shows a representative plot of function  $h(\delta_{ijk})$ . With this choice of  $h(\delta_{ijk})$ , weight  $w_{V_{ijk}}$  can then be chosen as

$$w_{V_{ijk}}(\delta_{ijk}, \eta_{ijk}) = (1 - h(\delta_{ijk}))g(\eta_{ijk}) + h(\delta_{ijk}), \quad (6.6)$$

and symmetrically for  $w_{V_{ikj}}(\delta_{ikj}, \eta_{ikj})$ . Because of (6.4–6.5), one can easily verify that the expression in (6.6) correctly realizes the above-mentioned requirements 1)–3) for weights  $w_{V_{ijk}}$  and  $w_{V_{ikj}}$ .

The last step is to generalize this procedure for producing the (cumulative) weight  $w_{V_{ij}}$  on the edge  $(i, j)$  to be plugged in (6.1). The previous  $w_{V_{ijk}}$  accounts for the occlusions on edge  $(i, j)$  by a *single* specific agent  $k \in \mathcal{N}_i/\{j\}$ . Clearly, if  $|\mathcal{N}_i| < 2$  then one should have  $w_{V_{ij}} \equiv 1$  (no occlusions are possible if agent  $i$  has less than two neighbors). If, on the other hand,  $|\mathcal{N}_i| \geq 2$  then, in order to take into account *all* the possible occlusions on edge  $(i, j)$  by *any* neighbor  $k \in \mathcal{N}_i/\{j\}$ , one can just take the product sequence

$$w_{V_{ij}} = \prod_{k \in \mathcal{N}_i/\{j\}} w_{V_{ijk}}(\delta_{ijk}, \eta_{ijk}). \quad (6.7)$$

### 6.3.2 The bearing rigidity eigenvalue

Having introduced the weighting machinery meant to encode the sensing constraints considered in this work, we now discuss a suitable measure of bearing rigidity that is exploited by the rigidity maintenance controller. We first consider the *unweighted* case ( $\mathbf{W} = \mathbf{I}_{3|\mathcal{E}|}$ ) and then explicitly introduce the weights  $w_{ij}$  in the design.

As discussed, a framework is infinitesimally bearing rigid if and only if  $\text{rank}(\mathbf{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q})) = \text{rank}(\mathbf{B}_{\mathcal{G}}(\mathbf{q})) = 4N - 5$ . This rank condition can be translated into an equivalent spectral condition on the eigenvalues of two suitable corresponding matrices. To this end, define  $\mathbf{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q}) = \mathbf{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q})^T \mathbf{B}_{\mathcal{G}}^{\mathcal{W}}(\mathbf{q})$  and  $\mathbf{B}_{\mathcal{G}}(\mathbf{q}) = \mathbf{B}_{\mathcal{G}}^T(\mathbf{q}) \mathbf{B}_{\mathcal{G}}(\mathbf{q})$  as the  $4N \times 4N$

world-frame and body-frame (weighted) *symmetric rigidity matrices*, respectively. These two matrices are *similar*<sup>3</sup> since from (4.9) it follows

$$\mathbf{B}_G(\mathbf{q}) = \mathbf{T}^T(\boldsymbol{\psi})\mathbf{B}_G^{\mathcal{W}}(\mathbf{q})\mathbf{T}(\boldsymbol{\psi}) \quad (6.8)$$

and  $\mathbf{T}(\boldsymbol{\psi})$  is orthonormal<sup>4</sup>. Letting  $\lambda_i(\mathbf{A})$  represent the  $i$ -th smallest eigenvalue of a square symmetric matrix  $\mathbf{A}$ , infinitesimal bearing rigidity then translates into the condition  $\lambda_6(\mathbf{B}_G^{\mathcal{W}}(\mathbf{q})) = \lambda_6(\mathbf{B}_G(\mathbf{q})) > 0$  (where the similarity between  $\mathbf{B}_G^{\mathcal{W}}(\mathbf{q})$  and  $\mathbf{B}_G(\mathbf{q})$  has been used). Let then  $\lambda_6^B(\mathbf{q})$  represent the sixth smallest eigenvalue of the world/body-frame symmetric rigidity matrix:  $\lambda_6^B(\mathbf{q})$  is a natural measure of infinitesimal rigidity and is denoted from now on as the *bearing rigidity eigenvalue*. The purpose of this section is to detail the main properties and explicit expressions of  $\lambda_6^B(\mathbf{q})$  and of its gradient with respect to the  $i$ -th agent configuration  $\mathbf{q}_i$ , which is later used in Sect. 6.3.3.

A first observation is formalized by the following Proposition.

**Proposition 6.1.** *The bearing rigidity eigenvalue does not depend on the agent orientations  $\boldsymbol{\psi}$ , i.e.,  $\lambda_6^B(\mathbf{q}) = \lambda_6^B(\mathbf{p})$ .*

*Proof.* The proof exploits the definition  $\lambda_6^B(\mathbf{q}) = \lambda_6(\mathbf{B}_G^{\mathcal{W}}(\mathbf{q}))$  based on the *world-frame* symmetric rigidity matrix. Let  $\bar{\mathbf{p}}_{ij} = \mathbf{p}_{ij}/d_{ij}$  and  $\bar{\mathbf{P}}_{ij} = \mathbf{I}_n - \bar{\mathbf{p}}_{ij}\bar{\mathbf{p}}_{ij}^T$ . Exploiting (4.2) and the identity  $\mathbf{S}\mathbf{R}_i = \mathbf{R}_i\mathbf{S}$ , it follows that

$$\mathbf{P}_{ij}\mathbf{R}_i^T = (\mathbf{I}_N - \beta_{ij}\beta_{ij}^T)\mathbf{R}_i^T = \mathbf{R}_i^T\bar{\mathbf{P}}_{ij}, \quad (6.9)$$

$$\mathbf{S}\beta_{ij} = \mathbf{R}_i^T\mathbf{S}\bar{\mathbf{p}}_{ij} \quad (6.10)$$

The  $k$ -th row block of  $\mathbf{B}_G^{\mathcal{W}}(\mathbf{q})$  in (4.7) associated to an edge  $e_k = (i, j)$  can then be factorized as

$$\mathbf{R}_i^T(\boldsymbol{\psi}) \begin{bmatrix} -\mathbf{0} & -\underbrace{\frac{\bar{\mathbf{P}}_{ij}}{d_{ij}}}_i & -\mathbf{0} & -\underbrace{\frac{\bar{\mathbf{P}}_{ij}}{d_{ij}}}_j & -\mathbf{0} & -\underbrace{\mathbf{S}\bar{\mathbf{p}}_{ij}}_{3N+i} & -\mathbf{0} \end{bmatrix} = \mathbf{R}_i^T(\boldsymbol{\psi})\bar{\mathbf{B}}_k(\mathbf{p}_{ij}), \quad (6.11)$$

where  $\bar{\mathbf{B}}_k(\mathbf{p}_{ij}) \in \mathbb{R}^{3 \times 4N}$  does *not* depend on  $\boldsymbol{\psi}$ . If we define

$$\bar{\mathbf{B}} = \text{stack}(\bar{\mathbf{B}}_1, \dots, \bar{\mathbf{B}}_k) \in \mathbb{R}^{3|\mathcal{E}| \times 4N}, \quad (6.12)$$

<sup>3</sup>Two matrixes  $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times n}$  are similar if there exists a matrix  $\mathbf{P} \in \mathbb{R}^{n \times n}$  such that  $\mathbf{B} = \mathbf{P}^{-1}\mathbf{A}\mathbf{P}$ . One of the (several) properties of *similar* matrixes which is exploited in the math of this chapter is that the two matrixes have the same eigenvalues (but, in general, not the same eigenvectors).

<sup>4</sup>An orthonormal matrix  $\mathbf{A} \in \mathbb{R}^{n \times n}$  is such that  $\mathbf{A}^T\mathbf{A} = \mathbf{A}\mathbf{A}^T = \mathbf{I}_n$  and equivalently  $\mathbf{A}^T = \mathbf{A}^{-1}$ .

it follows that the world-frame symmetric bearing rigidity matrix can be actually rewritten as

$$\mathbf{B}_G^{\mathcal{W}}(\mathbf{q}) = \mathbf{B}_G^{\mathcal{W}}(\mathbf{q})^T \mathbf{B}_G^{\mathcal{W}}(\mathbf{q}) = \bar{\mathbf{B}}(\mathbf{p})^T \bar{\mathbf{B}}(\mathbf{p}). \quad (6.13)$$

This shows that  $\mathbf{B}_G^{\mathcal{W}}(\mathbf{q}) = \mathbf{B}_G^{\mathcal{W}}(\mathbf{p})$  and, as a consequence, that  $\lambda_6^B(\mathbf{q}) = \lambda_6^B(\mathbf{p})$ , thus concluding the proof.  $\square$

We now provide two explicit expressions of  $\lambda_6^B(\mathbf{p})$  that is instrumental in the following developments. Let  $\mathbf{v}_6 \in \mathbb{R}^{4N}$  represent the *world-frame* normalized eigenvector associated to  $\lambda_6^B$ , i.e., such that (using (6.13))

$$\lambda_6^B = \mathbf{v}_6^T \mathbf{B}_G^{\mathcal{W}} \mathbf{v}_6 = \mathbf{v}_6^T \bar{\mathbf{B}}^T \bar{\mathbf{B}} \mathbf{v}_6 \quad (6.14)$$

Consider the partition  $\mathbf{v}_6 = [\mathbf{v}_{p_1}^T \dots \mathbf{v}_{p_N}^T \ v_{\psi_1} \dots v_{\psi_N}]^T$ , where  $\mathbf{v}_{p_i} \in \mathbb{R}^3$  and  $v_{\psi_i} \in \mathbb{R}$  are the eigenvector components associated to the position and orientation of agent  $i$ . By exploiting (6.11), the properties  $\bar{\mathbf{P}}_{ij} \bar{\mathbf{P}}_{ij} = \bar{\mathbf{P}}_{ij}$  and  $\bar{\mathbf{P}}_{ij} \mathbf{S} \mathbf{p}_{ij} = \mathbf{S} \mathbf{p}_{ij}$  (see the proof of Prop. 6.1 and [27]), it is possible to obtain the following expression for  $\lambda_6^B$  (applying (6.14))

$$\lambda_6^B = \sum_{(i,j) \in \mathcal{E}} \left( \mathbf{v}_{p_{ij}}^T \frac{\bar{\mathbf{P}}_{ij}}{d_{ij}^2} \mathbf{v}_{p_{ij}} + 2 \mathbf{v}_{p_{ij}}^T \mathbf{S} \frac{\bar{\mathbf{P}}_{ij}}{d_{ij}} v_{\psi_i} - v_{\psi_i}^2 \bar{\mathbf{p}}_{ij}^T \mathbf{S}^2 \bar{\mathbf{p}}_{ij} \right) \quad (6.15)$$

where  $\mathbf{v}_{p_{ij}} = \mathbf{v}_{p_i} - \mathbf{v}_{p_j}$ .

The expression (6.15) obviously depends on *world-frame* quantities. An equivalent expression in terms of *body-frame* quantities can be obtained as follows (look at Appendix A.1 for more details): let  $\boldsymbol{\nu}_6$  represent the *body-frame* normalized eigenvector, i.e., such that  $\lambda_6^B = \boldsymbol{\nu}_6^T \mathbf{B}_G \boldsymbol{\nu}_6$ . From (6.8) it follows that  $\mathbf{v}_6 = \mathbf{T} \boldsymbol{\nu}_6$  and, therefore,  $\mathbf{v}_{p_i} = \mathbf{R}_i \boldsymbol{\nu}_{p_i}$  and  $v_{\psi_i} = \nu_{\psi_i}$ . Since  $\bar{\mathbf{P}}_{ij} = \mathbf{R}_i \mathbf{P}_{ij} \mathbf{R}_i^T$  and  $\bar{\mathbf{p}}_{ij} = \mathbf{R}_i \boldsymbol{\beta}_{ij}$  (see Prop. 6.1 and (4.2)), the expression (6.15) can be reformulated as

$$\lambda_6^B = \sum_{(i,j) \in \mathcal{E}} \left( \boldsymbol{\nu}_{p_{ij}}^T \frac{\mathbf{P}_{ij}}{d_{ij}^2} \boldsymbol{\nu}_{p_{ij}} + 2 \boldsymbol{\nu}_{p_{ij}}^T \mathbf{S} \frac{\boldsymbol{\beta}_{ij}}{d_{ij}} \nu_{\psi_i} - \nu_{\psi_i}^2 \boldsymbol{\beta}_{ij}^T \mathbf{S}^2 \boldsymbol{\beta}_{ij} \right) = \sum_{(i,j) \in \mathcal{E}} l_{ij} \quad (6.16)$$

with  $\boldsymbol{\nu}_{p_{ij}} = \boldsymbol{\nu}_{p_i} - {}^i \mathbf{R}_j \boldsymbol{\nu}_{p_j}$ . Each term  $l_{ij}$  in (6.16) is now rewritten in terms of only body-frame quantities relative to agents  $i$  and  $j$  (measured bearings  $\boldsymbol{\beta}_{ij}$ , interdistances  $d_{ij}$ , relative orientations  ${}^i \mathbf{R}_j$  and relative components of the body-frame rigidity eigenvector  $\boldsymbol{\nu}_6$ ). Note that, in general,  $l_{ij} \neq l_{ji}$ . Furthermore, by looking at the (world-frame) expression of  $l_{ij}$  in (6.15), it follows that  $\partial l_{ij} / \partial \psi_k = 0, \forall k$  (in accordance with Prop. 6.1).

Let us now introduce the weights  $w_{ij}$  into the rigidity eigenvalue  $\lambda_6^B$  and proceed to obtain a closed-form expression for its gradient. As explained at the beginning of Sect. 6.3, the weights  $w_{ij}$  can be included by replacing the body-frame symmetric

bearing rigidity matrix with a weighted counterpart  $\mathbf{B}_G^T(\mathbf{q})\mathbf{W}\mathbf{B}_G(\mathbf{q})$  (and analogously for the world-frame symmetric bearing rigidity matrix  $\mathbf{B}^W$ ). Since  $\mathbf{W}$  is a diagonal matrix, repeating the previous steps simply results into

$$\lambda_6^B = \sum_{(i,j) \in \mathcal{E}} w_{ij} l_{ij} \quad (6.17)$$

in place of (6.16). Expression (6.17) can be leveraged for obtaining the gradient of  $\lambda_6^B$  with respect to the configuration  $\mathbf{q}_v = (\mathbf{p}_v, \psi_v)$  of a specific agent  $v$  in the group. Recalling the definition of set  $\mathcal{O}_i$  in (3.2), we first note that the only terms  $l_{ij}$  that depend on  $\mathbf{p}_v$  are those associated to the edges  $(v, j), \forall j \in \mathcal{N}_v$  and  $(j, v), \forall j \in \mathcal{O}_v$ . The computation of the gradient of these terms with respect to  $\mathbf{p}_v$  is based on the general formula for expressing the derivative of an eigenvalue with respect to a parameter (see, e.g., [155])  $d\lambda_6 = \boldsymbol{\nu}_6^T d\mathbf{B}_G \boldsymbol{\nu}_6$  (for more details refer to Appendix A.2). Therefore, it can be shown, after some (tedious) steps, that

$$\begin{aligned} \nabla_{\mathbf{p}_v} l_{vj} &= 2\mathbf{R}_v \left( \frac{1}{d_{vj}^3} (\boldsymbol{\beta}_{vj} \boldsymbol{\nu}_{p_{vj}}^T + \boldsymbol{\nu}_{p_{vj}}^T \boldsymbol{\beta}_{vj} \mathbf{I}_3) \mathbf{P}_{vj} \boldsymbol{\nu}_{p_{vj}} + \right. \\ &\left. + \frac{\nu_{\psi_v}}{d_{vj}^2} (\mathbf{P}_{vj} - \boldsymbol{\beta}_{vj} \boldsymbol{\beta}_{vj}^T) \mathbf{S} \boldsymbol{\nu}_{p_{vj}} + \frac{\nu_{\psi_v}^2}{d_{vj}} \mathbf{P}_{vj} \mathbf{S}^2 \boldsymbol{\beta}_{vj} \right) = 2\mathbf{R}_v \mathbf{L}_{vj} \end{aligned} \quad (6.18)$$

and  $\nabla_{\mathbf{p}_v} l_{jv} = -2\mathbf{R}_j \mathbf{L}_{jv}$ .

The gradient of weights  $w_{ij}$  is, however, more involved since the weight design introduces additional dependencies which must be correctly taken into account. Some details about the gradient of the weights are given in Appendix A.3. Consider again the configuration  $\mathbf{q}_v = (\mathbf{p}_v, \psi_v)$  of a specific agent  $v$  and the weight partition (6.1): clearly, the three subweights  $w_{R_{ij}}$  in (6.2),  $w_{F_{ij}}$  in (6.3) and  $w_{V_{ij}}$  in (6.7) depend on  $\mathbf{p}_v$  over the same set of edges  $(v, j), \forall j \in \mathcal{N}_v$  and  $(j, v), \forall j \in \mathcal{O}_v$  as for the terms  $l_{ij}$ . However, because of its definition, and unlike the terms  $l_{ij}$ , the third subweight  $w_{V_{ij}}$  depends on  $\mathbf{p}_v$  also over all the edges  $(j, m), \forall j \in \mathcal{O}_v, \forall m \in \mathcal{N}_j$ . Furthermore, weight  $w_{F_{ij}}$  depends on  $\psi_v$  over all the edges  $(v, j), \forall j \in \mathcal{N}_v$ , while weights  $w_{R_{ij}}$  and  $w_{V_{ij}}$  do not depend on  $\psi_v$  ( $w_{R_{ij}}$  and  $w_{V_{ij}}$  are functions of interdistances, and  $w_{V_{ij}}$  is also function of the angular quantity  $\eta_{ijk}$  which, however, does not depend on  $\psi$ ).

Therefore, after some (again tedious) algebra, and by exploiting (6.18), the

gradient of  $\lambda_6^B$  with respect to  $\mathbf{q}_v$  can be expanded as follows

$$\left\{ \begin{array}{l} \nabla_{\mathbf{p}_v} \lambda_6^B = \mathbf{R}_v \sum_{j \in \mathcal{N}_v} (l_{vj} \nabla_{\mathbf{p}_v} w_{vj} + 2\mathbf{L}_{vj} w_{vj}) + \\ + \sum_{j \in \mathcal{O}_v} \left( -2\mathbf{R}_j w_{jv} \mathbf{L}_{jv} + \sum_{m \in \mathcal{N}_j} \mathbf{R}_m l_{jm} \nabla_{\mathbf{p}_v} w_{jm} \right) \\ \nabla_{\psi_v} \lambda_6^B = \sum_{j \in \mathcal{N}_v} l_{vj} \nabla_{\psi_v} w_{vj} \end{array} \right. \quad (6.19)$$

### 6.3.3 The bearing rigidity maintenance controller

Having obtained an explicit expression for the gradient  $\nabla_{\mathbf{q}_v} \lambda_6^B$ , we can now present the proposed bearing rigidity maintenance controller. Similarly to [34, 119], we assume that the agents need to maintain a minimum level of bearing rigidity during motion, i.e., guarantee that  $\lambda_6^B(t) > \lambda_6^{\min} > 0$  where  $\lambda_6^{\min}$  is a suitable lower threshold depending on the particular application. We then introduce a potential function  $V_\lambda(\lambda_6^B) \geq 0$  such that  $V_\lambda(\lambda_6^B) \rightarrow \infty$  as  $\lambda_6^B \rightarrow \lambda_6^{\min}$  and  $V_\lambda(\lambda_6^B) \rightarrow 0$  (with vanishing derivative) as  $\lambda_6^B \rightarrow \lambda_6^{\max} > \lambda_6^{\min}$ . In particular the expression of  $V_\lambda(\lambda_6^B)$  is the following

$$V_\lambda(\lambda_6^B) = \begin{cases} 0 & 0 \leq \lambda_6^B \leq \lambda_6^{\min} \\ K_V \tan(a\lambda_6^B + b)^2 & \lambda_6^{\min} \leq \lambda_6^B \leq \lambda_6^{\max} \\ 0 & \lambda_6^{\max} \leq \lambda_6^B \end{cases} \quad (6.20)$$

where

$$a = \frac{0.5\pi}{\lambda_6^{\max} - \lambda_6^{\min}}, \quad b = -\frac{0.5\pi\lambda_6^{\max}}{\lambda_6^{\max} - \lambda_6^{\min}}. \quad (6.21)$$

Fig. 6.6 shows a possible shape for the potential function  $V_\lambda(\lambda_6^B)$ .

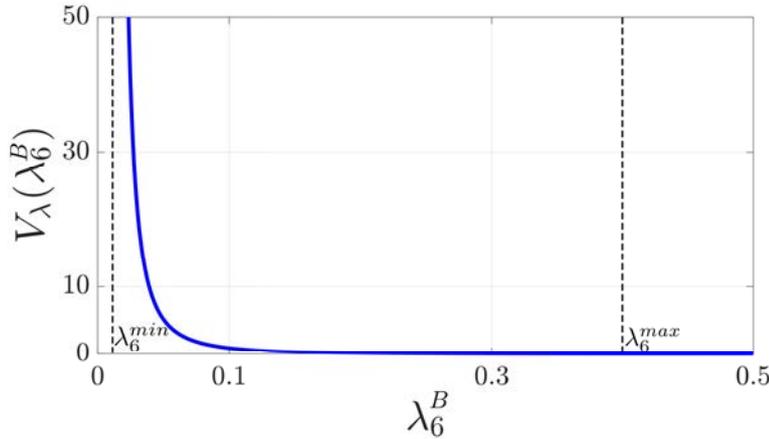


Figure 6.6 – A representative shape for the function  $V_\lambda(\lambda_6^B)$  with  $\lambda_6^{\min} = 0.01$  and  $\lambda_6^{\max} = 0.42$

Each agent  $v$  will then follow the anti-gradient of  $V_\lambda(\lambda_6^B)$  with respect to  $\mathbf{q}_v$  which, using (4.1), yields the following body-frame velocity commands

$$\begin{cases} \mathbf{u}_v &= -\mathbf{R}_v^T \nabla_{\mathbf{p}_v} V_\lambda(\lambda_6^B) \\ w_v &= -\nabla_{\psi_v} V_\lambda(\lambda_6^B) \end{cases} \quad (6.22)$$

By plugging (6.19) in (6.22), we then obtain

$$\begin{cases} \mathbf{u}_v = -\frac{\partial V_\lambda}{\partial \lambda_6^B} \left( \sum_{j \in \mathcal{N}_v} (l_{vj} \nabla_{\mathbf{p}_v} w_{vj} + 2\mathbf{L}_{vj} w_{vj}) + \right. \\ \left. + \sum_{j \in \mathcal{O}_v} \left( -2 {}^v \mathbf{R}_j w_{jv} \mathbf{L}_{jv} + \sum_{m \in \mathcal{N}_j} {}^v \mathbf{R}_m l_{jm} \nabla_{\mathbf{p}_v} w_{jm} \right) \right) \\ w_v = -\frac{\partial V_\lambda}{\partial \lambda_6^B} \sum_{j \in \mathcal{N}_v} l_{vj} \nabla_{\psi_v} w_{vj} \end{cases} \quad (6.23)$$

### 6.3.4 Discussion

We conclude with some remarks about the bearing rigidity maintenance controller (6.23). Let us first consider the issue of decentralization and scalability: by analyzing the explicit expressions of the various terms in (6.23) (given in the previous sections), it is possible to conclude that, in order to implement the rigidity maintenance action, agent  $v$  needs knowledge of  $\lambda_6^B$  (for evaluating  $\partial V_\lambda / \partial \lambda_6^B$ ), of  $(\nu_{p_v}, \nu_{\psi_v})$ , and of

- 1)  $\beta_{vj}, d_{vj}, \psi_{vj}, \nu_{p_j}, \nu_{\psi_j}, \forall j \in \mathcal{N}_v$ ;
- 2)  $\beta_{jv}, d_{jv}, \psi_{jv}, \nu_{p_j}, \nu_{\psi_j}, \forall j \in \mathcal{O}_v$ ;
- 3)  $\beta_{jm}, d_{jm}, \psi_{jm}, \nu_{p_m}, \nu_{\psi_m}, \forall j \in \mathcal{O}_v, \forall m \in \mathcal{N}_j / \{v\}$ .

Let us assume, for now, that each agent has access to the value of  $\lambda_6^B$  and to its own components of  $\nu_6$ : one can then verify that 1)–3) consist of (measured) bearings, interdistances, relative orientations, and components of the eigenvector  $\nu_6$ , i.e., all quantities available to the robot group. In particular, the quantities in items 1)–2) are either locally available to agent  $v$ , or can be computed<sup>5</sup> by exploiting communication with its 1-hop (communication) neighbors  $j \in \mathcal{N}_v$  (the agents measured by  $v$ ) and  $j \in \mathcal{O}_v$  (the agents measuring  $v$ ). The quantities in item 3) (which are ultimately due to weights  $w_{V_{ij}}$  in (6.7)) are instead relative to the 1-hop neighbors  $m \in \mathcal{N}_j / \{v\}$  of any agent  $j \in \mathcal{O}_v$ . Therefore, agent  $v$  needs to receive this information from some

<sup>5</sup>We recall that, as discussed in Sect. 6.2, the interdistances  $d_{ij}$  and relative orientations  $\psi_{ij}$  can be computed by any neighboring pair by exchanging the local estimates  $\hat{\mathbf{p}}_i, \hat{\mathbf{p}}_j, \hat{\psi}_i, \hat{\psi}_j$ .

of its 2-hop communication neighbors. In any case, since the amount of information in 1)–3) is *constant* per neighbor, an upper bound of the communication complexity for an agent  $v$  is  $O(|\mathcal{N}_v| + |\mathcal{O}_v| \cdot \max_{j \in \mathcal{O}_v} |\mathcal{N}_j|)$ . We can then conclude that the rigidity maintenance controller (6.23) admits a decentralized implementation by, however, assuming a 2-hop communication model.

A second remark is about the availability, for each agent  $v$ , of  $\lambda_6^B$  (a *global quantity*) and of its own eigenvector components  $(\nu_{p_v}, \nu_{\psi_v})$ : although these quantities are not directly measurable, it is in principle conceivable to *estimate* them in a decentralized way by adapting the estimators presented in [34, 155] for the connectivity case, and ported in [119] to the (distance) rigidity case. Indeed, these methods essentially require an explicit characterization of the null-space of the bearing rigidity matrix (which is well understood, see [27]), the use of some *PI consensus filters*, and the suitable exploitation of the structure of the symmetric bearing rigidity matrix. The estimation of  $\lambda_6^B$  and of  $(\nu_{p_v}, \nu_{\psi_v})$  is, however, left for future extensions of this work and in the following results we just assume availability of these quantities.

A final remark is about the well-posedness of controller (6.23): as well-known, the derivative of an eigenvalue is not well-defined for multiplicities larger than one (repeated eigenvalues), since in these cases one cannot reliably find/estimate a *unique* eigenvector to be plugged in the derivative computation [165]. This difficulty, which affects any method based on the optimization of a single eigenvalue associated to some matrix of interest, has already been recognized in [34, 119, 155] without, however, proposing an explicit solution to deal with it. A possible workaround is to replace the eigenvalue to be optimized with a ‘smoothed’ version [166] which is well-behaved for multiplicities larger than one. Following [146], a possibility is to replace  $\lambda_6^B$  with the quantity  $\bar{\lambda}^B = \sum_{i=6}^{4N} \sqrt[p]{(\lambda_i^B)^p}$  where  $\lambda_i^B$  is the  $i$ -th eigenvalue of the symmetric bearing rigidity matrix, and  $p \ll 0$ . Indeed, when  $\lambda_6^B \ll \lambda_7^B$  one has  $\bar{\lambda}^B \approx \lambda_6^B$  while when, instead,  $\lambda_6^B \approx \lambda_7^B \approx \lambda_m^B \ll \lambda_{m+1}^B$  it is  $\bar{\lambda}^B \approx \sqrt[p]{m-6} \lambda_6^B$ . Therefore, maximization of  $\bar{\lambda}^B$  results into maximization of  $\lambda_6^B$  with however the advantage of  $\bar{\lambda}^B$  being always differentiable. By evaluating  $V_\lambda(\cdot)$  on  $\bar{\lambda}^B$ , the controller (6.22) then becomes

$$\begin{cases} \mathbf{u}_v &= -\frac{\partial V_\lambda}{\partial \bar{\lambda}^B} \mathbf{R}_v^T \sum_{i=6}^{4N} \left( \frac{\lambda_i^B}{\bar{\lambda}^B} \right)^{p-1} \nabla_{\mathbf{p}_v} \lambda_i^B \\ \mathbf{w}_v &= -\frac{\partial V_\lambda}{\partial \bar{\lambda}^B} \sum_{i=6}^{4N} \left( \frac{\lambda_i^B}{\bar{\lambda}^B} \right)^{p-1} \nabla_{\psi_v} \lambda_i^B \end{cases} \quad (6.24)$$

Evaluation of (6.24) by an agent  $v$  would require the same quantities listed in 1)–3) and, in order to evaluate  $\partial V_\lambda / \partial \bar{\lambda}^B$  and  $\nabla_{\mathbf{q}_v} \lambda_i^B$ , the additional availability of  $\lambda_i^B$  and of  $(\nu_{p_v}^i, \nu_{\psi_v}^i)$  (the  $v$ -th components of the eigenvector  $\boldsymbol{\nu}^i$  associated to  $\lambda_i$ ) for  $i = 6 \dots 4N$ . Therefore any implementation of (6.24) would be decentralized but

not *scalable*, as the amount of information per agent would increase with the size of the whole agent group.

## 6.4 Experimental results

We employed a group of five quadrotors of the ones described in Sect. B.2. The TeleKyb framework [159] was used for implementing the low-level flight control receiving the body-frame velocity commands (6.24), and for exchanging data via Wi-Fi with the other robots in the group and the ground station. Each quadrotor was also equipped with an onboard camera, refer to Fig. B.5(b). The UAVs were flying in an indoor room equipped with the Vicon motion capture system (Fig. 6.7), refer to Sect. B.2 for more details. This was used for obtaining the ground truth and for reconstructing the body-frame bearing measurements  $\beta_{ij}$  that would have been obtained by the onboard cameras. Indeed, as others [105], we found the problem of detecting and tracking (in a reliable way) each quadrotor during flight to be non-trivial, and then exploited the onboard cameras only for the sake of verifying that, during motion, all the neighboring pairs remained indeed well visible in the corresponding camera field of view as predicted by the proposed machinery.

The on-board low-level controller ensures that the orientation of the UAV tracks the desired reference. As usual, this is accomplished by neglecting the couplings among the three body axes, and by treating each individual rotation (roll, pitch, yaw) as a separate channel modeled as a double integrator with input the correspondent body torque. The adopted controller is then a simple PID with saturated integral term in order to prevent wind-up issues. Readings from the on-board gyros are exploited as velocity feedback, while a complementary filter provides estimates of the UAV attitude by fusing together accelerometer and gyro readings from the IMU.

The experimental results are shown in Figs. 6.8 and 6.9: at the beginning the UAVs just followed the control action (6.24) and, as a consequence, the value of  $\bar{\lambda}^B(t)$  was maximized (until the vertical dashed line in Fig. 6.8(a)). Then two human operators started acting on two UAVs by sending two velocity commands added to (6.24) with the aim of triggering as much as possible the loss/gain of neighbors because of the considered sensing constraints. One can then note how presence of these two additional velocity inputs (which are representative of any external planner/algorithm in charge of steering the robot formation in the environment) did not threaten bearing rigidity of the formation thanks to the maintenance action (6.24) (indeed, as shown in Fig. 6.8(a),  $\bar{\lambda}(t)$  remains larger than the chosen threshold  $\bar{\lambda}^{\min} = 0.01$ ). This is also confirmed in Fig. 6.9(a) where the evolution of  $\lambda_6^B(t)$  for the *unweighted framework* is reported. Finally, Fig. 6.8(b) shows the number of edges  $|\mathcal{E}|$  over time, thus confirming the (intentionally induced) time-varying nature



Figure 6.7 – A group of five quadrotor UAVs flying in our flying arena at INRIA Rennes, Bretagne Atlantique, France

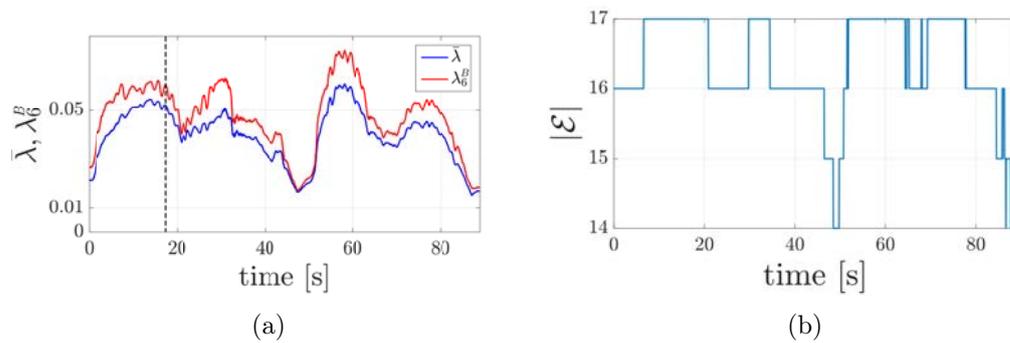


Figure 6.8 – **6.8(a)**: behavior of the  $p$ -norm  $\bar{\lambda}$ , of the rigidity eigenvalue  $\lambda_6^B$  **6.8(b)**: the number of edges  $|\mathcal{E}|$  of the considered framework.

of the sensing topology during motion, and Fig. 6.9(b) reports two of the graph topologies encountered during the experiment. The formation also goes through a rearranging process of its topology (Fig. 6.8(b)) by virtue of the *always* continuous evolution of the weights associated to its edges (Fig. 6.10).

## 6.5 Conclusions

In this chapter we illustrated a decentralized gradient-based controller able to enforce bearing rigidity maintenance for a group of quadrotor UAVs while coping with three typical sensing constraints of onboard cameras: minimum/maximum range, limited field of view, and possible occluded visibility. The proposed control action exploited a

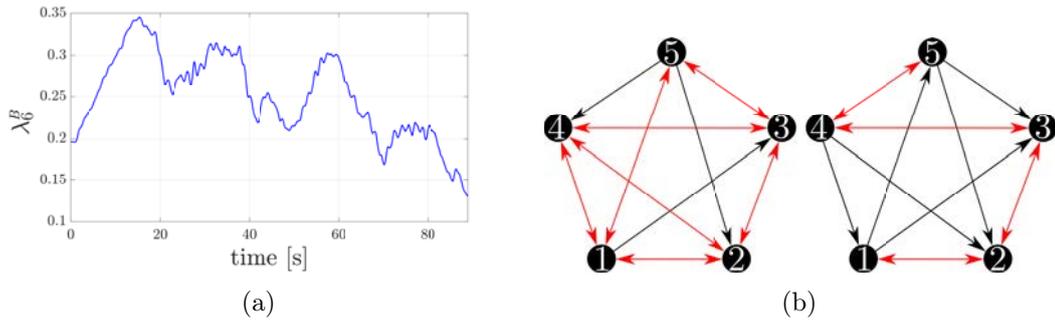


Figure 6.9 – 6.9(a): behavior of the *unweighted* rigidity eigenvalue  $\lambda_6^B$ , 6.9(b): two graph topologies corresponding to the maximum (left,  $|\mathcal{E}| = 17$ ) and minimum (right,  $|\mathcal{E}| = 14$ ) number of edges during the experiment.

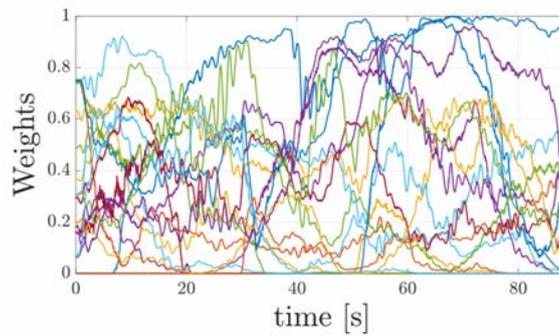


Figure 6.10 – Behavior of the weights over the edges in  $\mathcal{E}$  during the reported experiment. The reader should notice the *continuous nature* of the weights

closed-form expression of  $\lambda_6^B$ , the sixth smallest eigenvalue of the symmetric bearing rigidity matrix, and of its gradient with respect to the agent configuration. Moreover, a proof of independence of the bearing rigidity eigenvalue with respect to the agent orientations  $\psi$  was given. To the best of our knowledge, this represents the first work addressing the problem of bearing rigidity maintenance for a formation of UAVs by also explicitly taking into account (complex) sensing constraints. The reported experimental results showed the effectiveness of the proposed control strategy.

Several extensions of this work are possible: as pointed out in the previous chapter, a fully onboard implementation is still missing (refer to Chapt. 8 for more details). Furthermore, we would like to address the open points listed in Sect. 6.3.4, that is, decentralized estimation of the components of the body-frame eigenvector  $\nu_6$ , and the scalability issue of exploiting the  $p$ -norm  $\bar{\lambda}$  in place of  $\lambda_6^B$ . Finally, we are also considering the possibility of extending the proposed weighting machinery for including a *collision avoidance* action among quadrotors (which is now not guaranteed), similarly to what done in [119].

# Nonlinear observability and estimation for multi-agent systems

## Contents

7.1	Introduction . . . . .	94
7.1.1	Prior work . . . . .	94
7.1.2	Main contributions . . . . .	96
7.1.3	Chapter overview . . . . .	97
7.2	Preliminaries . . . . .	97
7.2.1	General notation . . . . .	97
7.2.2	Formation, agent and measurement model . . . . .	98
7.2.3	Elements of Riemannian geometry . . . . .	99
7.2.4	Elements of local nonlinear observability . . . . .	101
7.3	<i>Dynamic Bearing Observability Matrix</i> . . . . .	102
7.3.1	Matrix $\tilde{\mathbf{R}}_A$ . . . . .	103
7.3.2	Matrix $\tilde{\mathbf{R}}_B$ . . . . .	105
7.3.3	Numerical verification of the ranks of $\tilde{\mathbf{R}}_A$ and $\tilde{\mathbf{R}}$ . . . . .	109
7.4	A multi-agent Extended Kalman Filter . . . . .	110
7.5	Experimental results . . . . .	112
7.6	Conclusions and future works . . . . .	113

IN this chapter we still consider the problem of localization in multi-agent formations with bearing only measurements but specifically analyze the fundamental observability properties for dynamic agents. The current well-established approach is based on the so-called *rigidity matrix*, and its algebraic properties (e.g., its rank and nullspace). This method is typically motivated using first-order derivatives, and shows, among other facts, that the *global scale* of the formation is not observable.

This chapter shows that current results represent an incomplete view of the problem. In particular, we show that

- current methods are a particular instantiation of nonlinear observability theory
- we can introduce the concept of the *dynamic bearing observability matrix* from higher order derivatives to study the observability of dynamic formations
- the global scale is, in fact, generally observable when the agents move according to known inputs.

We use tools from Riemannian geometry and Lie group theory to tackle, in a general and principled way, the general formulation of the localization problem with states that include both rotations and translations. Finally, we verify our theoretical results by deriving and applying, in both simulations and real experiments on UAVs, a centralized Extended Kalman Filter on Lie groups that is able to estimate the global scale of a moving formation.

## 7.1 Introduction

As already discussed in Chapt. 5 and Sect. 2.5, a fundamental problem in the multi-robot applications is the one of *localization*.<sup>1</sup> This problem becomes of paramount importance, for instance, when a team of robots has to navigate autonomously in an unknown environment, or needs to collaborate on a physical task (e.g., transporting a load [57, 167, 168]). In this case, the robots need a precise localization with respect to other robots of the formation. In this chapter, as in the whole Thesis, we consider the case of bearing measurements coming from onboard calibrated monocular cameras. Our goal is to show that, despite the very limited information provided by this type of measurements (direction of relative translations alone), it is possible to reconstruct the full 3-D pose (rotation and translation) of the agents, including the global scale, up to a global gauge ambiguity. We demonstrate this 1) theoretically through an application of nonlinear observability analysis and Riemannian geometry, and 2) practically by applying an Extended Kalman Filter in both simulations and experiments.

### 7.1.1 Prior work

The problem of localization from bearing-only measurements has appeared in a variety of domains, such as (to cite a few) sensor network localization [88, 124, 169] and

---

<sup>1</sup>Remember that in this Thesis with *localization* we refer to the process of determining the configuration of the agents in a common reference frame from a sparse set of relative measurements between them, and without the aid of an external centralized system (such as Vicon/GNSS).

formation control [170–172] in controls and robotics, Structure from Motion [92, 93] in computer vision, and graph drawings [94] in discrete mathematics. Most of the literature has focused on the development of distributed algorithms (especially in the sensor network and robotics communities), but centralized solutions have also been considered (mostly in the computer vision community). In this chapter, rather than specific algorithms, we are interested in analyzing the fundamental aspects of the localization problem. In this direction, for our case of interest involving bearing-only measurement, there has been a considerable amount of work for developing a *theory of rigidity* [27, 28, 31, 72, 126, 126], which can predict what information can be recovered from the available measurements (i.e., whether the solution is “unique”). While most of these works considered only agents in 2-D, recent work has also studied the 3-D case [173]. The commonly accepted result is that when the number and connectivity of the measurement graph is sufficiently high (that is, when the graph is *rigid*), then, *for static agents*, the solution to the bearing-only localization problem is unique up to a *rototranslation* and a *contraction/expansion* of the whole formation. This is determined by considering the nullspace of a so-called *rigidity matrix*. Most of the existing works, however, do not explicitly consider the case of *dynamic* agents. At a high level, one could expect that if the agents know their own velocities in their own local frames (e.g., because they control them, or measure them using the onboard IMU), then they could use this *metric* information to avoid the scale ambiguity. Recent works [146] have pursued this idea, but do not provide a full, rigorous analysis rooted in nonlinear observability analysis and Riemannian geometry. One disadvantage of [146] is the presence of two estimators in *series* for estimating the scale of the formation (one filter estimates the distances over a selection of edges, and the second filter recovers a correctly scaled estimation of the formation configuration). Stability of this cascaded structure is difficult to prove (indeed, nothing is said in [146]), while our algorithm achieves the same result with only one *single* EKF. On the other hand, [146] is able to determine the optimal motion for the agents in order to maximize the observability of the scale factor. We plan to exploit the ideas in [146] for a similar characterization in the context of our EKF estimation. Nonetheless, ideas related to this approach have been successfully explored in the context of single-agent Simultaneous Localization and Mapping (SLAM) [144, 174] and localization from distance measurements [175].

In this regard, nonlinear observability (the problem of determining if the state of a nonlinear dynamical system can be reconstructed by knowing its inputs and outputs) is a classical topic in automatic controls [102, 154, 176], and it now constitutes textbook material [177]. However, it has never been explicitly applied to the problem of localization from bearing-only measurements. On the other hand, Riemannian geometry has been applied in the context of geometric control and estimation of

mechanical systems in general [178, 179], and quadrotors in particular [167, 180, 181]. A Riemannian geometry formulation has also been used for multiagent localization with unscaled relative poses [182], but it has never been applied together with nonlinear observability analysis to the bearing-only case.

Finally, in this work, we propose a validation of our theoretical results using an Extended Kalman Filter (EKF) for statistical filtering of states evolving on Riemannian manifolds. The main advantage of the EKF formulation is that it is relatively easy to derive; in fact, both centralized [183] and decentralized [184] implementations have been proposed specifically for multiagent systems, although without considering states evolving on Riemannian manifolds. On the other hand, it is known that the EKF is not optimal for nonlinear systems. Developing filtering techniques with optimality guarantees on Riemannian manifolds (and Lie groups in particular) is still an active field of research [185, 186]. Since the goal of this work is merely to use filtering as a validation of the theoretical derivations, we opted for a straight (although suboptimal), centralized (as opposed to distributed) application of the EKF, albeit with the explicit consideration of the Riemannian geometry of the states.

### 7.1.2 Main contributions

In this work, we make several contributions to the state of the art

- we show how the study of rigidity is, in fact, a particular instance of classical nonlinear observability analysis;
- using this insight, we propose the notion of *dynamic bearing observability matrix* (DBOM), which extends the standard notion of rigidity matrix for the case of moving agents with known inputs;
- by (numerically) analyzing the rank of the DBOM, we show that *the global scale of the formation is generally observable*;
- we show how tools from Riemannian geometry can be employed to carry out the observability analysis for states evolving in the space of rigid body motions  $SE(3)$ ;
- we derive and apply a centralized Extended Kalman Filter (EKF) on Riemannian manifolds that empirically verifies the theory (i.e., that shows that the global scale can be indeed recovered).

Overall, we show how bearing-only measurements (which, taken individually, do not contain any scale information) and local linear and angular velocity information

can be used to recover the entire state of the agents. The recovered state is up to a common rotation and translation (since all measurements are relative and do not have any relation to external reference systems, this last ambiguity appears to be unavoidable, even with dynamic agents). Moreover, by explicitly using the Riemannian geometry of the space of poses (which is based, among other elements, on the use of rotations matrices) throughout the chapter (both for the observability analysis and the statistical filter), we avoid the problems given by other representations (e.g., the singularities of Euler angles, and the non-uniqueness of quaternions). Finally, in this work, we do not perform a full, analytical characterization of the nullspace of the DBOM, and we do not consider distributed filtering solutions. However, these are interesting future directions that are enabled by the present work.

### 7.1.3 Chapter overview

This chapter is organized as follows. Sect. 7.2 reviews notions from several areas that are necessary to carry out our analysis. In Sect. 7.3 we introduce the novel concept of *Dynamic Bearing Observability Matrix*. Sect. 7.4 illustrates our EKF design. Finally, Sect. 7.5 reports experimental results with a group of quadrotor UAVs, followed by Sect. 7.6 that concludes the chapter and gives possible future directions.

## 7.2 Preliminaries

### 7.2.1 General notation

The localization problem was solved in the previous Chapt. 5 and 6 through the estimator described in Sect. 5.2.2 and by assuming that at least one distance was known (either measured or estimated through [146]). As shown in Sects. 5.3, 5.4 and 6.4 this solution works well, but something better can be done if we think of employing only one EKF which would estimate all the poses of the agents of the formation with the right scale and not the *cascade* of two estimators like in [146].

To design this EKF, as we said in the introduction of this chapter, we conducted an observability analysis which shows that the poses of the agents (with the right scale) are actually observable. This will then motivate the convergence of the designed EKF. We decided to conduct the observability analysis directly on the Lie group  $SE(3)$ . This was done for the sake of completeness, to avoid singularity problems related to the orientation and to use a representation of the orientation which is also global. Moreover, this will allow us to extend this analysis to complete models of the quadrotors (also with their full dynamics) and/or to other robots too.

Note that the representation of the pose of the quadrotor as an element of  $SE(3)$  is done both for the observability analysis and for the localization while the control can still be done through the model detailed in Sect. 4.2 once the localization phase is over. Then, as usual, let  $\mathcal{W}$  represent an absolute 3-D world reference frame, and  $\mathcal{A}_i$  represent a body reference frame attached to the  $i$ -th agent.

### 7.2.2 Formation, agent and measurement model

This chapter shares the same formation model in terms of graph theory of the previous Chapt. 5 and 6 which is described in Sect. 4.3. But, as just said, in this chapter there is a different modeling assumption with respect to Chapt. 4. Indeed, here we represent the orientation of the single agents in a different way. More specifically, we model the state of an agent  $i \in \mathcal{V}$  as a pose  $\mathbf{q}_i = (\mathbf{p}_i, \mathbf{R}_i)$ , where  $\mathbf{p}_i \in \mathbb{R}^3$  represents the translation of the origin of  $\mathcal{A}_i$  expressed in  $\mathcal{W}$ , and  $\mathbf{R}_i \in SO(3)$  represents the rotation transforming directions from  $\mathcal{A}_i$  to  $\mathcal{W}$ . Note that here  $\mathbf{R}_i$  is a function of the full orientation of the rigid body and not only of the yaw angle  $\psi_i$  as in Sect. 4.2. We denote the space of rigid poses as  $SE(3)$  (the detailed definition of  $SO(3)$  and its geometry is postponed to Sect. 7.2.3). Let us denote with  $\{\mathbf{e}_m\}_{m=1}^3$  the standard  $\mathbb{R}^3$  basis. We also let  $\mathbf{1}_N$  and  $\mathbf{I}_N$  represent a vector of all ones and the identity matrix of dimension  $N$ , respectively. The operator  $\text{stack}(\cdot)$  returns a matrix containing a vertical stacking of the arguments. We assume a simple first order model for the 6-D dynamics of each agent

$$\dot{\mathbf{q}}_i = \begin{pmatrix} \dot{\mathbf{p}}_i \\ \dot{\mathbf{R}}_i \end{pmatrix} = \begin{pmatrix} \mathbf{R}_i \mathbf{v}_i \\ \mathbf{R}_i \hat{\mathbf{w}}_i \end{pmatrix} = \quad (7.1)$$

$$= \sum_{k=1}^3 \begin{pmatrix} \mathbf{R}_i \mathbf{e}_k \\ 0 \end{pmatrix} v_{i_k} + \sum_{k=1}^3 \begin{pmatrix} 0 \\ \mathbf{R}_i \hat{\mathbf{e}}_k \end{pmatrix} w_{i_k}, \quad (7.2)$$

where  $\mathbf{v}_i, \mathbf{w}_i \in \mathbb{R}^3$  represent, respectively, the linear and angular velocities expressed in  $\mathcal{A}_i$ , and  $v_{i_k}, w_{i_k}$  represent their components along the  $\mathbf{e}_k$  basis vector. We use this model for generality, but the results of this work could be easily specialized to other cases (e.g., considering only the positions of the agents, or just the 2-D yaw angle, as done in the majority of previous works).

In this work, as in the previous chapters, we assume that each robot is equipped with a sensor (onboard calibrated camera) that allows it to measure the *relative bearing vector* (4.2) where the  $\mathbf{R}_i$  is now the *full* rotation matrix.

As in [27,30,31,146], we assume that we have only available the inputs  $\{\mathbf{v}_i, \mathbf{w}_i\}_{i \in \mathcal{V}}$ , and the measurements  $\{\beta_{ij}\}_{(i,j) \in \mathcal{E}}$ . In particular, we do not have access neither to the absolute states  $\mathbf{q}_i$ , nor to the global reference frame  $\mathcal{W}$ . Throughout the nonlinear observability analysis we will refer to the different components of the

vector  $\beta_{ij}$  as:

$$\beta_{ij_m} = \mathbf{e}_m^T \beta_{ij} \in \mathbb{R}, \quad m \in \{1, 2, 3\}. \quad (7.3)$$

**Remark 7.1.** *While we will individually consider each one of the three elements of each bearing  $\beta_{ij}$  as a separate output, in reality, the fact that  $\beta_{ij} \in \mathbb{S}^2$  implies that, in general, only two outputs are algebraically independent. The effect of this is that the bearing rigidity matrix that we will derive will contain more rows than strictly needed (i.e., some rows will be automatically linearly dependent). However, this does not change the result of the rank-based observability test.*

### 7.2.3 Elements of Riemannian geometry

This section covers the basic Riemannian geometry notions that are used in the derivations below. We will be mostly concerned with three manifolds: the Euclidean space  $\mathbb{R}^3$ , the space of 3-D rotations  $\text{SO}(3) = \{\mathbf{R} \in \mathbb{R}^{3 \times 3} : \mathbf{R}^T \mathbf{R} = \mathbf{I}, \det(\mathbf{R}) = 1\}$ , the space of 3-D poses  $\text{SE}(3) = \{(\mathbf{p}, \mathbf{R}) : \mathbf{R} \in \text{SO}(3), \mathbf{p} \in \mathbb{R}^3\}$ , and the space of  $N$  3-D poses  $\text{SE}(3)^N$ . These manifold are in fact *Lie groups*, but we will not make use of this fact.

**Tangent spaces** We denote as  $T_x \mathcal{M}$  the *tangent space* of a manifold  $\mathcal{M}$  at a point  $x \in \mathcal{M}$ . The tangent space at a point can be identified as the vector space spanned by the tangents of the curves passing through that point; for instance, if  $\mathbf{R}(t) : I \rightarrow \text{SO}(3)$  is a parametrized curve in  $\text{SO}(3)$  defined on some interval  $I \subset \mathbb{R}$  around zero, then  $\dot{\mathbf{R}}(0) \in T_{\mathbf{R}(0)} \text{SO}(3)$ . For  $\mathbb{R}^3$ , the tangent space at each point can be identified with  $\mathbb{R}^3$  itself. For  $\text{SO}(3)$  however, we first need to define the usual *hat*  $(\cdot)^\wedge$  and *vee*  $(\cdot)^\vee$  operators between  $\mathbb{R}^3$  and the set of skew-symmetric matrices in  $\mathbb{R}^{3 \times 3}$  as follows, with  $\mathbf{v} = [v_1, v_2, v_3]^T$ :

$$\mathbf{v}^{\mathbf{R}\wedge} = \mathbf{R} \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix} = \mathbf{R}\hat{\mathbf{v}}, \quad \mathbf{V}^{\mathbf{R}\vee} = (\mathbf{R}^T \mathbf{V})^\vee \quad (7.4)$$

where  $\mathbf{V}$  is any vector  $\mathbf{V} \in T_{\mathbf{R}} \text{SO}(3)$ . Note that, if  $\mathbf{R}$  is not present in the superscript, or if the  $\mathbf{R} = \mathbf{I}$ , the definitions of *hat* and *vee* operators are the classical ones present in the literature. It can be shown that the tangent space of  $\text{SO}(3)$  is given by

$$T_{\mathbf{R}} \text{SO}(3) = \{\mathbf{R}\hat{\mathbf{v}} : \mathbf{v} \in \mathbb{R}^3\}. \quad (7.5)$$

We define a basis for  $T_{\mathbf{R}} \text{SO}(3)$  as  $\{\mathbf{e}_m^{\mathbf{R}\vee}\}_{m=1}^3$ ; it follows that any vector  $\mathbf{V} \in T_{\mathbf{R}} \text{SO}(3)$  can be expressed as a vector  $\mathbf{v} \in \mathbb{R}^3$  of *local coordinates* in this basis with the relation  $\mathbf{v} = \mathbf{V}^{\mathbf{R}\vee}$ . For instance, for the curve  $\mathbf{R}(t)$  defined above, letting  $\mathbf{R}_0 = \mathbf{R}(0)$  we will have  $\dot{\mathbf{R}}(0) = \mathbf{w}^{\mathbf{R}_0\wedge}$  for some  $\mathbf{w} \in \mathbb{R}^3$ , or, equivalently,  $\mathbf{w} = \dot{\mathbf{R}}(0)^{\mathbf{R}_0\vee}$ . In the case

where  $\mathbf{R}(t)$  represents a physical time-varying rotation, using the convention given in Sec. 7.2.2, the vector  $\mathbf{w}$  coincides with a vector of angular velocities expressed in the body frame (see also (7.1)). The tangent space of  $T_q\text{SE}(3)$  can be identified with the direct sum  $\mathbb{R}^3 \oplus T_{\mathbf{R}}\text{SO}(3)$  (i.e., a tangent for  $\text{SE}(3)$  is simply a tangent for  $\mathbb{R}^3$  together with a tangent for  $\text{SO}(3)$ ). Similarly, the tangent space of  $\text{SE}(3)^N$  is simply the direct sum of  $N$  copies of  $T_q\text{SE}(3)$ . A representation in local coordinates of a vector  $T_q\text{SE}(3)$  can be obtained by stacking the local coordinate representation of each rotational component (as discussed above), with the translational components.

**Riemannian metrics** A Riemannian metric  $\langle \cdot, \cdot \rangle$  smoothly assigns an inner product to each tangent space. The standard Riemannian metric for  $\mathbb{R}^3$  is the usual inner product. The standard Riemannian metric for  $\text{SO}(3)$  is defined as

$$\langle \mathbf{R}\hat{\mathbf{v}}_1, \mathbf{R}\hat{\mathbf{v}}_2 \rangle = \frac{1}{2} \text{tr}(\hat{\mathbf{v}}_1^T \hat{\mathbf{v}}_2) = \mathbf{v}_1^T \mathbf{v}_2, \quad (7.6)$$

where  $\mathbf{R}\hat{\mathbf{v}}_1, \mathbf{R}\hat{\mathbf{v}}_2 \in T_{\mathbf{R}}\text{SO}(3)$  are two tangent vectors. For  $\text{SE}(3)$ , we use the metric given by the sum of the two previous metrics, and for  $\text{SE}(3)^N$ , the sum of the metrics for each copy of  $\text{SE}(3)$ .

**Gradients and how to compute them** The *gradient* of a differentiable function  $f(x)$ ,  $f: \mathcal{M} \rightarrow \mathbb{R}$  computed at a point  $x_0$  on a manifold  $\mathcal{M}$  is defined as the unique tangent vector  $\nabla_x f(x_0)$  such that, for all curves  $x(t)$  with  $x(0) = x_0$ ,

$$\langle \nabla_x f(x_0), \dot{x}(0) \rangle = \left. \frac{d}{dt} f(x(t)) \right|_{t=0}. \quad (7.7)$$

For  $\mathbb{R}^3$ , it can be shown that this definition coincides with the more common definition as a vector of partial derivatives. For  $\text{SO}(3)$ , we can use (7.7) to compute gradients of any arbitrary function  $f$  in a few steps. First, we consider a fictitious parametrized curve  $\mathbf{R}(t)$  in  $\text{SO}(3)$ . Then we use the chain rule to compute  $\frac{d}{dt} f(\mathbf{R}(t))$ , the derivative of the function along the curve. It can be shown that this derivative (when it exists) can always be written as

$$\frac{d}{dt} f(\mathbf{R}) = \text{tr}(\mathbf{M}^T \dot{\mathbf{R}}) = \text{tr}(\text{skew}(\mathbf{R}^T \mathbf{M})^T \mathbf{R}^T \dot{\mathbf{R}}), \quad (7.8)$$

where  $\mathbf{M}$  is some matrix in  $\mathbb{R}^{3 \times 3}$  which in general depends on  $\mathbf{R}$ ,  $\text{skew}(\mathbf{A}) = \frac{1}{2}(\mathbf{A} - \mathbf{A}^T)$  is an operator that extracts the skew-symmetric component of a matrix, and the explicit dependency on  $t$  has been omitted for brevity. The first equality in (7.8) comes from the fact that the differential of a map (gradients are a particular case of differentials) are always linear maps [187], and they can be expressed as linear functionals using the trace operator [188]. The second equality in (7.8) comes from the characterization of  $T_{\mathbf{R}}\text{SO}(3)$  given in (7.5) and the fact that  $\text{tr}(\mathbf{S}\hat{\mathbf{v}}) = 0$

for any symmetric matrix  $\mathbf{S} \in \mathbb{R}^{3 \times 3}$  and  $\mathbf{v} \in \mathbb{R}^3$ . Comparing (7.8) with (7.7), we then obtain  $\nabla_{\mathbf{R}} f(\mathbf{R}_0) = 2\mathbf{R} \text{skew}(\mathbf{R}^T \mathbf{M})$  (this is equivalent to the formula given, e.g., in [189]); in local coordinates, this becomes  $2 \text{skew}(\mathbf{R}^T \mathbf{M})^\vee$ . Informally, we refer to this set of steps as the *trace trick*. For  $\text{SE}(3)$  and  $\text{SE}(3)^N$ , the computation of the gradient reduces to a separate computation for each component.

**Remark 7.2.** *In this chapter we represent rotations using rotation matrices. Compared to other representations (such as Euler angles or quaternions), this representation is unambiguous, does not have singularities, and, as shown above, provides a relatively straightforward way to compute gradients (see, e.g., [181] for additional insights).*

### 7.2.4 Elements of local nonlinear observability

Let  $\mathbf{q} = (\mathbf{q}_1, \dots, \mathbf{q}_N) \in \text{SE}(3)^N$  (according to remark 2),  $\boldsymbol{\beta} = \text{stack}(\{\beta_{ij}\}) \in \mathbb{R}^{3|\mathcal{E}|}$ , and the *input vector fields*  $\mathbf{g}_{v_{i_k}}, \mathbf{g}_{w_{i_k}}$  on  $\text{SE}(3)^N$  which are obtained by appropriately padding with zeros the corresponding vectors in (7.2). Then, the dynamical model of the entire network can be considered as a nonlinear system with affine inputs:

$$\dot{\mathbf{q}} = \sum_{i \in \mathcal{V}} \sum_{k=1}^3 (\mathbf{g}_{v_{i_k}} v_{i_k} + \mathbf{g}_{w_{i_k}} w_{i_k}), \quad (7.9)$$

$$\boldsymbol{\beta} = \mathbf{h}(\mathbf{q}),$$

where  $\mathbf{h} : \text{SE}(3)^N \rightarrow \mathbb{R}^{3|\mathcal{E}|}$ ,  $\mathbf{h} = \text{stack}(\{\mathbf{h}_{ij}\})$ .

The goal of nonlinear observability theory [174] applied to problem (7.9) is to determine what parts of the state  $\mathbf{q}$  can be reconstructed from the outputs  $\boldsymbol{\beta}$  and the inputs  $\{\mathbf{v}_i, \mathbf{w}_i\}$ . Following the notation of [174], we indicate the  $k$ th order *Lie derivative* of a function  $h$  along the vector fields  $\mathbf{f}_1, \dots, \mathbf{f}_k$  as  $L_{\mathbf{f}_1, \dots, \mathbf{f}_k}^k h$ . The definition of Lie derivative is given by letting  $L^0 h = h$ , and then, recursively:

$$dL_{\mathbf{f}_1, \dots, \mathbf{f}_k}^k h = \nabla_{\mathbf{q}} L_{\mathbf{f}_1, \dots, \mathbf{f}_k}^k h, \quad (7.10)$$

$$L_{\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_{k+1}}^{k+1} h = \langle dL_{\mathbf{f}_1, \dots, \mathbf{f}_k}^k h, \mathbf{f}_{k+1} \rangle, \quad (7.11)$$

where  $\langle \cdot, \cdot \rangle$  is the Riemannian metric on  $\text{SE}(3)^N$  described in Sect. 7.2.3, and  $dL_{\mathbf{f}_1, \dots, \mathbf{f}_k}^k h$  is a shorthand notation for the gradient of a Lie derivative. In nonlinear observability, the function  $h$  is set to be an output of the system, and the vector fields  $\mathbf{f}_1, \dots, \mathbf{f}_k$  are taken to be the input vector fields.

In our case, we consider each  $\beta_{ijm}$  (that is, each element of  $\boldsymbol{\beta}$ ) as a separate output of the system. In order to carry out the local nonlinear observability analysis at a particular configuration  $\mathbf{q}$ , it is necessary to define the subspace

$$d\Omega = \text{span}(\{dL_{\mathbf{f}_1, \dots, \mathbf{f}_k}^k \beta_{ijm}\}), \quad (7.12a)$$

$$\text{where } k \in \{0, 1, \dots\}, (i, j) \in \mathcal{E}, m \in \{1, 2, 3\}, \quad (7.12b)$$

$$\mathbf{f}_1, \dots, \mathbf{f}_k \in \{\mathbf{g}_{v_{i_k}}, \mathbf{g}_{w_{i_k}}\}. \quad (7.12c)$$

As discussed in [177, Sect. 1.9], it is sufficient to consider Lie derivatives up to the order  $k = 6N - 1$  (where  $6N$  is the dimension of the system (7.9)). In practice, we will numerically verify that  $k = 1$  is already sufficient under general conditions to show that the global scale of the system can be recovered. We define  $d\Omega^\perp$  to be the *annihilator* of  $d\Omega$ , that is, the subspace of  $T_{\mathbf{q}}\text{SE}(3)^N$  such that

$$\langle v, n \rangle = 0 \quad \forall v \in d\Omega, n \in N \quad (7.13)$$

The annihilator tells us the *locally unobservable modes* of the system [174, 177], that is, what variations of the state  $\mathbf{q}$  cannot be observed under any choice of the inputs (and their Lie brackets). In practice, to find  $d\Omega^\perp$  and compute its dimension, we need to switch to a local coordinate representation. To avoid introducing additional notation, we redefine  $dL_{\mathbf{f}_1, \dots, \mathbf{f}_k}^k h$  to be the vector in  $\mathbb{R}^{6N}$  of local coordinates (as opposed to an abstract tangent vector in  $T_{\mathbf{q}}\text{SE}(3)^N$ ). Similarly, we redefine  $d\Omega$  as a matrix (the original subspace  $d\Omega$  is given by the row span of this matrix):

$$d\Omega = \text{stack}(\{dL_{\mathbf{f}_1, \dots, \mathbf{f}_k}^k \beta_{ijm}\}), \quad (7.14)$$

where the indexes are the same as in (7.12). Intuitively, the matrix  $d\Omega$  generalizes the classical observability matrix used for linear systems [154, 176, 177]. Finally, the annihilator  $d\Omega^\perp$  is redefined to be the nullspace of  $d\Omega$ ,  $d\Omega^\perp = \text{null}(d\Omega)$ .

In the next section we will give the details of the computation of  $d\Omega$  for Lie derivatives of order up to  $k = 1$  for our system (7.9).

### 7.3 Dynamic Bearing Observability Matrix

In this section we introduce the notion of *Dynamic Bearing Observability Matrix*<sup>2</sup> (DBOM). We define the DBOM  $\tilde{\mathbf{R}}$  to be equal to the matrix  $d\Omega$  computed with the gradients of Lie derivatives of order up to  $k = 1$ . More explicitly:

$$\tilde{\mathbf{R}} = \text{stack}(\{dL^0 \beta_{ijm}\}, \{dL_{\mathbf{f}_1}^1 \beta_{ijm}\}) = \text{stack}(\tilde{\mathbf{R}}_A, \tilde{\mathbf{R}}_B), \quad (7.15)$$

where the matrix  $\tilde{\mathbf{R}}_A$  and  $\tilde{\mathbf{R}}_B$  contain the gradients of the Lie derivative of order, respectively,  $k = 0$  and  $k = 1$ , and will be described in detail in Sects. 7.3.1 and 7.3.2.

We anticipate here that the matrix  $\tilde{\mathbf{R}}_A$  is equivalent to the traditional *bearing rigidity matrix* first introduced in [150], and then expanded upon in various works (e.g., [27, 28, 173]). However, here we

<sup>2</sup>We chose this name to stress that the scale of a formation based on bearing measurements is retrievable through *dynamic* information.

1. derive it for full 6-D states in SE(3), using a clear interpretation with respect to the Riemannian geometry of the space, and
2. give it the interpretation of a first step in a full nonlinear observability analysis.

Intuitively, since this matrix includes only zeroth order Lie derivatives, its properties tell us which parts of the state can be estimated without providing any input (static agents). As expect from previous works, and compatibly with the intuition, global scaling (contraction/expansion) of the formation generate tangent vectors that are in the nullspace of  $\tilde{\mathbf{R}}_A$ , meaning that the global scale is not observable.

The matrix  $\tilde{\mathbf{R}}_B$  constitutes the main novelty in our analysis. Intuitively, since this matrix includes first order Lie derivatives, its properties tell us what can be estimated by moving the agents with constant inputs. As we will numerically verify in the following, this matrix contributes to reducing the dimension of the nullspace of  $\tilde{\mathbf{R}}$  by one. The direction that is removed corresponds exactly to the contraction/expansion motion. This is compatible with the intuition above: when agents move, there is a parallax effect that can be exploited to get an estimate of the unknown scales.

### 7.3.1 Matrix $\tilde{\mathbf{R}}_A$

In order to compute the matrix  $\tilde{\mathbf{R}}_A$ , since the zeroth-order Lie derivatives  $L^0\beta_{ijm}$  are simply equal to the function themselves, we can directly focus on computing their gradients  $\nabla_{\mathbf{q}}L^0\beta_{ijm}$ . For this purpose, we will use the trace-trick method. Assuming that  $\beta_{ij}$  moves along a fictitious curve  $\beta_{ij}(t)$ , we compute the following:

$$\begin{aligned} \frac{d}{dt}L^0\beta_{ijm} &= \langle \nabla_{\mathbf{p}_i}L^0\beta_{ijm}, \dot{\mathbf{p}}_i \rangle + \langle \nabla_{\mathbf{p}_j}L^0\beta_{ijm}, \dot{\mathbf{p}}_j \rangle \\ &\quad \langle \nabla_{\mathbf{R}_i}L^0\beta_{ijm}, \dot{\mathbf{R}}_i \rangle + \langle \nabla_{\mathbf{R}_j}L^0\beta_{ijm}, \dot{\mathbf{R}}_j \rangle = \mathbf{e}_m^T \dot{\beta}_{ij} \end{aligned} \quad (7.16)$$

which can be written also as

$$\mathbf{e}_m^T \dot{\beta}_{ij} = \mathbf{e}_m^T \dot{\mathbf{R}}_i^T \frac{\mathbf{p}_{ij}}{d_{ij}} + \mathbf{e}_m^T \frac{d}{dt} \left( \frac{\mathbf{p}_{ij}}{d_{ij}} \right). \quad (7.17)$$

The  $\nabla_{\mathbf{p}_i}L^0\beta_{ijm}$  comes from the computation of  $\frac{\partial\beta_{ij}}{\partial\mathbf{p}_i}$  which is reported for clarity here

$$\frac{\partial\beta_{ij}}{\partial\mathbf{p}_i} = \frac{\partial}{\partial\mathbf{p}_i} \left( \mathbf{R}_i^T \frac{\mathbf{p}_{ij}}{d_{ij}} \right) = \frac{\mathbf{R}_i^T}{d_{ij}^2} \left[ \left( -\frac{\partial\mathbf{p}_i}{\partial\mathbf{p}_i} \right) d_{ij} - \left( \frac{\partial\mathbf{p}_i}{\partial d_{ij}} \right) \mathbf{p}_{ij} \right] = \frac{-\mathbf{P}_{ij}\mathbf{R}_i^T}{d_{ij}} \quad (7.18)$$

where  $\mathbf{P}_{ij} = \mathbf{I}_3 - \beta_{ij}\beta_{ij}^T$  is the orthogonal projector onto the orthogonal complement of  $\beta_{ij}$ . In the same way one can compute  $\frac{\partial\beta_{ij}}{\partial\mathbf{p}_j}$  which gives birth to  $\nabla_{\mathbf{p}_j}L^0(\mathbf{e}_m^T\beta_{ij})$ .

On the other hand, regarding the gradient  $\nabla_{\mathbf{R}_i}L^0\beta_{ijm}$  we need to focus our attention on the part of  $\mathbf{e}_m^T\dot{\beta}_{ij}$  which multiplies the  $\dot{\mathbf{R}}_i$ .

Before going into the details of the computation of  $\nabla_{\mathbf{R}_i} L^0 \beta_{ijm}$  we want to highlight some useful identities. Let  $\mathbf{u}, \mathbf{v} \in \mathbb{R}^d$  represent two generic vectors and  $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{d \times d}$  two generic matrices. Then, the following identities are satisfied

$$\mathbf{u}^T \mathbf{v} = \text{tr}(\mathbf{u}^T \mathbf{v}) = \text{tr}(\mathbf{v} \mathbf{u}^T) \quad (7.19)$$

$$\text{tr}(\hat{\mathbf{u}}^T \hat{\mathbf{v}}) = -\text{tr}(\hat{\mathbf{u}} \hat{\mathbf{v}}) \quad (7.20)$$

$$\text{tr}(\mathbf{A} \mathbf{B}) = \text{tr}(\mathbf{B} \mathbf{A}) \quad (7.21)$$

and that

$$\dot{\mathbf{R}}_i = \mathbf{R}_i \hat{\boldsymbol{\omega}}_i \implies \dot{\mathbf{R}}_i^T = \hat{\boldsymbol{\omega}}_i^T \mathbf{R}_i^T \implies \dot{\mathbf{R}}_i^T \mathbf{R}_i = \hat{\boldsymbol{\omega}}_i^T. \quad (7.22)$$

Therefore, we can write

$$\begin{aligned} \mathbf{e}_m^T \dot{\mathbf{R}}_i^T \frac{\mathbf{P}_{ij}}{d_{ij}} &= \text{tr} \left( \mathbf{e}_m^T \dot{\mathbf{R}}_i^T \frac{\mathbf{P}_{ij}}{d_{ij}} \right) = \text{tr} \left( \dot{\mathbf{R}}_i^T \frac{\mathbf{P}_{ij}}{d_{ij}} \mathbf{e}_m^T \right) = \text{tr} \left( \dot{\mathbf{R}}_i^T \mathbf{R}_i \mathbf{R}_i^T \frac{\mathbf{P}_{ij}}{d_{ij}} \mathbf{e}_m^T \right) = \\ &= \text{tr} \left( \hat{\boldsymbol{\omega}}_i^T \mathbf{R}_i^T \mathbf{R}_i \boldsymbol{\beta}_{ij} \mathbf{e}_m^T \right) = \text{tr} \left( \hat{\boldsymbol{\omega}}_i^T \boldsymbol{\beta}_{ij} \mathbf{e}_m^T \right) = -\text{tr} \left( \hat{\boldsymbol{\omega}}_i \boldsymbol{\beta}_{ij} \mathbf{e}_m^T \right) = -\text{tr} \left( \boldsymbol{\beta}_{ij} \mathbf{e}_m^T \hat{\boldsymbol{\omega}}_i \right) \end{aligned} \quad (7.23)$$

The last term of (7.23) does not represent an inner product because it is missing a  $\frac{1}{2}$ . We can therefore rewrite the (7.23) as

$$-\text{tr} \left( \boldsymbol{\beta}_{ij} \mathbf{e}_m^T \hat{\boldsymbol{\omega}}_i \right) = -\frac{1}{2} \text{tr} \left( 2 \boldsymbol{\beta}_{ij} \mathbf{e}_m^T \hat{\boldsymbol{\omega}}_i \right) \quad (7.24)$$

the argument of the trace in the (7.24) has the *shape* of a *tangent vector* of  $\text{SO}(3)$  and therefore we can finally extract the gradient  $\nabla_{\mathbf{R}_i} L^0 \beta_{ijm}$  through the *trace trick* described in Sect. 7.2.3.

So, we can extract the desired gradients (in local coordinates):

$$\nabla_{\mathbf{p}_i} L^0 (\mathbf{e}_m^T \boldsymbol{\beta}_{ij}) = -\frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T, \quad (7.25)$$

$$\nabla_{\mathbf{p}_j} L^0 (\mathbf{e}_m^T \boldsymbol{\beta}_{ij}) = +\frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T, \quad (7.26)$$

$$\nabla_{\mathbf{R}_i} L^0 (\mathbf{e}_m^T \boldsymbol{\beta}_{ij}) = -\left( \text{skew} \left( 2 \boldsymbol{\beta}_{ij} \mathbf{e}_m^T \right)^\vee \right)^T, \quad (7.27)$$

$$\nabla_{\mathbf{R}_j} L^0 (\mathbf{e}_m^T \boldsymbol{\beta}_{ij}) = \mathbf{0}, \quad (7.28)$$

Equations (7.25)–(7.28) then give the  $(1 \times 6N)$  row block of  $\tilde{\mathbf{R}}_A$  associated to the edge  $(i, j) \in \mathcal{E}$ , which has the following form

$$\tilde{\mathbf{R}}_{A_{ijm}} = \left[ \mathbf{0} \quad -\frac{\mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T}{d_{ij}} \quad -\left( \text{skew} \left( 2 \boldsymbol{\beta}_{ij} \mathbf{e}_m^T \right)^\vee \right)^T \quad \mathbf{0} \quad \frac{\mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T}{d_{ij}} \quad \mathbf{0} \quad \mathbf{0} \right], \quad (7.29)$$

where the blocks are ordered following the ordering of the translational and rotational states in  $\mathbf{q}$ . Therefore the matrix  $\tilde{\mathbf{R}}_A$  can be seen as

$$\tilde{\mathbf{R}}_A = \text{stack}(\nabla_{\mathbf{q}} L^0 \beta_{ijm}) = \text{stack}(\tilde{\mathbf{R}}_{A_{ijm}}), \quad (i, j) \in \mathcal{E}, \quad m \in \{1, 2, 3\} \quad (7.30)$$

Notice that the block row in (7.29) is expressed in the frame  $\mathcal{W}$ . It is also possible to express it in the local frame using the same ideas as [28].

### 7.3.2 Matrix $\tilde{\mathbf{R}}_B$

In order to compute the matrix  $\tilde{\mathbf{R}}_B$ , it is necessary to first compute the first-order Lie derivatives  $L_{\mathbf{f}_1}^1 \beta_{ijm} = \langle \nabla_{\mathbf{q}} L^0 \beta_{ijm}, \mathbf{f}_1 \rangle$ , where  $\mathbf{f}_1 \in \{\mathbf{g}_{v_{i_k}}, \mathbf{g}_{w_{i_k}}\}_{i \in \mathcal{V}}$ . Note that, for a given  $i, j$ , all these Lie derivative are zero except for  $\mathbf{f}_1 \in \{\mathbf{g}_{v_{i_k}}, \mathbf{g}_{w_{i_k}}, \mathbf{g}_{v_{j_k}}\}$  (for  $\mathbf{f}_1 = \mathbf{g}_{w_{j_k}}$ , the Lie derivative is zero due to (7.28)). We can therefore consider only the latter ones. For instance, let us focus on the case  $\mathbf{f}_1 = \mathbf{g}_{v_{i_k}}$ . To compute the gradient of  $L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}$  we can employ again the *trace-trick* method described in Sect. 7.2.3.

$$\begin{aligned} \frac{d}{dt} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm} &= \langle \nabla_{\mathbf{p}_i} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}, \dot{\mathbf{p}}_i \rangle + \langle \nabla_{\mathbf{p}_j} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}, \dot{\mathbf{p}}_j \rangle + \\ &\quad + \langle \nabla_{\mathbf{R}_i} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}, \dot{\mathbf{R}}_i \rangle + \langle \nabla_{\mathbf{R}_j} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}, \dot{\mathbf{R}}_j \rangle \end{aligned} \quad (7.31)$$

Notice that, however, while expanding above we will obtain terms that depend on  $\dot{\beta}_{ij}$ . Similarly to what we mentioned in Sect. 7.2.3, one can show that this dependency is linear. More explicitly, we can rewrite (7.31) as

$$\begin{aligned} \frac{d}{dt} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm} &= \langle \bar{\nabla}_{\mathbf{p}_i} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}, \dot{\mathbf{p}}_i \rangle + \langle \bar{\nabla}_{\mathbf{p}_j} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}, \dot{\mathbf{p}}_j \rangle + \\ &\quad + \langle \bar{\nabla}_{\mathbf{R}_i} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}, \dot{\mathbf{R}}_i \rangle + \langle \bar{\nabla}_{\mathbf{R}_j} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}, \dot{\mathbf{R}}_j \rangle + \mathbf{K}_{ij} \dot{\beta}_{ij}, \end{aligned} \quad (7.32)$$

where  $\mathbf{K}_{ij}$  is a matrix in  $\mathbb{R}^{3 \times 3}$ . We can exploit (7.32) to compute the rank of the DBOM  $\tilde{\mathbf{R}}$ , while avoiding the explicit computation of the terms in  $\mathbf{K}_{ij}$ , thus simplifying the analytical expressions involved; this is because we can collect all the  $\{\mathbf{K}_{ij}\}_{(i,j) \in \mathcal{E}}$  into a  $3|\mathcal{E}| \times 3|\mathcal{E}|$  matrix  $\mathbf{K} = \text{diag}(\{\mathbf{K}_{ij}\})$ , and then rewrite (7.15) as

$$\tilde{\mathbf{R}} = \begin{bmatrix} \tilde{\mathbf{R}}_A \\ \tilde{\mathbf{R}}_C + \mathbf{K} \tilde{\mathbf{R}}_A \end{bmatrix}, \quad (7.33)$$

where  $\tilde{\mathbf{R}}_C$  is defined in the same way as  $\tilde{\mathbf{R}}_B$ , but by using the modified (and analytically simpler) gradients  $\bar{\nabla}_{\mathbf{q}}$  instead of the full gradients  $\nabla_{\mathbf{q}}$ . Eq. (7.33) implies that

$$\text{rank} \begin{bmatrix} \tilde{\mathbf{R}}_A \\ \tilde{\mathbf{R}}_B \end{bmatrix} = \text{rank} \left( \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{K} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{R}}_A \\ \tilde{\mathbf{R}}_C \end{bmatrix} \right) = \text{rank} \begin{bmatrix} \tilde{\mathbf{R}}_A \\ \tilde{\mathbf{R}}_C \end{bmatrix} \quad (7.34)$$

Hence, for our purposes, we can compute the rows of  $\tilde{\mathbf{R}}_C$  instead of those of  $\tilde{\mathbf{R}}_B$ . As for the  $\tilde{\mathbf{R}}_A$  in the (7.30), the  $\tilde{\mathbf{R}}_C$  can be written as

$$\tilde{\mathbf{R}}_C = \text{stack}(\bar{\nabla}_q L_{\mathbf{f}_1}^1 \beta_{ijm}) = \text{stack}(\tilde{\mathbf{R}}_{C_{ijm}}), \quad (7.35)$$

where  $(i, j) \in \mathcal{E}$ ,  $m \in \{1, 2, 3\}$ ,  $\mathbf{f}_1 \in \{\mathbf{g}_{v_k}, \mathbf{g}_{w_k}\}_{k \in \mathcal{V}}$

where  $\tilde{\mathbf{R}}_{C_{ijm}}$  is

$$\begin{aligned} \tilde{\mathbf{R}}_{C_{ijm}} = & \\ \text{stack} & \left( \bar{\nabla}_q L_{\mathbf{g}_{v_{i_1}}}^1 \beta_{ijm}, \bar{\nabla}_q L_{\mathbf{g}_{v_{i_2}}}^1 \beta_{ijm}, \bar{\nabla}_q L_{\mathbf{g}_{v_{i_3}}}^1 \beta_{ijm}, \right. \\ & \bar{\nabla}_q L_{\mathbf{g}_{v_{j_1}}}^1 \beta_{ijm}, \bar{\nabla}_q L_{\mathbf{g}_{v_{j_2}}}^1 \beta_{ijm}, \bar{\nabla}_q L_{\mathbf{g}_{v_{j_3}}}^1 \beta_{ijm}, \\ & \bar{\nabla}_q L_{\mathbf{g}_{w_{i_1}}}^1 \beta_{ijm}, \bar{\nabla}_q L_{\mathbf{g}_{w_{i_2}}}^1 \beta_{ijm}, \bar{\nabla}_q L_{\mathbf{g}_{w_{i_3}}}^1 \beta_{ijm}, \\ & \left. \bar{\nabla}_q L_{\mathbf{g}_{w_{j_1}}}^1 \beta_{ijm}, \bar{\nabla}_q L_{\mathbf{g}_{w_{j_2}}}^1 \beta_{ijm}, \bar{\nabla}_q L_{\mathbf{g}_{w_{j_3}}}^1 \beta_{ijm} \right) \end{aligned} \quad (7.36)$$

Just to give an example, the first argument of the  $\text{stack}(\cdot)$  operator in the (7.36) would be

$$\begin{aligned} \nabla_q L_{\mathbf{g}_{v_k}}^1 \beta_{ijm} = & \\ \text{stack} & \left( -\mathbf{0} - \bar{\nabla}_{\mathbf{p}_i} L_{\mathbf{g}_{v_k}}^1 \beta_{ijm} \quad -\mathbf{0} - \bar{\nabla}_{\mathbf{p}_j} L_{\mathbf{g}_{v_k}}^1 \beta_{ijm} \right. \\ & \left. -\mathbf{0} - \bar{\nabla}_{\mathbf{R}_i} L_{\mathbf{g}_{v_k}}^1 \beta_{ijm} \quad -\mathbf{0} - \bar{\nabla}_{\mathbf{R}_j} L_{\mathbf{g}_{v_k}}^1 \beta_{ijm} \right)^T. \end{aligned} \quad (7.37)$$

The other terms of the (7.36), of course, will have the same structure.

In order to show an analytical expression of the matrix  $\tilde{\mathbf{R}}_C$  we need to rewrite the (7.32) as

$$\begin{aligned} \frac{d}{dt} L_{\mathbf{g}_{v_k}}^1 \beta_{ijm} &= \frac{d}{dt} \left[ -\frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \mathbf{R}_i \mathbf{e}_k \right] = \\ & - \left[ \left( \frac{\partial}{\partial \mathbf{p}_i} \frac{1}{d_{ij}} \right) \cdot \dot{\mathbf{p}}_i + \left( \frac{\partial}{\partial \mathbf{p}_j} \frac{1}{d_{ij}} \right) \cdot \dot{\mathbf{p}}_j \right] \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \mathbf{R}_i \mathbf{e}_k + \\ & \frac{1}{d_{ij}} \mathbf{e}_m^T \underbrace{\left[ \left( \dot{\beta}_{ij} \right) \beta_{ij}^T + \beta_{ij} \left( \dot{\beta}_{ij}^T \right) \right]}_{\dot{\mathbf{P}}_{ij}} \mathbf{R}_i^T \mathbf{R}_i \mathbf{e}_k + \\ & - \frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \dot{\mathbf{R}}_i^T \mathbf{R}_i \mathbf{e}_k - \frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \dot{\mathbf{R}}_i \mathbf{e}_k = \\ & - \left[ \left( \frac{\mathbf{R}_i \beta_{ij}}{d_{ij}^2} \right) \cdot \dot{\mathbf{p}}_i + \left( -\frac{\mathbf{R}_i \beta_{ij}}{d_{ij}^2} \right) \cdot \dot{\mathbf{p}}_j \right] \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \mathbf{R}_i \mathbf{e}_k + \\ & + \frac{1}{d_{ij}} \mathbf{e}_m^T \dot{\beta}_{ij} \beta_{ij}^T \mathbf{R}_i^T \mathbf{R}_i \mathbf{e}_k + \frac{1}{d_{ij}} \mathbf{e}_m^T \beta_{ij} \dot{\beta}_{ij}^T \mathbf{R}_i^T \mathbf{R}_i \mathbf{e}_k + \\ & - \frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \dot{\mathbf{R}}_i^T \mathbf{R}_i \mathbf{e}_k - \frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \dot{\mathbf{R}}_i \mathbf{e}_k. \end{aligned} \quad (7.38)$$

Focusing on the right hand side of the (7.38)

- from the first two terms, which multiply respectively  $\dot{\mathbf{p}}_i, \dot{\mathbf{p}}_j$ , we will extract  $\bar{\nabla}_{\mathbf{p}_i} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}, \bar{\nabla}_{\mathbf{p}_j} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}$
- the third and fourth term, which multiply respectively  $\dot{\beta}_{ij}, \dot{\beta}_{ij}^T$ , are the terms which give birth to the part of the matrix  $\mathbf{K}$  corresponding to  $\mathbf{g}_{v_{i_k}}$
- from the fifth and sixth term, which multiply respectively  $\dot{\mathbf{R}}_i^T, \dot{\mathbf{R}}_i$ , we will extract the  $\bar{\nabla}_{\mathbf{R}_i} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}$

Extracting  $\bar{\nabla}_{\mathbf{p}_i} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}, \bar{\nabla}_{\mathbf{p}_j} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}$  from (7.38) is straightforward and therefore we will focus on the computation of  $\bar{\nabla}_{\mathbf{R}_i} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}$ . The first term to consider is

$$\begin{aligned} -\frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \dot{\mathbf{R}}_i^T \mathbf{R}_i \mathbf{e}_k &= -\frac{1}{2d_{ij}} \text{tr} (2\mathbf{e}_m^T \mathbf{P}_{ij} \hat{\mathbf{w}}_i^T \mathbf{R}_i^T \mathbf{R}_i \mathbf{e}_k) = -\frac{1}{2d_{ij}} \text{tr} (2\mathbf{e}_k \mathbf{e}_m^T \mathbf{P}_{ij} \hat{\mathbf{w}}_i^T) = \\ &= -\frac{1}{2d_{ij}} \text{tr} (2\hat{\mathbf{w}}_i^T \mathbf{e}_k \mathbf{e}_m^T \mathbf{P}_{ij}) = \frac{1}{2d_{ij}} \text{tr} (2\hat{\mathbf{w}}_i \mathbf{e}_k \mathbf{e}_m^T \mathbf{P}_{ij}) = \frac{1}{2d_{ij}} \text{tr} (2\mathbf{e}_k \mathbf{e}_m^T \mathbf{P}_{ij} \hat{\mathbf{w}}_i). \end{aligned} \quad (7.39)$$

The second term to consider is

$$\begin{aligned} -\frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \dot{\mathbf{R}}_i \mathbf{e}_k &= -\frac{1}{2d_{ij}} \text{tr} (2\mathbf{e}_m^T \mathbf{P}_{ij} \hat{\mathbf{w}}_i \mathbf{e}_k) = \\ &= -\frac{1}{2d_{ij}} \text{tr} (2\hat{\mathbf{w}}_i \mathbf{e}_k \mathbf{e}_m^T \mathbf{P}_{ij}) = -\frac{1}{2d_{ij}} \text{tr} (2\mathbf{e}_k \mathbf{e}_m^T \mathbf{P}_{ij} \hat{\mathbf{w}}_i). \end{aligned} \quad (7.40)$$

Thus, from (7.39–7.40) it is possible to notice that

$$-\frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \dot{\mathbf{R}}_i^T \mathbf{R}_i \mathbf{e}_k - \frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \dot{\mathbf{R}}_i \mathbf{e}_k = \mathbf{0} \implies \bar{\nabla}_{\mathbf{R}_i} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm} = \mathbf{0} \quad (7.41)$$

and therefore, for  $\mathbf{f}_1 = \mathbf{g}_{v_{i_k}}$ , we have:

$$\bar{\nabla}_{\mathbf{p}_i} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm} = - \left[ \frac{1}{d_{ij}^2} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \mathbf{R}_i \mathbf{e}_k \mathbf{R}_i \beta_{ij} \right]^T \quad (7.42)$$

$$\bar{\nabla}_{\mathbf{p}_j} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm} = \left[ \frac{1}{d_{ij}^2} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \mathbf{R}_i \mathbf{e}_k \mathbf{R}_i \beta_{ij} \right]^T \quad (7.43)$$

$$\bar{\nabla}_{\mathbf{R}_i} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm} = \mathbf{0}^T \quad (7.44)$$

$$\bar{\nabla}_{\mathbf{R}_j} L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm} = \mathbf{0}^T \quad (7.45)$$

where the (7.45) is zero due to the fact that  $L_{\mathbf{g}_{v_{i_k}}}^1 \beta_{ijm}$  does not depend on  $\mathbf{R}_j$ .

Analogously to (7.38), the following holds with respect to  $\frac{d}{dt}L_{\mathbf{g}_{v_{jk}}}^1 \beta_{ijm}$

$$\begin{aligned} \frac{d}{dt}L_{\mathbf{g}_{v_{jk}}}^1 \beta_{ijm} &= \frac{d}{dt} \left[ \frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \mathbf{R}_j \mathbf{e}_k \right] = \\ & \left[ \left( \frac{\mathbf{R}_i \beta_{ij}}{d_{ij}^2} \right) \cdot \dot{\mathbf{p}}_i + \left( -\frac{\mathbf{R}_i \beta_{ij}}{d_{ij}^2} \right) \cdot \dot{\mathbf{p}}_j \right] \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \mathbf{R}_j \mathbf{e}_k + \\ & - \frac{1}{d_{ij}} \mathbf{e}_m^T \dot{\beta}_{ij} \beta_{ij}^T \mathbf{R}_i^T \mathbf{R}_j \mathbf{e}_k - \frac{1}{d_{ij}} \mathbf{e}_m^T \beta_{ij} \dot{\beta}_{ij}^T \mathbf{R}_i^T \mathbf{R}_j \mathbf{e}_k + \\ & + \frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \dot{\mathbf{R}}_i^T \mathbf{R}_j \mathbf{e}_k + \frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \dot{\mathbf{R}}_j \mathbf{e}_k. \end{aligned} \quad (7.46)$$

Focusing on the right hand side of the (7.46)

- from the first two terms, which multiply respectively  $\dot{\mathbf{p}}_i, \dot{\mathbf{p}}_j$ , we will extract  $\bar{\nabla}_{\mathbf{p}_i} L_{\mathbf{g}_{v_{jk}}}^1 \beta_{ijm}, \bar{\nabla}_{\mathbf{p}_j} L_{\mathbf{g}_{v_{jk}}}^1 \beta_{ijm}$
- the third and fourth term, which multiply respectively  $\dot{\beta}_{ij}, \dot{\beta}_{ij}^T$ , are the terms which give birth to the part of the matrix  $\mathbf{K}$  corresponding to  $\mathbf{g}_{v_{jk}}$
- from the fifth and sixth term, which multiply respectively  $\dot{\mathbf{R}}_i^T, \dot{\mathbf{R}}_j$ , we will extract the  $\bar{\nabla}_{\mathbf{R}_i} L_{\mathbf{g}_{v_{jk}}}^1 \beta_{ijm}, \bar{\nabla}_{\mathbf{R}_j} L_{\mathbf{g}_{v_{jk}}}^1 \beta_{ijm}$ .

Also in this case, extracting  $\bar{\nabla}_{\mathbf{p}_i} L_{\mathbf{g}_{v_{jk}}}^1 \beta_{ijm}, \bar{\nabla}_{\mathbf{p}_j} L_{\mathbf{g}_{v_{jk}}}^1 \beta_{ijm}$  from (7.46) is straightforward and therefore we will focus on the computation of  $\bar{\nabla}_{\mathbf{R}_i} L_{\mathbf{g}_{v_{jk}}}^1 \beta_{ijm}, \bar{\nabla}_{\mathbf{R}_j} L_{\mathbf{g}_{v_{jk}}}^1 \beta_{ijm}$ . For this purpose, let us write the following (with  $\mathbf{R}_i^T \mathbf{R}_j = {}^i\mathbf{R}_j$ )

$$\begin{aligned} \frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \dot{\mathbf{R}}_i^T \mathbf{R}_j \mathbf{e}_k &= \frac{1}{2d_{ij}} \text{tr} \left( 2\mathbf{e}_m^T \mathbf{P}_{ij} \dot{\mathbf{R}}_i^T \mathbf{R}_j \mathbf{e}_k \right) = \frac{1}{2d_{ij}} \text{tr} \left( 2\dot{\mathbf{R}}_i^T \mathbf{R}_j \mathbf{e}_k \mathbf{e}_m^T \mathbf{P}_{ij} \right) = \\ &= \frac{1}{2d_{ij}} \text{tr} \left( 2\hat{\mathbf{w}}_i^T {}^i\mathbf{R}_j \mathbf{e}_k \mathbf{e}_m^T \mathbf{P}_{ij} \right) = \frac{1}{2d_{ij}} \text{tr} \left( 2{}^i\mathbf{R}_j \mathbf{e}_k \mathbf{e}_m^T \mathbf{P}_{ij} \hat{\mathbf{w}}_i^T \right) \end{aligned} \quad (7.47)$$

and

$$\begin{aligned} \frac{1}{d_{ij}} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \dot{\mathbf{R}}_j \mathbf{e}_k &= \frac{1}{2d_{ij}} \text{tr} \left( 2\mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \dot{\mathbf{R}}_j \mathbf{e}_k \right) = \frac{1}{2d_{ij}} \text{tr} \left( 2\mathbf{e}_k \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \mathbf{R}_j \hat{\mathbf{w}}_j \right) = \\ &= \frac{1}{2d_{ij}} \text{tr} \left( 2\hat{\mathbf{w}}_j \mathbf{e}_k \mathbf{e}_m^T \mathbf{P}_{ij} {}^i\mathbf{R}_j \right) = \frac{1}{2d_{ij}} \text{tr} \left( 2{}^i\mathbf{R}_j^T \mathbf{P}_{ij} \mathbf{e}_m \mathbf{e}_k^T \hat{\mathbf{w}}_j^T \right). \end{aligned} \quad (7.48)$$

from the (7.47–7.48), through the *trace-trick* described in Sect. 7.2.3, we can then extract the  $\bar{\nabla}_{\mathbf{R}_i} L_{\mathbf{g}_{v_{j_k}}}^1 \beta_{ijm}, \bar{\nabla}_{\mathbf{R}_j} L_{\mathbf{g}_{v_{j_k}}}^1 \beta_{ijm}$ . Accordingly, for  $\mathbf{f}_1 = \mathbf{g}_{v_{j_k}}$ , we have:

$$\bar{\nabla}_{\mathbf{p}_i} L_{\mathbf{g}_{v_{j_k}}}^1 \beta_{ijm} = \left[ \frac{\mathbf{R}_i \beta_{ij}}{d_{ij}^2} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \mathbf{R}_j \mathbf{e}_k \right]^T \quad (7.49)$$

$$\bar{\nabla}_{\mathbf{p}_j} L_{\mathbf{g}_{v_{j_k}}}^1 \beta_{ijm} = - \left[ \frac{\mathbf{R}_i \beta_{ij}}{d_{ij}^2} \mathbf{e}_m^T \mathbf{P}_{ij} \mathbf{R}_i^T \mathbf{R}_j \mathbf{e}_k \right]^T \quad (7.50)$$

$$\bar{\nabla}_{\mathbf{R}_i} L_{\mathbf{g}_{v_{j_k}}}^1 \beta_{ijm} = \left[ \frac{1}{d_{ij}} \text{skew} (2\mathbf{R}_i^T \mathbf{R}_j \mathbf{e}_k \mathbf{e}_m^T \mathbf{P}_{ij})^\vee \right]^T \quad (7.51)$$

$$\bar{\nabla}_{\mathbf{R}_j} L_{\mathbf{g}_{v_{j_k}}}^1 \beta_{ijm} = \left[ \frac{1}{d_{ij}} \text{skew} (2\mathbf{R}_j^T \mathbf{R}_i \mathbf{P}_{ij} \mathbf{e}_m \mathbf{e}_k^T)^\vee \right]^T \quad (7.52)$$

Regarding the case  $\mathbf{f}_1 = \mathbf{g}_{w_{i_k}}$ , the Lie derivative  $L_{\mathbf{g}_{w_{i_k}}}^1 \beta_{ijm}$  depends on the states only through  $\beta_{ij}$ . Since we have separated the contribution of  $\dot{\beta}_{ij}$  in (7.32), we have that  $\bar{\nabla}_q L_{\mathbf{g}_{w_{i_k}}}^1 \beta_{ijm} = 0$ . Finally, as already mentioned, all the other first order Lie derivatives are zero, so they cannot contribute to the rank of  $\tilde{\mathbf{R}}$ .

### 7.3.3 Numerical verification of the ranks of $\tilde{\mathbf{R}}_A$ and $\tilde{\mathbf{R}}$

Sects. 7.3.1 and 7.3.2 provided a complete analytical expression of the different terms of the matrix  $\tilde{\mathbf{R}}$ . Notice that these matrices (and their ranks) depend only on the position and rotations of the agents and on their velocities (i.e., inputs).

After building the matrix (7.33) we found that, for a formation of 3 agents in random positions, the rank of the  $\tilde{\mathbf{R}}_A$  is equal to 11 ( $6N - 7$ ) while the one of the  $\tilde{\mathbf{R}}$  to 12 ( $6N - 6$ ). As an example, we plot in Fig. 7.1 the last 8 singular values of the two matrices as a function of time along the trajectories of the experiment which will be discussed in Sect. 7.5. Note that, as with any dynamic observability result, this result does not imply that the global scale can be recovered under *any arbitrary* choice of inputs. Instead, it is necessary to choose inputs that “excite” this mode (intuitively, where the agents move approximately perpendicularly to the bearing). This is in line with what was found in [146] and in any other work concerned with Structure from Motion.

Note that at the beginning of this chapter we claimed that the machineries exploited in this work are based only on bearing measurements while in the previous terms, almost everywhere, it appears the distance  $d_{ij}$ . This quantity is retrievable through the EKF which is the topic of next Sect. 7.4

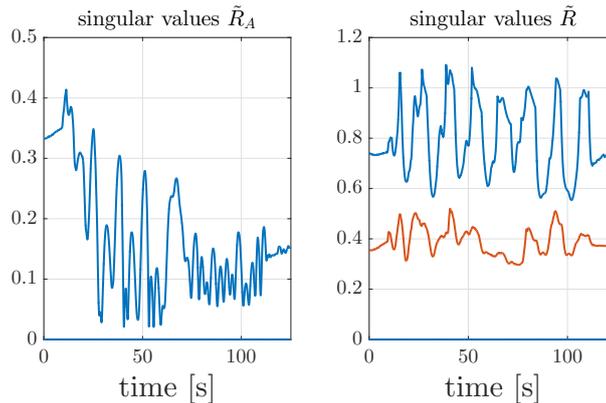


Figure 7.1 – Behavior of the last 8 singular values of the matrix  $\tilde{\mathbf{R}}_A$  (left) and of the matrix  $\tilde{\mathbf{R}}$  (right) during the experiment described in Sect. 7.5

## 7.4 A multi-agent Extended Kalman Filter

This section describes the design of an Extended Kalman Filter (EKF) on Lie groups to empirically verify the ideas of the previous sections. In particular, the EKF will provide an estimate  $\hat{\mathbf{q}}_i = (\hat{\mathbf{p}}_i, \hat{\mathbf{R}}_i)$  of the configuration  $\mathbf{q}_i$  of each agent. There are, however, three issues that need to be addressed. First, as already seen, the state of the system can be estimated only up to a global rotation and translation. We fix this ambiguity by choosing a moving reference frame that moves with the first agent. Of course, the uncertainty associated with this agent will always be zero by construction. A more representative choice would probably be to consider an “average” reference frame placed at the centroid of the formation; this, however, would significantly complicate the derivation of the filter, and it is therefore outside the scope of this work. Second, actual implementations of any filter must work in discrete time. We therefore need to discretize our dynamical model (7.1). As in [181], we use a simple 1-step Euler forward approach, which is equivalent to assuming constant input velocities  $\mathbf{v}_i, \mathbf{w}_i$  between discretization instants. With this assumption, our system (7.1) becomes

$$\mathbf{q}_{i,k+1} = \begin{bmatrix} \mathbf{p}_{i,k} + dt\mathbf{v}_{i,k} \\ \exp_{\mathbf{R}_{i,k}}(dt\mathbf{R}_{i,k}(\mathbf{w}_{i,k})^\wedge) \end{bmatrix} = \mathbf{f}_i(\mathbf{q}_{i,k}, \mathbf{u}_{i,k}), \quad (7.53)$$

where  $\mathbf{u}_{i,k} = \text{stack}(\mathbf{v}_{i,k}, \mathbf{w}_{i,k})$ ,  $dt$  is the length of time of the discretization interval, and  $\exp_{\mathbf{R}}(\cdot)$  is the exponential map at  $\mathbf{R} \in \text{SO}(3)$ ; the exponential map can be computed as  $\exp_{\mathbf{R}}(\mathbf{R}\hat{\mathbf{w}}) = \mathbf{R}\exp_{\mathbf{I}}(\hat{\mathbf{w}})$ , where the exponential at the identity  $\exp_{\mathbf{I}}$ , which corresponds to the matrix exponential, can be computed using the Rodriguez’s formula [142]. Moreover, we define an operator  $\exp_{\mathbf{q}_k}(\tau_k)$  with  $\mathbf{q}_k = (\mathbf{p}_k, \mathbf{R}_k) \in \text{SE}(3)$  and  $\tau_k = (\mathbf{v}_k, \mathbf{w}_k) \in T_{\mathbf{q}}\text{SE}(3)$ . This operator for the position part of  $\mathbf{q}_k$  corresponds to the operation  $\mathbf{q}_k + \mathbf{v}_k dt$  and for the rotation part of  $\mathbf{q}_k$  corresponds to the usual

$\exp_{\mathbf{R}_k}(\hat{\mathbf{w}}_k)$  (this notation simplifies the filter equations). Note that (7.53) is used for  $i > 1$ ; for agent  $i = 1$  we have  $\mathbf{q}_{i,k} = \mathbf{0}$ , given our choice of the reference frame. Note also that the output function  $\mathbf{h}_{ij}$  in (4.2), corresponding to the edge  $(i, j)$ , should be redefined to explicitly take into account the measurement noise (denoted with  $\mathbf{o}_{ij}$ )

$$\tilde{\mathbf{h}}_{ij}(\mathbf{q}_i, \mathbf{q}_j, \mathbf{o}_{ij}) = \mathbf{R}_i^\top \frac{\mathbf{p}_j - \mathbf{p}_i + \mathbf{o}_{ij}}{\|\mathbf{p}_j - \mathbf{p}_i + \mathbf{o}_{ij}\|} \quad (7.54)$$

Similarly, we define  $\tilde{\mathbf{f}}_i(\mathbf{q}_i, \mathbf{u}_i, \mathbf{n}_i) = \mathbf{f}(\mathbf{q}_i, \mathbf{u}_i + \mathbf{n}_i)$ , where  $\mathbf{n}_i \in T_{\mathbf{q}_i}SE(3)$  is the noise.

The overall system for the entire network is then

$$\hat{\mathbf{q}}_k = \tilde{\mathbf{f}}(\hat{\mathbf{q}}_{k-1}, \mathbf{u}_k, \mathbf{n}_k), \quad \beta_{ij} = \tilde{\mathbf{h}}(\mathbf{q}_j, \mathbf{q}_j, \mathbf{o}_k), \quad (7.55)$$

with  $\mathbf{u}_k = \text{stack}(\{u_{i,k}\})$ , and where we added process and measurement noises  $\mathbf{n}_k, \mathbf{o}_k$  ( $\mathbf{o}_k = \text{stack}(\{o_{ij}\})$ ) introduced in (7.54), which are assumed to be zero mean multivariate Gaussian with covariance (block diagonal) matrixes  $\mathbf{Q}_k, \mathbf{R}_k$ .

The EKF is based on the linearization of the system (7.55):

$$\mathbf{F}_k = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{q}} \right|_{\substack{\hat{\mathbf{x}}_{k-1|k-1}, \\ \mathbf{u}_k, \mathbf{n}_k = \mathbf{0}}}, \quad \mathbf{H}_k = \left. \frac{\partial \mathbf{h}}{\partial \mathbf{q}} \right|_{\hat{\mathbf{x}}_{k|k-1}, \mathbf{o}_k = \mathbf{0}}, \quad (7.56)$$

$$\mathbf{L}_k = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{n}} \right|_{\substack{\hat{\mathbf{x}}_{k-1|k-1}, \\ \mathbf{u}_{k-1}, \mathbf{n}_k = \mathbf{0}}}, \quad \mathbf{M}_k = \left. \frac{\partial \mathbf{h}}{\partial \mathbf{o}} \right|_{\hat{\mathbf{x}}_{k|k-1}, \mathbf{o}_k = \mathbf{0}}. \quad (7.57)$$

In our specific case, the matrix  $\mathbf{F}$  is block diagonal with blocks given by

$$\mathbf{F}_i \hat{\mathbf{q}}_i = \frac{\partial \mathbf{f}_i}{\partial \mathbf{q}_i} \hat{\mathbf{q}}_i = \begin{pmatrix} \dot{\mathbf{p}}_i + dt \dot{\mathbf{R}}_i \mathbf{v}_i + dt \mathbf{R}_i \dot{\mathbf{v}}_i \\ D_{\mathbf{R}_i} \dot{\mathbf{R}}_i + dt D_{\mathbf{w}_i} \dot{\mathbf{w}}_i \end{pmatrix}. \quad (7.58)$$

where  $D_{\mathbf{R}_i} = D_{\mathbf{R}_i} \exp_{\mathbf{R}_i}(dt \mathbf{R}_i \hat{\mathbf{w}}_i)$  and  $D_{\mathbf{w}_i} = D_{\mathbf{w}_i} \exp_{\mathbf{R}_i}(dt \mathbf{w}_i)$  denote the differentials of the exponential map with respect to  $\mathbf{R}$  and  $\mathbf{w}$ , respectively; details on the implementation of these differential can be found in [190]. The matrix  $\mathbf{H}_k$  is actually the same as the matrix  $\tilde{\mathbf{R}}_A$  computed in Sect. 7.3.1. The matrices  $\mathbf{L}_k$  and  $\mathbf{M}_k$  can be computed similarly to  $\mathbf{F}_k$  and  $\mathbf{H}_k$ .

The *prediction* step of the EKF is given by the state estimate

$$\hat{\mathbf{q}}_{k|k-1} = \mathbf{f}(\hat{\mathbf{q}}_{k-1|k-1}, \mathbf{u}_{k-1}) \quad (7.59)$$

and the covariance matrix estimate

$$\mathbf{P}_{k|k-1} = \mathbf{F}_{k-1} \mathbf{P}_{k-1|k-1} \mathbf{F}_{k-1}^\top + \mathbf{L}_{k-1} \mathbf{Q}_{k-1} \mathbf{L}_{k-1}^\top. \quad (7.60)$$

The *update* step of the filter consists of the state estimate

$$\hat{\mathbf{q}}_{k|k} = \exp_{\hat{\mathbf{q}}_{k|k-1}} \left( \mathbf{P}_{k|k-1} \mathbf{H}_k^\top + \underbrace{(\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^\top + \mathbf{M}_k \mathbf{R}_k \mathbf{M}_k^\top)^{-1}}_{\mathbf{S}_k} \right) \tilde{\mathbf{y}}_k \quad (7.61)$$

where  $\tilde{\mathbf{y}}_k$  is the difference between the measured and estimated bearings.

Moreover, the covariance matrix estimate is given by

$$\mathbf{P}_{k|k} = (\mathbf{I} - (\mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{S}_k^{-1}) \mathbf{H}_k) \mathbf{P}_{k|k-1} \quad (7.62)$$

In the next Sect. 7.5 it is shown a result of our (centralized) EKF.

## 7.5 Experimental results

In order to validate the presented ideas, extensive simulations in a Matlab/Simulink environment have been performed considering cases with both process and measurement noise. In addition to simulations also experiments have been performed with real quadrotor UAVs. In this section an experiment with three agents (two real quadrotor UAVs and a fixed *virtual* agent) will be presented. The experiments were performed in the flying arena described in Sect. B.2.2. Furthermore, we used two of the quadrotors described in Sect. B.2 and a third *virtual* agent of the formation which fix the ambiguity discussed in Sect. 7.4 has been placed, for convenience, in the origin of the motion capture system reference frame. This agent will be the one with respect to which the other two UAVs will refer their measurements.

During the experiment reported in Fig. 7.2 the agent 2 was moving mainly back and forth on the  $x$  axis while the agent 3 was following an ellipsoidal trajectory. A user was also able to give velocity inputs to the UAVs through a joystick. The trajectories described above allow to highlight the fact that, in order to collectively estimate the scale, the agents need to move along *exciting* trajectories [146]. Indeed, from Figs. 7.2(a) and 7.2(b) it is possible to notice that the variance associated to the coordinates which are only *lightly excited* ( $y, z$  coordinates on the agent 2 and 3) remains higher than the ones which are *strongly excited* ( $x$  coordinates on the agents 2 and 3). Fig. 7.2(c) shows the behavior of the true and estimated distance between the agents 2 and 3 and it is clear that the EKF, even starting far from the true values (both for positions and orientations (Fig. 7.2(d))) it is able to recover the true distance  $d_{23}$ , hence the scale of the formation.

The orientation error showed in Fig. 7.2(d) is defined as:

$$\mathbf{R}_{err_i} = \frac{1}{2} \text{tr} \left( \mathbf{I} - \hat{\mathbf{R}}_i^T \mathbf{R}_i \right), \quad \forall i \in \mathcal{V} \quad (7.63)$$

where  $\hat{\mathbf{R}}_i$  represents the estimation of the real rotation matrix  $\mathbf{R}_i$  corresponding to the agent  $i$ .

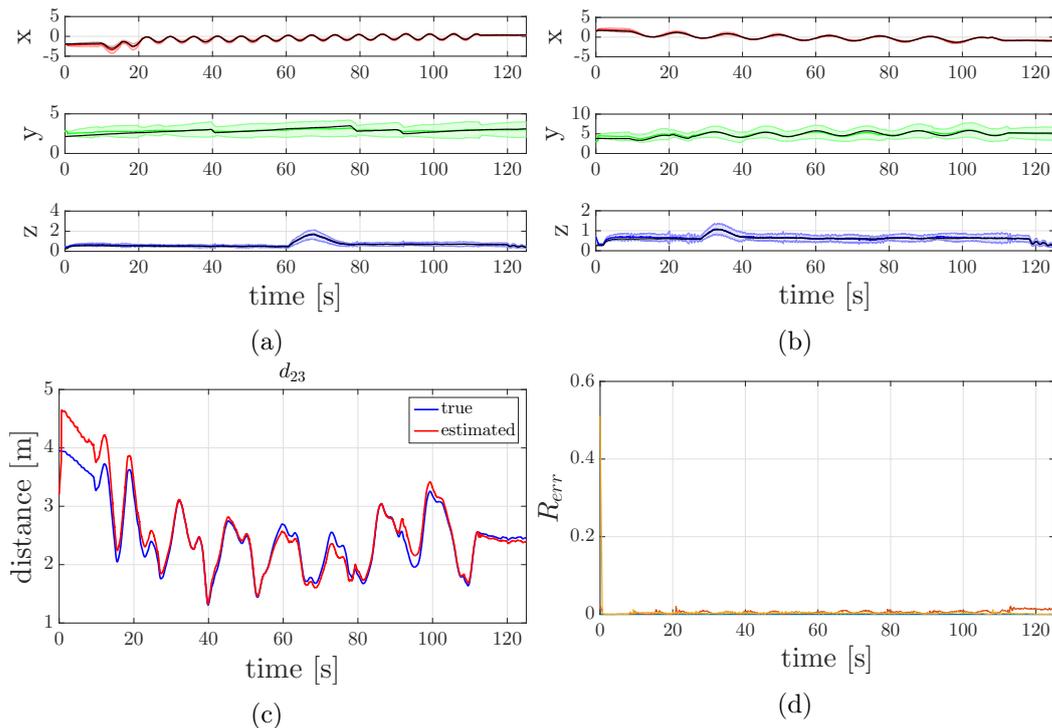


Figure 7.2 – Fig. 7.2(a) shows the behavior of the real position  $(x, y, z)$  (in meters) of the agent 2, in black, and the estimated one with the associated covariance respectively in red  $(x)$ , green  $(y)$  and blue  $(z)$ ; Fig. 7.2(b) is the dual of Fig. 7.2(a) for the agent 3; Fig. 7.2(c) shows the behavior of the true (blue) and estimated (red) distance  $d_{23}$  between the agents 2 and 3. Fig. 7.2(d) shows the behavior of the orientation error introduced in (7.63)

## 7.6 Conclusions and future works

In this chapter we considered the problem of scale estimation and localization with bearing-only measurements and known agent velocities. By applying nonlinear observability theory and Riemannian geometry, we extended existing results on the theory of rigidity and introduced the notion of *Dynamic Bearing Observability Matrix* (DBOM). Using experiments with two quadrotor UAVs, we have shown that the global scale is indeed observable and that it can be recovered by employing a centralized EKF implemented directly on SE(3).

The preliminary results of this work open several interesting future research direction, such as<sup>3</sup>

- providing a full analytical characterization of the nullspace of the *Dynamic Bearing Observability Matrix* and critical agent configurations

<sup>3</sup>As for the previous chapters, a fully onboard implementation is missing, refer to Chapt. 8 for more details.

- verifying if it is possible to achieve additional insights by including higher order Lie derivatives
- designing input signals that minimize the uncertainty in localization (using [146] as inspiration),
- studying distributed implementations of the EKF (building upon, e.g., [184, 191]) and, finally,
- applying similar ideas to different types of agents (e.g. UAVs with full dynamics, or unicycle dynamics) and measurements (e.g., distance-only),
- extending the proposed ideas to the case of multi-agent systems with unknown input as in [192].

## Summary

The material presented in the previous chapters has been published to different international conferences:

- The work presented in Chapt. 5 has been published in [27] and a related video can be found at <https://goo.gl/ZwwMGa>
- The work presented in Chapt. 6 has been published in [28] and a related video can be found at <https://goo.gl/z4v7Bu>
- The work presented in Chapt. 7 has been published in [29] and a related video can be found at <https://goo.gl/jqB1kX>



## Part III

# Conclusions and Future Work



---

## Conclusions and future work

IN this last chapter, we wish to review the main theoretical and experimental results achieved in the Thesis and point out some issues which are still left open. Regarding the latter, we also intend to indicate possible directions to follow for further investigation and research.

### 8.1 Summary and contributions

The goal of this Thesis was mainly to study the problems of decentralized formation control and localization of a group of quadrotor UAVs through relative measurements. The relative measurements on which we focused our attention were 3D bearings, that is, what one can recover from monocular cameras mounted onboard the robots. Note that in the whole Thesis we assumed that the underlying topology of the sensing graph associated with the formation is of a *directed* type to make our approach more prone to a real-world application. On the other hand, the underlying communication graph is supposed to be *undirected*, that is, if there is an edge, we assume that the robots connected through that edge are able to communicate with each other.

In Chapt. 5 we showed how to stabilize, through a decentralized controller, a group of quadrotor UAVs towards a desired formation specified in terms of bearing constraints. This was done by extending the theory of rigidity from  $SE(2)$  to the case of  $\mathbb{R}^3 \times \mathbb{S}^1$  and by combining some control and estimation algorithms. All this was done, up to our knowledge for the first time, in the case of non-stationary agents under the assumption that the formation remains infinitesimally bearing rigid. Furthermore, we fully characterized the null-space of the BRM. This allowed us to steer the group of quadrotors in 3D space through motions which do not alter the specified bearing constraints. Specifically, these motions consist of the so-called rigid-body-motions which consist of collective roto-translation and expansion/contraction of the whole formation.

Having understood the importance of bearing rigidity in the formation control and localization problems, we worked on the maintenance of this essential property. Therefore, in Chapt. 6, we proposed a bearing maintenance strategy which required the analytical expression of the rigidity eigenvalue and of its derivatives with respect to the configuration which we detailed in closed-form in the Thesis. This strategy was built on our previous results and was able to cope with some (realistic) sensor limitations as limited field of view of the camera, limited range of the camera and lastly the possibility of occlusions between the quadrotors during motion (a complex constraint which is quite seldom considered in the literature).

Bearing-only localization intrinsically does not allow to remove the scale ambiguity because bearings consist of non-metric information. Indeed, our localization scheme through bearing measurements, used for both the contributions mentioned above, *suffered* of the assumption that at least one agent needs to be able to measure its distance with respect to another agent. Because of this and under the (realistic) assumption that each robot can know its ego-motion (body frame linear/angular velocity) we decided to conduct an observability analysis of the whole system. This showed that the scale of the formation is indeed observable. To this end, we designed an EKF, directly on SE(3) which was able to estimate the poses of the agents of the formation and therefore delete the "scale ambiguity". Note that the finding that the scale is observable is in line with other works such as [146] and with the literature of Structure from Motion [92, 93]. However, unlike the majority of other works, we tackled these problems with tools from nonlinear observability and Riemannian geometry.

Again, all the presented theoretical claims were supported by extensive simulations and experimental campaigns. The experiments were run in our flying arena (see Sect. B.2) with multiple quadrotor UAVs.

Note that, during his Ph.D., the author extensively contributed to the setup of all the needed software/hardware infrastructure which was not in place at the beginning of his Ph.D. These experimental activities allowed him to get in contact with several implementation issues, such as network issues, limited flying time and the choice of the right software infrastructure which is responsible for the overall control of the UAVs. Due to this experience, the author believes that substantial work is still needed to allow a group of flying robots for robust autonomous navigation in an unknown environment without a centralized localization system and relying only on their *local* skills. As we mentioned, a good example of the implementation issues which continues to be a significant bottleneck is the limited flying time of quadrotor UAVs. Indeed, current battery technology only enables flights of 10-30 min, depending on many factors such as wind and payload. There is much research going on with battery refueling on UAVs with some solutions being battery swapping [193, 194] or

automated landing on a recharging platform as presented in [195]. We believe that improvements on this side of the technology are also fundamental for deploying a *swarm* of UAVs in real-world missions.

On the other hand, the numerous technological advancements regarding vision sensors, computational power and miniaturization of the electronics give big hopes of seeing these systems becoming a *robust* reality in the *not-so-far future*.

## 8.2 Open issues and future perspectives

However, as we mentioned, even if our approach presented many promising results it also highlighted some limitations both theoretical and practical. Note that some difficulties are specific to flying robots and especially quadrotor UAVs and this is the reason why in a part of the multi-agent community they are not even mentioned.

### Entirely onboard implementation

As we pointed out in the conclusions of Chaps. 5 to 7 one limitation of this work (and in our opinion the most important) is that we did not implement a fully onboard solution to retrieve bearing measurements from onboard cameras. This obstacle was due to the complex nature of the problems of robust detection, tracking, and identification, in real-time of the agents of the formation. However, we believe to have found the right tool to tackle this problem: the AprilTag detector [196, 197]. This tool is used extensively in different applications (such as exploration and mapping for ground robots [198], camera calibration [199, 200] and autonomous landing on moving platforms for UAVs [201]) but not yet, up to our knowledge, in scenarios like the one described in this Thesis. An alternative to the AprilTag fiducial markers is to use the WhyCon localization system proposed in [64] and already successfully employed in several works as [202, 203]. For all these reasons, during the experiments described in Chaps. 5 to 7 the bearing measurements were retrieved from the Vicon motion capture system. Note that the development of a fully onboard implementation is the topic of current research of the author (see Fig. 8.1). The previous problem is also related to the one of onboard computational power of each quadrotor. At the time of writing, as explained in Appendix B, we equipped our robots with an ODROID-XU4. However, we think that the new, and more powerful, NVIDIA JETSON TX2 module (Fig. 8.2) could represent a significant improvement to our system and this is why we are looking into switching to a platform with this computer module onboard.

### Collision avoidance

Another aspect which we did not take into account and plays a pivotal role in all kinds of formation control tasks is the one of collision avoidance. It is impossible to think about a real-world application which is not embedded with a collision



Figure 8.1 – First tests with the AprilTag fiducial markers. Fig. 8.1(a): two quadrotor UAVs flying in our flying arena with some Apriltag [197] mounted on them. Note that we plan to reduce the size of the fiducial markers used in our applications to make them more compact and easy to embed on each UAV. Fig. 8.1(b): a snapshot of an onboard camera which shows the detection of an AprilTag fiducial marker mounted on another quadrotor UAV and the consequent *translation* to a bearing vector.

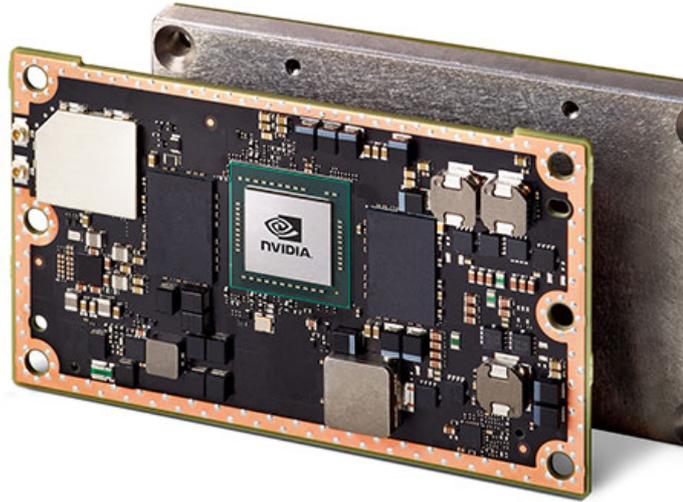


Figure 8.2 – The NVIDIA JETSON TX2 Module, from <http://www.nvidia.com>

avoidance technique. However, in this work collision avoidance is not taken into account in the design of the controller. Despite the capability of the maintenance strategy (Chapt. 6) of making an edge disappear when two agents become too close, this (unfortunately) does not ensure (in general) the avoidance of collision between the robots. This is because the interested edge could be one which is not necessary to maintain the rigidity of the formation over a certain threshold. Indeed, consider the scenario in which two agents start to become too close, but the rigidity of the framework is still well over the desired threshold. In this case, the weight associated with the corresponding edge will gradually vanish leaving the considered agents without any mean of avoiding each other. This is an intrinsic property of our design of the weights. Indeed, it encodes the concept that the disappearance of an edge

corresponds to the vanishing of the capability of a robot to sense the other robot to which it is *connected*. For this reason, we believe that our framework needs an explicit consideration of the collision avoidance problem. In the literature, there are many works [204] which deal with this problem, and we realized that finding the right method is of paramount importance. As a starting point towards this aim, we plan to adapt to our scenario the collision avoidance strategies used in [34, 62, 63, 205] which allow preventing inter-agent and obstacle collisions. Another choice would be following up what proposed in [122] where, besides extending their results of stabilization of bearing-only formations to arbitrary dimensions, they also provide a sufficient condition to ensure collision avoidance between any pair of agents under the action of their controller.

### ***Stochasticity of the control system***

Another interesting direction is the explicit modeling of the effect of noise on our algorithms. Noise was added in our simulations and it was intrinsically present in all the experiments, but no formal study was conducted to claim specific robustness properties of the proposed methods. This direction was partially followed in Chapt. 7 in which we took into account (see the equation (7.55)) both process and measurement noises. Therefore, we are aware that to go towards a real-world application we need to deal with the *stochasticity* of the overall system which could include not entirely reliable communication, tracking of the other robots through cameras and so on. Regarding the communication, during our experiments, we experienced several problems with the WiFi interferences, and therefore an excellent future direction would be to investigate alternatives to this communication technology. Regarding the problem of tracking other quadrotors, one of the dilemmas we would need to deal with is the presence of outliers and therefore choose the right algorithm (between the many which are present in the literature) is essential. A well-known way to deal with outliers, which would be our first shot at tackling these problems is the random sample consensus (RANSAC) algorithm [206] (and its extensions), which has been successfully explicitly used for AprilTags in [207]. All this could compromise the designed controller and therefore study the robustness of the controller is essential. A specific study of the gradient-based control law used to steer a formation to the desired bearing formation which could be taken as a starting point for future developments is [118]. In this work, they try to give a solution to the specific problem of *measurements discrepancies*. It is highlighted that this issue can sometimes produce undesirable collective motions of the formation, and therefore dealing with this problem is also of paramount importance.

### **Multiplicity of the rigidity eigenvalue**

Both in Chaps. 5 and 6 we relied on the sixth smallest eigenvalue  $\lambda_6$  of the

symmetric bearing rigidity matrix  $\mathbf{B}_{\mathcal{G}}^W(\mathbf{q})$  as a measure of rigidity of the considered framework  $(\mathcal{G}, \mathbf{q})$ . Already in Sect. 6.3.4 we alluded to the fact that what we said is valid as soon as the *rigidity eigenvalue* has multiplicity equal to one. This is easy to understand if we remember that we exploited, in our computations, several times the well-known formula

$$\lambda_i = \mathbf{v}_i^T \mathbf{A} \mathbf{v}_i \quad (8.1)$$

where  $\lambda_i$  is an eigenvalue of the matrix  $\mathbf{A}$  and  $\mathbf{v}_i$  is the corresponding eigenvector. Now suppose that we design a controller which is based on the  $\lambda_i$  and that this eigenvalue has multiplicity equal to two. This means that there are two different eigenvectors  $\mathbf{v}_{i_1}, \mathbf{v}_{i_2}$  corresponding to this eigenvalue and this would mean that even a linear combination of these two vectors is an eigenvector corresponding to the considered eigenvalue  $\lambda_i$ . Therefore, it will likely happen that the controller, at each time step, would switch between different eigenvectors, with all the possible instabilities caused by this discontinuing switching. This is a very relevant problem which was already encountered, as we already said, in [34, 119, 155] without proposing a solution. To find a solution, we believe that it could be helpful to understand, in the scenarios analyzed in this Thesis, what motivates the occurrence of multiple eigenvalues and if it is associable to a specific structure of the framework and/or a specific topology of the graph  $\mathcal{G}$ . For this reason, we believe that this problem represents an additional future direction of this work.

### Agent dynamics

In the multi-robot literature, as in this Thesis, we often encounter works<sup>1</sup> in which the dynamics of the single agent is neglected and therefore each agent is seen as a simple-integrator. Hence, we believe that another interesting future direction, also highlighted in Sect. 7.6, is extending our results to systems with different types of agents, especially second-order integrators, in order to be able to include also the dynamics associated to each agent.

### Final remarks

A last open issue, which is a result of the solution of the open directions mentioned above, is the need of tackling more complex problems than the simple steering of the formation by a human operator in an indoor room equipped with a Vicon motion capture system. As we pointed out many times, this would include, but it is not limited to, exploration and mapping in collapsed buildings, cooperative transportation, surveillance, and so forth. Furthermore, moving outdoor, in an uncontrolled environment is a big step which is needed to make these systems undoubtedly more useful for humankind. A first step towards all these aims would

---

<sup>1</sup>Especially when dealing with multiple quadrotor UAVs.

be to enhance the framework described in Chapt. 6 with the one of Chapt. 7 for the scale estimation.

Overall, we believe that the contributions of this Thesis allow the field of multi-robot coordination in unknown environments to take a step further in the understanding of the concepts of theory of rigidity and its application to the decentralized formation control and localization. These problems revealed to be challenging and at the same time full of potential for real-world applications which we sincerely hope to see in the near future.



## Additional technical details associated to Chapt. 6

### Contents

---

A.1 How to go from (6.15) to (6.16) . . . . .	127
A.2 Useful derivatives for the computation of the derivative of $\lambda_6^B$ .	128
A.3 Towards the derivatives of the weights . . . . .	129

---

**T**HIS appendix includes additional technical details about the derivations relative to Chapt. 6. The mathematical details presented here are not essential for the understanding of the maintenance strategy but give to the interested reader and idea on how we proceeded to arrive to certain results.

### A.1 How to go from (6.15) to (6.16)

In this section we want to show how to go from (6.15), which is reported below for convenience,

$$\lambda_6^B = \sum_{(i,j) \in \mathcal{E}} \left( \mathbf{v}_{p_{ij}}^T \frac{\bar{\mathbf{P}}_{ij}}{d_{ij}^2} \mathbf{v}_{p_{ij}} + 2\mathbf{v}_{p_{ij}}^T \mathbf{S} \frac{\bar{\mathbf{p}}_{ij}}{d_{ij}} \nu_{\psi_i} - \nu_{\psi_i}^2 \bar{\mathbf{p}}_{ij}^T \mathbf{S}^2 \bar{\mathbf{p}}_{ij} \right) \quad (\text{A.1})$$

to the (6.16) reported below

$$\lambda_6^B = \sum_{(i,j) \in \mathcal{E}} \left( \boldsymbol{\nu}_{p_{ij}}^T \frac{\mathbf{P}_{ij}}{d_{ij}^2} \boldsymbol{\nu}_{p_{ij}} + 2\boldsymbol{\nu}_{p_{ij}}^T \mathbf{S} \frac{\boldsymbol{\beta}_{ij}}{d_{ij}} \nu_{\psi_i} - \nu_{\psi_i}^2 \boldsymbol{\beta}_{ij}^T \mathbf{S}^2 \boldsymbol{\beta}_{ij} \right). \quad (\text{A.2})$$

This is really important because, as explained in Sect. 6.3.2, the (A.1) contains quantities expressed in the world frame (not available to the  $i$ -th agent) while (A.2) contains quantities only relative to agents  $i$  and  $j$ .

Therefore, we can rewrite (A.1), knowing that  $\mathbf{v}_{p_i} = \mathbf{R}_i \boldsymbol{\nu}_{p_i}$  and  $v_{\psi_i} = \nu_{\psi_i}$ , as

$$\begin{aligned} \lambda_6^B = & \sum_{(i,j) \in \mathcal{E}} \left[ (\mathbf{R}_i \boldsymbol{\nu}_{p_i} - \mathbf{R}_j \boldsymbol{\nu}_{p_j})^\top \frac{\bar{\mathbf{P}}_{ij}}{d_{ij}^2} (\mathbf{R}_i \boldsymbol{\nu}_{p_i} - \mathbf{R}_j \boldsymbol{\nu}_{p_j}) + \right. \\ & \left. 2 (\mathbf{R}_i \boldsymbol{\nu}_{p_i} - \mathbf{R}_j \boldsymbol{\nu}_{p_j})^\top \mathbf{S} \frac{\bar{\mathbf{P}}_{ij}}{d_{ij}} \nu_{\psi_i} - \nu_{\psi_i}^2 \boldsymbol{\beta}_{ij}^\top \mathbf{S}^2 \boldsymbol{\beta}_{ij} \right] = \\ & \sum_{(i,j) \in \mathcal{E}} \left[ (\boldsymbol{\nu}_{p_i} - {}^i \mathbf{R}_j \boldsymbol{\nu}_{p_j})^\top \mathbf{R}_i^\top \frac{\bar{\mathbf{P}}_{ij}}{d_{ij}^2} \mathbf{R}_i (\boldsymbol{\nu}_{p_i} - {}^i \mathbf{R}_j \boldsymbol{\nu}_{p_j}) + \right. \\ & \left. 2 (\boldsymbol{\nu}_{p_i} - {}^i \mathbf{R}_j \boldsymbol{\nu}_{p_j})^\top \mathbf{R}_i^\top \mathbf{S} \frac{\bar{\mathbf{P}}_{ij}}{d_{ij}} \nu_{\psi_i} - \nu_{\psi_i}^2 \boldsymbol{\beta}_{ij}^\top \mathbf{S}^2 \boldsymbol{\beta}_{ij} \right]. \end{aligned} \quad (\text{A.3})$$

where the term  $\nu_{\psi_i}^2 \bar{\mathbf{p}}_{ij}^\top \mathbf{S}^2 \bar{\mathbf{p}}_{ij}$  has been rewritten as

$$\nu_{\psi_i}^2 \bar{\mathbf{p}}_{ij}^\top \mathbf{S}^2 \bar{\mathbf{p}}_{ij} = \nu_{\psi_i} \mathbf{R}_i^\top \bar{\mathbf{p}}_{ij}^\top \mathbf{R}_i \mathbf{S}^2 \mathbf{R}_i^\top \bar{\mathbf{p}}_{ij} \mathbf{R}_i \nu_{\psi_i} = \nu_{\psi_i} \mathbf{R}_i^\top \boldsymbol{\beta}_{ij}^\top \mathbf{S}^2 \boldsymbol{\beta}_{ij} \mathbf{R}_i \nu_{\psi_i} = \nu_{\psi_i}^2 \boldsymbol{\beta}_{ij}^\top \mathbf{S}^2 \boldsymbol{\beta}_{ij}. \quad (\text{A.4})$$

Noticing that

$$\mathbf{R}_i^\top \bar{\mathbf{P}}_{ij} \mathbf{R}_i = \mathbf{R}_i^\top (\mathbf{I} - \bar{\mathbf{p}}_{ij} \bar{\mathbf{p}}_{ij}^\top) \mathbf{R}_i = \mathbf{I} - \boldsymbol{\beta}_{ij} \boldsymbol{\beta}_{ij}^\top = \mathbf{P}_{ij} \quad (\text{A.5})$$

and with  $\boldsymbol{\nu}_{p_{ij}} = \boldsymbol{\nu}_{p_i} - {}^i \mathbf{R}_j \boldsymbol{\nu}_{p_j}$  we have that (A.1) is equal to the (A.2) as wanted.

## A.2 Useful derivatives for the computation of the derivative of $\lambda_6^B$

This section contains some useful derivatives for the computation of the  $\lambda_6^B$  in Chapt. 6.

The distance  $d_{ij}$  between two robots can be written as

$$d_{ij} = \|\mathbf{p}_j - \mathbf{p}_i\| = ((\mathbf{p}_j - \mathbf{p}_i) \cdot (\mathbf{p}_j - \mathbf{p}_i))^{1/2} \quad (\text{A.6})$$

therefore, its derivative will be

$$\dot{d}_{ij} = \frac{1}{2} ((\mathbf{p}_j - \mathbf{p}_i) \cdot (\mathbf{p}_j - \mathbf{p}_i))^{-1/2} \frac{d}{dt} ((\mathbf{p}_j - \mathbf{p}_i) \cdot (\mathbf{p}_j - \mathbf{p}_i)) \quad (\text{A.7})$$

(A.7) follows from (A.6) by an easy application of the chain rule. We further have

$$\frac{d}{dt} [(\mathbf{p}_j - \mathbf{p}_i) \cdot (\mathbf{p}_j - \mathbf{p}_i)] = 2(\mathbf{p}_j - \mathbf{p}_i) \cdot (\dot{\mathbf{p}}_j - \dot{\mathbf{p}}_i) \quad (\text{A.8})$$

substituting (A.8) into (A.7) yields

$$\dot{d}_{ij} = ((\mathbf{p}_j - \mathbf{p}_i) \cdot (\mathbf{p}_j - \mathbf{p}_i))^{-1/2} (\mathbf{p}_j - \mathbf{p}_i) \cdot (\dot{\mathbf{p}}_j - \dot{\mathbf{p}}_i) = \frac{\mathbf{p}_j - \mathbf{p}_i}{d_{ij}} \cdot (\dot{\mathbf{p}}_j - \dot{\mathbf{p}}_i). \quad (\text{A.9})$$

If we multiply (on the left) (A.9) by  $\mathbf{R}_i \mathbf{R}_i^T$  we have

$$\dot{d}_{ij} = \mathbf{R}_i \boldsymbol{\beta}_{ij} (\dot{\mathbf{p}}_j - \dot{\mathbf{p}}_i) = \frac{\partial d_{ij}}{\partial \mathbf{p}_i} \dot{\mathbf{p}}_i + \frac{\partial d_{ij}}{\partial \mathbf{p}_j} \dot{\mathbf{p}}_j = -\mathbf{R}_i \boldsymbol{\beta}_{ij} \dot{\mathbf{p}}_i + \mathbf{R}_i \boldsymbol{\beta}_{ij} \dot{\mathbf{p}}_j \quad (\text{A.10})$$

and therefore it follows

$$\frac{\partial d_{ij}}{\partial \mathbf{p}_i} = -\mathbf{R}_i \boldsymbol{\beta}_{ij}, \quad \frac{\partial d_{ij}}{\partial \mathbf{p}_j} = \mathbf{R}_i \boldsymbol{\beta}_{ij}. \quad (\text{A.11})$$

In the same way we can find

$$\frac{\partial}{\partial \mathbf{p}_i} \frac{1}{d_{ij}} = \frac{\mathbf{R}_i \boldsymbol{\beta}_{ij}}{d_{ij}^2}, \quad \frac{\partial}{\partial \mathbf{p}_j} \frac{1}{d_{ij}} = -\frac{\mathbf{R}_i \boldsymbol{\beta}_{ij}}{d_{ij}^2} \quad (\text{A.12})$$

and

$$\frac{\partial}{\partial \mathbf{p}_i} \frac{1}{d_{ij}^2} = 2 \frac{\mathbf{R}_i \boldsymbol{\beta}_{ij}}{d_{ij}^3}, \quad \frac{\partial}{\partial \mathbf{p}_j} \frac{1}{d_{ij}^2} = -2 \frac{\mathbf{R}_i \boldsymbol{\beta}_{ij}}{d_{ij}^3}. \quad (\text{A.13})$$

Regarding the derivative of  $\boldsymbol{\beta}_{ij}$ , it holds

$$\dot{\boldsymbol{\beta}}_{ij} = \frac{\partial \boldsymbol{\beta}_{ij}}{\partial \mathbf{p}_i} \dot{\mathbf{p}}_i + \frac{\partial \boldsymbol{\beta}_{ij}}{\partial \mathbf{p}_j} \dot{\mathbf{p}}_j + \frac{\partial \boldsymbol{\beta}_{ij}}{\partial \psi_i} \dot{\psi}_i. \quad (\text{A.14})$$

Note that (A.14) is valid only if the matrix  $\mathbf{R}_i$  considered is a function of one angle  $\psi_i$ . where

$$\frac{\partial \boldsymbol{\beta}_{ij}}{\partial \mathbf{p}_i} = -\frac{\mathbf{P}_{ij} \mathbf{R}_i}{d_{ij}}, \quad \frac{\partial \boldsymbol{\beta}_{ij}}{\partial \mathbf{p}_j} = \frac{\mathbf{P}_{ij} \mathbf{R}_i}{d_{ij}}, \quad \frac{\partial \boldsymbol{\beta}_{ij}}{\partial \psi_i} = -\mathbf{S} \boldsymbol{\beta}_{ij} \quad (\text{A.15})$$

Furthermore, to compute  $\dot{\mathbf{P}}_{ij}$  we need to use the following

$$\dot{\mathbf{P}}_{ij} = \frac{d}{dt} (\mathbf{I} - \boldsymbol{\beta}_{ij} \boldsymbol{\beta}_{ij}^T) = -\frac{d}{dt} (\boldsymbol{\beta}_{ij} \boldsymbol{\beta}_{ij}^T) = -\left[ \left( \dot{\boldsymbol{\beta}}_{ij} \right) \boldsymbol{\beta}_{ij}^T + \boldsymbol{\beta}_{ij} \left( \dot{\boldsymbol{\beta}}_{ij}^T \right) \right] \quad (\text{A.16})$$

and combine it with the (A.14–A.15).

### A.3 Towards the derivatives of the weights

In Sect. 6.3.2, specifically in (6.19), we refer to the gradients of the weights  $w_{ij}$  with respect to  $\mathbf{p}_i, \mathbf{p}_j, \psi_i, \psi_j$  without giving further details. This section is intended to give some more details about the math involved in the computation of these weights. This should give the interested reader a hint of the rationale behind the math exploited by the controller proposed in (6.23).

Let us start from the (6.17) which is reported below for convenience

$$\lambda_6^B = \sum_{(i,j) \in \mathcal{E}} w_{ij} l_{ij}. \quad (\text{A.17})$$

Remember that  $w_{ij}$  is composed of different weights

$$w_{ij} = w_{R_{ij}} w_{F_{ij}} w_{V_{ij}} \quad (\text{A.18})$$

therefore the  $\nabla_{\mathbf{q}_i} \lambda_6^B$  would be equal to

$$\begin{aligned} \nabla_{\mathbf{q}_i} \lambda_6^B = \sum_{(i,j) \in \mathcal{E}} & (\nabla_{\mathbf{q}_i} w_{R_{ij}}) w_{F_{ij}} w_{V_{ij}} l_{ij} + w_{R_{ij}} (\nabla_{\mathbf{q}_i} w_{F_{ij}}) w_{V_{ij}} l_{ij} + \\ & w_{R_{ij}} w_{F_{ij}} (\nabla_{\mathbf{q}_i} w_{V_{ij}}) l_{ij} + w_{R_{ij}} w_{F_{ij}} w_{V_{ij}} (\nabla_{\mathbf{q}_i} l_{ij}). \end{aligned} \quad (\text{A.19})$$

From the (A.19) it is then clear that to compute the gradient of the  $\lambda_6^B$  when the weights are involved, we need to compute the terms  $\nabla_{\mathbf{q}_i} w_{R_{ij}}, \nabla_{\mathbf{q}_i} w_{F_{ij}}, \nabla_{\mathbf{q}_i} w_{V_{ij}}$ . Remember that  $\mathbf{q}_i = (\mathbf{p}_i, \psi_i)$  and therefore each of the previous terms will break down to a  $\nabla_{\mathbf{p}_i}(\cdot)$  and a  $\nabla_{\psi_i}(\cdot)$ .

As an example, let us look at the gradient of the weight  $w_{F_{ij}}(\alpha_{ij})$  defined in (6.3). In particular it will be

$$\frac{\partial}{\partial \mathbf{p}_i} [w_{F_{ij}}(\alpha_{ij})] = \frac{\partial}{\partial \alpha_{ij}} w_{F_{ij}} \frac{\partial}{\partial \mathbf{p}_i} \alpha_{ij}. \quad (\text{A.20})$$

The first term of the right-hand-side of the (A.20) depends just from the shape we decided for the function expressed by (6.3) and it involves really easy derivatives. On the other side, the second term of the right-hand-side of the (A.20), because  $\alpha_{ij} = \mathbf{o}_C^T \boldsymbol{\beta}_{ij}$ , it will be just equal to the first equation of (A.15)<sup>1</sup>. The same thing goes for the derivative with respect to  $\mathbf{p}_j$  and for  $\psi_i$  it holds

$$\frac{\partial}{\partial \psi_i} [w_{F_{ij}}(\alpha_{ij})] = \frac{\partial}{\partial \alpha_{ij}} w_{F_{ij}} \frac{\partial}{\partial \psi_i} \alpha_{ij}. \quad (\text{A.21})$$

where the second term of the right-hand-side of the (A.21) is equal to the third equation of (A.15).

---

<sup>1</sup>Where  $\mathbf{o}_C \in \mathbb{S}^2$  is the (constant and known) direction of the camera optical axis in the quadrotor body-frame

---

# Simulation and experimental architecture

## Contents

---

<b>B.1</b>	<b>Simulations architecture</b>	<b>131</b>
<b>B.2</b>	<b>Real experiments architecture</b>	<b>132</b>
<b>B.2.1</b>	<b>The quadrotor</b>	<b>132</b>
<b>B.2.2</b>	<b>The Vicon motion capture system</b>	<b>134</b>

---

**I**N THIS appendix we present the overall software/hardware architecture used throughout this work both for the simulations and for the experiments. Note that, during his Ph.D., the author has been extensively involved in the setup of the whole architecture both on the software and on the hardware level.

## B.1 Simulations architecture

The simulation architecture can be divided into two main parts depending on how the single quadrotor UAV is modeled:

- (1) Quadrotor UAV modeled as a simple integrator using *only* MATLAB and Simulink to run each simulation
- (2) Quadrotor UAV modeled with a more complex model through the V-REP robot simulator [158] and by using the TeleKyb framework to run each simulation (Figs. B.1 and B.2). For more details about TeleKyb the reader is referred to [159].

In both cases MATLAB and Simulink represent the main programming environment where we implemented:

- the formation control and estimation algorithms described in Chaps. 5 and 6
- the observability analysis (with the corresponding EKF) described in Chapt. 7.

In the case (2) above TeleKyb was used to interface the control/estimation algorithms with the V-REP simulator. Using TeleKyb also in simulations allowed us to have a similar architecture between simulations and real experiments. This enabled us to switch easily between the simulation and experimental architectures. The module which provides the state of the formation is different, of course, between the simulation architecture and the experimental one. In the first case, it is the V-REP simulator (or MATLAB) which provides the state, while during real experiments is the Vicon motion capture system which tracks the pose of real quadrotor UAVs.

The communication between the modules present in Fig. B.1 is made possible through the Robotic Operating System (ROS<sup>1</sup>) and, more in details:

- the *bridge* described in [208] has been used to build a communication between V-REP and ROS
- another *bridge*<sup>2</sup> has been used to build a communication between MATLAB, Simulink and ROS

## B.2 Real experiments architecture

As explained in Appendix B.1 the overall architecture used for the real experiments is similar to the one used in simulations. It is possible to appreciate this similarity by comparing Fig. B.1 with Fig. B.3. From Fig. B.3 it is also possible to notice that we used a Vicon motion capture system (see Sect. B.2.2) to track the pose of the quadrotor UAVs. The Vicon was used for obtaining a reliable ground truth with a sub-millimeter accuracy, to retrieve the linear/angular velocities of the real robots and, when an edge was existing, the body-frame bearing measurements  $\beta_{ij}$  between the considered robots which would have been obtained by the onboard cameras. To mimic bearing measurements coming from a camera we also exploited the option to decrease the frequency at which the Vicon was giving data.

### B.2.1 The quadrotor

The quadrotor platform used for the experiments of this work consists of a MK-Quadro by Mikrokopter<sup>3</sup>. The usual Mk-Quadro was extended with an ODROID-XU4 computing device (Fig. B.4) running Ubuntu (14.04 or 16.04) ROS and TeleKyb.

---

<sup>1</sup><http://www.ros.org/>

<sup>2</sup>[http://wiki.ros.org/matlab\\_ros\\_bridge](http://wiki.ros.org/matlab_ros_bridge)

<sup>3</sup><http://www.mikrokopter.de/en/home>

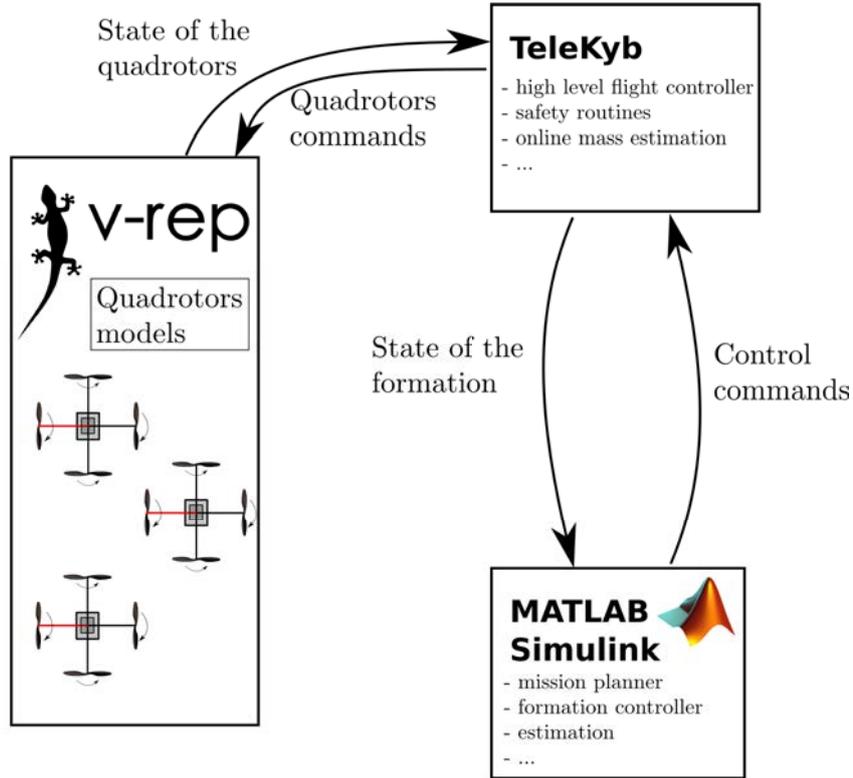


Figure B.1 – Scheme of the software architecture used during a simulation when the V-REP robot simulator is used to model the quadrotor UAVs

For the experiments described in Chapt. 6 each quadrotor was also equipped with an onboard camera (a Flea FL3-U3-32S2C-CS by FLIR, previously Point-Grey<sup>4</sup>) as shown in Fig. B.5(b). Note that this camera was not used to retrieve the bearing measurements during the experiments but only as an additional ground-truth and to verify that the maintenance strategy of Sect. 6.3 and the associated weight functions are working correctly.

Note also that in order to have a smooth trajectory and to match better the dynamics capabilities of the UAVs we used a fourth order linear filter for the inputs given to each UAV:

$$\ddot{\mathbf{p}}_i^f = -k_1 \ddot{\mathbf{p}}_i^f - k_2 \dot{\mathbf{p}}_i^f - k_3 (\dot{\mathbf{p}}_i^f - \dot{\mathbf{p}}_i) - k_4 (\mathbf{p}_i^f - \mathbf{p}_i) \quad (\text{B.1})$$

where with  $\ddot{\mathbf{p}}_i^f$ ,  $\dot{\mathbf{p}}_i^f$ ,  $\mathbf{p}_i^f$ ,  $\mathbf{p}_i$  we indicate respectively the filtered snap, jerk, acceleration and velocity given to the  $i$ -th UAV. The gains  $k_1$ ,  $k_2$ ,  $k_3$ ,  $k_4$  were chosen in order to have the poles in  $-7, -6, -5, -4$  and therefore have a convenient settling time.

Moreover, note that towards the end of this work we decided to *upgrade* our system from using TeleKyb to a set of packages based on GenoM3 [209] for implementing

<sup>4</sup><https://www.ptgrey.com/>

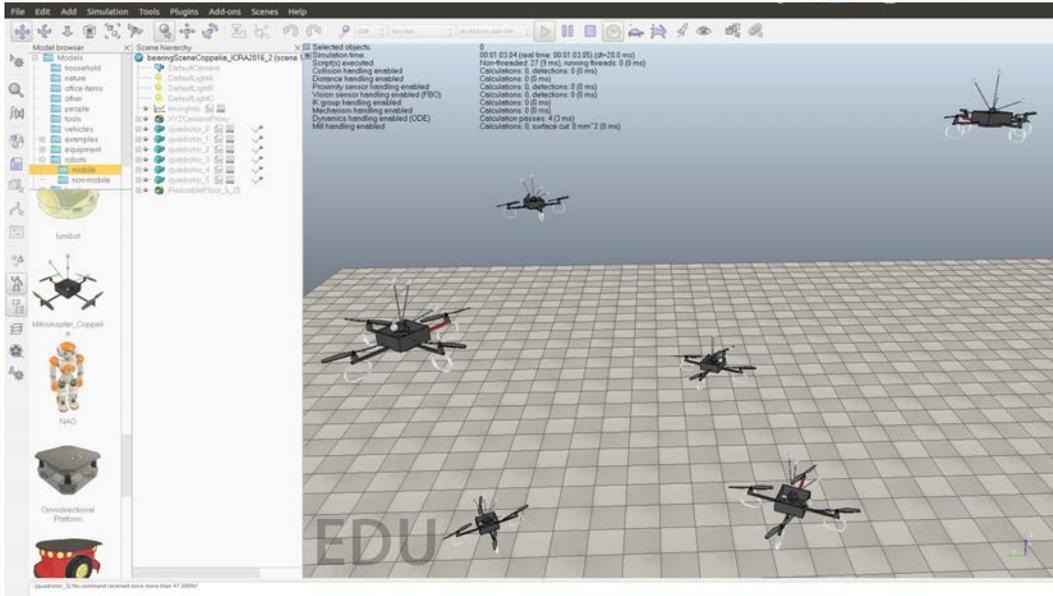


Figure B.2 – Screenshot of a simulation of six quadrotor UAVs in the V-REP robot simulator

the low-level flight control. This set of packages is mainly developed at LAAS-CNRS of Toulouse. This choice was done in order to build a common *better* tool for flying UAVs and mainly because of the many improvements that were made over the TeleKyb framework. This *new* framework has been successfully used for the experiments showed in Chapt. 7. However, the high level part of the algorithm described in Chapt. 7 (as in the other chapters) was still implemented through Matlab/Simulink.

### B.2.2 The Vicon motion capture system

During the experiments, in order to get the pose (position and orientation) of the robots in the environment we used a Vicon motion capture system. This system is made of multiple infrared cameras (Fig. B.7), a router to receive all the camera measurements and a software (Vicon Tracker) to analyze these measurements. We used this system in two different configurations according to the room in which the experiments were conducted:

- for early-stage tests we used a room with a flying volume of about  $7m \times 4.5m \times 2.5m$  of about  $7 \text{ m} \times 4.5 \text{ m} \times 2.5 \text{ m}$  (Fig. B.6(a)) equipped with a Vicon system made of eight *Bonita 10* cameras (Fig. B.7(a))
- for final experiments we used a room with a bigger flying volume (with respect to the previous one) of about  $6.5 \text{ m} \times 6.5 \text{ m} \times 3 \text{ m}$  (Fig. B.6(b)) equipped with

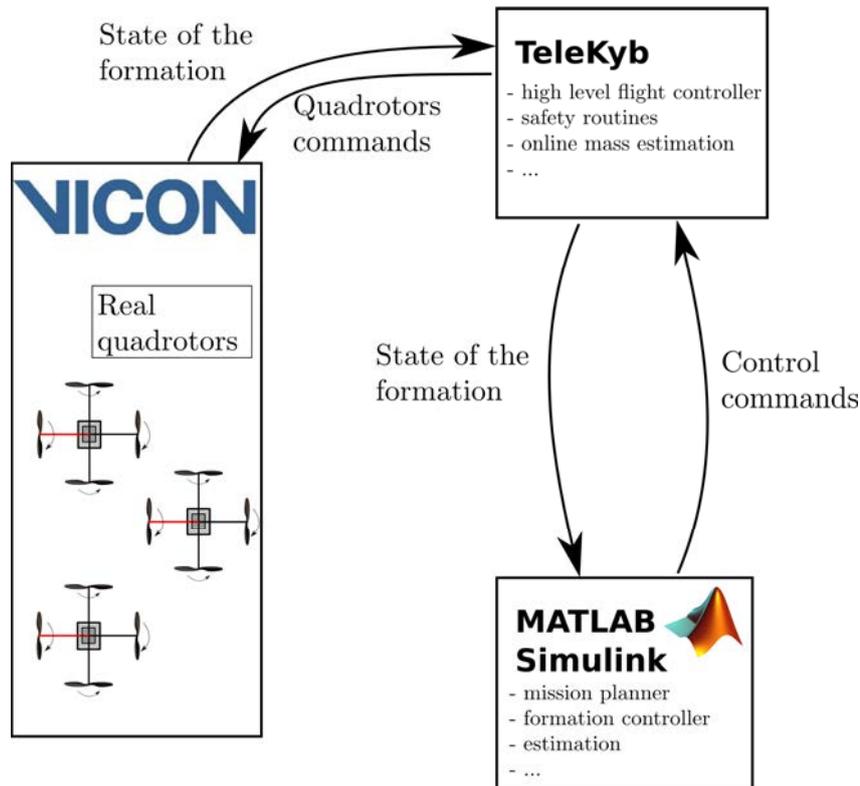


Figure B.3 – Scheme of the overall architecture used during the experiments with real quadrotor UAVs and the Vicon motion capture system

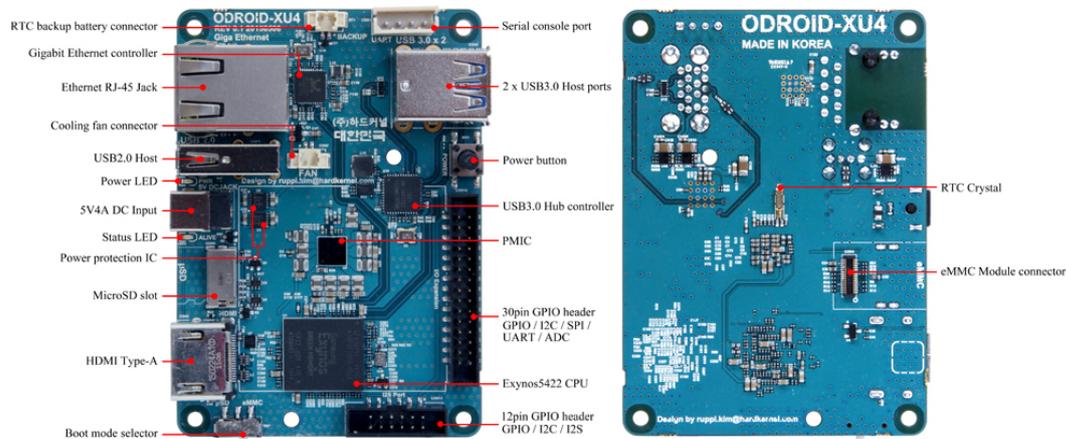


Figure B.4 – Odroid XU4, from <http://www.hardkernel.com/>

a Vicon system made of eight *Bonita 10* cameras (Fig. B.7(a)) and four *Vero 1.3* cameras (Fig. B.7(b))<sup>5</sup>.

A *Bonita 10* camera has the following specifications

<sup>5</sup>The experiments described in Chapt. 5 to 7 were performed in the room with a bigger flying volume, in Fig. B.6(b).

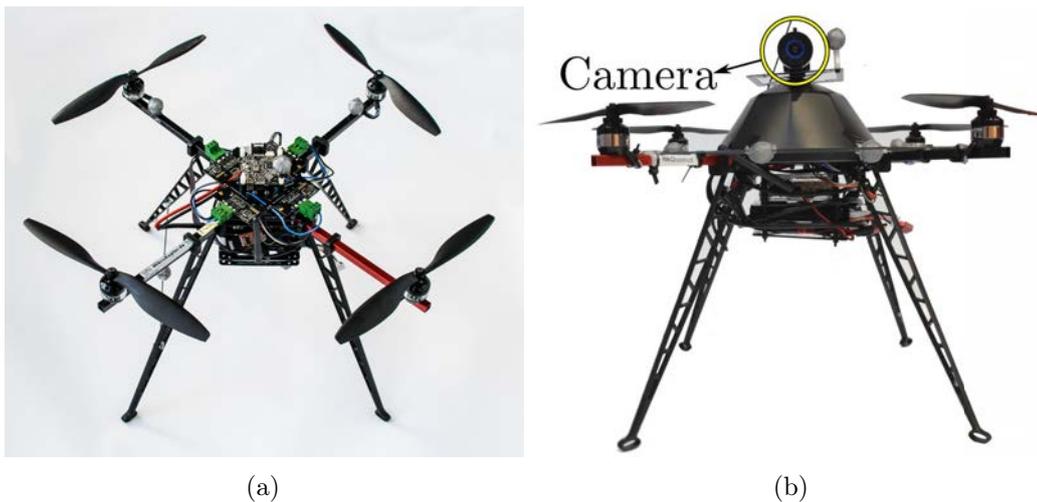


Figure B.5 – The MkQuadro without (Fig. B.5(a)) and with (Fig. B.5(b)) a Point Grey camera mounted on top of it.

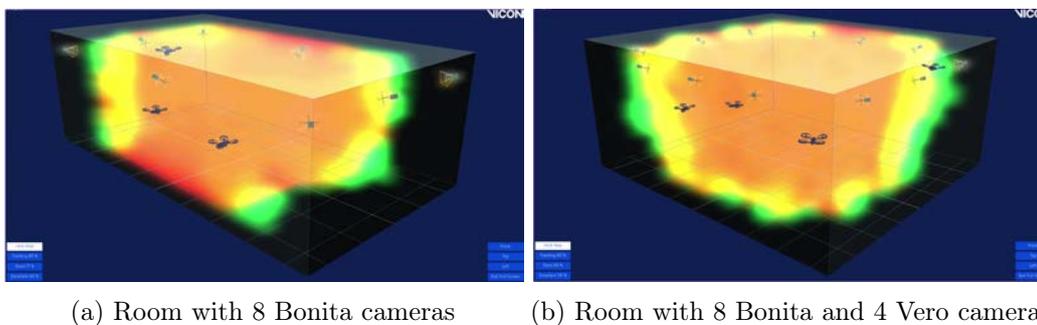


Figure B.6 – Screenshots of the coverage tool present in the Vicon Tracker software which show the two available rooms for conducting experiments, at INRIA Rennes Bretagne Atlantique.



Figure B.7 – Vicon cameras used during the experiments, from <https://www.vicon.com/>

- frame rate: 250 fps
- resolution: 1 megapixel
- angle of view wide:  $70.29^\circ \times 70.29^\circ$
- angle of view narrow:  $26.41^\circ \times 26.41^\circ$ ,

while a *Vero v1.3* camera has the following specifications

- frame rate: 250 fps
- resolution: 1.3 megapixel
- angle of view wide:  $60.8^\circ \times 50.3^\circ$
- angle of view narrow:  $32.7^\circ \times 26.4^\circ$ .



---

## Bibliography

- [1] N. Joseph, “Science and Civilization in China: Volume 2, History of Scientific Thought,” 1991.
- [2] S. M. Douglas, I. Bachelet, and G. M. Church, “A logic-gated nanorobot for targeted transport of molecular payloads,” *Science*, vol. 335, no. 6070, pp. 831–834, 2012.
- [3] B. Jang, E. Gutman, N. Stucki, B. F. Seitz, P. D. Wendel-Garcia, T. Newton, J. Pokki, O. Ergeneman, S. Pane, Y. Or *et al.*, “Undulatory locomotion of magnetic multilink nanoswimmers,” *Nano letters*, vol. 15, no. 7, pp. 4829–4833, 2015.
- [4] H.-T. Lin, G. G. Leisk, and B. Trimmer, “Goqbot: a caterpillar-inspired soft-bodied rolling robot,” *Bioinspiration & biomimetics*, vol. 6, no. 2, p. 026007, 2011.
- [5] G. C. de Croon, M. Groen, C. De Wagter, B. Remes, R. Ruijsink, and B. W. van Oudheusden, “Design, aerodynamics and autonomy of the delfly,” *Bioinspiration & biomimetics*, vol. 7, no. 2, p. 025003, 2012.
- [6] M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, K. Bodie, P. Fankhauser, M. Bloesch *et al.*, “Anymal-a highly mobile and dynamic quadrupedal robot,” in *IEEE International Conference on Intelligent Robots and Systems*, 2016, pp. 38–44.
- [7] K. E. Peyer, E. C. Siringil, L. Zhang, M. Suter, and B. J. Nelson, “Bacteria-inspired magnetic polymer composite microrobots,” in *Conference on Biomimetic and Biohybrid Systems*. Springer, 2013, pp. 216–227.

- [8] M. Dorigo, V. Maniezzo, and A. Colorni, “Ant system: optimization by a colony of cooperating agents,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 26, no. 1, pp. 29–41, 1996.
- [9] A. Martinoli, A. J. Ijspeert, and F. Mondada, “Understanding collective aggregation mechanisms: From probabilistic modelling to experiments with real robots,” *Robotics and Autonomous Systems*, vol. 29, no. 1, pp. 51–63, 1999.
- [10] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm intelligence: from natural to artificial systems*. Oxford university press, 1999, no. 1.
- [11] F. Mondada, G. C. Pettinaro, A. Guignard, I. W. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L. M. Gambardella, and M. Dorigo, “Swarm-bot: A new distributed robotic concept,” *Autonomous Robots*, vol. 17, no. 2-3, pp. 193–221, 2004.
- [12] A. Martinoli, K. Easton, and W. Agassounon, “Modeling swarm robotic systems: A case study in collaborative distributed manipulation,” *The Int. J. of Robotics Research*, vol. 23, no. 4-5, pp. 415–436, 2004.
- [13] M. Dorigo, M. Birattari, and T. Stutzle, “Ant colony optimization,” *IEEE computational intelligence magazine*, vol. 1, no. 4, pp. 28–39, 2006.
- [14] J. Halloy, G. Sempo, G. Caprari, C. Rivault, M. Asadpour, F. Tâche, I. Saïd, V. Durier, S. Canonge, J. M. Amé *et al.*, “Social integration of robots into groups of cockroaches to control self-organized choices,” *Science*, vol. 318, no. 5853, pp. 1155–1158, 2007.
- [15] D. Karaboga and B. Akay, “A survey: algorithms simulating bee swarm intelligence,” *Artificial intelligence review*, vol. 31, no. 1-4, pp. 61–85, 2009.
- [16] N. J. Mlot, C. A. Tovey, and D. L. Hu, “Fire ants self-assemble into waterproof rafts to survive floods,” *Proceedings of the National Academy of Sciences*, vol. 108, no. 19, pp. 7669–7673, 2011.
- [17] L. E. Parker *et al.*, “Current state of the art in distributed autonomous mobile robotics.” in *DARS*, 2000, pp. 3–14.
- [18] R. M. Murray, “Recent research in cooperative control of multi-vehicle systems,” *ASME Journal on Dynamic Systems, Measurement, and Control*, vol. 129, no. 5, pp. 571–583, 2006.
- [19] A. Howard, L. E. Parker, and G. S. Sukhatme, “Experiments with a large heterogeneous mobile robot team: Exploration, mapping, deployment and detection,” *The Int. J. of Robotics Research*, vol. 25, no. 5-6, pp. 431–447, 2006.

- 
- [20] Z. Yan, N. Jouandeau, and A. A. Cherif, “A survey and analysis of multi-robot coordination,” *International Journal of Advanced Robotic Systems*, vol. 10, no. 12, p. 399, 2013.
- [21] B. Charrow, N. Michael, and V. Kumar, “Cooperative multi-robot estimation and control for radio source localization,” *The Int. J. of Robotics Research*, vol. 33, no. 4, pp. 569–580, 2014.
- [22] M. Schwager, B. J. Julian, M. Angermann, and D. Rus, “Eyes in the sky: Decentralized control for the deployment of robotic camera networks,” *Proceedings of the IEEE*, vol. 99, no. 9, pp. 1541–1561, 2011.
- [23] “Forecast: Personal and Commercial Drones, Worldwide, 2016,” <https://www.gartner.com/doc/3557717>, accessed: 2017-09-04.
- [24] “ARCAS: Aerial Robotics Cooperative Assembly System,” <http://www.arcas-project.eu/>, accessed: 2017-09-04.
- [25] “SHERPA: Smart collaboration between Humans and ground-aerial Robots for improving rescuing activities in Alpine environments,” <http://www.sherpa-project.eu/sherpa/>, accessed: 2017-09-04.
- [26] “AEROWORKS: Collaborative Aerial Workers,” <http://www.aeroworks2020.eu/>, accessed: 2017-09-04.
- [27] F. Schiano, A. Franchi, D. Zelazo, and P. Robuffo Giordano, “A rigidity-based decentralized bearing formation controller for groups of quadrotor UAVs,” in *IEEE International Conference on Intelligent Robots and Systems*, 2016.
- [28] F. Schiano and P. Robuffo Giordano, “Bearing rigidity maintenance for formations of quadrotor uavs,” in *IEEE International Conference on Robotics and Automation*, 2017.
- [29] F. Schiano and R. Tron, “The Dynamic Bearing Observability Matrix, Nonlinear Observability and Estimation for Multi-Agent Systems,” in *IEEE International Conference on Robotics and Automation*, 2018.
- [30] D. Zelazo, A. Franchi, and P. Robuffo Giordano, “Rigidity Theory in  $SE(2)$  for Unscaled Position Estimation using only Bearing Measurements,” in *2014 ECC*, Strasbourg, France, Jun. 2014, pp. 2703–2708.
- [31] D. Zelazo, P. Robuffo Giordano, and A. Franchi, “Bearing-Only Formation Control Using an  $SE(2)$  Rigidity Theory,” in *IEEE International Conference on Decision and Control*, Osaka, Japan, 2015, pp. 6121–6126.

- [32] D. Fox, J. Ko, K. Konolige, B. Limketkai, D. Schulz, and B. Stewart, “Distributed multirobot exploration and mapping,” *Proc. of the IEEE*, vol. 94, no. 7, pp. 1325–1339, 2006.
- [33] N. Michael, J. Fink, and V. Kumar, “Cooperative manipulation and transportation with aerial robots,” in *2009 Robotics: Science and Systems*, Seattle, WA, Jun. 2009.
- [34] P. Robuffo Giordano, A. Franchi, C. Secchi, and H. H. Bühlhoff, “A passivity-based decentralized strategy for generalized connectivity maintenance,” *The Int. J. of Robotics Research*, vol. 32, no. 3, pp. 299–323, 2013.
- [35] K. Hausman, J. Müller, A. Hariharan, N. Ayanian, and G. S. Sukhatme, “Cooperative multi-robot control for target tracking with onboard sensing,” *IEEE Transactions on Automatic Control*, vol. 34, no. 13, pp. 1660–1677, 2015.
- [36] E. Montijano, E. Cristofalo, D. Zhou, M. Schwager, and C. Sagues, “Vision-based distributed formation control without an external positioning system,” *IEEE Transactions on Automatic Control*, vol. 32, no. 2, pp. 339–351, Jul. 2016.
- [37] P. R. Wurman, R. D’Andrea, and M. Mountz, “Coordinating hundreds of cooperative, autonomous vehicles in warehouses,” *AI magazine*, vol. 29, no. 1, p. 9, 2008.
- [38] A. Jaimes, S. Kota, and J. Gomez, “An approach to surveillance an area using swarm of fixed wing and quad-rotor unmanned aerial vehicles uav (s),” in *IEEE International Conference on System of Systems Engineering*. IEEE, 2008, pp. 1–6.
- [39] A. Franchi, L. Freda, G. Oriolo, and M. Vendittelli, “The sensor-based random graph method for cooperative robot exploration,” *IEEE/ASME Transactions on Mechatronics*, vol. 14, no. 2, pp. 163–175, 2009.
- [40] A. Renzaglia, L. Doitsidis, A. Martinelli, and E. B. Kosmatopoulos, “Multi-robot three-dimensional coverage of unknown areas,” *The Int. J. of Robotics Research*, vol. 31, no. 6, pp. 738–752, 2012.
- [41] F. Rafi, S. Khan, K. Shafiq, and M. Shah, “Autonomous target following by unmanned aerial vehicles,” in *Proceedings of SPIE*, vol. 6230, 2006, pp. 325–332.
- [42] Q. Lindsey, D. Mellinger, and V. Kumar, “Construction of cubic structures with quadrotor teams,” in *2011 Robotics: Science and Systems*, Los Angeles, CA, Jun. 2011.

- 
- [43] H. Joo, C. Son, K. Kim, K. Kim, and J. Kim, “A study on the advantages on high-rise building construction which the application of construction robots take (iccas 2007),” in *International Conference on Control, Automation and Systems (ICCAS)*. IEEE, 2007, pp. 1933–1936.
- [44] S. Berman, Q. Lindsey, M. S. Sakar, V. Kumar, and S. Pratt, “Study of group food retrieval by ants as a model for multi-robot collective transport strategies,” in *Robotics: Science and Systems*, 2010.
- [45] K. Sreenath and V. Kumar, “Dynamics control and planning for cooperative manipulation of payloads suspended by cables from multiple quadrotor robots,” *rm*, vol. 1, no. r2, p. r3, 2013.
- [46] Z. Wang and M. Schwager, “Force-amplifying n-robot transport system (force-ants) for cooperative planar manipulation without communication,” *The Int. J. of Robotics Research*, vol. 35, no. 13, pp. 1564–1586, 2016.
- [47] P. R. Chandler, M. Pachter, and S. Rasmussen, “Uav cooperative control,” in *American Control Conference, 2001. Proceedings of the 2001*, vol. 1. IEEE, 2001, pp. 50–55.
- [48] G. Antonelli, “Interconnected dynamic systems: An overview on distributed control,” *IEEE Control Systems*, vol. 33, no. 1, pp. 76–88, 2013.
- [49] C. Belta and V. Kumar, “Abstraction and control for groups of robots,” *IEEE Transactions on robotics*, vol. 20, no. 5, pp. 865–875, 2004.
- [50] G. Antonelli and S. Chiaverini, “Kinematic control of platoons of autonomous vehicles,” *IEEE Transactions on Robotics*, vol. 22, no. 6, pp. 1285–1292, 2006.
- [51] G. Antonelli, “Stability analysis for prioritized closed-loop inverse kinematic algorithms for redundant robotic systems,” *IEEE Transactions on Robotics*, vol. 25, no. 5, pp. 985–994, 2009.
- [52] A. Korchenko and O. Illyash, “The generalized classification of unmanned air vehicles,” in *Actual Problems of Unmanned Air Vehicles Developments Proceedings (APUAVD), 2013 IEEE 2nd International Conference*. IEEE, 2013, pp. 28–34.
- [53] M. Ryll, H. H. Bühlhoff, and P. R. Giordano, “Modeling and control of a quadrotor uav with tilting propellers,” in *IEEE International Conference on Robotics and Automation*, 2012, pp. 4606–4613.
- [54] A. Vuruskan, B. Yuksek, U. Ozdemir, A. Yukselen, and G. Inalhan, “Dynamic modeling of a fixed-wing vtol UAV,” in *International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2014, pp. 483–491.

- [55] C. Stöcker, R. Bennett, F. Nex, M. Gerke, and J. Zevenbergen, “Review of the current state of uav regulations,” *Remote Sensing*, vol. 9, no. 5, p. 459, 2017.
- [56] F. Augugliaro, S. Lupashin, M. Hamer, C. Male, M. Hehn, M. W. Mueller, J. S. Willmann, F. Gramazio, M. Kohler, and R. D’Andrea, “The flight assembled architecture installation: Cooperative construction with flying machines,” *IEEE Control Systems*, vol. 34, no. 4, pp. 46–64, 2014.
- [57] G. Gioioso, A. Franchi, G. Salvietti, S. Scheggi, and D. Prattichizzo, “The flying hand: A formation of UAVs for cooperative aerial tele-manipulation,” in *IEEE International Conference on Robotics and Automation*, 2014, pp. 4335–4341.
- [58] D. Mellinger, M. Shomin, N. Michael, and V. Kumar, “Cooperative grasping and transport using multiple quadrotors,” in *Distributed autonomous robotic systems*. Springer, 2013, pp. 545–558.
- [59] Q. Lindsey, D. Mellinger, and V. Kumar, “Construction with quadrotor teams,” *Autonomous Robots*, vol. 33, no. 3, pp. 323–336, 2012.
- [60] A. Mirjan, F. Augugliaro, R. D’Andrea, F. Gramazio, and M. Kohler, “Building a bridge with flying robots,” in *Robotic Fabrication in Architecture, Art and Design 2016*. Springer, 2016, pp. 34–47.
- [61] R. Ritz, M. W. Müller, M. Hehn, and R. D’Andrea, “Cooperative quadcopter ball throwing and catching,” in *IEEE International Conference on Intelligent Robots and Systems*, 2012, pp. 4972–4978.
- [62] P. Robuffo Giordano, A. Franchi, C. Secchi, and H. H. Bühlhoff, “Bilateral teleoperation of groups of UAVs with decentralized connectivity maintenance,” in *Robotics: Science and Systems*, Los Angeles, CA, Jun. 2011.
- [63] D. Lee, A. Franchi, H. I. Son, H. H. Bühlhoff, and P. Robuffo Giordano, “Semi-autonomous haptic teleoperation control architecture of multiple unmanned aerial vehicles,” *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 4, pp. 1334–1345, 2013.
- [64] T. Krajník, M. Nitsche, J. Faigl, P. Vanek, M. Saska, L. Preucil, T. Duckett, and M. Mejail, “A practical multirobot localization system,” *Journal of Intelligent & Robotic Systems*, vol. 76, no. 3-4, pp. 539–562, 2014.
- [65] G. Loianno, C. Brunner, G. McGrath, and V. Kumar, “Estimation, control, and planning for aggressive flight with a small quadrotor with a single camera and imu,” *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 404–411, 2017.

- 
- [66] K.-K. Oh, M.-C. Park, and H.-S. Ahn, “A survey of multi-agent formation control,” *Automatica*, vol. 53, pp. 424–440, 2015.
- [67] R. Olfati-Saber, J. A. Fax, and R. M. Murray, “Consensus and cooperation in networked multi-agent systems,” *Proc. of the IEEE*, vol. 1, no. 95, pp. 215–233, 2007.
- [68] N. A. Lynch, *Distributed algorithms*. Morgan Kaufmann, 1996.
- [69] R. Olfati-Saber and R. M. Murray, “Consensus problems in networks of agents with switching topology and time-delays,” *IEEE Transactions on automatic control*, vol. 49, no. 9, pp. 1520–1533, 2004.
- [70] R. A. Freeman, P. Yang, and K. M. Lynch, “Stability and convergence properties of dynamic average consensus estimators,” in *IEEE International Conference on Decision and Control*, San Diego, CA, Dec. 2006, pp. 338–343.
- [71] W. Ren, R. W. Beard, and E. M. Atkins, “A survey of consensus problems in multi-agent coordination,” in *American Control Conference, 2005. Proceedings of the 2005*. IEEE, 2005, pp. 1859–1864.
- [72] B. D. O. Anderson, C. Yu, B. Fidan, and J. M. Hendrickx, “Rigid graph control architectures for autonomous formations,” *IEEE Control Systems Magazine*, vol. 28, no. 6, pp. 48–63, 2008.
- [73] Y. Hou, “Control of formations with non-rigid and hybrid graphs,” Ph.D. dissertation, The Australian National University, 2016.
- [74] M. Basiri, A. N. Bishop, and P. Jensfelt, “Distributed control of triangular formations with angle-only constraints,” *Systems & Control Letters*, vol. 59, no. 2, pp. 147–154, 2010.
- [75] A. N. Bishop, “A very relaxed control law for bearing-only triangular formation control,” *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 5991–5998, 2011.
- [76] A. N. Bishop, I. Shames, and B. D. O. Anderson, “Stabilization of rigid formations with direction-only constraints,” in *IEEE International Conference on Decision and Control*, Orlando, FL, Dec. 2011, pp. 746–752.
- [77] A. Franchi, C. Masone, V. Grabe, M. Ryll, H. H. Bühlhoff, and P. Robuffo Giordano, “Modeling and control of UAV bearing-formations with bilateral high-level steering,” *The Int. J. of Robotics Research, Special Issue on 3D Exploration, Mapping, and Surveillance*, vol. 31, no. 12, pp. 1504–1525, 2012.

- [78] L. E. Parker, “Designing control laws for cooperative agent teams,” in *IEEE International Conference on Robotics and Automation*. IEEE, 1993, pp. 582–587.
- [79] K. Fathian, D. I. Rachinskii, T. H. Summers, M. W. Spong, and N. R. Gans, “Distributed formation control under arbitrarily changing topology,” in *American Control Conference (ACC), 2017*. IEEE, 2017, pp. 271–278.
- [80] B. A. Francis and M. Maggiore, *Flocking and rendezvous in distributed robotics*. Springer, 2016.
- [81] R. Kurazume, S. Nagata, and S. Hirose, “Cooperative positioning with multiple robots,” in *IEEE International Conference on Robotics and Automation*, 1994, pp. 1250–1257.
- [82] D. Fox, W. Burgard, H. Kruppa, and S. Thrun, “Collaborative multi-robot localization,” in *Mustererkennung 1999*. Springer, 1999, pp. 15–26.
- [83] ———, “A probabilistic approach to collaborative multi-robot localization,” *Autonomous Robots*, vol. 8, no. 3, pp. 325–344, 2000.
- [84] S. Panzieri, F. Pascucci, and R. Setola, “Multirobot localisation using interlaced extended kalman filter,” in *IEEE International Conference on Intelligent Robots and Systems*, 2006, pp. 2816–2821.
- [85] S. Roumeliotis and G. A. Bekey, “Distributed multirobot localization,” *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 781–795, 2002.
- [86] S. S. Kia, S. F. Rounds, and S. Martinez, “A centralized-equivalent decentralized implementation of extended kalman filters for cooperative localization,” in *IEEE International Conference on Intelligent Robots and Systems*, 2014, pp. 3761–3766.
- [87] A. Prorok, A. Bahr, and A. Martinoli, “Low-cost collaborative localization for large-scale multi-robot systems,” in *IEEE International Conference on Robotics and Automation*. Ieee, 2012, pp. 4236–4241.
- [88] B. Shirmohammadi and C. Taylor, “Self localizing smart camera networks,” *ACM Transactions on Sensor Networks*, 2010.
- [89] T. Eren, “Graph invariants for unique localizability in cooperative localization of wireless sensor networks: rigidity index and redundancy index,” *Ad Hoc Networks*, vol. 44, pp. 32–45, 2016.
- [90] R. K. Williams, A. Gasparri, A. Priolo, and G. S. Sukhatme, “Evaluating network rigidity in realistic systems: Decentralization, asynchronicity, and parallelization,” *IEEE Transactions on Automatic Control*, vol. 4, no. 30, pp. 950–965, 2014.

- 
- [91] I. Shames, A. N. Bishop, and B. D. O. Anderson, "Analysis of Noisy Bearing-Only Network Localization," *IEEE Transactions on Automatic Control*, vol. 58, no. 1, pp. 247–252, Jan. 2013.
- [92] M. Brand, M. Antone, and S. Teller, "Spectral solution of large-scale extrinsic camera calibration as a graph embedding problem," in *IEEE European Conference on Computer Vision*, 2004, pp. 262–273.
- [93] H. Li, "Multi-view structure computation without explicitly estimating motion," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2777–2784.
- [94] B. Servatius and W. Whiteley, "Constraining plane configurations in computer-aided design: Combinatorics of directions and lengths," *SIAM Journal on Discrete Mathematics*, vol. 12, no. 1, pp. 136–153, 1999.
- [95] A. Franchi, G. Oriolo, and P. Stegagno, "Probabilistic mutual localization in multi-agent systems from anonymous position measures," in *IEEE International Conference on Decision and Control*. IEEE, 2010, pp. 6534–6540.
- [96] M. Cagnetti, P. Stegagno, A. Franchi, G. Oriolo, and H. H. Bühlhoff, "3-D mutual localization with anonymous bearing measurements," in *IEEE International Conference on Robotics and Automation*, St. Paul, MN, May 2012, pp. 791–798.
- [97] X. S. Zhou and S. I. Roumeliotis, "Determining 3-d relative transformations for any combination of range and bearing measurements," *IEEE Transactions on Robotics*, vol. 29, no. 2, pp. 458–474, 2013.
- [98] A. Martinelli, "Closed-form solutions for attitude, speed, absolute scale and bias determination by fusing vision and inertial measurements," INRIA Grenoble Rhone-Alpes, Tech. Rep., 2011.
- [99] G. C. Calafiore, L. Carlone, and M. Wei, "A distributed gradient method for localization of formations using relative range measurements," in *IEEE Int. Symp. on Computer-Aided Control System Design*, Yokohama, Japan, Sep. 2010, pp. 1146–1151.
- [100] T. Eren, "Using angle of arrival (bearing) information for localization in robot networks," *Turkish J. of Elec. Eng. & Comput. Sci.*, vol. 15, no. 2, pp. 169–186, 2007.
- [101] M. Ye, B. D. Anderson, and C. Yu, "Bearing-only measurement self-localization, velocity consensus and formation control," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 2, pp. 575–586, 2017.

- [102] A. Martinelli and R. Siegwart, “Observability analysis for mobile robot localization,” in *IEEE International Conference on Intelligent Robots and Systems*. IEEE, 2005, pp. 1471–1476.
- [103] X. S. Zhou and S. Roumeliotis, “Determining the robot-to-robot 3d relative pose using combinations of range and bearing measurements: 14 minimal problems and closed-form solutions to three of them,” in *IEEE International Conference on Intelligent Robots and Systems*, 2010, pp. 2983–2990.
- [104] P. Stegagno, “Mutual localization from anonymous measurements in multi-robot systems,” Ph.D. dissertation, Università di Roma La Sapienza, 2012.
- [105] A. Franchi, G. Oriolo, and P. Stegagno, “Mutual localization in a multi-robot system with anonymous relative position measures,” in *IEEE International Conference on Intelligent Robots and Systems*, 2009, pp. 3974–3980.
- [106] P. Stegagno, M. Cagnetti, A. Franchi, and G. Oriolo, “Mutual localization using anonymous bearing measurements,” in *IEEE International Conference on Intelligent Robots and Systems*, 2011, pp. 469–474.
- [107] P. Stegagno, M. Cagnetti, G. Oriolo, H. H. Bühlhoff, and A. Franchi, “Ground and aerial mutual localization using anonymous relative-bearing measurements,” *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1133–1151, 2016.
- [108] N. Biggs, E. K. Lloyd, and R. J. Wilson, *Graph Theory, 1736-1936*. Oxford University Press, 1976.
- [109] A.-L. Cauchy, “Recherche sur les polyèdres-premier mémoire,” *Journal de l’Ecole Polytechnique*. v9, pp. 66–86, 1813.
- [110] G. L’Huillier, “Recherche sur les polyèdres-premier mémoire,” *Annales de Mathématiques pures et appliquées, tome 3*, pp. 169–189, 1812-1813.
- [111] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks*. Princeton University Press, 2010.
- [112] A. Franchi and P. Robuffo Giordano, “Decentralized control of parallel rigid formations with direction constraints and bearing measurements,” in *IEEE International Conference on Decision and Control*, Maui, HI, Dec. 2012, pp. 5310–5317.
- [113] F. Bullo, J. Cortes, and S. Martinez, *Distributed control of robotic networks: a mathematical approach to motion coordination algorithms*. Princeton University Press, 2009.

- 
- [114] M. Fiedler, “Algebraic connectivity of graphs,” *Czechoslovak mathematical journal*, vol. 23, no. 2, pp. 298–305, 1973.
- [115] —, “Laplacian of graphs and algebraic connectivity,” *Banach Center Publications*, vol. 25, no. 1, pp. 57–70, 1989.
- [116] L. Krick, M. E. Broucke, and B. A. Francis, “Stabilisation of infinitesimally rigid formations of multi-robot networks,” *International Journal of Control*, vol. 82, no. 3, pp. 423–439, Mar. 2009.
- [117] B. D. Anderson, I. Shames, G. Mao, and B. Fidan, “Formal theory of noisy sensor network localization,” *SIAM Journal on Discrete Mathematics*, vol. 24, no. 2, pp. 684–698, 2010.
- [118] H. Garcia de Marina, M. Cao, and B. Jayawardhana, “Controlling rigid formations of mobile agents under inconsistent measurements,” *IEEE Transactions on Robotics*, vol. 31, no. 1, pp. 31–39, Feb 2015.
- [119] D. Zelazo, A. Franchi, H. H. Bühlhoff, and P. Robuffo Giordano, “Decentralized Rigidity Maintenance Control with Range-only Measurements for Multi-Robot Systems,” *The Int. J. of Robotics Research*, vol. 34, no. 1, pp. 105–128, 2015.
- [120] T. Eren, “Formation shape control based on bearing rigidity,” *International Journal of Control*, vol. 9, no. 85, pp. 1361–1379, 2012.
- [121] A. Cornejo, A. J. Lynch, E. Fudge, S. Bilstein, M. Khabbazi, and J. McLurkin, “Scale-free coordinates for multi-robot systems with bearing-only sensors,” *The Int. J. of Robotics Research*, vol. 12, no. 32, pp. 1459–1474, 2013.
- [122] S. Zhao and D. Zelazo, “Bearing rigidity and almost global bearing-only formation stabilization,” *IEEE Transactions on Automatic Control*, vol. 61, no. 5, pp. 1255–1268, 2016.
- [123] —, “Translational and scaling formation maneuver control via a bearing-based approach,” *IEEE Transactions on Control of Network Systems*, vol. 4, no. 3, pp. 429–438, 2017.
- [124] J. Aspnes, W. Whiteley, and Y. R. Yang, “A theory of network localization,” *IEEE Transactions on Mobile Computing*, vol. 5, no. 12, pp. 1663–1678, 2006.
- [125] I. Shames, A. N. Bishop, and B. D. O. Anderson, “Analysis of noisy bearing-only network localization,” *IEEE Trans. on Automatic Control*, vol. 1, no. 58, pp. 247–252, 2013.

- [126] R. Tron, L. Carlone, F. Dellaert, and K. Daniilidis, “Rigid components identification and rigidity enforcement in bearing-only localization using the graph cycle basis,” in *IEEE American Control Conference*, 2015.
- [127] D. Zelazo, A. Franchi, F. Allgöwer, H. H. Bühlhoff, and P. Robuffo Giordano, “Rigidity maintenance control for multi-robot systems,” in *2012 Robotics: Science and Systems*, Sydney, Australia, Jul. 2012.
- [128] G. Laman, “On graphs and rigidity of plane skeletal structures,” *Journal of Engineering Mathematics*, vol. 4, no. 4, pp. 331–340, 1970.
- [129] T. Tay and W. Whiteley, “Generating isostatic frameworks,” *Structural Topology*, vol. 11, no. 1, pp. 21–69, 1985.
- [130] W. Whiteley, “Some matroids from discrete applied geometry,” *Contemporary Mathematics*, vol. 197, pp. 171–312, 1996.
- [131] R. K. Williams, A. Gasparri, M. Soffietti, and G. S. Sukhatme, “Redundantly rigid topologies in decentralized multi-agent networks,” in *IEEE International Conference on Decision and Control*. IEEE, 2015, pp. 6101–6108.
- [132] T. Eren, “Combinatorial measures of rigidity in wireless sensor and robot networks,” in *IEEE International Conference on Decision and Control*. IEEE, 2015, pp. 6109–6114.
- [133] L. Henneberg, *Die graphische Statik der starren Systeme*. BG Teubner, 1911, vol. 31.
- [134] S. Bereg, “Certifying and constructing minimally rigid graphs in the plane,” in *Proceedings of the twenty-first annual symposium on Computational geometry*. ACM, 2005, pp. 73–80.
- [135] L. Asimow and B. Roth, “The Rigidity of Graphs, II,” *Journal of Mathematical Analysis and Applications*, vol. 68, pp. 171–190, 1979.
- [136] B. D. O. Anderson, B. Fidan, C. Yu, and D. van der Walle, “UAV formation control: Theory and application,” in *Recent Advances in Learning and Control*, ser. Lecture Notes in Control and Information Sciences, V. D. Blondel, S. P. Boyd, and H. Kimura, Eds. Springer, 2008, vol. 371, pp. 15–34.
- [137] T. Eren, W. Whiteley, and P. N. Belhumeur, “Using angle of arrival (bearing) information in network localization,” in *IEEE International Conference on Decision and Control*. IEEE, 2006, pp. 4676–4681.

- 
- [138] T. Eren, “Formation shape control based on bearing rigidity,” *International Journal of Control*, vol. 85, no. 9, pp. 1361–1379, Sep. 2012.
- [139] J. N. Twigg, J. R. Fink, L. Y. Paul, and B. M. Sadler, “Rss gradient-assisted frontier exploration and radio source localization,” in *IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 889–895.
- [140] R. Tron, J. Thomas, G. Loianno, K. Daniilidis, and V. Kumar, “A distributed optimization framework for localization and formation control,” *IEEE Control Systems Magazine*, vol. 36, no. 4, pp. 22–44, 2016.
- [141] S. Zhao and D. Zelazo, “Bearing-constrained formation control using bearing measurements,” in *Israel Annual Conference on Aerospace Sciences*, 2015.
- [142] Y. Ma, *An invitation to 3-D vision: from images to geometric models*. Springer, 2004.
- [143] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.
- [144] E. S. Jones and S. Soatto, “Visual-inertial navigation, mapping and localization: A scalable real-time causal approach,” *International Journal of Computer Vision*, vol. 30, no. 4, pp. 407–430, 2011.
- [145] D. Scaramuzza, M. C. Achtelik, L. Doitsidis, F. Fraundorfer, E. B. Kosmatopoulos, A. Martinelli, M. W. Achtelik, M. Chli, S. A. Chatzichristofis, L. Kneip, D. Gurdan, L. Heng, G. H. Lee, S. Lynen, L. Meier, M. Pollefeys, A. Renzaglia, R. Siegwart, J. C. Stumpf, P. Tanskanen, C. Troiani, and S. Weiss, “Vision-controlled micro flying robots: from system design to autonomous navigation and mapping in GPS-denied environments,” *IEEE Robotics & Automation Magazine*, vol. 21, no. 3, pp. 26–40, 2014.
- [146] R. Spica and P. Robuffo Giordano, “Active decentralized scale estimation for bearing-based localization,” in *IEEE International Conference on Intelligent Robots and Systems*, 2016, pp. 5084–5091.
- [147] J. Fink, N. Michael, S. Kim, and V. Kumar, “Planning and control for cooperative manipulation and transportation with aerial robots,” *The Int. J. of Robotics Research*, vol. 30, no. 3, pp. 324–334, 2010.
- [148] T. Lee, M. Leokyand, and N. H. McClamroch, “Geometric tracking control of a quadrotor UAV on SE(3),” in *IEEE International Conference on Decision and Control*, 2010, pp. 5420–5425.

- [149] C. D. Godsil and G. Royle, *Algebraic Graph Theory*. Springer, 2001.
- [150] T. Eren, W. Whiteley, A. S. Morse, P. N. Belhumeur, and B. D. Anderson, “Sensor and network topologies of formations with direction, bearing, and angle information between agents,” in *IEEE International Conference on Decision and Control*, vol. 3. IEEE, 2003, pp. 3064–3069.
- [151] K.-K. Oh and H.-S. Ahn, “Formation control of mobile agents based on inter-agent distance dynamics,” *Automatica*, vol. 47, no. 10, pp. 2306 – 2312, 2011.
- [152] J. Cortés, “Global and robust formation-shape stabilization of relative sensing networks,” *Automatica*, vol. 12, no. 45, pp. 2754–2762, 2009.
- [153] E. Montijano, D. Zhou, M. Schwager, and C. Sagues, “Distributed formation control without a global reference frame,” in *IEEE American Control Conference*, Portland, OR, 2014, pp. 3862–3867.
- [154] H. K. Khalil, *Nonlinear Systems*. Prentice-Hall, New Jersey, 1996.
- [155] P. Yang, R. A. Freeman, G. J. Gordon, K. M. Lynch, S. S. Srinivasa, and R. Sukthankar, “Decentralized estimation and control of graph connectivity for mobile sensor networks,” *Automatica*, vol. 46, no. 2, pp. 390–396, 2010.
- [156] A. De Luca, G. Oriolo, and P. Robuffo Giordano, “Feature depth observation for image-based visual servoing: Theory and experiments,” *The Int. J. of Robotics Research*, vol. 27, no. 10, pp. 1093–1116, 2008.
- [157] R. Spica, P. Robuffo Giordano, and F. Chaumette, “Active Structure from Motion: Application to Point, Sphere and Cylinder,” *IEEE Transactions on Automatic Control*, vol. 30, no. 6, pp. 1499–1513, 2014.
- [158] E. Rohmer, S. P. N. Singh, and M. Freese, “V-rep: a versatile and scalable robot simulation framework,” in *IEEE International Conference on Intelligent Robots and Systems*, 2013.
- [159] V. Grabe, M. Riedel, H. H. Bulthoff, P. Robuffo Giordano, and A. Franchi, “The TeleKyb framework for a modular and extendible ros-based quadrotor control,” in *IEEE European Conference on Mobile Robots*, Barcelona, Spain, 2013, pp. 19–25.
- [160] R. Spica, P. R. Giordano, M. Ryll, H. H. Bülthoff, and A. Franchi, “An open-source hardware/software architecture for quadrotor uavs,” *IFAC Proceedings Volumes*, vol. 46, no. 30, pp. 198–205, 2013.

- 
- [161] L. Sabattini, C. Secchi, and N. Chopra, “Decentralized connectivity maintenance for networked lagrangian dynamical systems,” in *IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 2433–2438.
- [162] S. Zhao and D. Zelazo, “Bearing-Based Distributed Control and Estimation of Multi-Agent Systems,” in *European Control Conference*, Linz, Austria, 2015, pp. 2207–2212.
- [163] A. R. Vidal, H. Rebecq, T. Horstschaefer, and D. Scaramuzza, “Hybrid, frame and event based visual inertial odometry for robust, autonomous navigation of quadrotors,” *arXiv preprint arXiv:1709.06310*, 2017.
- [164] P. Lichtsteiner, C. Posch, and T. Delbruck, “A  $128 \times 128$  120 db  $15\mu\text{s}$  latency asynchronous temporal contrast vision sensor,” *IEEE Journal of solid-state circuits*, vol. 43, no. 2, pp. 566–576, 2008.
- [165] M. I. Friswell, “The derivatives of repeated eigenvalues and their associated eigenvectors,” *Journal of Vibration and Acoustics*, vol. 118, pp. 390–397, 1996.
- [166] X. Chen, H. Qi, L. Qi, and K.-L. Teo, “Smooth convex approximation to the maximum eigenvalue function,” *Journal of Global Optimization*, vol. 30, no. 2, pp. 253–270, 2004.
- [167] K. Sreenath, T. Lee, and V. Kumar, “Geometric control and differential flatness of a quadrotor uav with a cable-suspended load,” in *IEEE International Conference on Decision and Control*. IEEE, 2013, pp. 2269–2274.
- [168] Z. Wang and M. Schwager, “Kinematic multi-robot manipulation with no communication using force feedback,” in *IEEE International Conference on Robotics and Automation*, 2016, pp. 427–432.
- [169] T. Eren, W. Whiteley, and P. N. Belhumeur, “Using angle of arrival (bearing) information in network localization,” in *IEEE International Conference on Decision and Control*. IEEE, 2006, pp. 4676–4681.
- [170] A. N. Bishop, I. Shames, and B. Anderson, “Stabilization of rigid formations with direction-only constraints,” in *IEEE International Conference on Decision and Control*, 2011, pp. 746–752.
- [171] A. Franchi and P. R. Giordano, “Decentralized control of parallel rigid formations with direction constraints and bearing measurements,” in *IEEE International Conference on Decision and Control*, 2012, pp. 5310–5317.

- [172] S. Zhao, F. Lin, K. Peng, B. M. Chen, and T. H. Lee, “Distributed control of angle-constrained cyclic formations using bearing-only measurements,” *Systems & Control Letters*, vol. 63, pp. 12–24, 2014.
- [173] G. Michieletto, A. Cenedese, and A. Franchi, “Bearing rigidity theory in  $SE(3)$ ,” in *IEEE International Conference on Decision and Control*. IEEE, 2016, pp. 5950–5955.
- [174] A. Martinelli, “Vision and IMU data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination,” *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 44–60, 2012.
- [175] R. K. Williams and G. S. Sukhatme, “Observability in topology-constrained multi-robot target tracking,” in *IEEE International Conference on Robotics and Automation*. IEEE, 2015, pp. 1795–1801.
- [176] R. Hermann and A. Krener, “Nonlinear controllability and observability,” *IEEE Transactions on automatic control*, vol. 22, no. 5, pp. 728–740, 1977.
- [177] A. Isidori, *Nonlinear control systems*. Springer Science & Business Media, 2013.
- [178] F. Bullo and A. D. Lewis, *Geometric control of mechanical systems: modeling, analysis, and design for simple mechanical control systems*. Springer Science & Business Media, 2004, vol. 49.
- [179] F. L. Markley and J. L. Crassidis, *Fundamentals of spacecraft attitude determination and control*. Springer, 2014, vol. 33.
- [180] T. Lee, M. Leok, and N. H. McClamroch, “Nonlinear robust tracking control of a quadrotor UAV on  $SE(3)$ ,” *Asian Journal of Control*, vol. 15, no. 2, pp. 391–408, 2013.
- [181] G. Loianno, M. Watterson, and V. Kumar, “Visual inertial odometry for quadrotors on  $se(3)$ ,” in *IEEE International Conference on Robotics and Automation*. IEEE, 2016, pp. 1544–1551.
- [182] R. Tron and R. Vidal, “Distributed 3-D localization of camera sensor networks from 2-D image measurements,” *IEEE Transactions on Automatic Control*, 2014.
- [183] S. I. Roumeliotis and I. M. Rekleitis, “Propagation of uncertainty in cooperative multirobot localization: Analysis and experimental results,” *Autonomous Robots*, vol. 17, no. 1, pp. 41–54, 2004.

- 
- [184] L. Luft, T. Schubert, S. I. Roumeliotis, and W. Burgard, “Recursive Decentralized Collaborative Localization for Sparsely Communicating Robots,” *Proceedings of Robotics: Science and Systems*, 2016.
- [185] T. Lee, “Geometric control of multiple quadrotor uavs transporting a cable-suspended rigid body,” in *IEEE International Conference on Decision and Control*. IEEE, 2014, pp. 6155–6160.
- [186] A. Barrau and S. Bonnabel, “The invariant extended kalman filter as a stable observer,” *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1797–1812, 2017.
- [187] M. P. do Carmo, *Riemannian geometry*. Boston, MA: Birkhäuser, 1992.
- [188] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2012.
- [189] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. Princeton, NJ: Princeton University Press, 2008.
- [190] R. Tron, “Distributed optimization on manifolds for consensus algorithms and camera network localization,” Ph.D. dissertation, The Johns Hopkins University, 2012.
- [191] Z. Lendek, R. Babuska, and B. D. Schutter, “Distributed Kalman filtering for multiagent systems,” *IEEE European Control Conference*, vol. 19, pp. 2193–2200, 2007.
- [192] A. Martinelli, “Nonlinear Unknown Input Observability: Extension of the Observability Rank Condition,” *IEEE Transactions on Automatic Control*, vol. 9286, pp. 1–16, 2018.
- [193] T. Toksoz, J. Redding, M. Michini, B. Michini, J. P. How, M. Vavrina, and J. Vian, “Automated battery swap and recharge to enable persistent uav missions,” in *AIAA Infotech@ Aerospace Conference*, 2011.
- [194] N. K. Ure, G. Chowdhary, T. Toksoz, J. P. How, M. A. Vavrina, and J. Vian, “An automated battery management system to enable persistent missions with multiple aerial vehicles,” *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 1, pp. 275–286, 2015.
- [195] F. Cocchioni, E. Frontoni, G. Ippoliti, S. Longhi, A. Mancini, and P. Zingaretti, “Visual based landing for an unmanned quadrotor,” *The Int. J. of Robotics Research*, vol. 84, no. 1-4, pp. 511–528, 2016.

- [196] E. Olson, “AprilTag: A robust and flexible visual fiducial system,” in *IEEE International Conference on Robotics and Automation*. IEEE, May 2011, pp. 3400–3407.
- [197] J. Wang and E. Olson, “AprilTag 2: Efficient and robust fiducial detection,” in *IEEE International Conference on Intelligent Robots and Systems*. IEEE, 2016, pp. 4193–4198.
- [198] E. Olson, J. Strom, R. Goeddel, R. Morton, P. Ranganathan, and A. Richardson, “Exploration and mapping with autonomous robot teams,” *Communications of the ACM*, vol. 56, no. 3, pp. 62–70, 2013.
- [199] A. Richardson, J. Strom, and E. Olson, “Aprilcal: Assisted and repeatable camera calibration,” in *IEEE International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 1814–1821.
- [200] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, “The euroc micro aerial vehicle datasets,” *The Int. J. of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.
- [201] S. Kyriasis, A. Antonopoulos, T. Chaniakakis, E. Stefanakis, C. Linardos, A. Tripolitsiotis, and P. Partinevelos, “Towards autonomous modular uav missions: The detection, geo-location and landing paradigm,” *Sensors*, vol. 16, no. 11, p. 1844, 2016.
- [202] M. Saska, J. Chudoba, L. Preucil, J. Thomas, G. Loianno, A. Tresnak, V. Vonasek, and V. Kumar, “Autonomous deployment of swarms of micro-aerial vehicles in cooperative surveillance,” in *IEEE International Conference on Unmanned Aircraft Systems*, 2014, pp. 584–595.
- [203] M. Saska, T. Baca, J. Thomas, J. Chudoba, L. Preucil, T. Krajnik, J. Faigl, G. Loianno, and V. Kumar, “System for deployment of groups of unmanned micro aerial vehicles in gps-denied environments using onboard visual relative localization,” *Autonomous Robots*, pp. 1–26, 2016.
- [204] S. Zhao, “Control and navigation of multi-vehicle systems using visual measurements,” Ph.D. dissertation, National University of Singapore, 2014.
- [205] T. Nestmeyer, P. R. Giordano, H. H. Bühlhoff, and A. Franchi, “Decentralized simultaneous multi-target exploration using a connected network of multiple robots,” *Autonomous Robots*, vol. 41, no. 4, pp. 989–1011, 2017.
- [206] M. A. Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

- [207] C. Nissler, B. Stefan, Z.-C. Marton, L. Beckmann, U. Thomasy *et al.*, “Evaluation and improvement of global pose estimation with multiple apriltags for industrial manipulators,” in *Emerging Technologies and Factory Automation (ETFA), 2016 IEEE 21st International Conference on*. IEEE, 2016, pp. 1–8.
- [208] R. Spica, G. Claudio, F. Spindler, and P. Robuffo Giordano, “Interfacing Matlab/Simulink with V-REP for an easy development of sensor-based control algorithms for robotic platforms,” in *IEEE International Conference on Robotics and Automation Workshop on MATLAB/Simulink for Robotics Education and Research*, 2014.
- [209] A. Mallet, C. Pasteur, M. Herrb, S. Lemaignan, and F. Ingrand, “GenoM3: Building middleware-independent robotic components,” in *IEEE International Conference on Robotics and Automation*. IEEE, 2010, pp. 4627–4632.
- [210] F. Schiano, J. Alonso-Mora, K. Rudin, P. Beardsley, R. Siegwart, and S. B., “Towards estimation and correction of wind effects on a quadrotor UAV,” in *International Micro Air Vehicle Conference*, 2014, pp. 134–141.



---

## Vita

Fabrizio Schiano received his BSc degree in 2010 and MSc degree (highest honors) in 2013 in Automation Engineering from the *Università degli Studi di Napoli Federico II* in Naples, Italy. He spent 6 months as a visiting MSc student at the Swiss Federal Institute of Technology in Zurich (ETH) and Disney Research Zurich (DRZ) [210]. He worked for 13 months as a Research Scientist at the *Zentrum für Telematik* (ZFT) in Würzburg, Germany. He is currently a Ph.D. student at INRIA Rennes Bretagne Atlantique and *Université de Rennes 1* in Rennes, France. During his Ph.D. he spent 6 months at Boston University as a Ph.D. visiting researcher in Boston (MA), USA.

## Résumé

Le but de cette thèse est d'étendre l'état de l'art par des contributions sur le comportement collectif d'un groupe de robots volants, à savoir des quadrirotors UAV. Afin de pouvoir sûrement naviguer dans un environnement, ces derniers peuvent se reposer uniquement sur leurs capacités à bord et non sur des systèmes centralisés (e.g., Vicon ou GNSS). Nous réalisons cet objectif en offrant une possible solution aux problèmes de contrôle en formation et de localisation à partir de mesures à bord et via une communication locale. Nous abordons ces problèmes exploitant différents concepts provenant de la théorie des graphes algébriques et de la *théorie de la rigidité*. Cela nous permet de résoudre ces problèmes de façon décentralisée et de proposer des algorithmes décentralisés capables de prendre en compte également des limites sensorielles classiques. Les capacités embarquées que nous avons mentionnées plus tôt sont représentées par une caméra monoculaire et une centrale inertielle (IMU) auxquelles s'ajoute la capacité de chaque robot à communiquer (par RF) avec certains de ses voisins. Cela est dû au fait que l'IMU et la caméra représentent une possible configuration économique et légère pour la navigation et la localisation autonome d'un *quadrotor* UAV.

## Abstract

The aim of this Thesis is to give contributions to the state of the art on the collective behavior of a group of flying robots, specifically quadrotor UAVs, which can only rely on their onboard capabilities and not on a centralized system (e.g., Vicon or GNSS) in order to safely navigate in the environment. We achieve this goal by giving a possible solution to the problems of formation control and localization from onboard sensing and local communication. We tackle these problems exploiting mainly concepts from algebraic graph theory and the so-called *theory of rigidity*. This allows us to solve these problems in a decentralized fashion, and propose decentralized algorithms able to also take into account some typical sensory limitations. The onboard capabilities we referred to above are represented by an onboard monocular camera and an inertial measurement unit (IMU) in addition to the capability of each robot to communicate (through RF) with some of its neighbors. This is due to the fact that an IMU and a camera represent a possible minimal, lightweight and inexpensive configuration for the autonomous localization and navigation of a quadrotor UAV.

