Synthèse des travaux scientifiques présentée par

Fabio Morbidi

pour obtenir le diplôme d'

Habilitation à Diriger des Recherches

Discipline : Sciences pour l'Ingénieur

Université de Picardie Jules Verne
École Doctorale en Sciences, Technologie et Santé (EDSTS 585)

Cooperative Multi-robot Systems: From Perception to Action

Soutenue le 31 janvier 2024 devant le jury composé de :

- M. El Mustapha Mouaddib, PU, Université de Picardie Jules Verne (parrain HDR)
- M. Simon Lacroix, DR CNRS, LAAS-CNRS Toulouse (rapporteur)
- M. Youcef Mezouar, PU, Université Clermont Auvergne (rapporteur)
- M. Olivier Simonin, PU, INSA de Lyon (rapporteur)
- M. Paolo Robuffo Giordano, DR CNRS, IRISA/INRIA Rennes (examinateur)
- Mme Véronique Cherfaoui, PU, Univ. Technologie de Compiègne (examinatrice)
- M. Pascal Vasseur, PU, Université de Picardie Jules Verne (examinateur)

Année Universitaire 2023-2024

Cooperative Multi-robot Systems: From Perception to Action

Fabio Morbidi

Université de Picardie Jules Verne
Amiens, 1er février 2024

Une version numérique du mémoire est disponible à l'adresse :
`https://home.mis.u-picardie.fr/~fabio/Publications.html`
ou peut être obtenue sur demande auprès de l'auteur à l'adresse e-mail :
`fabio.morbidi@u-picardie.fr`

Contents

Remerciements

Une réunion, c'est un groupe de gens réunis pour compliquer un travail qu'une seule personne ferait sans mal.

"Try to Remember" – Frank Herbert
Amazing Stories, Fact and Science Fiction, 1961

Vu le grand nombre de collaborateurs, étudiants, amis et collègues qui m'ont entouré et soutenu au cours des dernières années, l'écriture de quelques mots de remerciements s'avère être un exercice assez difficile et j'espère que les personnes que je vais malheureusement oublier ne m'en voudront pas trop.

Tout d'abord, je voudrais témoigner ma gratitude à mon parrain[1], El Mustapha Mouaddib, pour sa générosité, sa franchise, son amitié et sa bonne humeur contagieuse. Il a eu confiance en moi et c'est surtout grâce à lui que j'ai pu rejoindre l'UPJV en septembre 2014. J'espère avoir répondu à toutes ses attentes au cours de ces neuf années. Merci aussi pour les bons moments de convivialité et détente passés à discuter de science, histoire de l'art, cinéma, politique, et surtout gastronomie ! en dehors des heures de travail.

Je suis également très reconnaissant à toutes les personnes qui ont travaillé avec moi pendant mes années de post-doc : Gian Luca Mariottini (le premier à m'avoir initié au métier de chercheur et mon vrai maître), Luca Consolini (pour ses enseignements précieux en commande non linéaire et sa grande disponibilité et sympathie), Randy A. Freeman et Kevin M. Lynch (pour avoir partagé avec moi leurs immenses connaissances en automatique, mécanique et robotique), He Bai (pour les riches échanges scientifiques et pour m'avoir gentiment remercié dans la préface de son excellent livre sur la commande coopérative), Stefano Scheggi, Aaron Staranowicz, Gustavo A. Puerto-Souza, Federica Garin et Alain Y. Kibangou. J'ai beaucoup appris en travaillant avec chacun d'eux, sur comment mener des recherches originales et rédiger des articles de qualité.

[1] Irrésistiblement, ce mot évoque une des mélodies les plus célèbres du compositeur Nino Rota.

Je voudrais aussi témoigner ma gratitude aux membres du jury pour avoir accepté d'examiner ce mémoire d'HDR. Un merci tout spécial va à mes doctorants et collaborateurs récents à l'UPJV, une grande source d'inspiration et d'énergie pour moi : Jordan Caracotte, Devesh Adlakha, Guillaume Hardouin, Daniel Rodrigues da Costa, Anass El Moudni, Djessy Rossi, Sarah Delmas, Guillaume Caron, Pascal Vasseur, Dominique Groux, Claude Pégard, Cédric Demonceaux, Julien Marzat, Julien Moras, Adlane Habed, Rémi Boutteau, Sébastien Kramm. Je tiens aussi à remercier les membres de ma famille en Italie pour leur confiance inébranlable et leurs conseils toujours pertinents. *Last but not the least*, je serai à jamais reconnaissant à Sorina pour son amour au cours des dernières années et son encouragement pendant la rédaction de ce manuscrit.

La première page de chaque chapitre de ce manuscrit est agrémentée par une phrase d'un scientifique ou d'un auteur de science-fiction. Je trouve l'interaction entre la science et la science-fiction très stimulante, surtout en ce qui concerne la robotique.

L'extrait qui suit a été écrit par l'un des plus grands physiciens du XX^{e} siècle et il décrit très bien le travail quotidien du scientifique et son rôle essentiel dans notre société. On a souvent tendance à l'oublier :

> "*I would now like to turn to a third value that science has. It is a little less direct, but not much. The scientist has a lot of experience with ignorance and doubt and uncertainty, and this experience is of very great importance, I think. When a scientist doesn't know the answer to a problem, he is ignorant. When he has a hunch as to what the result is, he is uncertain. And when he is pretty darn sure of what the result is going to be, he is still in some doubt. We have found it of paramount importance that in order to progress we must recognize our ignorance and leave room for doubt. Scientific knowledge is a body of statements of varying degrees of certainty – some most unsure, some nearly sure, but none* absolutely *certain. Now, we scientists are used to this, and we take it for granted that it is perfectly consistent to be unsure, that it is possible to live and* not *know. But I don't know whether everyone realizes this is true. Our freedom to doubt was born out of a struggle against authority in the early days of science. It was a very deep and strong struggle: permit us to question – to doubt – to not be sure. I think that it is important that we do not forget this struggle and thus perhaps lose what we have gained. Herein lies a responsibility to society.*"

"*The Value of Science*," public address of Richard P. Feynman at the National Academy of Sciences (autumn 1955); published in "*What Do You Care What Other People Think?*" (1988), page 245; republished in "*The Pleasure of Finding Things Out: The Best Short Works of Richard P. Feynman*" (1999) edited by Jeffrey Robbins.

Amiens
1er février 2024

Curriculum vitæ

Généralités

Nom : Morbidi
Prénom : Fabio
Date de naissance : 17 août 1981
Lieu de naissance : Siena, Italie
Nationalité : Italienne

Coordonnées

Université de Picardie Jules Verne
Laboratoire MIS (Modélisation, Information & Systèmes)
Équipe PR (Perception Robotique)
33, rue Saint-Leu, 80039 Amiens
E-mail : `fabio.morbidi@u-picardie.fr`
Page web : `http://home.mis.u-picardie.fr/~fabio`
Téléphone : +33 (0)3 22 82 59 02

Parcours universitaire et professionnel

• **Laurea L1, équivalente à un BAC+3 (octobre 2000 − octobre 2003)** :
Magna cum Laude en Ingénierie Informatique, spécialisation en Automatique et Systèmes pour l'Automatisation Industrielle, Dipartimento d'Ingegneria dell'Informazione, Università di Siena, Italie.

Titre du mémoire : *Visual Servoing for Mobile Robots based on Multiple Views Geometry*.
Date de la soutenance : 20 octobre 2003
Encadrant : Prof. Domenico Prattichizzo

• **Laurea L2, équivalente à un BAC+5 (octobre 2003 – octobre 2005)** :
Magna cum Laude en Ingénierie Informatique, spécialisation en Robotique et Automatique, Dipartimento d'Ingegneria dell'Informazione, Università di Siena, Italie.

Titre du mémoire : *Nonlinear Localizability and Panoramic Cameras: Applications to Robot Teams*.
Date de la soutenance : 26 septembre 2005
Encadrant : Prof. Domenico Prattichizzo
Co-encadrant : Prof. Antonio Vicino

Habilitation professionnelle à exercer les fonctions d'ingénieur (janvier 2006) :
Facoltà d'Ingegneria, Università di Firenze, Italie.

• **Doctorat (1er octobre 2005 – 1er octobre 2008)** :
Doctorat en *Automatique et Robotique* (cycle XXI), Dipartimento d'Ingegneria dell'Informazione, Università di Siena, Italie.

Titre de la thèse de doctorat : *Leader-Follower Formation Control and Visibility Maintenance of Nonholonomic Mobile Robots*.
Date de la soutenance : 27 mars 2009
Directeur de thèse : Prof. Domenico Prattichizzo
Co-directeur de thèse : Prof. Antonio Vicino

Composition du jury de thèse : Prof. K.M. Lynch (Northwestern University, USA), Prof. G. Antonelli (Università di Cassino, Italie), Prof. A. Garulli (Università di Siena, Italie).

Expérience de recherche à l'étranger pendant le doctorat :

- **Visiting researcher (1er octobre 2007 – 31 mars 2008)**
  Department of Mechanical Engineering and Center for Control Dynamical Systems and Computation (CCDC), University of California, Santa Barbara, USA.
  Supervisor : Prof. Francesco Bullo

Fonctions exercées :

- **Assegno di ricerca (1er janvier 2009 – 15 octobre 2009)**
  Dipartimento d'Ingegneria dell'Informazione, Università di Siena, Italie.
  Encadrant : Prof. Domenico Prattichizzo

- **Postdoctoral fellow (16 octobre 2009 – 26 septembre 2010)**
  Center for Robotics and Biosystems, Department of Mechanical Engineering,

Northwestern University, Evanston, Illinois, USA.
Supervisor : Prof. Kevin M. Lynch
Co-supervisor : Prof. Randy A. Freeman

- **Postdoctoral fellow (27 septembre 2010 – 24 août 2011)**
  Department of Computer Science and Engineering, University of Texas at Arlington, Arlington, Texas, USA.
  Supervisor : Dr. Gian-Luca Mariottini

- **Universitätsassistent mit doktorat (1er septembre 2011 – 31 octobre 2012)**
  Institute for Design and Control of Mechatronical Systems,
  Johannes Kepler University, Linz, Autriche.
  Supervisor : Prof. Luigi del Re

- **Post-doctorat (1er novembre 2012 – 31 août 2014)**
  Équipe projet NeCS, Inria Grenoble - Rhône-Alpes,
  Montbonnot-Saint-Martin, France.
  Encadrant : Dr. Carlos Canudas-de-Wit

Fonction actuelle

Depuis le **1er septembre 2014**, je suis **Maître de Conférences** (classe normale) en 61ème section, "Génie Informatique, Automatique et Traitement du Signal", à l'Université de Picardie Jules Verne (UPJV) à Amiens. Je suis rattaché au laboratoire MIS (Unité de Recherche 4290) et j'effectue mon activité d'enseignement au département EEA (Électronique, Électrotechnique et Automatique) de l'UFR des Sciences.

Chapter 1

Activités scientifiques et pédagogiques

My colleague Watson is limited in its thinking to rather narrow confines, but possesses the utmost tenacity.

"A Wild Sheep Chase" – Haruki Murakami, 1982

Résumé

Ce premier chapitre présente l'ensemble de mes activités activités scientifiques, pédagogiques et administratives depuis ma soutenance de thèse à l'Università di Siena, Italie, en mars 2009.

1.1 Recherche

1.1.1 Résumé des activités de recherche

Mes activités de recherche au cours des quinze dernières années (2009 - aujourd'hui), se sont concentrées dans les domaines de la *Robotique* et de l'*Automatique*. En particulier, mon travail s'est articulé autour des deux axes suivants :

1. **Systèmes multi-robots**
   (a) Contrôle de formations de robots non holonômes
   (b) Analyse et commande de systèmes robotiques en réseaux
2. **Vision omnidirectionnelle**
   (a) Caméras catadioptriques et twin-fisheye pour la navigation de robots.

Une description détaillée de ces deux axes est fournie dans la suite. Il est à noter que les renvois mentionnées dans cette section concernent exclusivement mes publications, dont la liste complète est donnée dans la Sect. 1.1.2. Un bilan des travaux publiés après ma thèse dans chaque axe et des encadrements, est présenté dans les Tableaux 1.1 et 1.2.

Il est enfin important de souligner que dans ce mémoire d'HDR, pour des raisons d'espace et de cohérence thématique, on a fait le choix délibéré de privilégier le 1er axe ("*Systèmes multi-robots*") qui sera développé davantage dans les prochains chapitres.

Systèmes multi-robots

"*E pluribus unum*", locution latine signifiant "un seul à partir de plusieurs" (Moretum, Virgile), ou plus librement "l'union fait la force", est la devise qui apparaît sur le grand sceau des États-Unis, mais elle peut être aussi considérée comme une des lignes directrices de ma carrière de chercheur. En fait, à partir de la fin de mon Master à l'Università di Siena en 2005, je me suis intéressé à la commande de systèmes multi-robots. Cela a coïncidé avec une explosion d'activité autour des systèmes multi-agents qui a commencée au début des années 2000 en Théorie des Systèmes et s'est propagée rapidement, en moins d'une décennie, dans de nombreux autres domaines, comme la Robotique, les Télécommunications, le Traitement du Signal, l'Intelligence Artificielle et la Science des Réseaux. Les raisons de ce succès sont multiples et variées, et elles incluent une plus grande disponibilité des technologies de communication sans fil et de processeurs à faible coût, l'intérêt croissant porté au calcul parallèle et à l'informatique embarquée, la miniaturisation des capteurs et la définition de tâches de plus en plus complexes nécessitant l'action conjointe de groupes de robots pour être menées à bien.

(a) Pendant mes études de doctorat (2005 - 2009), j'ai travaillé sur le *problème du contrôle de formation*, qui a suscité un vif intérêt dans la littérature des systèmes multi-agents, à cause de son vaste potentiel d'application. L'idée de base est assez simple et consiste à contrôler la position et l'orientation relative des robots dans un groupe, tout en permettant au groupe de se déplacer d'un seul bloc. En particulier, mes travaux de recherche ont porté sur formations de robots *non holonômes* qui suivent un paradigme de type *leader-follower*. Dans ce cadre, un robot (le leader) guide les suiveurs (les followers), et tous les véhicules ont des contraintes cinématiques. Ces contraintes entraînent une réduction de l'ensemble des déplacements instantanés que les robots peuvent effectuer. L'effet des contraintes de vitesse sur la stabilité d'une formation de deux robots, a été étudié dans [C1, C3, C5, J2] d'un point de vue purement géométrique. Ces résultats ont été étendus à des formations leader-follower organisées selon une *structure hiérarchique* dans [C7, J4] (une collaboration avec L. Consolini, Università di Parma, Italie), et à des systèmes constitués d'un tractor-trailer et d'un robot tricycle, dans [BC1]. Enfin, sur le même sujet, dans [C15, C18, J9], la notion inédite de *dynamique interne* d'une formation de robots a été introduite pour décrire la "rigidité" des structures hiérarchiques étudiées en [C7, J4].

Une autre série de travaux connexes, [C2, C8, J5, J6], a porté sur les formations leader-follower de robots de type unicycle, equipés de *caméras panoramiques non calibrées*. Le fait que ces caméras sont les seuls capteurs embarqués, rend le problème de contrôle de formation particulièrement difficile car les robots ne disposent pas directement des mesures de distance. En [C2, C8, J5, J6], nous avons pu montrer qu'il est possible d'estimer la distance entre le leader et les followers (par le biais du filtre de Kalman

étendu ou d'autres observateurs non-linéaires), en utilisant uniquement les mesures angulaires relatives entre les robots fournies par les caméras panoramiques. Notamment, une analyse de l'observabilité du système dynamique non-linéaire qui décrit la formation, a permis d'identifier analytiquement les trajectoires pour lesquelles on a une meilleure estimation de la distance entre le leader et les followers.

Le problème du *maintien de la visibilité* a été largement étudié en théorie des jeux (problème de pursuit-evasion) et en géométrie algorithmique (problème de la galerie d'art et de l'illumination), il a aussi des liens forts avec le contrôle de formation. Dans [C9, J8], en collaboration avec le Prof. F. Bullo (University of California Santa Barbara, USA), une nouvelle stratégie a été conçue pour le contrôle de la visibilité de robots leader-follower de type Dubins. L'utilisation de la notion géométrique d'invariance contrôlée constitue un apport original de ces deux publications.

D'autres travaux menés pendant mon post-doc à l'Università di Siena, Italie [C27, J15], ont abordé le problème de coordination du mouvement d'une *équipe mixte* constituée d'un leader humain et d'une cohorte de robots mobiles. Chaque robot est équipé d'une caméra RGB-D et le leader porte un bracelet vibro-tactile doté de connexion Bluetooth, grâce auquel ses mouvements peuvent être contrôlés à distance. Pour valider la loi de commande proposée, la première à notre connaissance à faire intervenir un *retour haptique*, une campagne d'expérimentation a été réalisée avec un opérateur humain et deux robots Pioneer. Enfin, une nouvelle *caractérisation de la mobilité* de formations de robots de type unicycle avec contraintes de distance/orientation relative, a été présentée récemment dans [C33]. La fameuse condition de "roulement sans glissement" d'un roue fixe standard, joue un rôle crucial dans la classification hiérarchique par "type" introduite dans ce travail.

(b) Durant mon post-doctorat à la Northwestern University, USA (2009 - 2010), mes intérêts de recherche se sont progressivement réorientés vers l'*estimation et la commande distribuées de systèmes dynamiques en réseaux*. Au cours du post-doc au sein du "Center for Robotics and Biosystems" (N×R), j'ai eu l'occasion d'approfondir mes connaissances et d'interagir avec quelques grands experts du domaine (le Prof. K.M. Lynch et le Prof. R.A. Freeman). Cette expérience m'a permis de gagner en maturité et elle a eu un impact profond et durable sur ma carrière. Mon premier travail dans ce nouveau domaine, [C19], propose une stratégie basée sur l'estimation et la commande distribuées pour une volée d'UAVs (Unmanned Aerial Vehicles). Les véhicules sont modélisés comme unicycles avec une vitesse de translation positive et ils doivent se reconfigurer dans l'espace pour surveiller l'apparition d'événements d'intérêt dans un environnement 2D. Une première contribution originale de [C19] est l'utilisation de moments géométriques pour décrire la forme de la cohorte de robots et les événements à surveiller (par exemple, un déversement de pétrole dérivant dans l'océan). La deuxième contribution est

l'emploi d'estimateurs proportionnels-intégraux basés sur le consensus dynamique en moyenne (ou "dynamic average consensus estimators") pour la reconstruction locale des moments désirés.

Plus récemment (avec A. Kibangou, Université Grenoble Alpes), un algorithme des moindres carrés basé sur la même famille d'estimateurs PI distribués que [C19] a été utilisé pour *reconstruire la topologie d'un réseau inconnu*. Le nouvel algorithme, qui a fait l'objet de la publication [J14], fait appel à des séries temporelles entachées de bruit au niveau de chaque nœud du réseau, et contrairement à ce qui est fait dans la littérature, il est totalement décentralisé.

Inspirés par une série de travaux antérieurs du groupe de recherche de la Northwestern University, les articles [C21, J12] proposent une solution originale au *problème du suivi actif d'une cible* par une équipe d'UAVs modelisés comme double intégrateurs. Dans [C20] (avec G.L. Mariottini et C. Ray, University of Texas at Arlington, USA), la théorie introduite dans [J12] a été revisitée pour une application médicale (analyse automatique de la démarche de personnes à mobilité réduite) et adaptée à une cohorte hétérogène d'UAVs et de robots terrestres équipés de télémetres 2D. Enfin, une formulation à temps discret du problème traité dans [J12] est présentée et validée expérimentalement à l'aide d'un AR.Drone de Parrot, dans [C25].

Dans la littérature des systèmes multi-agents, la *contrainte de connexité* est bien connue car elle est essentielle pour garantir l'exécution correcte de nombreuses tâches distribuées (par exemple, dans le problème de l'accord ou "agreement problem", les états des agents convergent vers la même valeur uniquement si le réseau de communication reste connexe en permanence). Plusieurs solutions existent dans la littérature pour garantir le maintien de la connexité dans une cohorte de robots, comme par exemple les champs de potentiel, les fonctions de navigation ou les approches basées sur l'optimisation géométrique. Les articles [C12, C17] proposent des solutions purement géométriques au problème de la connexité pour des robots de type intégrateur et unicycle, en poursuite cyclique. Plus récemment, des algorithmes innovants ont été mis au point dans [C23] pour contrôler la connexité algébrique et pour subdiviser automatiquement un réseau de robots en deux groupes, grâce au partitionnement spectral (ou "spectral clustering").

Pendant mon séjour à la Johannes Kepler University, Autriche (2011 - 2012), mon activité pédagogique m'a permis d'approfondir mes connaissances en commande optimale, et j'ai eu l'opportunité de collaborer avec le Prof. P. Colaneri (Politecnico di Milano, Italie) au sujet du *contrôle optimal décentralisé* de pelotons de voitures [C24]. La source principale d'originalité de la méthode proposée, réside dans la mannière d'incorporer la contrainte de "string stability" dans un problème de commande linéaire quadratique (LQ) classique. Mon intérêt pour la théorie du consensus n'a pas diminué pendant le séjour autrichien, et à partir de 2012 j'ai travaillé sur *deux variantes de la ma-*

trice laplacienne d'un graphe : le laplacien déformé [J13, C22] et le laplacien paramétrique (ou "Laplacian pencil") [C28]. J'estime que [J13] est un de mes travaux les plus originaux et c'est remarquable que certaines des idées présentées dans cet article aient stimulé la recherche dans des domaines assez éloignés de la robotique coopérative (par exemple, en apprentissage semi-supervisé ou dans la conception de nouvelles mesures de centralité pour des réseaux non orientés et orientés).

Pendant la période passée à l'Université de Picardie Jules Verne comme Maître de Conférences (2014 - aujourd'hui), j'ai poursuivi mes activités de recherche autour des systèmes robotiques en réseaux. Cette démarche a abouti aux travaux [C35], [J21] et [C36, C38]. Le protocole de consensus en temps discret a été étudié en [C35], et une nouvelle approche basée sur la *projection sur un sous-espace vectoriel* a été proposée pour incorporer facilement les contraintes sur l'état du système (par exemple, contraintes sur la perception, la communication ou sur l'évitement d'obstacle), sans modifier la valeur de consensus. Dans le même esprit que [J13], une classe de fonctions de la matrice laplacienne d'un graphe qui préservent ses propriétés (semi-définie positivité, somme des lignes égale à zéro et éléments hors-diagonale non positifs) a été étudiée dans [J21]. Ces fonctions de la matrice laplacienne ont trouvé leur emploi dans la commande distribuée de formations de robots de type intégrateur.

Enfin, un nouvel algorithme "Next-Best-View" basé surface pour la planification de trajectoire, a été introduit dans [C36] pour la *reconstruction d'un environnement 3D de grande envergure* avec un UAV équipé d'une caméra de profondeur ou d'un banc stéréo (un travail conjoint avec J. Marzat et J. Moras, ONERA DTIS). Le planificateur TSGA ("TSP-Greedy Allocation") proposé, a été récemment adapté en [C38] à un cohorte d'UAVs travaillant de concert, pour rendre la reconstruction 3D plus rapide. Une version *complètement décentralisée* de l'algorithme est actuellement à l'étude et une campagne d'expérimentations avec un groupe de robots terrestres (WiFibots) est en cours. Ce travail s'inscrit dans la thèse de G. Hardouin : un article de revue qui présente l'ensemble des nos résultats a été publié en août 2023 [J24]. Le doctorant a été cofinancé par la région Hauts-de-France et l'ONERA DTIS dans le cadre du projet ScanBot. L'objectif final de ce projet sera d'appliquer les outils de planification développés au patrimoine monumental numérique (cf. le programme e-Cathédrale de l'équipe PR).

Il est utile de préciser ici que l'activité de recherche autour des systèmes multi-robots menée à l'UPJV depuis 2014, n'est pas isolée mais elle fait partie intégrante de mon projet professionnel. En fait, elle est cohérente avec mon activité pédagogique : j'ai dispensé deux cours de Master sur la théorie des systèmes multi-agents au département EEA en 2018-2021, "Systèmes Robotiques Hétérogènes et Coopératifs" et "Surveillance Distribuée de Systèmes Multi-Agents", dérivés du cours de Master intitulé "Distributed and Cooperative Control for Multi-agent Systems" que j'ai donné en 2012 à la Johannes Kepler University. Par ailleurs, un parcours centré sur les systèmes multi-agents et la

Publications	[J24], [J21], [J15], [J14], [J13], [J12], [J9], [C38], [C35], [C33], [C28], [C27], [C25], [C24], [C23], [C22], [C21], [C20], [C19], [C18], [C17], [C15]
Encadrements	1 doctorant (G. Hardouin)

Table 1.1: *Bilan des publications après ma thèse dans l'axe "*Systèmes Multi-Robots*" et encadrements.*

robotique collaborative, appelé "SDD : Systèmes Distribués et Dynamiques" (Master mention Informatique) a ouvert dans l'année universitaire 2022-2023 à l'UPJV et une action inter-équipe (équipe PR et équipe SDMA du laboratoire MIS), appelée "Cohortes de Robots", existe depuis 2007. Enfin, je travaille régulièrement comme Éditeur Associé avec un profile "Multi-Robot Systems", pour les IEEE Transactions on Robotics, pour le Conference Review Board de IROS et le Conference Editorial Board de ICRA tous les 6 mois, et je suis un reviewer spécialisé dans ce même domaine pour plusieurs revues internationales (cf. Sect. 1.1.4).

Vision omnidirectionnelle

Les caméras sont largement utilisées aujourd'hui en robotique mobile, car elles sont petites, légères, peu coûteuses et elles fournissent des informations sur le milieu environnant, bien plus riches que celles d'autres capteurs extéroceptifs. Cependant, leur champ de vue limité est problématique pour la navigation d'un robot dans un environnement inconnu.

(a) Les *caméras catadioptriques* combinent des éléments qui réfractent (lentilles) et réfléchissent (miroirs convexes) la lumière, et elles ont un champ de vue qui est bien plus large (supérieur à 180°) que celui d'une caméra perspective traditionnelle. Cette caractéristique rend ces capteurs fort séduisants pour de nombreuses applications robotiques, comprenant la localisation, l'estimation d'orientation 3D et l'asservissement visuel, entre autres. Un algorithme robuste et rapide qui s'appuie sur des nouvelles propriétés géométriques des images acquises par une caméra catadioptrique avec miroir parabolique (caméra para-catadioptrique), est proposé dans [BC2, J11] pour estimer l'orientation d'un robot mobile à roues. Le compas visuel non calibré a été modifié dans [C26], afin d'estimer le cap d'un robot aérien avec une caméra para-catadioptrique embarquée.

Dans la même veine, un compas visuel omnidirectionnel basé sur la corrélation de phase dans le domaine de Fourier 2D, a été présenté dans [J18]. Le compas est direct

(à savoir, il traite la totalité de l'image panoramique sans aucune extraction de points d'intérêt), précis et robuste au bruit dans l'image, et il n'a besoin que d'une connaissance partielle des paramètres internes de la caméra. Des tests en conditions réelles, effectués avec une caméra hyper-catadioptrique (une caméra dotée d'un miroir hyperbolique) montée sur l'effecteur d'un manipulateur Stäubli et sur un robot Pioneer, ont démontré que la nouvelle approche affiche des performances très satisfaisantes.

Les *caméras twin-fisheye* sont des capteurs de vision émergents qui comprennent deux lentilles fisheye montées dos à dos sur le même support, et une série de prismes qui dirigent les faisceaux lumineux vers deux éléments photosensibles. Plusieurs produits grand public sont disponibles aujourd'hui sur le marché, comme par exemple la Gear 360 de Samsung, la 360 CAM de LG, la Theta de Ricoh et la ONE X2 de Insta360. Dans [C31, CN1], un nouveau gyroscope visuel sphérique a été introduit pour la navigation de robots mobiles. En transformant l'intensité lumineuse en mélanges de potentiels photométriques, une nouvelle mesure de similarité de l'image a été proposée. Elle peut être facilement intégrée dans une procédure classique d'optimisation non-linéaire aux moindres carrées et offre un domaine de convergence élargi. La méthode fournit une estimation de l'orientation 3D précise et robuste, et elle est simple d'utilisation car elle comporte un seul paramètre de réglage, la largeur des potentiels photométriques (fonctions gaussiennes dans [C31]), qui contrôle le pouvoir d'attraction de chaque pixel. Le gyroscope a été testé avec des séquences d'images générées par une caméra twin-fisheye montée sur l'effectuer d'un manipulateur et sur un UAV à voilure fixe (Disco FPV de Parrot). Récemment, les images sphériques capturées par la même caméra (la Theta S de Ricoh), ont été utilisées dans un nouveau système d'aide à la conduite

Publications	[B1], [J23], [J22], [J20], [J19], [J18], [J11], [C40], [C37], [C31], [C26], [CN1], [O6]
Encadrements	• 1 post-doc (H.-E. Benseddik) • 2 doctorants (J. Caracotte, D. Rodrigues da Costa) • 1 ingénieur de recherche (S. Delmas) • 3 étudiants de Master (J. Albrand, M. Jeanne-Rose, M. Selmi)

Table 1.2: *Bilan des publications après ma thèse dans l'axe "*Vision Omnidirectionnelle*" et encadrements.*

destiné aux utilisateurs de fauteuils roulants électriques [C40],[J22],[J23]. La contribution de l'ingénieur de recherche S. Delmas, que j'ai encadré, a été fondamentale pour le développement logiciel et pour la validation du dispositif sur fauteuil roulant. Dans le cadre du même projet (Interreg VA ADAPT, post-doc de H.-E. Benseddik), une vaste base de données d'images catadioptriques et twin-fisheye avec une vérité terrain précise, nommée PanoraMIS [J19], a été créée et mise à disposition sur Internet pour la communauté de vision pour la robotique.

Pendant la thèse de D. Rodrigues da Costa, la première *caméra événementielle omnidirectionnelle* a été conçue [O6], et un gyroscope visuel basé sur l'observation de droites dans un environnement de type Manhattan a été mis à point et il est actuellement en cours de validation sur un robot Stäubli TX-60, qui nous fournit une vérité-terrain très précise, au département EEA de l'UPJV.

D'autre part, dans la thèse de J. Caracotte, nous avons pu mettre à profit nos connaissances au sujet des caméras catadioptriques et twin-fisheye, en *vision par ordinateur*. En fait, dans [J20], nous avons introduit une théorie générale de la stéréophotométrie pour les caméras à projection centrale, qui a été adaptée aux caméras twin-fisheye dans [C37].

Enfin, sur le thème de la vision omnidirectionnelle, j'ai coordonné avec un collègue du laboratoire MIS (Prof. P. Vasseur) un ouvrage, [B1], qui sera publié par ISTE-Wiley en décembre 2023.

1.1.2 Production scientifique

Le Tableau 1.3 fournit une synthèse de la production scientifique pendant (2005-2009) et après ma thèse de doctorat (2009 - aujourd'hui).

	B	J	BC	C	JN	CN	**Total**
Pendant la thèse	0	6	2	13	0	0	**21**
Après la thèse	1	18	2	29	2	2	**54**

Table 1.3: Nombre de publications par type *: B = ouvrage, J = revue internationale, BC = chapitre d'ouvrage, C = conférence internationale, JN = revue nationale, CN = conférence nationale.*

Ouvrages

B1. P. Vasseur, F. Morbidi (editors), "*Omnidirectional Vision: From Theory to Applications*", 10 authors, ISTE - Wiley, Series: SCIENCES - Sensors and Image Processing, December 2023, `www.iste.co.uk/book.php?id=2075`

Revues internationales à comité de lecture

J24. G. Hardouin, J. Moras, F. Morbidi, J. Marzat, E. Mouaddib, "*A Multi-Robot System for 3D Surface Reconstruction with Centralized and Distributed Architectures*", IEEE Transactions on Robotics, vol. 39, n. 4, pp. 2623-2638, August 2023. Journal Impact Factor (IF), November 2023: 7.8

J23. S. Delmas, F. Morbidi, G. Caron, M. Babel, F. Pasteau, "*SpheriCol: A Driving Assistant for Power Wheelchairs based on Spherical Vision*", IEEE Transactions on Medical Robotics and Bionics, vol. 5, n. 2, pp. 387-400, May 2023. IF: 3.7

J22. F. Morbidi, L. Devigne, C.S. Teodorescu, B. Fraudet, É. Leblong, T. Carlson, M. Babel, G. Caron, S. Delmas, F. Pasteau, G. Vailland, V. Gouranton, S. Guégan, R. Le Breton, N. Ragot, "*Assistive Robotic Technologies for Next-Generation Smart Wheelchairs: Codesign and Modularity to Improve Users' Quality of Life*", IEEE Robotics & Automation Magazine, vol. 30, n. 1, pp. 24-35, March 2023 (also accepted for presentation at the IEEE International Conference on Robotics and Automation, London, May 29 - June 2, 2023). IF: 5.7

J21. F. Morbidi, "*Functions of the Laplacian matrix with application to distributed formation control*", IEEE Transactions on Control of Network Systems, vol. 9, n. 3, pp. 1459-1467, September 2022. IF: 4.2

J20. J. Caracotte, F. Morbidi, E. Mouaddib, "*Photometric stereo with central panoramic cameras*", Computer Vision and Image Understanding, vol. 201, article 103080, December 2020. IF : 4.5

J19. H.-E. Benseddik, F. Morbidi, G. Caron, "*PanoraMIS: An Ultra-wide Field of View Image Dataset for Vision-based Robot-Motion Estimation*", The International Journal of Robotics Research, vol. 39, n. 9, pp. 1037-1051, August 2020. IF: 9.2

J18. F. Morbidi, G. Caron, "*Phase Correlation for Dense Visual Compass from Omnidirectional Camera-Robot Images*", IEEE Robotics and Automation Letters, vol. 2, n. 2, pp. 688-695, April 2017 (also accepted for presentation at the IEEE International Conference on Robotics and Automation, Singapore, May 29 - June 3, 2017). IF: 5.2

J17. A.N. Staranowicz, G.R. Brown, F. Morbidi, G.L. Mariottini, "*Practical and accurate calibration of RGB-D cameras using spheres*", Computer Vision and Image Understanding, vol. 137, pp. 102-114, August 2015. IF: 4.5

J16. C. Canudas de Wit, F. Morbidi, L. León Ojeda, A.Y. Kibangou, I. Bellicot, P. Bellemain, "*Grenoble Traffic Lab: An experimental platform for advanced traffic monitoring*

and forecasting", IEEE Control Systems Magazine, vol. 35, n. 3, pp. 23-39, June 2015. IF: 5.7

J15. S. Scheggi, F. Morbidi, D. Prattichizzo, "*Human-robot formation control via visual and vibrotactile haptic feedback*", IEEE Transactions on Haptics, vol. 7, n. 4, pp. 499-511, December 2014. IF: 2.9

J14. F. Morbidi, A.Y. Kibangou, "*A Distributed Solution to the Network Reconstruction Problem*", Systems & Control Letters, vol. 70, pp. 85-91, August 2014. IF: 2.6

J13. F. Morbidi, "*The Deformed Consensus Protocol*", Automatica, vol. 49, n. 10, pp. 3049-3055, October 2013. IF: 6.4

J12. F. Morbidi, G.L. Mariottini, "*Active Target Tracking and Cooperative Localization for Teams of Aerial Vehicles*", IEEE Transactions on Control Systems Technology, vol. 21, n. 5, pp. 1694-1707, September 2013. IF: 4.8

J11. G.L. Mariottini, S. Scheggi, F. Morbidi, D. Prattichizzo, "*An Accurate and Robust Visual-Compass Algorithm for Robot-mounted Omnidirectional Cameras*", Robotics and Autonomous Systems, vol. 60, n. 9, pp. 1179-1190, September 2012. IF: 4.3

J10. G.L. Mariottini, S. Scheggi, F. Morbidi, D. Prattichizzo, "*Planar mirrors for image-based robot localization and 3-D reconstruction*", Mechatronics, Special Issue on Visual Servoing, vol. 22, n. 4, pp. 398-409, June 2012. IF: 3.3

J9. L. Consolini, F. Morbidi, D. Prattichizzo, M. Tosques, "*On a class of hierarchical formations of unicycles and their internal dynamics*", IEEE Transactions on Automatic Control, vol. 57, n. 4, pp. 847-859, April 2012. IF: 6.8

J8. F. Morbidi, F. Bullo, D. Prattichizzo, "*Visibility maintenance via controlled invariance for leader-follower vehicle formations*", Automatica, vol. 47, n. 5, pp. 1060-1067, May 2011. IF: 6.4

J7. F. Chinello, S. Scheggi, F. Morbidi, D. Prattichizzo, "*The KUKA Control Toolbox: motion control of KUKA robot manipulators with MATLAB*", IEEE Robotics and Automation Magazine, vol. 18, n. 4, pp. 69-79, December 2011. IF: 3.591

J6. F. Morbidi, G.L. Mariottini, D. Prattichizzo, "*Observer design via Immersion and Invariance for vision-based leader-follower formation control*" Automatica, vol. 46, n. 1, pp. 148-154, January 2010. IF: 6.4

J5. G.L. Mariottini, F. Morbidi, D. Prattichizzo, N. Vander Valk, N. Michael, G.J. Pappas, K. Daniilidis, "*Vision-based Localization for Leader-Follower Formation Control*", IEEE Transactions on Robotics, vol. 25, n. 6, pp. 1431-1438, December 2009. IF: 7.8

J4. L. Consolini, F. Morbidi, D. Prattichizzo, M. Tosques, "*Stabilization of a hierarchical formation of unicycle robots with input constraints*", IEEE Transactions on Robotics, vol. 25, n. 5, pp. 1176-1184, October 2009. IF: 7.8

J3. F. Morbidi, A. Garulli, D. Prattichizzo, C. Rizzo, S. Rossi,"*Application of Kalman filter to remove TMS-induced artifacts from EEG recordings*", IEEE Transactions on Control Systems Technology, vol. 16, n. 6, pp. 1360-1366, November 2008. IF: 4.8

J2. L. Consolini, F. Morbidi, D. Prattichizzo, M. Tosques, "*Leader-Follower Formation Control of Nonholonomic Mobile Robots with Input Constraints*", Automatica, vol. 44, n. 5, pp. 1343-1349, May 2008. IF: 6.4

J1. F. Morbidi, A. Garulli, D. Prattichizzo, C. Rizzo, P. Manganotti, S. Rossi, "*Off-line removal of TMS-induced artifacts on human electroencephalography by Kalman filter*", Journal of Neuroscience Methods, vol. 162, pp. 293-302, May 2007. IF: 3

Chapitres d'ouvrage

BC4. A. Staranowicz, G.R. Brown, F. Morbidi, G.L. Mariottini, "*Easy-to-Use and Accurate Calibration of RGB-D Cameras from Spheres*", in Proc. 6th Pacific-Rim Symposium on Image and Video Technology, R. Klette, M. Rivera, S. Satoh, Eds., Lecture Notes in Computer Science, Springer, Volume 8333, pp. 265-278, 2014.

BC3. G.L. Mariottini, S. Scheggi, F. Morbidi, D. Prattichizzo, "*Catadioptric Stereo with Planar Mirrors: Multiple-View Geometry and Camera Localization*", in Visual Servoing via Advanced Numerical Methods, G. Chesi and K. Hashimoto, Eds., Lecture Notes in Computer Science, Springer, pp. 3-22, 2010.

BC2. G.L. Mariottini, S. Scheggi, F. Morbidi, D. Prattichizzo,"*A Robust Uncalibrated Visual Compass Algorithm from Paracatadioptric Line Images*" in Proc. 1st Workshop on Omnidirectional Robot Vision, E. Menegatti and T. Pajdla, Eds., Lecture Notes in Computer Science, Springer, pp. 242-255, November 2008.

BC1. F. Morbidi, D. Prattichizzo,"*Sliding mode formation tracking control of a tractor and trailer-car system*", in Proc. Robotics: Science and Systems III, W. Burgard, O. Brock and C. Stachniss Eds., MIT press, Cambridge MA, pp. 113-120, March 2008.

Conférences internationales avec actes et comité de lecture

C42. A. El Moudni, F. Morbidi, S. Kramm, R. Boutteau, "*An Event-based Stereo 3D Mapping and Tracking Pipeline for Autonomous Vehicles*", in Proc. 26th IEEE International

Conference on Intelligent Transportation Systems, Workshop "Beyond Traditional Sensing for Intelligent Transportation", Bilbao, Spain, September 24-28, 2023.

C41. F. Morbidi, D. Pisarski, "*Practical and Accurate Generation of Energy-Optimal Trajectories for a Planar Quadrotor*", in Proc. IEEE International Conference on Robotics and Automation, Xi'an, China, pp. 355-361, May 30 - June 5, 2021.

C40. S. Delmas, F. Morbidi, G. Caron, J. Albrand, M. Jeanne-Rose, L. Devigne, M. Babel, "*SpheriCol: A Driving Assistance System for Power Wheelchairs Based on Spherical Vision and Range Measurements*", in Proc. IEEE/SICE International Symposium on System Integration, Iwaki, Japan, pp. 505-510, January 11-14, 2021.

C39. D. Adlakha, A. Habed, F. Morbidi, C. Demonceaux, M. de Mathelin, "*Stratified Autocalibration of Cameras with Euclidean Image Plane*", in Proc. 31st British Machine Vision Conference, Manchester, UK, paper 26, September 7-11, 2020.

C38. G. Hardouin, J. Moras, F. Morbidi, J. Marzat, E. Mouaddib, "*Next-Best-View planning for surface reconstruction of large-scale 3D environments with multiple UAVs*", in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, USA, October 25-29, 2020.

C37. J. Caracotte, F. Morbidi, E. Mouaddib, "*Photometric Stereo with Twin-Fisheye Cameras*", in Proc. 25th International Conference on Pattern Recognition, Milan, Italy, pp. 5270-5277, January 10-15, 2021.

C36. G. Hardouin, F. Morbidi, J. Moras, J. Marzat, E. Mouaddib, "*Surface-driven Next-Best-View planning for exploration of large-scale 3D environments*", in Proc. 21st IFAC World Congress, Berlin, Germany, pp. 15501-15507, 12-17 July 2020.

C35. F. Morbidi, "*Subspace Projectors for State-Constrained Multi-Robot Consensus*, in Proc. IEEE International Conference on Robotics and Automation, Paris, France, pp. 7705-7711, May 31 - June 4, 2020.

C34. D. Adlakha, A. Habed, F. Morbidi, C. Demonceaux, M. de Mathelin, "*QUARCH: A New Quasi-Affine Reconstruction Stratum from Vague Relative Camera Orientation Knowledge*, in Proc. 17th International Conference on Computer Vision, Seoul, Republic of Korea, pp. 1082-1090, October 29 - November 1, 2019.

C33. F. Morbidi, E. Bretagne, "*A New Characterization of Mobility for Distance-Bearing Formations of Unicycle Robots*, in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, Madrid, Spain, pp. 4833-4839, October 1-5, 2018.

C32. F. Morbidi, D. Bicego, M. Ryll, A. Franchi, "*Energy-Efficient Trajectory Generation for a Hexarotor with Dual-Tilting Propellers*", in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, Madrid, Spain, pp. 6226-6232, October 1-5, 2018.

C31. G. Caron, F. Morbidi, "*Spherical Visual Gyroscope for Autonomous Robots using the Mixture of Photometric Potentials*", in Proc. IEEE International Conference on Robotics and Automation, Brisbane, Australia, pp. 820-827, May 21-25, 2018.

C30. F. Morbidi, R. Cano, D. Lara, "*Minimum-Energy Path Generation for a Quadrotor UAV*", in Proc. IEEE International Conference on Robotics and Automation, Stockholm, Sweden, pp. 1492-1498, May 16-21, 2016.

C29. F. Morbidi, L. León Ojeda, C. Canudas de Wit, I. Bellicot, "*Robust mode selection for highway traffic density estimation*", in Proc. European Control Conference, invited session "Traffic Control and Estimation: New Trends and Opportunities", Strasbourg, France, pp. 2576-2580, June 24-27, 2014.

C28. F. Morbidi, "*The Second-order Parametric Consensus Protocol*", in Proc. European Control Conference, Strasbourg, France, pp. 202-207, June 24-27, 2014.

C27. S. Scheggi, M. Aggravi, F. Morbidi, D. Prattichizzo, "*Cooperative human-robot haptic navigation*", in Proc. IEEE International Conference on Robotics and Automation, Hong Kong, China, pp. 2693-2698, May 31 - June 5, 2014.

C26. S. Scheggi, F. Morbidi, D. Prattichizzo, "*Uncalibrated Visual Compass from Omnidirectional Line Images with Application to Attitude MAV Estimation*", in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, pp. 1602-1607, November 3-7, 2013.

C25. U. Gürcüoğlu, G.A. Puerto-Souza, F. Morbidi, G.L. Mariottini, "*Hierarchical Control of a Team of Quadrotors for Cooperative Active Target Tracking*", in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, pp. 5730-5735, November 3-7, 2013.

C24. F. Morbidi, P. Colaneri, T. Stanger, "*Decentralized optimal control of a vehicle platoon with guaranteed string stability*", in Proc. European Control Conference, Zurich, Switzerland, pp. 3494-3499, July 17-19, 2013.

C23. F. Morbidi, "*On the control of the algebraic connectivity and clustering of a network of mobile agents*", in Proc. European Control Conference, Zurich, Switzerland, pp. 2801-2806, July 17-19, 2013.

C22. F. Morbidi, "*On the Properties of the Deformed Consensus Protocol*", in Proc. 51st IEEE Conference on Decision and Control, Maui, Hawaii, USA, pp. 812-817, December 10-13, 2012.

C21. F. Morbidi, G.L. Mariottini, "*On Active Target Tracking and Cooperative Localization for Multiple Aerial Vehicles*", in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, invited session on "Aerial Robotics", San Francisco, CA, USA, pp. 2229-2234, September 25-30, 2011.

C20. F. Morbidi, C. Ray, G.L. Mariottini, "*Cooperative active target tracking for heterogeneous robots with application to gait monitoring*", in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, CA, USA, pp. 3608-3613, September 25-30, 2011.

C19. F. Morbidi, R.A. Freeman, K.M. Lynch, "*Estimation and Control of UAV Swarms for Distributed Monitoring Tasks*", in Proc. American Control Conference, San Francisco, CA, USA, pp. 1069-1075, June 29-July 1, 2011.

C18. L. Consolini, F. Morbidi, D. Prattichizzo, M. Tosques, "*On internal dynamics of formations of unicycle robots*", in Proc. 49th IEEE Conference on Decision and Control, pp. 5462–5467, Atlanta, GA, USA, December 15-17, 2010.

C17. F. Morbidi, A. Giannitrapani, D. Prattichizzo, "*Maintaining connectivity among multiple agents in cyclic pursuit: a geometric approach*", in Proc. 49th IEEE Conference on Decision and Control, pp. 7461–7466, Atlanta, GA, USA, December 15-17, 2010.

C16. G. Marro, F. Morbidi, L. Ntogramatzidis, D. Prattichizzo, "*Geometric Control Theory for Linear Systems: a Tutorial*", in Proc. 19th International Symposium on Mathematical Theory of Networks and Systems, pp. 1579–1590, Budapest, Hungary, July 5-9, 2010.

C15. L. Consolini, F. Morbidi, D. Prattichizzo, M. Tosques, "*Non-rigid Formations of Nonholonomic Robots*", in Proc. IEEE International Conference on Robotics and Automation, pp. 4976-4981, Anchorage, AK, USA, May 3-8, 2010.

C14. F. Chinello, S. Scheggi, F. Morbidi, D. Prattichizzo, "*KCT: a MATLAB toolbox for motion control of KUKA robot manipulators*", in Proc. IEEE International Conference on Robotics and Automation, pp. 4603-4608, Anchorage, AK, USA, May 3-8, 2010.

C13. G. Marro, F. Morbidi, D. Prattichizzo, "*A Geometric Solution to the Cheap Spectral Factorization Problem*", in Proc. European Control Conference, pp. 814-819, Budapest, Hungary, August 23-26, 2009.

C12. F. Morbidi, G. Ripaccioli, D. Prattichizzo, "*On Connectivity Maintenance in Linear Cyclic Pursuit*", in Proc. IEEE International Conference on Robotics and Automation, pp. 363-368, Kobe, Japan, May 12-17, 2009.

C11. F. Morbidi and D. Prattichizzo, "*Range estimation from a moving camera: an I&I approach*", in Proc. IEEE International Conference on Robotics and Automation, pp. 2810-2815, Kobe, Japan, May 12-17, 2009.

C10. G.L. Mariottini, S. Scheggi, F. Morbidi, D. Prattichizzo, "*Planar Catadioptric Stereo: Single and Multi-View Geometry for Calibration and Localization*", in Proc. IEEE International Conference on Robotics and Automation, pp. 1510-1515, Kobe, Japan, May 12-17, 2009.

C9. F. Morbidi, F. Bullo, D. Prattichizzo, "*On leader-follower visibility maintenance for Dubins-like vehicles via controlled invariance*", in Proc. 47th IEEE Conference on Decision and Control, pp. 1821-1826, Cancun, Mexico, December 9-11, 2008.

C8. F. Morbidi, G.L. Mariottini, D. Prattichizzo, "*Vision-based range estimation* via *Immersion and Invariance for robot formation control*", in Proc. of the IEEE International Conference on Robotics and Automation, pp. 504-509, Pasadena, CA, USA, May 19-23, 2008.

C7. L. Consolini, F. Morbidi, D. Prattichizzo, M. Tosques, "*Steering hierarchical formations of unicycle robots*", in Proc. 46th IEEE Conference on Decision and Control, pp. 1410-1415, New Orleans, LA, USA, December 12-14, 2007.

C6. G. Marro, F. Morbidi, D. Prattichizzo, "*H_2-Pseudo Optimal Model Following: A Geometric Approach*", in Proc. 3rd IFAC Symposium on System, Structure and Control, Foz do Iguaçu, Brazil, October 17-19, 2007.

C5. F. Morbidi, L. Consolini, D. Prattichizzo, M. Tosques, "*Leader-Follower Formation Control as a Disturbance Decoupling Problem*", in Proc. European Control Conference, pp. 1492-1497, Kos, Greece, July 2-5, 2007.

C4. F. Morbidi, A. Garulli, D. Prattichizzo, C. Rizzo, S.Rossi, "*A Kalman filter approach to remove TMS-induced artifacts from EEG recordings*", in Proc. European Control Conference, pp. 2201-2206, Kos, Greece, July 2-5, 2007.

C3. L. Consolini, F. Morbidi, D. Prattichizzo, M. Tosques, "*A Geometric Characterization of Leader-Follower Formation Control*", in Proc. of the IEEE International Conference on Robotics and Automation, pp. 2397-2402, Rome, April 10-14, 2007.

C2. G.L. Mariottini, F. Morbidi, D. Prattichizzo, G.J. Pappas, K. Daniilidis, "*Leader-Follower Formations: Uncalibrated Vision-Based Localization and Control*", in Proc. of the IEEE International Conference on Robotics and Automation, pp. 2403-2408, Rome, April 10-14, 2007.

C1. L. Consolini, F. Morbidi, D. Prattichizzo, M. Tosques, "*On the Control of a Leader-Follower Formation of Nonholonomic Mobile Robots*", in Proc. 45th IEEE Conference on Decision and Control, pp. 5992–5997, San Diego, CA, USA, December 13-15, 2006.

Revues nationales à comité de lecture

JN2. L. Ménard, S. Achille-Fauveau, M. Babel, R. Khemmar, L. Lecrosnier, F. Morbidi, D. Laval-Quesney, P. Gallien, É. Leblong, "*Mieux connaître les nouvelles technologies: la formation des professionnels de santé par le projet ADAPT*", ergOThérapies, Tome 2, Editorial n. 87, pp. 35-45, October 2022 (dossier spécial: "Les aides techniques et l'ergothérapie: enjeux et perspectives"), `https://revue.anfe.fr`

JN1. E. Mouaddib, G. Caron, D. Leclet-Groux, F. Morbidi, "*Le patrimoine in silico. Exemple de la cathédrale d'Amiens*", In Situ. Revue des patrimoines, n. 39, numéro spécial intitulé "Imagerie numérique et patrimoine culturel : enjeux scientifiques et opérationnels", P. Liévaux, L. De Luca (eds.), May 2019.

Conférences nationales avec actes et comité de lecture

CN2. F. Morbidi, E. Mouaddib, "*E-Cathédrale: numérisation de la cathédrale d'Amiens et reconstruction des trajectoires balistiques de l'artillerie allemande en avril 1918*", actes du colloque "*Cathédrales en Guerre: XVI^{e} - XX^{e} siècles*", Amiens, June 8-9, 2017, Presses Universitaires du Septentrion, X. Boniface, L. Dessaivre (eds.), pp. 23–44, 1re édition, January 31, 2020, `www.septentrion.com/fr/livre/?GCOI=27574100809690`

CN1. G. Caron, F. Morbidi, "*Gyroscope visuel sphérique basé mélange de potentiels photométriques*", in Reconnaissance des Formes, Image, Apprentissage et Perception (RFIAP), Marne-la-Vallée, France, June 25-28, 2018.

Communications orales

O6. D. Rodrigues da Costa, P. Vasseur, F. Morbidi, "*Vision Événementielle Omnidirectionnelle : Théorie et Applications*", in Journées Francophones des Jeunes Chercheurs en Vision par Ordinateur (ORASIS), Carqueiranne, France, May 22-26, 2023.

O5. A. El Moudni, S. Kramm, F. Morbidi, R. Boutteau, "*Suivi et estimation de profondeur avec un banc stéréo événementiel embarqué sur un véhicule autonome*", in Journées Francophones des Jeunes Chercheurs en Vision par Ordinateur (ORASIS), Carqueiranne, France, May 22-26, 2023.

O4. J. Caracotte, F. Morbidi, E. Mouaddib, "*Photometric Stereo with Twin-Fisheye Cameras*", in Journées Francophones des Jeunes Chercheurs en Vision par Ordinateur (ORASIS), Lac de Saint-Ferréol, France, September 13-17, 2021.

O3. J. Caracotte, F. Morbidi, E. Mouaddib, "*Photometric Stereo with Twin-Fisheye Cameras*", in Journées Francophones des Jeunes Chercheurs en Vision par Ordinateur (ORASIS), Lac de Saint-Ferréol, France, September 13-17, 2021.

O2. G. Hardouin, F. Morbidi, J. Moras, J. Marzat, E. Mouaddib, "*Surface-driven Next-Best-View planning for exploration of large-scale 3D environments*", in Reconnaissance des Formes, Image, Apprentissage et Perception et Conférence sur l'Apprentissage automatique (CAp), Vannes, France, June 23-26, 2020.

O1. S. Scheggi, F. Morbidi, D. Prattichizzo, "*Mixed human-robot formation control with vibrotactile haptic feedback*" in IEEE International Conference on Robotics and Automation Workshop "Haptic Teleoperation of Mobile Robots: Theory, Applications and Perspectives", St. Paul, MN, USA, May 14-18, 2012.

Soumis

- D. Rodrigues da Costa, P. Vasseur, F. Morbidi, "*Gyrevento: Event-based Omnidirectional Visual Gyroscope in a Manhattan World*", submitted to the International Journal of Robotics Research, January 30, 2024.
- A.N. André, F. Morbidi, G. Caron, "*UniSphorM: A New Uniform Spherical Image Representation with Application to Direct Attitude Estimation*", submitted to the IEEE Transactions on Robotics, December 8, 2023.

N.B. Mes articles ont été cités 2990 fois ([J2] a été cité 760 fois) et mon h-index est 27 (février 2024). Pour plus de détails, veuillez consulter mon profil Google Scholar : `http://scholar.google.it/citations?user=THZ3Y7MAAAAJ&hl=en`.

1.1.3 Encadrement de jeunes chercheurs

- Post-doctorants :

  - **Houssem-Eddine Benseddik**, "*Omnidirectional Vision for Orientation Estimation and Localization of a Semi-Autonomous Wheelchair*", **post-doc** financé par le conseil régional des Hauts-de-France en support du projet Interreg VA ADAPT (1er janvier 2018 - 4 décembre 2019). Co-encadrement à 50% avec G. Caron. Une publication a été réalisée : [J19].

- Doctorants :

  - **Djessy Rossi**, "*Event-based detection for intelligent transportation systems: Convolutional vs Spiking Neural Networks*", thèse financée par le projet ANR PRCE CERBERE (janvier 2022 - janvier 2026). Directeur de thèse : Prof. P. Vasseur (taux d'encadrement 50%). Co-encadrant : F. Morbidi (taux d'encadrement 50%).
  - **Anass El Moudni**, "*Reconstruction 3D par vision événementielle stéréoscopique pour le véhicule*", thèse financée par le projet ANR PRCE CERBERE (janvier 2022 - janvier 2026). Directeur de thèse: Prof. R. Boutteau (taux d'encadrement 50%). Co-encadrants : F. Morbidi (taux d'encadrement 25%) et S. Kramm (taux d'encadrement 25%). Une publication a été réalisée : [C42].
  - **Daniel Rodrigues da Costa**, "*Evento : Caméras évènementielles omnidirectionnelles pour les robots à très haute dynamique*", thèse cofinancée par l'AID (Agence de l'Innovation de Défense - Ministère des Armées) et par l'UPJV (octobre 2021 - octobre 2024). Directeur de thèse : Prof. P. Vasseur (taux d'encadrement 50%). Co-encadrant : F. Morbidi (taux d'encadrement 50%). Une présentation orale a été réalisée : [O6].
  - **Guillaume Hardouin**, "*ScanBot : Scanners Robotisés pour la Numérisation Automatique du Patrimoine*", thèse cofinancée par la région Hauts-de-France et l'ONERA DTIS, Palaiseau (octobre 2018 - décembre 2021). Directeur de thèse: Prof. E. Mouaddib (taux d'encadrement 33%). Co-encadrants : F. Morbidi (taux d'encadrement 33%), J. Moras (taux d'encadrement 33%). La thèse a été soutenue le 22 mars 2022 à l'ONERA, Palaiseau. Trois publications ont été réalisées : [C36], [C38] et [J24].
  - **Devesh Adlakha**, "*Robust Globally Optimal Optimization in Computer Vision: Geometric Transformations Estimation with Semantic Cues and Vague Motion Knowledge*", thèse cofinancée par le projet ANR SUMUM et l'Université de Stras-

bourg (janvier 2018 - janvier 2021). Directeur de thèse : M. de Mathelin (iCube, Université de Strasbourg). Co-encadrant : A. Habed (Université de Strasbourg). J'ai suivi le doctorant et j'ai participé au co-encadrement tout au long de sa thèse (une réunion tous les 15 jours en moyenne). La thèse a été soutenue le 12 décembre 2022 à Strasbourg. Deux publications ont été réalisées : [C34] et [C39].

- **Jordan Caracotte**, "*Reconstruction 3D par stéréophotométrie pour la vision omnidirectionnelle*", thèse ministérielle (octobre 2016 - juillet 2021). La thèse a été soutenue le 6 juillet 2021. Directeur de thèse : Prof. E. Mouaddib (taux d'encadrement 30%). Co-encadrant : F. Morbidi (taux d'encadrement 70%). Deux publications ont été réalisées : [J20] et [C37].

N.B. Tous les co-encadrements de thèse ont été autorisés par la CR (Commission recherche) du conseil académique de l'UPJV.

▶ Ingénieurs de recherche :

- **Sarah Delmas**, "*SpheriCol: Augmented reality for driving assistance of an electric wheelchair*", **ingénieur de recherche** financée par le projet Interreg VA ADAPT (15 octobre 2019 - 15 octobre 2021). Trois publications ont été réalisées : [C40], [J22] et [J23].

▶ Étudiants en Master :

- **Anass El Moudni** (Master 2/ingénieur, Université de Franche-Comté) : Stage intitulé "*Reconstruction 3D par vision évènementielle stéréoscopique pour le véhicule autonome*", 21 février 2022 - 31 août 2022. Co-encadrement avec S. Framm et R. Boutteau, laboratoire LITIS, Université de Rouen. Stage financé par le projet ANR CERBERE.
- **Daniel Rodrigues da Costa** (Master 2, Dépt. EEA, UPJV) : Stage intitulé "*EVENTO : Caméras évènementielles omnidirectionnelles pour les robots à très haute dynamique*", 8 février 2021 - 31 août 2021. Co-encadrement (à 50%) avec P. Vasseur.
- **Julien Albrand** (Master 2, Dépt. EEA, UPJV) : Stage intitulé "*Estimation d'angle de cap d'un véhicule par vision*", 17 février - 31 août 2020. Co-encadrement avec A. Rabhi et G. Caron.

- **Méven Jeanne-Rose** (Master 2, Dépt. EEA, UPJV) : Stage intitulé "*Conception d'un système twin-fisheye stéréo pour l'aide à la conduite d'un fauteuil roulant électrique*", 17 février - 31 août 2020. Co-encadrement (à 50%) avec S. Delmas.
- **Julien Albrand** et **Méven Jeanne-Rose** (Master 1, Dépt. EEA, UPJV) : Projet intitulé "*SpheriCol: Augmented reality for driving assistance of an electric wheelchair*" juillet 2019. Co-encadrement (à 50%) avec G. Caron. Une publication a été réalisée : [C40].
- **Mohamed Zerrouki** (Master 2, Dépt. EEA, UPJV) : Stage intitulé "*Commande prédictive pour la stabilité du véhicule*", 16 avril 2018 - 17 septembre 2018. Co-encadrement avec A. Rabhi et A. El Hajjaji.
- **Jean Hoffmann** (Master 1, Dépt. EEA, UPJV) : Stage intitulé "*Reconnaissance d'Endive par Vision et Pick-and-Place par Robot Industriel*", avril-août 2016 en collaboration avec l'entreprise Godé.
- **Roel Ramiro Cano Cisneros** (Master 1, Universidad Autónoma de Tamaulipas, Mexico) : Stage intitulé "*Energy-optimal path planning for a quadrotor UAV*", mai-août 2015. Co-encadrant (Universidad Autónoma de Tamaulipas, Mexico) : David Lara. Une publication a été réalisée : [C30].
- **Maroua Selmi** (Master 2, Dépt. EEA, UPJV) : Stage intitulé "*Correlation de phase pour l'odométrie visuelle omnidirectionnelle d'un robot mobile*", mai-août 2015. Ce travail a été valorisé, en partie, dans l'article [J18].

1.1.4 Rayonnement et visibilité

Distinctions scientifiques et primes

- **Outstanding Reviewer** de la revue IEEE Transactions on Robotics pour les annés 2013 et 2014.
- Bénéficiaire d'une **PEDR**, "Prime d'Encadrement Doctoral et de Recherche" en octobre 2017, pour une durée de 4 ans.
- Bénéficiaire d'un **CRCT**, "Congé pour Recherches ou Conversions Thématiques" au titre de l'établissement (UPJV) d'une durée d'un semestre (1er février 2021 – 1er septembre 2021). Titre du projet de recherche : "*Optimal Energy Management for Extended Flight Endurance of Battery-Powered Multi-rotor UAVs*".

Conférences et séminaires sur invitation

- "*Event-based vision for robotics*", laboratoire CRIStAL, Université de Lille, 27 octobre 2023.

- "*Event-based vision: A new sensing paradigm for agile drones and autonomous cars*", CNRS-AIST Joint Robotics Laboratory, Tsukuba, Japan, 17 février 2023.
- "*Minimum-Energy Trajectory Generation for Battery-Powered Multirotor UAVs*", Dipartimento di Ingegneria e Architettura, Università di Parma, Italie, 5 novembre 2021 et GT-UAV (Véhicules Aériens), GdR Robotique, Webinar, 10 décember 2021.
- "*Vision non conventionnelle pour la robotique mobile: caméras catadioptriques, plénoptiques et event-based*", session Perception, Journées Nationales de la Recherche Robotique (JNRR), Domaine de Françon, Biarritz, France, 9 novembre 2017.
- "*E-Cathédrale : numérisation de la cathédrale d'Amiens et reconstruction des trajectoires balistiques de l'artillerie allemande en avril 1918*", Colloquium "Les Cathédrales en Guerre (XVIe-XXIe siècles)", organisé par la Société des Amis de la Cathédrale d'Amiens en partenariat avec l'Université de Picardie Jules Verne, Amiens, 8-9 juin 2017.
- "*Feature-based and Dense Omnidirectional Visual Compass for Autonomous Robots*", International Multidisciplinary Workshop on Sensing Reconstruction, and Recognition of Environment, Optical Media Interface Lab, Nara Institute of Science and Technology (NAIST), Japon, 16-17 mars 2017, `http://omilab.naist.jp/WS20170316.html`
- "*Minimum-Energy Path Generation for Battery-Powered Multirotor UAVs*", équipe RIS, LAAS-CNRS, Toulouse, France, 12 décembre 2016.
- "*Digital Modeling of Cultural Heritage: Toward Cloud-based Robotic Solutions*", Keynote talk, International Conference on Cloud and Robotics (ICCR16), Saint Quentin, France, 22 novembre 2016.
- "*Cooperative Active Target Tracking for Multiple Aerial Vehicles*", équipe Lagadic, Inria Rennes - Bretagne Atlantique, France, 30 juin 2016.
- "*Calibration précise et robuste de caméras RGB-D basée sur les objets sphériques*", Journée GdR ISIS-Robotique "Analyse des données RGB-Depth pour l'analyse de scènes", Telecom Paris, France, 31 mai 2016.
- "*Robust traffic estimation over graphs*", EU HYCON2 WP5 meeting - Traffic Showcase, Seville, Spain, 17 mai 2013.
- "*The Deformed Consensus Protocol*", GIPSA-lab, CNRS, Université Grenoble Alpes, France, 19 septembre 2012.

- "*Distributed Estimation and control of UAV Swarms for Environmental monitoring*", Dept. of Mechanical Engineering, Northwestern University, USA, 11 mai 2010.

Comités de rédaction

► **Revues internationales**

- Editeur Associé des "*IEEE Transactions on Robotics*" depuis mars 2022.
- Editeur Associé des "*IEEE Robotics and Automation Letters*" depuis mars 2022.
- Membre de l'Editorial Board (comme Review Editor) de "*Frontiers in Robotics and AI*" spécialité "Robotic Control Systems", depuis juin 2015.
- Membre de l'Editorial Board de "*International Journal of Advanced Robotic Systems*", depuis le 26 avril 2013.

► **Conférences internationales**

- Editeur Associé pour le Conference Editorial Board (CEB) de l'IEEE Robotics and Automation Society pour ICRA 2017-2024.
- Editeur Associé du IROS Conference Paper Review Board (CPRB) en 2020, 2021, 2022 et 2024.
- Program Co-Chair of the *3rd International Conference on Robotics, Computer Vision and Intelligent Systems*, ROBOVIS 2022, 26-27 octobre 2022, Valletta, Malta
- Member of the Program Committee of the "*10th European Conference on Mobile Robots*", ECMR 2021, 31 août - 3 septembre 2021, Bonn, Germany `https://ecmr2021.org`
- Member of the International Program Committee of the "*2nd International Conference on Robotics, Computer Vision and Intelligent Systems*", ROBOVIS 2021, 27-29 octobre 2021, Valletta, Malta
- Member of the Program Committee of the "*9th European Conference on Mobile Robots*" ECMR 2019, 4-6 septembre 2019, Prague, Czech Republic, `www.ecmr2019.eu`
- Member of the Program Committee of the "*International Symposium on Multi-robot and Multi-Agent Systems*", MRS 2017, 4-5 décembre 2017, University of Southern California, Los Angeles, USA.
- Member of the International Program Committee of the "*14th International Conference on Informatics in Control, Automation and Robotics*", ICINCO 2017, 29-31 juillet 2017, Madrid, Spain.

Appartenance à des sociétés savantes

- **IEEE Membership** (IEEE Robotics & Automation Society - IEEE RAS) :
  - IEEE *Student Member* depuis janvier 2007
  - IEEE *Associate Member* depuis octobre 2009
  - IEEE *Member* depuis mai 2012
  - IEEE **Senior Member** depuis avril 2019
- Membre du IEEE RAS Technical Committee on **Multi-Robot Systems**, depuis septembre 2014.
- Secrétaire du Chapitre Italien de l'IEEE RAS : octobre 2006 - octobre 2010.
- Membre nommé suppléant du CNU, collège B, section 61, mandature 2023-2027.
- Membre du GdR MACS ("*Groupement de Recherche Modélisation, Analyse et Conduite des Systèmes dynamiques*"), depuis janvier 2018, `http://gdr-macs.cnrs.fr`.

Activités de reviewer

- Expertise de **projets de recherche** nationaux et internationaux :
  - Expertise pour une bourse de doctorat, Région Normandie, 10 mars 2023.
  - Expertise d'un dossier scientifique dans le cadre de l'AAP IRS 2020 de l'IDEX Communauté Université Grenoble Alpes, 21 mars 2021.
  - Expertise d'un projet ANR générique (2^{e} phase, comité d'évaluation "CE33 - Interaction, Robotique"), 10 mai 2019.
  - Expertise d'un projet pour le "Research Grants Council" (RGC) de Hong Kong, Chine, 16 février 2018.
- Membre du comité pour l'Annual European Systems & Control PhD Thesis Award, mai 2023, `www.eeci-igsc.eu/eeciphdaward`
- Reviewer pour les **revues** les plus importantes dans le domaine de la Robotique et de l'Automatique, notamment : *IEEE Transactions on Robotics* (environ 40 soumissions évaluées), *International Journal of Robotics Research*, *Robotics and Autonomous Systems*, *Autonomous Robots*, *IEEE Robotics and Automation Letters*, *IEEE Transactions on Automatic Control*, *Automatica*, *Systems & Control letters*, *IEEE Transactions on Control Systems Technology*, *IEEE Transactions on Control of Network Systems*.

- Reviewer pour les **conférences** internationales de Robotique et d'Automatique les plus importantes, notamment : *IEEE Int. Conf. Robotics and Automation*, *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, *Robotics: Science and Systems*, *IEEE Conf. on Decision and Control*, *American Control Conf.*, *European Control Conf.*, *IFAC World Congress*.

Collaborations externes

- Visite de l'Institute of Fundamental Technological Research of the Polish Academy of Sciences (IPPT PAN), Varsovie, Pologne, 3-7 février 2020 et 20-24 juillet 2021. La visite a permis de lancer une **collaboration de recherche** avec le Dr. Dominik Pisarski. Une publication commune, [C41], a été réalisée.

- Optical Media Interface Lab (Mukaigawa Lab), Nara Institute of Science and Technology (NAIST), Japon, 7-15 avril 2019. Réunions de travail avec les collaborateurs du NAIST, du CNRS-AIST JRL (Tsukuba) et de l'Université de Tokyo, pour le **montage d'un projet Franco-Japonais** JST-ANR (CREST). Les doctorants J. Caracotte et D. Rodrigues da Costa ont effectué un séjour de recherche de deux mois au Optical Media Interface Lab en 2019 et 2023, respectivement.

- Institute of Computer Graphics and Vision, Graz University of Technology, Autriche. **Collaboration de recherche** avec le Prof. Friedrich Fraundorfer dans le cadre du projet ANR-FWF EVELOC (2024-2028), voir la Sect. 1.1.5 pour plus de détails.

Jurys de thèse

- "*Generation of Reactive Trajectories for Multi-Drone Systems*", **Nicola De Carli**, Université de Rennes 1, IRISA, INRIA Rennes-Bretagne Atlantique, 18 avril 2024.

  Directeur de thèse : P. Robuffo Giordano
  Co-directeur de thèse : P. Salaris (Università di Pisa, Italie)
  Examinateur : F. Morbidi

- "*Robust Trajectory Planning Algorithms for Robotic Tasks with Parametric Uncertainties*", **Pascal Brault**, Université de Rennes 1, IRISA, INRIA Rennes - Bretagne Atlantique, 27 avril 2023.

  Directeur de thèse : P. Robuffo Giordano
  Examinateur : F. Morbidi

- "*Segmentation dimages omnidirectionnelles de scènes routières*", **Ahmed Rida Sekkat**, Université de Rouen Normandie, 8 décembre 2022.

Directeurs de thèse : P. Honeine et P. Vasseur
Co-directeur de thèse : Y. Dupuis
Examinateur : F. Morbidi

- "*Perception embarquée/débarquée pour la navigation autonome*", **Antoine Caillot**, Université de Rouen, 22 novembre 2022.

  Directeurs de thèse : Y. Dupuis et R. Boutteau
  Co-directeurs de thèse : S. Ouerghi et P. Vasseur
  Examinateur : F. Morbidi

- "*Planar Pushing: a study of non-prehensile manipulation with single and multiple mobile robots*", **Filippo Bertoncelli**, Università degli studi di Modena e Reggio Emilia, Italie, mai 2022.

  Directeur de thèse : L. Sabattini
  Examinateur externe : F. Morbidi

- "*Algorithmes d'estimation et de commande pour des quadrirotors en intéraction physique avec l'environnement*", **Quentin Delamare**, Université de Rennes 1, IRISA, INRIA Rennes-Bretagne Atlantique, 9 décembre 2019.

  Directeur de thèse : P. Robuffo Giordano
  Co-directeur de thèse : A. Franchi
  Examinateur : F. Morbidi

- "*The Shape of Damping: Optimizing Damping Coefficients to Improve Transparency on Bilateral Telemanipulation*", **Olmo Alonso Moreno Franco**, Università degli studi di Genova et Istituto Italiano di Tecnologia (IIT), Italie, 14 février 2019.

  Directeur de thèse : D. Prattichizzo
  Membre du jury : L. Pallottino
  Rapporteurs externes : F. Chinello, F. Morbidi

- "*Synthesis and Analysis of Virtual Holonomic Constraints for Lagrangian Mechanical Systems*", **Alessandro Costalunga**, Università di Parma, Italie, mars 2017.

  Directeur de thèse : M. Locatelli
  Co-directeur de thèse : L. Consolini
  Membre du jury : A. Visioli
  Rapporteurs externes : M. Maggiore, F. Morbidi

- "*Distributed Methods for Finite-time Average Consensus Protocol Design and Network Robustness Assessment*", **Thi-Minh Dung Tran**, Université Grenoble Alpes, 26 mars 2015.

Directeur de thèse : C. Canudas-de-Wit
Co-encadrant : A. Kibangou
Examinateur : F. Morbidi

Comités de suivi individuel (CSI)

- "*Conduite assistée d'un fauteuil roulant : navigation par intervalles à l'aide de beacons Ultra Large Bande*", **Julien Albrand**, IRISA/INSA de Rennes (octobre 2020). Encadrants : V. Drevelle, E. Marchand, M. Babel.

Comités de sélection

- Poste de Maître de Conférences (Section CNU 61), Université de Technologie de Compiègne, laboratoire Heudiasyc, 24 avril et 22 mai 2024. *Rôle* : rapporteur.
- Poste de Maître de Conférences (Section CNU 61), Université Grenoble Alpes, laboratoire GIPSA, 2 et 16 mai 2023. *Rôle* : rapporteur. Présidente du comité de recrutement : Marie-Laure Espinouse.
- Poste de Maître de Conférences (Section CNU 61), INSA de Rouen, laboratoire LITIS, 9 avril 2021. *Rôle* : rapporteur. Présidente du comité de recrutement : Ouiddad Labbani-Igbida.

1.1.5 Responsabilités scientifiques

Projets et contrats de recherche

- Université de Picardie Jules Verne (UPJV) :
  - **EVELOC** [janvier 2024-janvier 2028] "*Event-based Visual Localization*", ANR-FWF (projet collaboratif franco-autrichien). Coordinateurs : P. Vasseur (UPJV, France), F. Fraundorfer (Graz University of Technology, Autriche). Trois laboratoires sont impliqués dans le projet. Budget total : 705 k€. *Rôle* : **Participant** dans l'équipe de l'UPJV.
  - **M^{2}SV** [janvier 2024-janvier 2026] "*Mobile Manipulator control based on Spherical Vision*". Projet exploratoire du GdR 720 ISIS. Porteur : Dr. Nathan Crombez (Université de Technologie de Belfort-Montbéliard). Trois laboratoires sont impliqués dans le projet. Budget : 7 k€. *Rôle* : **Participant** dans l'équipe de l'UPJV.

- **DEVIN** [janvier 2024-janvier 2028] "*Drones with Omni-Event Vision for Drone Neutralization*", ANR PRC, AAP TSIA (Thématiques Spécifiques en Intelligence Artificielle), Thématique 2 : Flottes intelligentes de robots. Porteur : Dr. Guillaume Allibert (I3S, Université Côte d'Azur). Trois laboratoires sont impliqués dans le projet. *Rôle* : **Participant** dans l'équipe de l'UPJV.
- **HaDROs** [janvier 2022-janvier 2024] "*Vision Omni-HDR pour la localisation et cartographie visuelles en environnement à large gamme de radiance*". Projet de recherche exploratoire 2021 du GdR 720 ISIS. Porteur : Dr. Guillaume Caron (CNRS-AIST JRL). Trois laboratoires sont impliqués dans le projet. Budget : 7 k€. *Rôle* : **Responsable** de l'équipe de l'UPJV.
- **CERBERE** [janvier 2021-décembre 2025] "*Caméra évènementielle pour la pERception d'oBjEts Rapides autour du véhicule autonomE*", ANR PRCE. Porteur : R. Boutteau (LITIS, Université de Rouen). Le consortium est constitué de trois laboratoires et d'une entreprise. Budget total : 656 k€. *Rôle* : **Responsable** de l'équipe de l'UPJV, `https://litis-cerbere.univ-rouen.fr`
- **ScanBot** [octobre 2018-octobre 2021] "*Scanners RoBotisés pour la Numérisation Automatique du Patrimoine*" : Projet de recherche cofinancé par la Région Hauts-de-France et par ONERA DTIS (Palaiseau). Budget : 100 k€. *Rôle* : **Porteur**.
- **FullScan** [décembre 2018-décembre 2019] "*Combining Complementary Sensors on a Mobile Robotic System to Fully and Autonomously Digitize Complex Heritage Buildings with Ultra High Definition*", projet PHC (Partenariat Hubert Curien) Sakura. Porteurs : G. Caron (UPJV, France) et T. Funatomi (NAIST, Japon). Budget : 15 k€. *Rôle* : **Participant** dans l'équipe de l'UPJV.
- **SUMUM** [septembre 2017-mars 2021] "*Le numérique au service de la conservation-restauration du patrimoine artistique*", ANR PRC, Challenge 7 : Information and Communication Society. Porteur : Prof. A. Mansouri (Le2I, Université Bourgogne Franche-Comté). Quatre laboratoires partenaires : Le2I, GREYC, CICRP/LABCOM, MIS. Budget total : 687 k€. *Rôle* : **Participant** dans l'équipe de l'UPJV, `https://anr-sumum.fr`
- **ADAPT** [janvier 2017-juin 2022] "*Assistive Devices for empowering disAbled People through robotic Technologies*", Interreg VA France (Channel) England programme, 16 partenaires : 8 français et 8 anglais. Porteur : Dr. N. Ragot (ESIGELEC Rouen, France). Budget total du European Region Development Fund (ERDF) : 5.944 M€. *Rôle* : **Participant** et depuis septembre 2019 **responsable local** (l'UPJV était le leader du WP2), `http://adapt-project.com`

- Inria, Grenoble Rhône-Alpes, France :

  - Entre novembre et décembre 2012, j'ai participé à la rédaction du projet EU STREP - FP7 ICT intitulé "*Systems-of-Systems control management of heterogeneous traffic networks*" (SoSmart). Partenaires du projet : Inria Grenoble Rhône-Alpes (porteur), EPFL (Suisse), KTH (Suède) et l'entreprise Karrus.
  - J'ai été impliqué dans le projet national "*MOCoPo: Measuring and mOdelling traffic COngestion and POllution*". Ce projet de trois ans a débuté en janvier 2011 et il a été financé par le Ministrère chargé des Transports via le PREDIT (Programme de Recherche et d'Innovation dans les Transports Terrestres).
  - J'ai été impliqué dans le projet HYCON2 NoE, WP5 "*Benchmarks: traffic showcase*". Ce projet européen de quatre ans, coordonné par le CNRS (F. Lamnabhi-Lagarrigue) a débuté en septembre 2010.
  - J'ai rédigé le projet intitulé "*Mobile Devices for Intelligent Road Traffic Prediction*" pour l'AAP iC LSI (Appel à projet, fonds de l'Institut Carnot, Logiciel et Systemés Intelligents). Ce projet de 9 mois, accepté le 29 novembre 2013, a démarré en janvier 2014.

- Johannes Kepler University (JKU), Autriche :

  - J'ai été impliqué pendant 9 mois dans le projet intitulé "*DVKUP: Demonstration und Validierung eines Kollisionsvermeidungssystems zur Unterstützung von Piloten*" ("*DVKUP: Demonstration and Validation of a Collision Avoidance System for the Support of Pilots*") financé par la FFG, l'agence autrichienne pour la recherche appliquée (janvier 2012 - janvier 2014). Partenaires du projet : AeroSpy Sense & Avoid Technology GmbH (porteur), JKU, Austrian Institute of Technology (AIT), HB Flugtechnik GmbH, Austro Control.
  - À l'automne 2011, j'ai participé à l'écriture du projet intitulé "*A RObotic SErvice System (ROSES) for the Science Park of Johannes Kepler University*". Le projet (2013-2017) a été rédigé dans le cadre du ACCM "Austrian Center of Competence in Mechatronics" (extension K2). Partenaires du projet (JKU): H. Bremer, L. del Re, A. Stelzer.

- University of Texas at Arlington (UTA), USA :

  - En mars 2011, j'ai participé à la rédaction du projet intitulé "*Mobile Multi-Agent Human-Capturing System for Monitoring Human Motion*", National Science Foundation (NSF), Cyber-Physical Systems (CPS) Program. PI : G.L. Mariottini (CSE Dept., UTA). Co-Pis : C. Ray, M. Ricard (Kinesiology Dept., UTA), F. Kamangar, V. Athitsos (CSE Dept., UTA).

- Northwestern University (NU), USA :

– J'ai été impliqué dans le projet financé par l'Office of Naval Research, intitulé : "*Automatic Compilation of Global Objectives into Local Controllers for Cooperative Mobile Agents*" (1 octobre 2008 - 30 novembre 2012). Porteurs du projet : K.M. Lynch (Dept. of Mechanical Engineering, NU) et R.A. Freeman (Dept. of Electrical Engineering, NU).

Financements pour la mobilité et la recherche

- **AAP S2R** (juillet 2022) de l'UPJV, Action 4.5 ("Aide aux mobilités sortantes des doctorants") : financement de 1.3 k€ pour une visite de deux mois du doctorant D. Rodrigues da Costa au NAIST (Nara Institute of Technology), Japon.
- **AAP S2R** (juillet 2021) de l'UPJV, Action 4.3 ("Aide aux mobilités sortantes des enseignants-chercheurs - recherche") : financement de 375 € pour une visite d'une semaine (20-24 juillet 2021) de l'IPPT PAN (Polish Academy of Sciences) Varsovie, Pologne.
- **AAP S2R** (février 2020) de l'UPJV, Action 4.3 ("Aide aux mobilités sortantes des enseignants-chercheurs - recherche") : financement de 800 € pour une visite d'une semaine (2-7 février 2020) de l'IPPT PAN (Polish Academy of Sciences), Varsovie, Pologne.
- **AAP S2R** (avril 2018) de l'UPJV, Action 4.5 ("Aide aux mobilités sortantes des doctorants") : financement de 1.3 k€ pour la visite de deux mois du doctorant J. Caracotte au NAIST (Nara Institute of Technology), Japon.

Organisation de workshops et activité de *Chair*

- European robotic projects, H2020 program showcase at ICRA'20, 31 mai - 31 août 2020, Paris, France. **Présentation du projet ADAPT** (avec M. Babel).
- **Organisateur local** de la Journée Régionale des Doctorants en Automatique (JRDA 2018), 3 juillet 2018, Amiens, France.
- IEEE/RSJ International Conference on Intelligent Robots and Systems, Madrid, Spain, 1er octobre 2018. **Co-organisateur** avec M. Babel, D. Daney, S. Mohammed, F. Colas et Y. Amirat du workshop intitulé "*Assistance and Service Robotics in a Human Environment: From Personal Mobility Aids to Rehabilitation-Oriented Robotics*", `www.iros-ar2018.lissi.fr/doku.php`
- IEEE International Conference on Robotics and Automation, Singapore, 31 mai 2017. **Chair** of the regular session "*Semantic Understanding*".

- "Journée des Jeunes Chercheurs en Robotique" (JJCR) et "Journées Nationales de la Recherche en Robotique" (JNRR), 20-23 octobre 2015. Membre du **comité local d'organisation**.

- European Control Conference, Strasbourg, France, juin 2014. **Organisateur** et **Chair** (avec A. Ferrara) de l'invited session intitulée "*Traffic Control and Estimation: New Trends and Opportunities*".

- European Control Conference, Zurich, Switzerland, juillet 2013. **Chair** of the regular session "*Vehicle Formation Control*".

- IEEE International Conference on Decision and Control, Maui, HI, USA, décembre 2012. **Co-Chair** of the regular session "*Agents and Autonomous Systems II*".

- American Control Conference, San Francisco, CA, USA, juin 2011. **Co-Chair** of the regular session "*Cooperative Control II*".

- **Judge** for the FIRST LEGO League, Grades 4-8. Museum of Nature & Science, Dallas, TX, USA, 12 novembre 2010. Department of Computer Science and Engineering, University of Texas at Arlington, USA, 4 décembre 2010, `www.firstlegoleague.org`

- **Co-organisateur** (avec G. Marro, L. Ntogramatzidis et D. Prattichizzo) du minicours (8 exposés, 4 heures) intitulé "*The Geometric Approach to Systems & Control Theory*", 19th Int. Symposium on Mathematical Theory of Networks and Systems (MTNS), Budapest, Hongrie, 8 juillet 2010.

- IEEE International Conference on Robotics and Automation, Anchorage, USA, mai 2010. **Co-Chair** of the regular session "*Software Tools for Robotics*".

Responsabilités au laboratoire MIS

- Responsable de l'équipe Perception Robotique (PR) du laboratoire MIS depuis le 1er mars 2022.

- Membre du conseil du laboratoire MIS depuis le 19 janvier 2017.

- Membre du comité des finances du laboratoire MIS entre mars 2017 et janvier 2022.

- Responsable (avec Corinne Lucet) des séminaires du laboratoire MIS depuis mai 2018. Responsable des séminaires de l'équipe PR depuis 2019.

Remarque : Le laboratoire MIS compte environ 90 membres (permanents, post-docs et doctorants) pour une dotation annuelle de l'UPJV d'environ 90 k€. Le conseil du laboratoire MIS et le comité des finances comptent entre 5 et 15 personnes. L'équipe PR a entre 15 et 20 membres (9 permanents) et un budget annuel de l'établissement d'environ 9 k€.

Activités de valorisation et vulgarisation

- "*L'œil de Terminator : Vision événementielle pour la robotique*", Conférence Flash pour les étudiants de Licence 2, UFR des Sciences, Université de Picardie Jules Verne, 7 novembre 2023, `https://webtv.u-picardie.fr`
- "*Introduction au Traitement du Signal sur Graphe*", Laboratoire MIS, Université de Picardie Jules Verne, 20 mars 2023.
- "*Handicap. Un fauteuil électrique intelligent conçu à Rennes*", interview de M. Babel, E. Leblong et F. Morbidi dans le cadre du projet ADAPT, par la journaliste du quotidien régional Ouest France, Nathalie Flochlay. L'interview a été publiée le 4 novembre 2020, `www.ouest-france.fr/bretagne/rennes-35000`

1.2 Enseignement

1.2.1 Cours dispensés

- À l'**Université de Picardie Jules Verne**, département EEA, j'ai dispensé des cours de niveau Licence et Master (parcours ViRob et RoVA : "Robotique et Vision Artificielle") pour des étudiants en alternance (contrat d'apprentissage ou de professionnalisation). L'effectif se situait entre 10 et 25 étudiants (en moyenne 15 étudiants). Tous mes cours sont disponibles en libre accès sur Internet à l'adresse suivante :
  `https://home.mis.u-picardie.fr/~fabio/Teaching.html`

- Année universitaire 2023-2024 : 192.75 UC (Prévisionnel)
  - Module de Master 2 "*Perception Avancée et Robotique Mobile*", parcours RoVA, semestre 9, hiver 2023-2024. CM, TD, TP.
  - Module de Licence Pro "*Initiation à la Robotique*", ME 1.1, parcours Robotique et Automatisme, avec Promeo, semestre 1, hiver 2023-2024. CM, TP.
  - Module de Master 2 "*Localisation et Navigation de Robots*", parcours RoVA, semestre 9, hiver 2023-2024. CM, TD, TP.

  - Module de Master 2 "*Systèmes Robotiques Hétérogènes et Coopératifs*", parcours RoVA, semestre 9, hiver 2023-2024. CM, TD, TP.
  - Module de Master 2 "*Projet Transversal - L'Usine du Futur : Industrie 4.0*", parcours RoVA, semestre 9, hiver 2023-2024. TP.
  - Module de Master 1 "*Robotique Industrielle*", parcours RoVA, semestre 8, printemps 2024. CM, TD.

- Année universitaire 2022-2023 : 229.62 UC

  - Module de Master 2 "*Perception Avancée et Robotique Mobile*", parcours RoVA, semestre 9, hiver 2022-2023. CM, TD, TP.
  - Module de Licence Pro "*Initiation à la Robotique*", ME 1.1, parcours Robotique et Automatisme, avec Promeo, semestre 1, hiver 2022-2023. CM.
  - Module de Master 2 "*Localisation et Navigation de Robots*", parcours RoVA, semestre 9, hiver 2022-2023. CM, TD, TP.
  - Module de Master 2 "*Surveillance Distribuée de Systèmes Multi-Agents*", parcours RoVA, semestre 10, printemps 2023. CM, TD, TP.
  - Module de Master 2 "*Systèmes Robotiques Hétérogènes et Coopératifs*", parcours RoVA, semestre 9, hiver 2022-2023. CM, TD, TP (mutualisé avec le parcours "SDD : Systèmes Distribués et Dynamiques" du Master mention Informatique de l'UPJV).
  - Module de Master 2 "*Projet Transversal - L'Usine du Futur : Industrie 4.0*", parcours RoVA, semestre 9, hiver 2022-2023. TP.
  - Module de Master 1 "*Robotique Industrielle*", parcours RoVA, semestre 8, printemps 2023. CM, TD.

- Année universitaire 2021-2022 : 216.75 UC

  - Module de Master 2 "*Perception Avancée et Robotique Mobile*", parcours RoVA, semestre 9, hiver 2021-2022. CM, TD, TP.
  - Module de Licence Pro "*Initiation à la Robotique*", ME 1.1, parcours Robotique et Automatisme, avec Promeo, semestre 1, hiver 2021-2022. CM/
  - Module de Master 2 "*Localisation et Navigation de Robots*", parcours RoVA, semestre 9, hiver 2021-2022. CM, TD, TP.
  - Module de Master 2 "*Surveillance Distribuée de Systèmes Multi-Agents*", parcours RoVA, semestre 9, hiver 2021-2022. CM, TD, TP.

  - Module de Master 2 "*Systèmes Robotiques Hétérogènes et Coopératifs*", parcours RoVA, semestre 9, hiver 2021-2022. CM, TD, TP.
  - Module de Master 2 "*Projet Transversal - L'Usine du Futur : Industrie 4.0*", parcours RoVA, semestre 9, hiver 2021-2022. TP
  - Module de Master 1 "*Robotique Industrielle*", parcours RoVA, semestre 8, printemps 2022. CM, TD.

- Année universitaire 2020-2021 : 95.25 UC (CRCT de 6 mois à partir du 1er février 2021)
  - Module de Licence Pro "*Initiation à la Robotique*", ME 1.1, parcours Robotique et Automatisme, avec Promeo, semestre 1, hiver 2020-2021. CM.
  - Module de Master 2 "*Localisation et Navigation de Robots*", parcours RoVA, semestre 10, hiver 2020-2021. CM, TD, TP.
  - Module de Master 2 "*Systèmes Robotiques Hétérogènes et Coopératifs*", parcours RoVA, semestre 10, hiver 2020-2021. CM, TD, TP.
  - Module de Master 2 "*Projet Transversal - L'Usine du Futur : Industrie 4.0*", parcours RoVA, semestre 9 et 10, hiver 2020 - printemps 2021. TP

- Année universitaire 2019-2020 : 226.25 UC
  - Module de Master 2 "*Perception Avancée et Robotique Mobile*", parcours RoVA, semestre 9, hiver 2019-2020. CM, TD, TP.
  - Module de Licence Pro "*Initiation à la Robotique*", ME 1.1, parcours Robotique et Automatisme, avec Promeo, semestre 1, hiver 2019-2020. CM.
  - Module de Master 2 "*Localisation et Navigation de Robots*", parcours RoVA, semestre 10, hiver 2019-2020. CM, TD.
  - Module de Master 2 "*Surveillance Distribuée de Systèmes Multi-Agents*", parcours RoVA, semestre 10, hiver 2019-2020. CM, TD, TP.
  - Module de Master 2 "*Systèmes Robotiques Hétérogènes et Coopératifs*", parcours RoVA, semestre 9, hiver 2019-2020. CM, TD, TP.
  - Module de Master 2 "*Projet Transversal - L'Usine du Futur : Industrie 4.0*", parcours RoVA, semestre 9 et 10, hiver 2019 - printemps 2020. TP
  - Module de Master 1 "*Robotique Industrielle*", parcours RoVA, semestre 8, printemps 2020. CM, TD.

- Année universitaire 2018-2019 : 215.25 UC
  - Module de Master 2 "*Perception Avancée et Robotique Mobile*", parcours RoVA, semestre 1, hiver 2018-2019. CM, TD, TP.

  - Module de Licence Pro "*Initiation à la Robotique*", ME 1.1, parcours Robotique et Automatisme, avec Promeo, semestre 1, hiver 2018-2019. CM.
  - Module de Master 2 "*Localisation et Navigation de Robots*", parcours RoVA, semestre 1, hiver 2018-2019. CM, TD.
  - Module de Master 2 "*Surveillance Distribuée de Systèmes Multi-Agents*", parcours RoVA, semestre 1, hiver 2018-2019. CM, TD, TP. À 50% avec M. Chadli.
  - Module de Master 2 "*Systèmes Robotiques Hétérogènes et Coopératifs*", parcours RoVA, semestre 1, hiver 2018-2019. CM, TD, TP. À 50% avec M. Chadli.
  - Module de Master 2 "*Projet Transversal - L'Usine du Futur : Industrie 4.0*", parcours RoVA, semestre 1 et 2, 2018-2019. TP
  - Module de Master 1 "*Robotique Industrielle*", parcours RoVA, semestre 2, printemps 2019. CM, TD.

- Année universitaire 2017-2018 : 201.5 UC

  - Module de Master 2 "*Perception Avancée et Robotique Mobile*", parcours ViRob, semestre 9, hiver 2017-2018. CM, TD, TP.
  - Module de Licence Pro "*Initiation à la Robotique*", ME 1.1, parcours Robotique et Automatisme, avec Promeo, semestre 1, hiver 2017-2018. CM.
  - Module de Master 2 "*Localisation et Navigation de Robots*", parcours ViRob, semestre 8, printemps 2018. CM, TD, TP.
  - Module de Master 1 "*Découverte de la Recherche*", parcours ViRob, semestre 8, printemps 2018. CM, TD.
  - Module de Master 1 "*Robotique Industrielle*", parcours ViRob, semestre 9, printemps 2018. CM.

- Année universitaire 2016-2017 : 194.5 UC

  - Module de Master 2 "*Perception Avancée et Robotique Mobile*", parcours ViRob, semestre 9, hiver 2016-2017. CM, TD, TP. À 50% avec D. Kachi.
  - Module de Licence Pro "*Initiation à la Robotique*", ME 1.1, parcours Robotique et Automatisme, avec Promeo, semestre 1, hiver 2016-2017. CM.
  - Module de Master 2 "*Localisation et Navigation de Robots*", parcours ViRob, semestre 8, printemps 2017. CM, TD, TP.
  - Module de Master 1 "*Découverte de la Recherche*", parcours ViRob, semestre 8, printemps 2017. CM, TD.

  - Module de Master 1 "*Robotique Industrielle*", parcours ViRob, semestre 9, printemps 2017. CM, TD, TP.

- Année universitaire 2015-2016 : 220 UC
  - Module de Master 2 "*Perception Avancée et Robotique Mobile*", parcours ViRob, semestre 9, hiver 2015-2016. CM, TD, TP. À 50% avec D. Kachi.
  - Module de Licence Pro "*Initiation à la Robotique*", ME 1.1, parcours Robotique et Automatisme, avec Promeo, semestre 1, hiver 2015-2016. CM.
  - Module de Licence Pro "*Phénomènes Dynamiques Engendrés par les Mouvements des Robots*", ME 4.2, parcours Robotique et Automatisme, en association avec Promeo, semestre 1, hiver 2015-2016. CM, TD, TP.
  - Module de Master 2 "*Localisation et Navigation de Robots*", parcours ViRob, semestre 8, printemps 2016. CM, TD, TP.
  - Module de Master 1 "*Découverte de la Recherche*", intitulé "Introduction aux Systémes Collaboratifs Multi-Agent", parcours ViRob, semestre 8, printemps 2016. CM, TD.
  - Module de Master 1 "*Robotique Industrielle*", parcours ViRob, semestre 9, printemps 2016. CM, TD, TP.

- Année universitaire 2014-2015 : 144 UC, décharge de 25% du service (48 UC)
  - Module de Master 2 "*Localisation et Navigation de Robots*", parcours ViRob, semestre 9, hiver 2014-2015. CM, TD, TP.
  - Module de Master 2 "*Perception Avancée et Robotique Mobile*", parcours ViRob, semestre 9, hiver 2014-2015. CM, TD, TP. À 50% avec D. Kachi.
  - Module de Master 1 "*Robotique Industrielle*", parcours ViRob, semestre 9, printemps 2015. CM, TD, TP.

▶ À la **Johannes Kepler University**, Institute for Design and Control of Mechatronical Systems, Linz, Autriche, j'ai dispensé des cours de niveau Licence et Master en anglais, pour un effectif de 10-15 étudiants :

  – Module de Master "*Distributed and Cooperative Control for Multi-agent Systems*", mars-juin 2012. Total : 30 heures.
  – Module de Licence "*Regelsysteme I*" (*Control systems I*), mars-juin 2012. Total : 15 heures.
  – Module de Licence "*Rechnerbasierter Entwurf von Regelkreisen*" (*Computer-based design of control systems*), octobre 2011-janvier 2012. Total : 15 heures.

1.2.2 Encadrement d'étudiants en Licence et Master

- Étudiants en alternance (Department EEA, UPJV) :
  - A.U. 2023-2024 : Un étudiant en Master 1, un étudiant en Master 2 et un étudiant en Licence Pro AutoRobo (Promeo-UPJV).
  - A.U. 2022-2023 : Deux étudiants en Master 2 et un étudiant en Licence Pro AutoRobo (Promeo-UPJV).
  - A.U. 2020-2021 : Un étudiant en Master 2 et un étudiant en Licence Pro AutoRobo (Promeo-UPJV).
  - A.U. 2019-2020 : Un étudiant en Master 1, un étudiant en Master 2 et un étudiant en Licence Pro AutoRobo (Promeo-UPJV).
  - A.U. 2018-2019 : Deux étudiants en Master 2.
  - A.U. 2017-2018 : Deux étudiants en Master 1, un étudiant en Master 2 et un étudiant en Licence Pro AutoRobo (Promeo-UPJV).
  - A.U. 2016-2017 : Un étudiant en Master 1, un étudiant en Master 2.
  - A.U. 2015-2016 : Deux étudiants en Master 1.
  - A.U. 2014-2015 : Un étudiant en Master 1.
- Stages au laboratoire MIS ou en entreprise :
  - Juliette Grosset (stagiaire à Inria Rennes) : Stage intitulé "*Caméra évènementielle pour l'aide à la conduite d'un fauteuil roulant électrique*", 9-13 mars 2020 (projet ADAPT). Encadrants à Inria : M. Babel, G. Vailland et V. Gouranton.
  - Ibrahima Gory (Master 1, Dépt. EEA, UPJV) : Stage intitulé "*Tracking vissage au défilé*", Segula Matra Automotive, Trappes, 16 avril 2018 - 28 septembre 2018. Tuteur d'entreprise : D. Lautram.
  - Pierrick Cédolin (DUT informatique, IUT Amiens) : 27 mars 2017 - 2 juin 2017. Co-encadrement (à 50%) avec E. Bretagne.
  - Abdallah Aguenaou (Master 2, Dépt. EEA, UPJV) : Stage chez EnerSys Sarl, Arras, avril-août 2016. Tuteur d'entreprise : L. Lucchini.
- Projets de Master (Département EEA, UPJV) :
  - Amath Ba et Lylia Djellouli (Master 1): Projet intitulé "*Planification probabiliste pour l'évitement d'obstacles: mise en œuvre de PRM et de RRT*", novembre 2023-juin 2024.

  - Thierry Huang (Master 1): Projet intitulé "*Détection de mouvement avec une caméra à événements*", novembre 2023-juin 2024.
  - Dyhia Immoune et Romane Huet (Master 1) : Projet intitulé "*Planification probabiliste pour l'évitement d'obstacles : mise en œuvre de PRM et de RRT*", octobre 2019-avril 2020.
  - Maxime Vambre et Timothé Deherre (Master 1) : Projet intitulé "*LumiBot : le robot artiste*", octobre 2019-avril 2020.
  - Julien Albrand et Méven Jeanne-Rose (Master 1) : Projet intitulé "*SpheriCol : réalité augmentée pour l'assistance à la conduite d'un fauteuil roulant électrique*", octobre 2018-avril 2019. Co-encadrement (à 50%) avec G. Caron.
  - Hakim Kadi et Ouardia Bourhala (Master 1) : Projet intitulé "*LumiBot : le robot artiste*", octobre 2018-avril 2019. Co-encadrement (à 50%) avec G. Caron.
  - Soukaina Benchchaoui et Salima Elbrahmi (Master 1) : Projet intitulé "*Conception d'un robot joueur de* Jungle Speed", octobre 2017-mars 2018.
  - Marwen Ben Hamada (Master 1) : Projet intitulé "*Cropbot : le robot agroglyphique*" octobre 2016-mars 2017. Co-encadrement (à 50%) avec G. Caron.
  - Laura Bailly et Damien Alary (Master 1) : Projet intitulé "*Robodoigts : Transport Coopératif d'Objets par Robots Mobiles*", octobre 2016-mars 2017.
  - Jean Hoffmann et Raphaël Plouard (Master 2) : Projet intitulé "*Lumidrone : le drone peintre*", octobre 2016-mars 2017. Co-encadrement (à 50%) avec G. Caron.
  - Jean Hoffmann et Chater Kawtar (Master 1) : Projet intitulé "*Système de localisation globale de robots mobiles basé vision*", octobre 2015-mars 2016.
  - Abdallah Aguenaou (Master 2) : Projet intitulé "*Odométrie assistée par centrale inertielle pour la localisation d'un robot mobile*", octobre 2015-mars 2016.

- Projets de fin d'études :

  - Zouhair Alaoui (Université Cadi Ayyad, Marrakesh, Maroc) : Projet intitulé "*Recalage dense de nuages de points 3D pour l'analyse comparative d'œuvres sculpturales*", janvier 2016-mars 2016.
    Co-encadrement avec le Prof. Hassan Ayad.

- Projets d'innovation pédagogique :

– En 2017-2018, j'ai participé au projet d'innovation pédagogique du parcours ViRob ("Vision Robotique") du Master 2 du département EEA de l'UPJV (2 UC). Le projet, intitulé "*L'Usine du Futur : Industrie 4.0*", a été coordonné par G. Caron. Sept membres du département EEA ont été impliqués dans ce projet, pour un budget total de 10.4 k€.

1.2.3 Responsabilités pédagogiques

- Responsable du parcours RoVA ("*Robotique et Vision Artificielle*") du Master 3EA, département EEA, UPJV, octobre 2018 - octobre 2020.

- Contact académique pour le programme Erasmus+ ("*Mobility for Learners and Staff*") entre l'UPJV et l'Università di Catania, Italie, 2020-aujourd'hui.

- J'ai fait partie du comité de recrutement de 10 ATER (section CNU 61) et d'un enseignant contractuel (LRU) au département EEA de l'UPJV : 19 juin 2019, 11 septembre 2019, 23 juin 2020, 14 juin 2021, 12 juillet 2021, 2 juin 2022, 4 juillet 2022, 23 septembre 2022.

Chapter 2

Introduction

The most exciting phrase to hear in science, the one that heralds new discoveries, is not "Eureka!" but "That's funny".

Isaac Asimov

Abstract

This introductory chapter presents the context and related work, and describes the general organization of this HDR thesis.

2.1 General context

Cooperative control of multi-agent systems is a popular research topic today, and it occupies a prominent place in robotics and systems theory (Parker et al. 2016, Cortés and Egerstedt 2017). The research in this area has been stimulated by major advances in sensing, control and wireless communications, and by the observation that multiple collaborative agents can perform tasks far beyond the capabilities of a single robot. A major effort has also been undertaken towards mimicking natural organisms, such as flocks of birds, schools of fish or swarms of insects (see Fig. 2.1), where self-organizing or emergent behaviors arise from agents which appear to act independently (Reynolds 1987, Jadbabaie et al. 2003, Vicsek and Zafeiris 2012, Strogatz 2012).

(a) (b) (c) (d)

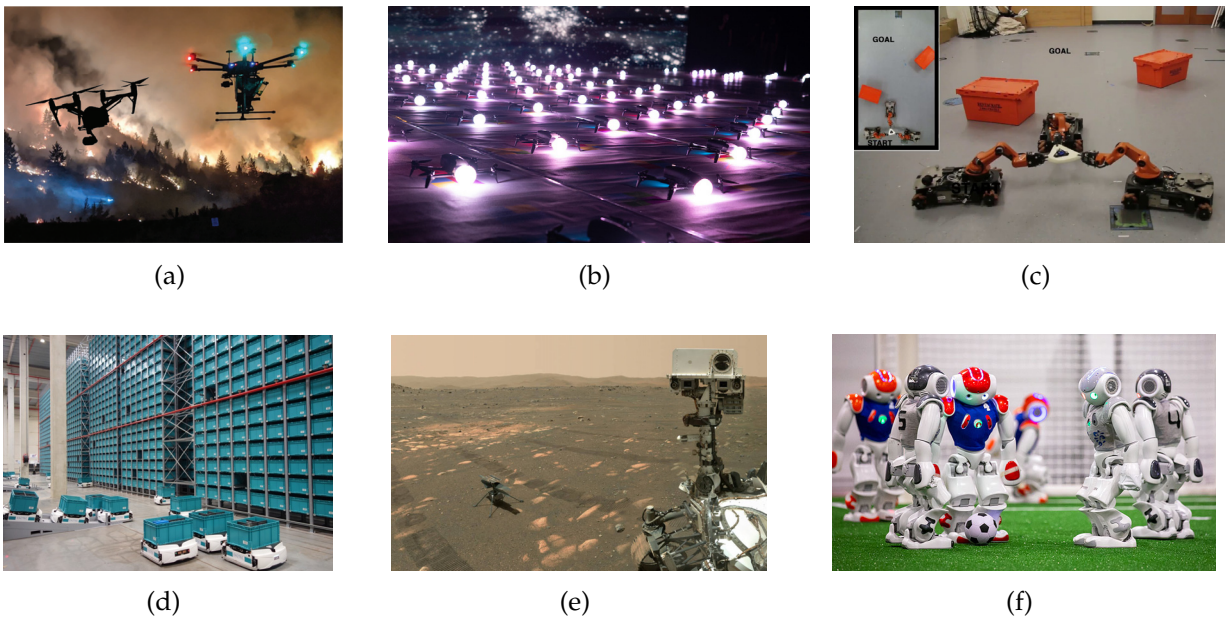
Figure 2.1: *Examples of biological systems exhibiting a cooperative behavior: (a) flock of geese, (b) school of fish, (c) swarm of bees, and (d) ant colony.*

Figure 2.2: *Books and monographs on multi-agent systems, published between 2009 and 2016.*

In the last two decades, several *books* and *monographs* dedicated to multi-agent systems have appeared in the literature (see Fig. 2.2). They have the great merit of having collected, systematized and consolidated a large body of research scattered in multiple domains: control systems, robotics, computer and network science, telecommunications, among others. These books cover different but complementary topics: the theoretical foundations (Shamma 2008, Bullo et al. 2009, Mesbahi and Egerstedt 2010, Fuhrmann and Helmke 2015, Fagnani and Frasca 2018, Hadjicostis et al. 2018, Bullo 2022), passivity-based estimation and control (Bai et al. 2011, Hatanaka et al. 2015), distributed optimization (Zhu and Martínez 2015), and optimal and adaptive control (Lewis et al. 2014). In (Francis and Maggiore 2016), the authors focused on the *rendezvous* and *flocking problems*, while (Queiroz et al. 2019, Ahn 2020, Roza et al. 2022) are devoted to the *formation control problem* (see also (Oh et al. 2015), for a recent authoritative survey on formation control). These prototypal problems have played an important role in the development of multi-agent systems theory, and they frequently serve, today, as benchmarks for testing new (distributed) coordination strategies.

In the recent past, mobile robot networks have found their way into numerous applications in the real world, as for example (see Fig. 2.3): terrain and utility inspection, disaster monitoring (Ghassemi and Chowdhury 2022), environmental surveillance and modeling (Casbeer et al. 2006, Susca et al. 2008, Salam and Hsieh 2023), ocean sampling (Leonard et al. 2007), traffic control, structures moving and assembling (Pereira et al. 2004, Alonso-Mora et al. 2017, Koung et al. 2021, Li, Ge and Loianno 2021, Hu et al. 2021), warehouse management (Wurman et al. 2008, Farinelli et al. 2017, Sabattini et al. 2017, Li and Ma 2023), military espionage and reconnaissance, mapping of unknown and hazardous environments (Hardouin et al. 2023, Simonin 2023–2027), planetary exploration, entertainment (e.g. drone-based light shows, robotic soccer (Parker et al. 2016)).

Research in multi-agent systems is not restricted to wheeled mobile robots (WMR), but deals with different typologies of vehicles at sea, in the air, and on land (see Fig. 2.4):



(a) (b) (c)

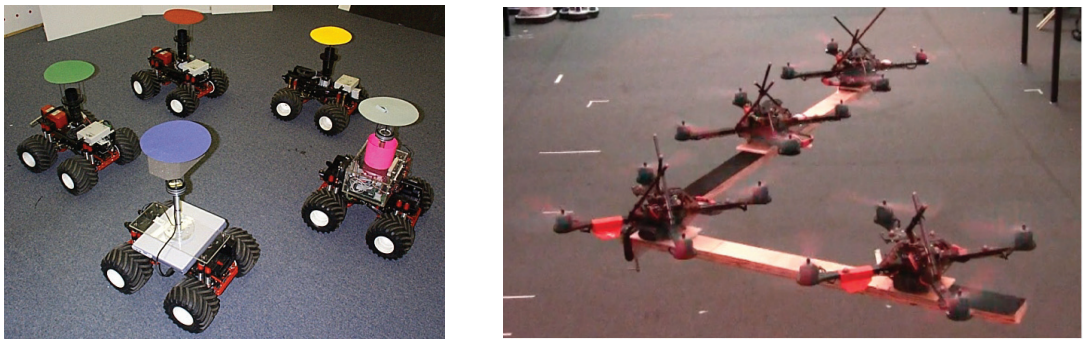
(d) (e) (f)

Figure 2.3: *Utilization of multi-robot systems: (a) Wildfire management with aerial robots (image courtesy of* Drone Amplified*); (b) drone-based light show*[2]*; (c) cooperative manipulation and assembly (image courtesy of* Autonomous Multi-Robots Lab*, Delft University of Technology); (d) automatic warehouse management (image courtesy of* Exotec*); (e) planetary exploration with a marsupial robotic system (Perseverance rover and Ingenuity helicopter, image courtesy of NASA* Jet Propulsion Laboratory*); (f) robotic soccer (RoboCup, Humanoid League, 2022). In each case, individual robots must decide on control actions that are conducive to accomplishing a collective task.*

autonomous underwater vehicles (AUVs) (Edwards et al. 2004, McConnell et al. 2022), unmanned aerial vehicles (UAVs, see Fig. 2.4(b)) (Hou and Fantoni 2017, McGuire et al. 2019, Yao et al. 2023), satellites (Massioni et al. 2010, Zhang and Gurfil 2016), humanoid robots (Gerndt et al. 2015), and micro robots (Seyfried et al. 2005, Ongaro et al. 2019). The latest trend goes towards larger and larger groups (from hundreds to thousands) of small, simple and expendable agents, so that the loss of a bunch of them during a mission does not have a critical impact on its achievement (Rubenstein et al. 2014, Saravanos et al. 2023).

Besides rendezvous, flocking and formation control mentioned above, other problems which have received special attention in the multi-agent systems literature are: *cyclic pursuit* (Marshall et al. 2004, Sinha and Ghose 2006, Pavone and Frazzoli 2007, Marshall and Broucke 2008, Morbidi et al. 2009), *containment control* and *herding* (Ji et al. 2008, Pierson and Schwager 2018, Chipade and Panagou 2023), *target encirclement*,

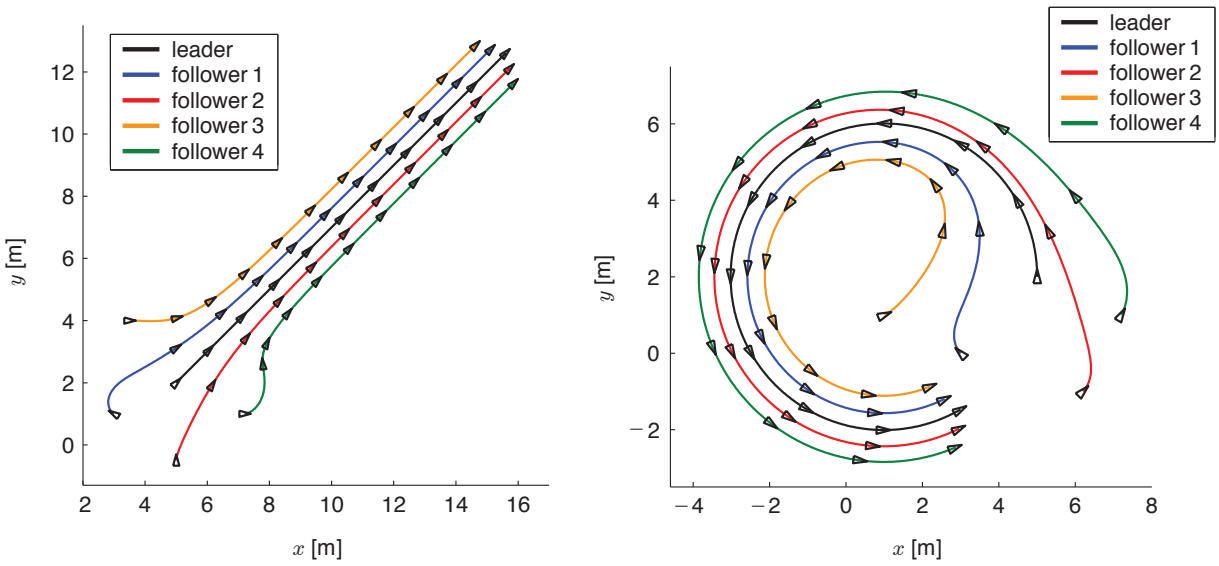
[2] `www.electricsky.io`



(a) (b)

Figure 2.4: *Team of (a) ground robots with colored markers, and (b) quadrotors transporting an L-shaped payload (images courtesy of GRASP lab, University of Pennsylvania).*

tracking or *entrapment* (Antonelli et al. 2008, Robin and Lacroix 2016, Franchi et al. 2016, López-Nicolás et al. 2020), *coverage* and *deployment* (Howard et al. 2002, Cortés et al. 2004, Santos et al. 2018, Mansouri et al. 2018), and *connectivity* and *visibility maintenance* (Meng and Egerstedt 2007, Dimarogonas and Kyriakopoulos 2008, Zavlanos et al. 2011, Gasparri et al. 2017).



(a) (b)

Figure 2.5: Leader-follower formation control (Morbidi et al. 2007): *Five unicycle robots achieve the desired wedge formation starting from a random initial configuration. In (a), the trajectory of the leader (solid black) is rectilinear, while in (b), is circular.*

For its wide range of applications, the formation control problem stands out in multi-agent systems research (Das et al. 2002, Tabuada et al. 2005, Dong and Farrell 2008, Mariottini et al. 2009, Consolini et al. 2012), and recent years have seen a growing effort towards the design of (consensus-based) *distributed* control strategies (Antonelli et al. 2014, García de Marina et al. 2016, Sun et al. 2018, Alonso-Mora et al. 2019, Stamouli et al. 2020, Fathian et al. 2021, Aranda et al. 2021, Nuño et al. 2022, Lin et al. 2022). Generally speaking, the formation control problem is the problem of controlling the relative pose (position and orientation) of the robots in a group while allowing the group to move as a whole to form a predefined pattern.

Classical approaches to formation control can be divided into three major categories: *behavior based*, *virtual structure* and *leader following*.

In the *behavior based* approach (Matarić 1992, Balch and Arkin 1998, Lawton et al. 2003, Antonelli et al. 2008) several desired behaviors (e.g. collision avoidance, formation keeping, target seeking) are prescribed to each robot. The final action of each robot is derived by weighting the relative importance of each behavior. The theoretical formalization and mathematical analysis of this approach is difficult and it is not easy to guarantee the convergence of the formation to a desired configuration, in general.

The *virtual structure* approach (Tan and Lewis 1997, Ren and Beard 2004, Belta and Kumar 2004) considers the robot formation as a single virtual rigid structure so that the behavior of the robotic system corresponds to that of a physical object. Desired trajectories are not assigned to each single robot but to the entire formation as a whole. In this case, the behavior of the robot formation is predictable and consequently the stability analysis can be easily performed. Nonetheless, a large inter-robot communication bandwidth is required.

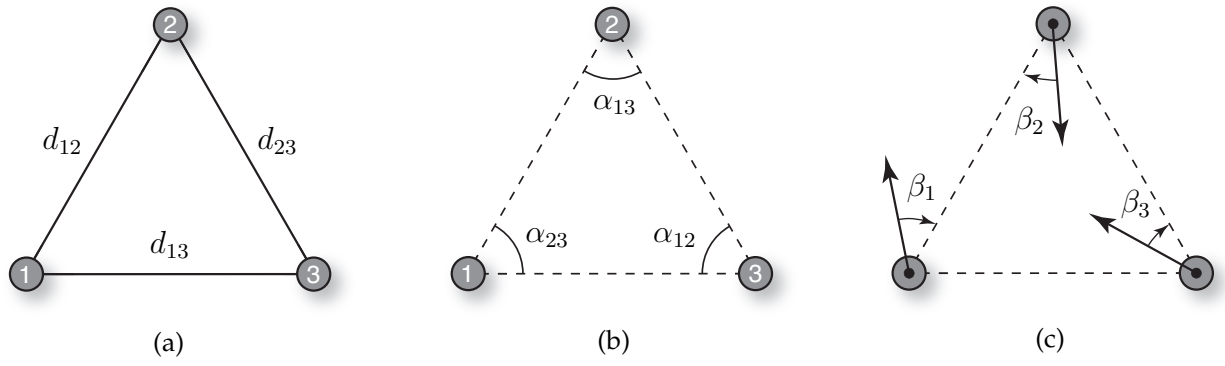


Figure 2.6: *Triangular formation of mobile robots (gray disks) defined by constraints on: (a) relative distance d_{ij}, (b) relative orientation α_{ij}, and (c) bearing angles β_j. A combination of distance and angular constraints is possible as well.*

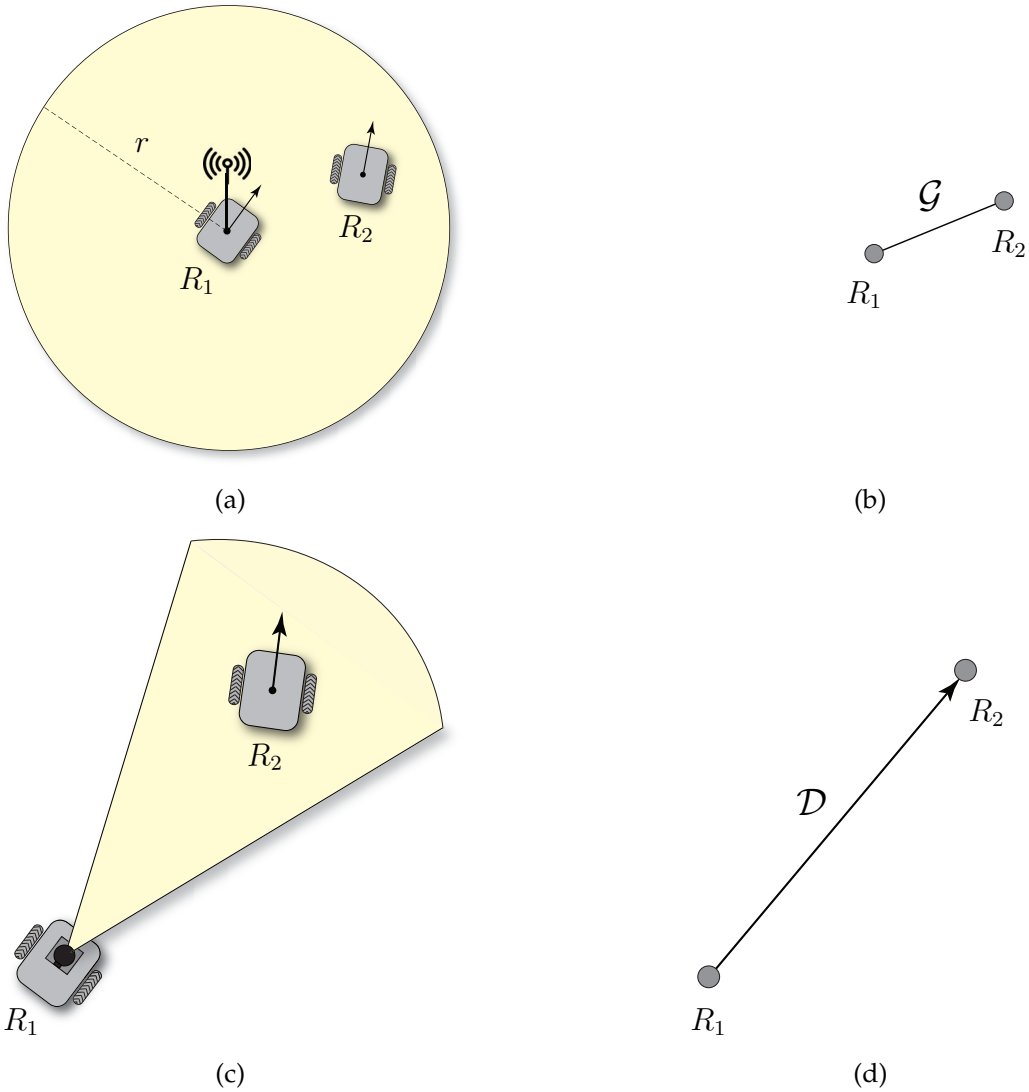


Figure 2.7: *Communication/sensory limitations and motion constraints: (a) Robot R_1 can communicate only with the robots which lie within a disk of radius r; (c) R_1 is equipped with a 2D Lidar with a limited field of view, and it can only sense the robots which lie within a narrow circular sector (line-of-sight constraint). In both cases, the robots (unicycles) cannot freely move around, but they are subject to kinematic constraints, that impose zero lateral velocity. In (b) and (d), the corresponding abstractions, in terms of undirected ($\mathcal{G}$) and directed ($\mathcal{D}$) graphs, are shown.*

In the *leader-follower* approach (see Fig. 2.5), a robot of the formation, designed as the leader, moves along a predefined trajectory while the others, the followers, are to maintain a desired pose (distance and orientation) to the leader (Das et al. 2002, Tanner et al. 2004, Léchevin et al. 2006, Gustavi and Hu 2008, Hou and Fantoni 2017, Zhao 2018,

Chen et al. 2020). A drawback of leader following is that the formation does not tolerate leader faults and exhibits poor disturbance rejection performances (this promoted the emergence of leaderless approaches: see for example (Aranda et al. 2023) and the references therein). In spite of these limitations, the leader-follower approach is highly valued by researchers, since it is simple and easy to implement. To show the idea behind leader following, two illustrative examples are reported in Fig. 2.5. Five unicycle robots, a leader and four followers, achieve a desired wedge formation (defined by pairs of distance and bearing constraints), starting from a random initial pose: in Fig. 2.5(a), the trajectory of the leader is rectilinear, while in Fig. 2.5(b), it is circular.

The type of *geometric constraints* to be enforced in a multi-robot system (e.g. relative distances, relative positions, relative orientations, bearing angles, a combination thereof, see Fig. 2.6), the *physical quantities to be preserved* during the desired collective maneuvers (e.g. position of the centroid, orientation, scale), and the *mobility* of the robots (linear robots such as single and double integrators versus vehicles with nonholonomic constraints (Kolmanovsky and McClamroch 1995), such as unicycle and car-like robots), obviously play an important role in the classification of formation control strategies.

When it comes to real environments and real robots, a cooperative control strategy operates in degraded conditions and contends with a number of practical issues. So far, we have not made any specific assumption on the capabilities of the robots:

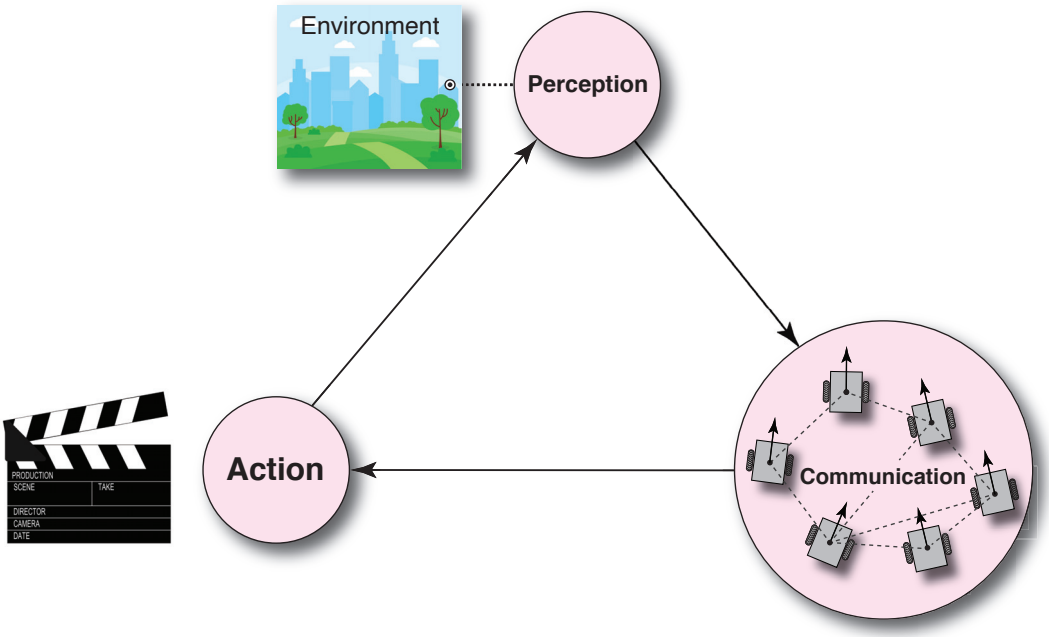


Figure 2.8: *"Perception-Action-Communication" (PAC) loop. The robots must perceive their local environment, communicate with the other robots, and take actions in real time.*

however, real robots de have *communication* and *sensing limitations*. For example, two robots can communicate only if the first lies within a circular region (representing the approximate extent of the electromagnetic field used for data exchange), centered on the second (see Fig. 2.7(a)). Moreover, communication is not instantaneous, and the channel might be noisy. Analogously, a robot can only detect an obstacle or another vehicle which lie within a circular sector corresponding to the footprint of a range sensor with limited field of view (e.g. a 2D Lidar or an RGB-D camera, see Fig. 2.7(c)). Graph theory is a useful means of abstracting these constraints: in fact, a robot can be associated with a node of a graph, and an edge will be present between two nodes if a communication link (or visual contact) has been established among them. More specifically, undirected (directed) edges can be used to model bidirectional (unidirectional) interactions or information exchanges among the robots (see Fig. 2.7(b) and Sect. 3.1). *Kinematic constraints*, *dynamic constraints* and *actuation noise* also limit the set of (instantaneous) feasible motions of a robot, and they cannot be ignored since they have a critical impact on the achievement and accuracy of a given mission.

The link between *perception* and *action* in a multi-robot system is not trivial, especially in unstructured or unknown environments (see Fig. 2.8). This calls for a deep understanding of the interplay between perception, communication and action, in order to endow robots with effective and robust (distributed) control laws. This challenging problem will take central stage in this manuscript, and several original solutions, motivated by different application scenarios, will be proposed, as detailed in Sect. 2.2.

2.2 Organization

The remainder of this manuscript is organized as follows (see the plan in Table 2.1):

- **Chapter 3** introduces some basic notions of algebraic graph theory, and presents the continuous- and discrete-time consensus protocols, and the most popular linear and nonlinear models of a mobile robot, for later use in Chapters 4 and 5.

- **Chapter 4** is devoted to the *formation control problem* and three different incarnations are studied. The chapter starts with a new characterization of mobility for distance-bearing formations of unicycle robots, and it then introduces new decentralized optimal strategies for cooperative adaptive cruise control of a car platoon, under string-stability constraints. The chapter concludes with a description of functions of Laplacian matrix, and their application, for the first time, to shape-based distributed formation control.

Chapter 3: Background	• 3.1: Graph theory • 3.2: Consensus protocol • 3.3: Mobile robot models
Chapter 4: Formation control	• 4.1: Mobility of unicycle formations • 4.2: Cooperative adaptive cruise control • 4.3: **Functions of Laplacian matrix**
Chapter 5: Part I Cooperative tasks with UAVs	• 5.1: **Environmental monitoring** • 5.2: Active target tracking • 5.3: 3D reconstruction
Chapter 5: Part II Extensions of the consensus protocol	• 5.4: **Deformed consensus** • 5.5: **State-constrained consensus**

Table 2.1: *Plan of the core of the HDR thesis and dependencies between Chapters 4 and 5 (in boldface).*

- **Chapter 5** presents several original contributions pertaining to the coordination of multiple collaborative robots. Compared to Chapter 4, we are interested here in designing *distributed control laws* based on the consensus protocol, to perform a variety of cooperative tasks: environmental monitoring, active target tracking, and 3D reconstruction of unknown environments. These tasks are performed by aerial robots and require interaction with the surrounding environment. In the second half of the chapter, two original extensions of the standard consensus protocol are presented: we introduce a generalization of combinatorial Laplacian which provides more flexibility in the execution of coordinated tasks (e.g. rendezvous in the 2D space). Finally, a geometric formulation which allows to naturally handle state constraints in the discrete-time consensus protocol while preserving the consensus value, is introduced and its pros and cons are discussed.

- **Chapter 6** closes the manuscript with a critical account of original contributions and with a discussion of promising directions for future work.

In order to keep the exposition simple, the proofs in Chapters 3, 4 and 5 are omitted: they can be found in the references listed in the bibliography.

2.3 Acknowledgements

The research work presented in this manuscript is not the result of the effort of a single person, but it has been made possible by multiple exchanges and fruitful discussions with several collaborators in France and abroad, with whom I had the honor to interact during my post-docs and during my stay at the UPJV. I accumulated a significant set of debts with them. In particular, I would like to express my sincere gratitude to the following individuals (in chronological order): G.L. Mariottini (Draper Laboratory, USA), K.M. Lynch and R.A. Freeman (Northwestern University, USA), P. Colaneri (Politecnico di Milano, Italy), J. Marzat and J. Moras (ONERA DTIS, France).

On a par with my research activity, the constructive comments of the students of my Master 2's courses "*Perception Avancée et Robotique Mobile*", "*Localisation et Navigation de Robots*", "*Systèmes Robotiques Hétérogènes et Coopératifs*" and "*Surveillance Distribuée de Systèmes Multi-agents*" that I have taught at the EEA department, UPJV, since September 2018, have been a continuous source of motivation and inspiration for me. These lectures also gave me the opportunity to collect and organize knowledge in a concise and accessible form, and to keep it updated with the latest trends in research and technology. I would like to thank F. Collet and A. Potelle, heads of EEA department, for the excellent working environment and the freedom they gave me in the preparation of my classes.

From a historical standpoint, it is worth recalling here that the Robotic Perception group at the UPJV, was a pioneer in the field of multi-robot systems in the early 90s (Pégard et al. 1990, Hutin et al. 1998), and this heritage is still very much alive today.

Finally, for the research reported in Chapters 4 and 5, we gratefully acknowledge support from the project "*Automatic Compilation of Global Objectives into Local Controllers for Cooperative Mobile Agents*" (2008-2012) and from the ScanBot project "*Scanners RoBotisés pour la Numérisation Automatique du Patrimoine*" (2018-2021), cf. Sect. 1.1.5.

2.4 Glossary

Symbols and abbreviations

a, α, x	Scalars
$\mathbf{a}$, $\boldsymbol{\alpha}$, $\mathbf{x}$	Column vectors
$\mathbf{A}$, $\mathbf{X}$	Matrices or linear transformations
$\mathcal{A}$, $\mathcal{X}$	Sets, subspaces or manifolds
$\mathcal{G}$, $\mathcal{D}$	Undirected, directed graph
$\mathbb{N}$	Set of natural numbers
$\mathbb{Z}$	Set of integers
$\mathbb{Z}_{>0}$, $\mathbb{Z}_{\geq 0}$	Set of positive (non-negative) integers
$\mathbb{R}$	Set of real numbers
$\mathbb{R}_{>0}$, $\mathbb{R}_{\geq 0}$	Set of positive (non-negative) real numbers
$\mathbb{C}$	Set of complex numbers
$\mathbb{N}^n$	Set of n-tuples of natural numbers
$\mathbb{S}^n$	Unit n-sphere
$\mathbb{R}^n$	Set of n-tuples of real numbers
$\mathbb{R}^{m \times n}$	Set of $m \times n$ matrices with real entries
$\mathrm{SO}(n)$	Special orthogonal group in dimension n
$\mathrm{SE}(n)$	Special Euclidean group in dimension n
$[a, b]$	Closed interval with endpoints a and b
(a, b)	Open interval with endpoints a and b
$f \in \boldsymbol{\mathcal{C}}^m$	Function f is of differentiability class $\boldsymbol{\mathcal{C}}^m$
$\mathrm{Re}(z)$, $\mathrm{Im}(z)$, $\arg(z)$	Real part, imaginary part, argument of complex number z
$\mathrm{sgn}(x)$	Sign function of real number x
$\lvert\mathcal{A}\rvert$	Cardinality of set $\mathcal{A}$ (i.e. number of elements of $\mathcal{A}$)
$\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{k=1}^{n} x_i \, y_i$	Scalar or dot product of vectors $\mathbf{x}$, $\mathbf{y} \in \mathbb{R}^n$
$\lVert\mathbf{x}\rVert_2 = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle}$	Euclidean norm of vector $\mathbf{x}$
$\mathcal{A} \times \mathcal{B}$	Cartesian product of sets $\mathcal{A}$ and $\mathcal{B}$
$\mathbb{1}$, $\mathbb{1}_n$	Vector of n ones
$\mathbb{k}$	Vector of n ones with alternating sign, $[-1, 1, \ldots, (-1)^{n-1}, (-1)^n]^T$
$\mathbf{I}_{n \times n}$, $\mathbf{I}_n$	Identity matrix of size $n \times n$
$\mathbf{0}_{m \times n}$, $\mathbf{0}_n$	Matrix of zeros of size $m \times n$ and of size $n \times n$

$\mathrm{diag}(a_1, \ldots, a_n)$	Diagonal matrix with entries $a_1, \ldots, a_n$ on the main diagonal
$\mathrm{blkdiag}(\mathbf{A}_1, \ldots, \mathbf{A}_n)$	Block diagonal matrix with square matrices $\mathbf{A}_1, \ldots, \mathbf{A}_n$ on the main diagonal
$[\mathbf{A}]_{ij}$	Entry (i, j) of matrix $\mathbf{A}$
$\mathbf{A}^T$	Transpose of matrix $\mathbf{A}$
$\mathbf{A}^H = \overline{\mathbf{A}}^T$	Conjugate (or Hermitian) transpose of matrix $\mathbf{A}$
$\mathbf{A}^{-1}$	Inverse of matrix $\mathbf{A}$ (square and invertible)
$\mathbf{A}^\dagger$	Pseudo-inverse of matrix $\mathbf{A}$
$\mathbf{A}^+$	Moore-Penrose inverse of matrix $\mathbf{A}$
$\mathbf{A}^\#$	Group generalized inverse of matrix $\mathbf{A}$
$\|\mathbf{A}\|_2$	Spectral norm of matrix $\mathbf{A}$
$\|\mathbf{A}\|_F$	Frobenius norm of matrix $\mathbf{A}$
$\log \mathbf{A}$	Principal logarithm of matrix $\mathbf{A}$
$\mathrm{Im}\, \mathbf{A}$	Image of matrix $\mathbf{A}$
$\ker \mathbf{A}$	Kernel of matrix $\mathbf{A}$
$\mathrm{nul}\, \mathbf{A}$	Nullity of matrix $\mathbf{A}$ (i.e. dimension of kernel of $\mathbf{A}$)
$\mathrm{tr}\, \mathbf{A}$	Trace of matrix $\mathbf{A}$
$\det \mathbf{A}$	Determinant of matrix $\mathbf{A}$
$\mathrm{spec}(\mathbf{A})$	Spectrum of matrix $\mathbf{A}$ (ignoring algebraic multiplicity)
$p_{\mathbf{A}}(\lambda)$	Characteristic polynomial of matrix $\mathbf{A}$
$\lambda_k(\mathbf{A})$, $\lambda_{\min}(\mathbf{A})$, $\lambda_{\max}(\mathbf{A})$	k-th, minimum, maximum eigenvalue of matrix $\mathbf{A}$
$\mathrm{span}\{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$	Linear span of vectors $\mathbf{x}_1, \ldots, \mathbf{x}_m$
$\mathbf{x} \perp \mathbf{y}$	Vectors $\mathbf{x}$, $\mathbf{y}$ are orthogonal (i.e. $\langle \mathbf{x}, \mathbf{y} \rangle = 0$)
$\mathbf{R}_x(\gamma)$, $\mathbf{R}_y(\beta)$, $\mathbf{R}_z(\alpha)$	Elementary 3×3 rotation matrices about the x-, y- and z-axis of an angle γ, β and α
$\mathbf{A} \otimes \mathbf{B}$	Kronecker product of matrices $\mathbf{A}$ and $\mathbf{B}$
$\mathcal{V}^\perp$	Orthogonal complement of linear subspace $\mathcal{V} \subset \mathbb{R}^n$
$\mathcal{U} \oplus \mathcal{V}$	Direct sum of linear subspaces $\mathcal{U}$, $\mathcal{V} \subset \mathbb{R}^n$
$\succ$, $\succeq$	Matrix inequality in the positive definite, positive semidefinite sense
$\lfloor \cdot \rfloor$, $\lceil \cdot \rceil$	Floor, ceiling function
$\triangleq$	Equality by definition
$\blacksquare$, $\diamond$	End of discussion

Acronyms

AUV	Autonomous Underwater Vehicle
CACC	Cooperative Adaptive Cruise Control
DoF	Degree of Freedom
EKF	Extended Kalman Filter
ESDF	Euclidean Signed Distance Field
FoV	Field of View
GNN	Graph Neural Network
GPU	Graphics Processing Unit
GSO	Graph Shift Operator
GVLD	Generalized Vehicle Longitudinal Dynamic
ICR	Instantaneous Center of Rotation
IMU	Inertial Measurement Unit
LQR	Linear Quadratic Regulation
MCTS	Monte-Carlo Tree Search
MPC	Model Predictive Control
NBV	Next Best View
PAC	Perception Action Communication
PRM	Probabilistic Roadmap
QEP	Quadratic Eigenvalue Problem
RDE	Riccati Differential Equation
RGB-D	Red Green Blue - Depth
ROS	Robot Operating System
RRT	Rapidly exploring Random Tree
SLAM	Simultaneous Localization and Map Building
SVD	Singular Value Decomposition
SWaP	Size Weight and Power
SWC	Singular Wheels Configuration
TSDF	Truncated Signed Distance Function
TSP	Traveling Salesman Problem
UAV	Unmanned Aerial Vehicle
WMR	Wheeled Mobile Robot

Chapter 3

Background material

The purpose of abstraction is not to be vague but to create a new semantic level in which one can be absolutely precise.

Edsger W. Dijkstra

Abstract

This chapter provides relevant background material on graph theory, distributed control and mobile robotics for later use in Chapters 4 and 5.

In this chapter, we first recall some notions of algebraic and spectral graph theory (Sect. 3.1), and then review the classical continuous-time and discrete-time consensus protocols for undirected communication graphs (Sect. 3.2), which are instrumental in designing distributed control and estimation algorithms. Finally, in Sect. 3.3, we briefly review the most popular linear and nonlinear models of a mobile robot.

3.1 Graph theory

3.1.1 Undirected and directed graphs

Let $\mathcal{G} = (V, E)$ be an *undirected graph* without self-loops and multiple edges, where $V = \{1, 2, \ldots, n\}$ is the set of nodes or vertices and $E \subseteq V \times V$ is the set of edges or arcs[1], where the cardinality of E is m. This definition applies to *simple* (or *schlicht*) *graphs* and it is not valid for multi-graphs, e.g. graphs which are permitted to have multiple edges (edges that have the same end nodes) or loops (Godsil and Royle 2001). In the following, we will assume that $\mathcal{G}$ is *unweighted* and *connected*, i.e. there is a path from any node to any other node in the graph. Two categories of graphs of interest are *bipartite* and *regular graphs*.

[1]The terms, *node*, *vertex*, and *edge*, *arc*, will be used interchangeably throughout the manuscript.

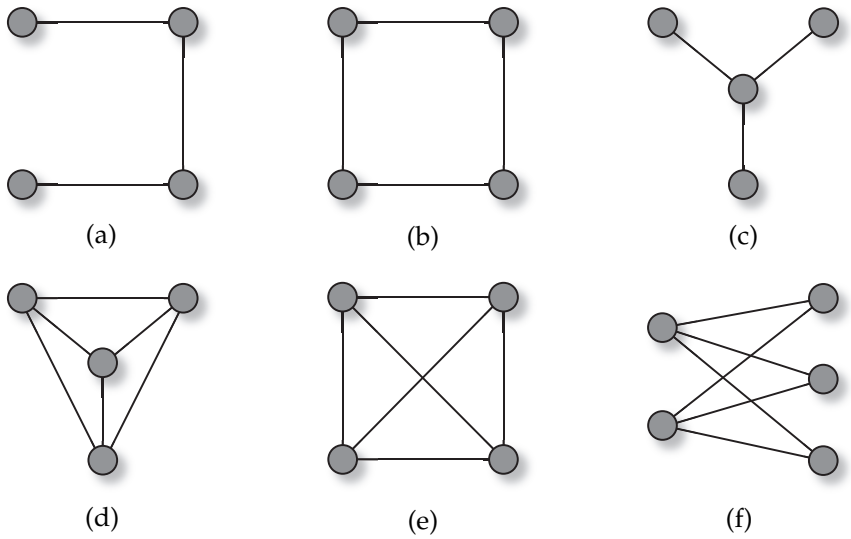


Figure 3.1: Special families of undirected graphs: *(a) Path graph P_4, (b) cycle graph C_4, (c) star graph S_4, (d) wheel graph W_4, (e) complete graph K_4, and (f) complete bipartite graph $K_{2,3}$. Graph C_4 is 2-regular and graphs W_4 and K_4 are 3-regular.*

3.1. DEFINITION (BIPARTITE GRAPH). *A graph $\mathcal{G}$ is called bipartite if its set of nodes V can be divided into two disjoint sets V_1 and V_2, such that every edge connects a node in V_1 to one in V_2. If $\mathcal{G}$ is connected, its bipartition $\{V_1, V_2\}$ is unique.* ◇

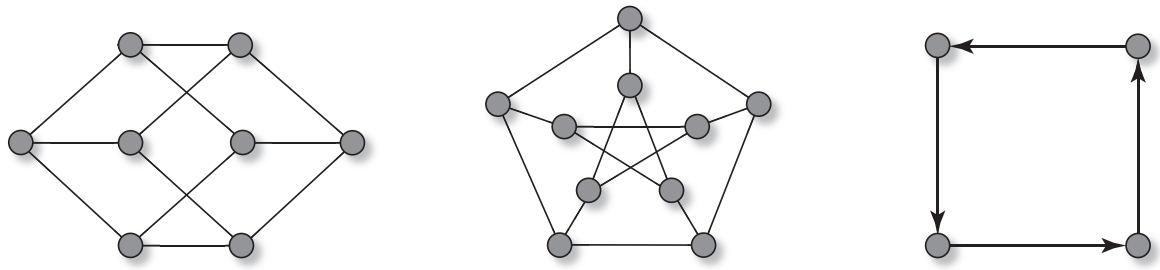
Equivalently, a graph is bipartite if and only if it does not contain cycles of odd length. A *complete bipartite graph* is a special kind of bipartite graph where every node of V_1 is connected to every node of V_2.

3.2. DEFINITION (κ-REGULAR GRAPH). *A graph $\mathcal{G}$ is called κ-regular, when every node has precisely $\kappa \leq n-1$ neighbors.* ◇

The following families of graphs will recur throughout this HDR thesis.

3.3. DEFINITION (SPECIAL FAMILIES OF UNDIRECTED GRAPHS). *The following symbols are conventionally used to denote special families of undirected graphs (see Fig. 3.1):*

- *P_n : Path graph with n nodes*
- *C_n : Cycle graph with n nodes*
- *$S_n \equiv K_{1,n-1}$: Star graph with n nodes*
- *W_n : Wheel graph with n nodes*
- *K_n : Complete graph with n nodes*



(a) (b) (c)

Figure 3.2: Special families of graphs: *(a) Hypercube graph Q_3, (b) Petersen graph, and (c) directed cycle graph D_4. The hypercube and Petersen graphs are 3-regular.*

- *$K_{p,n}$: Complete bipartite graph with $p + n$ nodes*
- *Q_n : n-cube (or hypercube graph).* ◇

The Petersen graph, $J(5, 2, 0)$, is a small graph that serves as a useful example and counterexample in many problems in graph theory[2]. It is a 3-regular graph with 10 vertices and 15 edges (Godsil and Royle 2001, Sect. 1.6), see Fig. 3.2(b).

Similarly to an undirected graph, a *directed graph* (or *digraph*, for short) $\mathcal{D} = (V, E)$ consists of a set of nodes $V = \{1, \ldots, n\}$ and a set of edges $E \subseteq V \times V$. Each edge is associated with a pair of nodes according to a direction represented by an arrow. We recall here the following important definition (Mesbahi and Egerstedt 2010):

3.4. DEFINITION (ROOTED OUT-BRANCHING). *A directed graph $\mathcal{D} = (V, E)$ is a rooted out-branching (or a directed rooted tree) if:*

1. *It does not contain a directed cycle.*
2. *It has a node v_{R} (root) such that for every other note $v \in V$, there is a directed path from v_{R} to v.* ◇

The directed counterpart of the cycle graph C_n is the *directed cycle* with n nodes, which we shall denote by D_n (see Fig. 3.2(c) and cf. Sect. 5.4).

3.1.2 Matrices associated with a graph

A graph $\mathcal{G}$ can be uniquely defined by using the list of its nodes and edges, but also by means of matrices of appropriate size. We will review here a few well-known matrices in graph theory (adjacency, incidence, degree and Laplacian matrices) and others that

[2]The Petersen graph belongs to the family of Kneser graphs, $J(i, k, 0)$, where $i, k \in \mathbb{Z}_{>0}$ and $i \geq k$.

have been recently introduced in the literature. By borrowing a term used in graph signal processing (Sandryhaila and Moura 2014, Gama et al. 2020), we can collectively refer to them as *Graph Shift Operators* (GSOs). More specifically, a GSO is a matrix $\mathbf{H} \in \mathbb{R}^{n \times n}$ that encodes the sparsity pattern of $\mathcal{G}$, whose entry $[\mathbf{H}]_{ij}$ can be nonzero only if $i = j$ or if $\{i,\, j\} \in E$.

To keep our presentation succinct, henceforth we focus on *undirected graphs*: the extension to directed graphs is straightforward as long as the orientation of the edges is taken into account.

3.5. DEFINITION (ADJACENCY MATRIX $\mathbf{A}$). *The adjacency matrix* $\mathbf{A}$ *of graph* $\mathcal{G}$ *is an* $n \times n$ *matrix defined as*

$$[\mathbf{A}]_{ij} = \begin{cases} 1 & \text{if } \{i,\, j\} \in E, \\ 0 & \text{otherwise.} \end{cases}$$

◇

3.6. DEFINITION (INCIDENCE MATRIX $\mathbf{B}$). *Let* $\mathcal{G}^o$ *the directed graph obtained by associating an arbitrary orientation with the edges of* $\mathcal{G}$*. The incidence matrix* $\mathbf{B}(\mathcal{G}^o)$ *is an* $n \times m$ *matrix, where* n *is the number of nodes of* $\mathcal{G}$ *and* m *is the number of edges of* $\mathcal{G}$*. The entry of the incidence matrix in row* i *and column* j *is defined as follows*

$$[\mathbf{B}(\mathcal{G}^o)]_{ij} = \begin{cases} -1 & \text{if } v_i \text{ is the tail of edge } e_j, \\ 1 & \text{if } v_i \text{ is the head of edge } e_j, \\ 0 & \text{otherwise.} \end{cases}$$

◇

We can notice that $\mathbf{B}(\mathcal{G}^o)$ captures not only the adjacency relationships in the graph, but also the orientation that the graph now enjoys.

3.7. DEFINITION (LAPLACIAN MATRIX $\mathbf{L}$). *The Laplacian matrix of graph* $\mathcal{G}$ *is an* $n \times n$ *symmetric positive semidefinite matrix defined as*

$$\mathbf{L} = \mathbf{D} - \mathbf{A},$$

where $\mathbf{D} = \text{diag}(\mathbf{A}\mathbb{1})$ *is the degree matrix of* $\mathcal{G}$ *and* $\mathbb{1} = [1, 1, \ldots, 1]^T \in \mathbb{R}^n$*.* ◇

From the previous definition, we deduce that for any graph $\mathcal{G}$, the row-sums of the Laplacian matrix are equal to zero (or in compact form, $\mathbf{L}\mathbb{1} = \mathbf{0}$), and that the off-diagonal entries of $\mathbf{L}$ are non-positive. Since $\mathbf{L}$ is a symmetric matrix, we have $\mathbf{L}\mathbf{v}_i = \lambda_i \mathbf{v}_i$, $i \in \{1, 2, \ldots, n\}$, where the eigenvalues $\{\lambda_1,\, \lambda_2, \ldots, \lambda_n\}$ of $\mathbf{L}$ are arranged in increasing order, i.e. $0 = \lambda_1 < \lambda_2 \leq \ldots \leq \lambda_n$ and $\{\mathbf{v}_1,\, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$ are the associated orthonormal eigenvectors, with $\mathbf{v}_1 = \mathbb{1}/\sqrt{n}$.

Given an arbitrary orientation of the set of edges E, the Laplacian matrix of the graph $\mathcal{G}$ can be alternatively defined by multiplying the incidence matrix $\mathbf{B}$ with its transpose

$$\mathbf{L}(\mathcal{G}) \;=\; \mathbf{B}(\mathcal{G}^o)\mathbf{B}^T(\mathcal{G}^o).$$

3.8. DEFINITION (REDUCED LAPLACIAN MATRIX $\mathbf{L}^\star$). *Let Orth$(\mathbb{1})$ denote the collection of $n\times(n-1)$ matrices $\mathbf{S}$ such that $\mathbf{S}^T\mathbf{S}=\mathbf{I}_{n-1}$, where $\mathbf{I}_{n-1}$ is the $(n-1)\times(n-1)$ identity matrix and $\mathbf{S}^T\mathbb{1}=\mathbf{0}_{(n-1)\times 1}$. Select a matrix $\mathbf{S}\in$ Orth$(\mathbb{1})$. The reduced Laplacian matrix of graph $\mathcal{G}$ is the $(n-1)\times(n-1)$ symmetric matrix*

$$\mathbf{L}^\star \;=\; \mathbf{S}^T\,\mathbf{L}\,\mathbf{S}.$$

◇

It is easy to verify that if the graph $\mathcal{G}$ is connected, then $\mathbf{L}^\star$ is positive definite and the spectrum of $\mathbf{L}^\star$ is strictly included in that of $\mathbf{L}$, i.e. spec$(\mathbf{L}^\star)\subset$ spec$(\mathbf{L})$.

Depending on the application, several variants of the Laplacian matrix $\mathbf{L}$ (a.k.a. combinatorial Laplacian) have appeared in the literature. We will not touch here on the popular *symmetric normalized Laplacian* $\mathbf{L}^{\text{sym}}=\mathbf{D}^{-1/2}\,\mathbf{L}\,\mathbf{D}^{-1/2}$ and *random-walk normalized Laplacian*, $\mathbf{L}^{\text{rw}}=\mathbf{D}^{-1}\,\mathbf{L}$ (Brouwer and Haemers 2012, Sect. 3.13), but we will focus on three variants that have received far less attention by the research community: the *signless Laplacian* $\mathbf{Q}$, the *parametric Laplacian* $\mathbf{L}_p(s)$, and the *deformed Laplacian* $\mathbf{\Delta}(s)$.

3.9. DEFINITION (SIGNLESS LAPLACIAN MATRIX $\mathbf{Q}$ (CVETKOVIĆ ET AL. 2007)). *The signless Laplacian matrix of graph $\mathcal{G}$ is an $n\times n$ matrix defined as*

$$\mathbf{Q} \;=\; \mathbf{D} \;+\; \mathbf{A}.$$

◇

As $\mathbf{L}$, the signless Laplacian $\mathbf{Q}$ is a symmetric positive semidefinite matrix, but it is *not* necessarily singular.

3.1. PROPERTY (SPECTRAL PROPERTIES OF $\mathbf{Q}$ (GOLDBERG AND KIRKLAND 2014)).

1. *Let $\mathcal{G}$ be a κ-*regular graph*. Then $p_{\mathbf{L}}(\lambda)=(-1)^n\,p_{\mathbf{Q}}(2\kappa-\lambda)$ where $p_{\mathbf{L}}(\lambda)$ denotes the characteristic polynomial of Laplacian $\mathbf{L}$. If $\mathcal{G}$ is a* bipartite graph*, then we have $p_{\mathbf{L}}(\lambda)=p_{\mathbf{Q}}(\lambda)$.*

2. *The least eigenvalue of $\mathbf{Q}$ of a connected graph is equal to 0 if and only if the graph is* bipartite*. In this case, 0 is a simple eigenvalue and the corresponding eigenvector is $\mathbf{v}=[v_1,v_2,\ldots,v_n]^T$ where $v_i=1$, $i\in V_1$ and $v_j=-1$, $j\in V_2$, being $\{V_1,V_2\}$ the bipartition of $\mathcal{G}$.*

3. *In any graph, the multiplicity of the eigenvalue 0 of $\mathbf{Q}$ is equal to the number of* bipartite connected components *of $\mathcal{G}$.* ◇

In a sense, the least eigenvalue of $\mathbf{Q}$ can be interpreted as a measure of how close $\mathcal{G}$ is to being a bipartite graph. A similar idea has been utilized for the second-smallest eigenvalue of $\mathbf{L}$, $\lambda_2(\mathbf{L})$, which is known as the *algebraic connectivity* (or *Fiedler value*) of $\mathcal{G}$ (Brouwer and Haemers 2012): the associated eigenvector, $\mathbf{v}_2$, is referred to as *Fiedler vector* of $\mathcal{G}$. For this reason, in (Fallat and Fan 2012), the least eigenvalue of $\mathbf{Q}$, is called the *algebraic bipartiteness* of $\mathcal{G}$.

In certain applications, it is desirable to smoothly change the structure of a GSO, by acting on a single parameter. To the best of our knowledge, the simplest example of such a GSO is the so-called *parametric Laplacian*

3.10. DEFINITION (PARAMETRIC LAPLACIAN $\mathbf{L}_p(s)$ (MORBIDI 2014)). *The parametric Laplacian matrix of graph $\mathcal{G}$ with parameter $s \in \mathbb{R}$, is an $n \times n$ matrix defined as*

$$\mathbf{L}_p(s) \;=\; \mathbf{D} \;-\; s\mathbf{A}.$$

◇

According to Definition 3.10, it is clear that $\mathbf{L}_p(1) = \mathbf{L}$ and $\mathbf{L}_p(-1) = \mathbf{Q}$. For the interpretation of parametric Laplacian, as a *matrix pencil* (Demmel 1997, Gantmacher 2000), see (Morbidi 2014). The interested reader is also referred to (Nikiforov 2017), where a convex linear combination of matrices $\mathbf{D}$ and $\mathbf{A}$, is considered.

The deformed Laplacian is more involved than the parametric Laplacian, but it also depends on a single real parameter s.

3.11. DEFINITION (DEFORMED LAPLACIAN MATRIX $\mathbf{\Delta}(s)$ (MORBIDI 2013A)). *The deformed Laplacian matrix of graph $\mathcal{G}$ with parameter $s \in \mathbb{R}$, is an $n \times n$ matrix defined as*

$$\mathbf{\Delta}(s) \;=\; \mathbf{I}_n \;-\; s\mathbf{A} \;+\; s^2(\mathbf{D} - \mathbf{I}_n).$$

◇

Note that $\mathbf{\Delta}(s)$ is a symmetric matrix (but *not* necessarily positive semidefinite for all s), and that $\mathbf{\Delta}(-1) = \mathbf{Q}$, $\mathbf{\Delta}(0) = \mathbf{I}_n$ and $\mathbf{\Delta}(1) = \mathbf{L}$. The main properties of this self-adjoint quadratic matrix polynomial in the variable s, have been studied in (Morbidi 2013a, Grindrod et al. 2018). We complement the existing results with a few additional properties:

1. *Additivity*: For $s_1, s_2 \in \mathbb{R}$,

$$\mathbf{\Delta}(s_1 + s_2) \;=\; \mathbf{\Delta}(s_1) \;+\; \mathbf{\Delta}(s_2) \;+\; 2s_1 s_2(\mathbf{D} - \mathbf{I}_n) \;-\; \mathbf{I}_n.$$

2. *Convexity*:

$$\frac{d^2\mathbf{\Delta}(s)}{ds^2} \;=\; 2(\mathbf{D} - \mathbf{I}_n) \;\succeq\; \mathbf{0},$$

where "$\succeq$" denotes the matrix inequality in the positive semidefinite sense. Therefore, if the domain of $\mathbf{\Delta}(s)$ is convex (e.g. an interval $\mathscr{U} \subset \mathbb{R}$), matrix-valued function $\mathbf{\Delta}(s)$ is convex (Boyd and Vandenberghe 2004, Sect. 3.1.4).

3. *Spectrum*: If $\mathcal{G}$ is κ-regular,

$$\boldsymbol{\Delta}(s)\mathbb{1} \;=\; (s-1)\big[(\kappa-1)s-1\big]\mathbb{1}, \tag{3.1}$$

i.e. the row-sums of $\boldsymbol{\Delta}(s)$ are all equal to $(s-1)[(\kappa-1)s-1]$. Equation (3.1) also implies that $(s-1)[(\kappa-1)s-1]$ is the eigenvalue associated with the eigenvector $\mathbb{1}$ of $\boldsymbol{\Delta}(s)$. For $\mathcal{G}=C_n$, this eigenvalue is simply $(s-1)^2$.

4. *Factorization*: If $(\mathbf{D}-\mathbf{I}_n) \succ \mathbf{0}$ (i.e. if the degree of the nodes of $\mathcal{G}$ is strictly greater than 1), then $\boldsymbol{\Delta}(s)$ can always be factorized into a pair of linear factors (Tisseur and Meerbergen 2001, Sect. 3.8.3) as follows

$$\boldsymbol{\Delta}(s) \;=\; (\mathbf{D}-\mathbf{I}_n)(s\mathbf{I}_n-\mathbf{U}_1)(s\mathbf{I}_n-\mathbf{U}_2),$$

where matrices $\mathbf{U}_1,\, \mathbf{U}_2 \in \mathbb{R}^{n\times n}$ satisfy

$$\mathbf{U}_1+\mathbf{U}_2 \;=\; (\mathbf{D}-\mathbf{I}_n)^{-1}\,\mathbf{A}, \quad \mathbf{U}_1\,\mathbf{U}_2 \;=\; (\mathbf{D}-\mathbf{I}_n)^{-1}.$$

An interesting link can be established between the deformed Laplacian and the *Parametrised Graph Shift Operator* (PGSO), based on the adjacency matrix, recently proposed in (Dasoulas et al. 2021).

3.12. DEFINITION (PARAMETRISED GRAPH SHIFT OPERATOR). *The parametrised graph shift operator of graph $\mathcal{G}$ is defined as*

$$\boldsymbol{\gamma}(\mathbf{A},\,\mathcal{S}) \;=\; m_1\,\mathbf{D}_a^{e_1} \;+\; m_2\,\mathbf{D}_a^{e_2}\mathbf{A}_a\,\mathbf{D}_a^{e_3} \;+\; m_3\,\mathbf{I}_n, \tag{3.2}$$

where $\mathbf{A}_a=\mathbf{A}+a\mathbf{I}_n$, $\mathbf{D}_a=\mathbf{D}+a\mathbf{I}_n$ and $\mathcal{S}=\{m_1,\, m_2,\, m_3,\, e_1,\, e_2,\, e_3,\, a\}$ is a 7-uple of scalar parameters (3 multiplicative parameters, 3 exponential parameters and 1 additive parameter). $\diamond$

The parametrised form in equation (3.2) spans the space of commonly-used GSOs and message-passing operators in Graph Neural Networks (GNNs) (Wu et al. 2021). It is easy to verify that

$$\boldsymbol{\gamma}(\mathbf{A},\,\mathcal{S}) \;=\; \boldsymbol{\Delta}(s) \text{ if } \mathcal{S}=\{s^2,\,-s,\,1-s,\,1,\,0,\,0,\,-1\} \text{ or } \mathcal{S}=\{s^2,\,-s,\,1-s^2,\,1,\,0,\,0,\,0\}.$$

The case of *directed graphs* asks for extra care and caution. We can define the adjacency matrix $\mathbf{A}$ as

$$[\mathbf{A}]_{ij} \;=\; \begin{cases} 1 & \text{if } (j,\,i)\in E,\\ 0 & \text{otherwise,}\end{cases}$$

and the degree matrix $\mathbf{D}=\operatorname{diag}(d_{\text{in}}(1),\ldots,d_{\text{in}}(n))$, where $d_{\text{in}}(i)$ denotes the in-degree of node i with $i\in\{1,\ldots,n\}$ (i.e., the number of directed edges pointing at node i).

With these definitions in hand, the in-degree Laplacian $\mathbf{L}(\mathcal{D})$, in-degree signless Laplacian $\mathbf{Q}(\mathcal{D})$ and in-degree signless parametric Laplacian $\mathbf{L}_p(\mathcal{D})$ of $\mathcal{D}$, can be defined as in the undirected case. A directed version of the deformed Laplacian has been considered in (Morbidi 2013a), and more recently in (Noferini and Quintana 2023, equ. 3) for studying the radius of convergence of the generating function of non-backtracking walks on directed graphs. Finally, taking inspiration from $\mathbf{L}^{\text{sym}}$ and $\mathbf{L}^{\text{rw}}$, normalized versions of signless, parametric and deformed Laplacian, can be defined as well.

3.2 Consensus protocol

3.2.1 Continuous time

For robots modeled as single integrators (see Sect. 3.3.1)

$$\dot{x}_i(t) \,=\, u_i(t), \quad i \in \{1, \ldots, n\}, \tag{3.3}$$

where $x_i(t) \in \mathbb{R}$ is the state of agent i, and $u_i(t) \in \mathbb{R}$ its input at time t, the *continuous-time consensus protocol* is defined as (Olfati-Saber and Murray 2004)

$$u_i(t) \,=\, \sum_{j \,\in\, \mathcal{N}(i)} (x_j(t) - x_i(t)), \quad i \in \{1, \ldots, n\}, \tag{3.4}$$

where $\mathcal{N}(i) \subset V$ denotes the set of nodes adjacent to node i in the fixed undirected communication graph $\mathcal{G}$. By applying control (3.4), equation (3.3) can be rewritten in matrix form as

$$\dot{\mathbf{x}}(t) \,=\, -\mathbf{L}\,\mathbf{x}(t), \tag{3.5}$$

where $\mathbf{x} = [x_1, \ldots, x_n]^T \in \mathbb{R}^n$ and $\mathbf{L}$ is the Laplacian matrix of graph $\mathcal{G}$. If $\mathcal{G}$ is connected, it is well known that

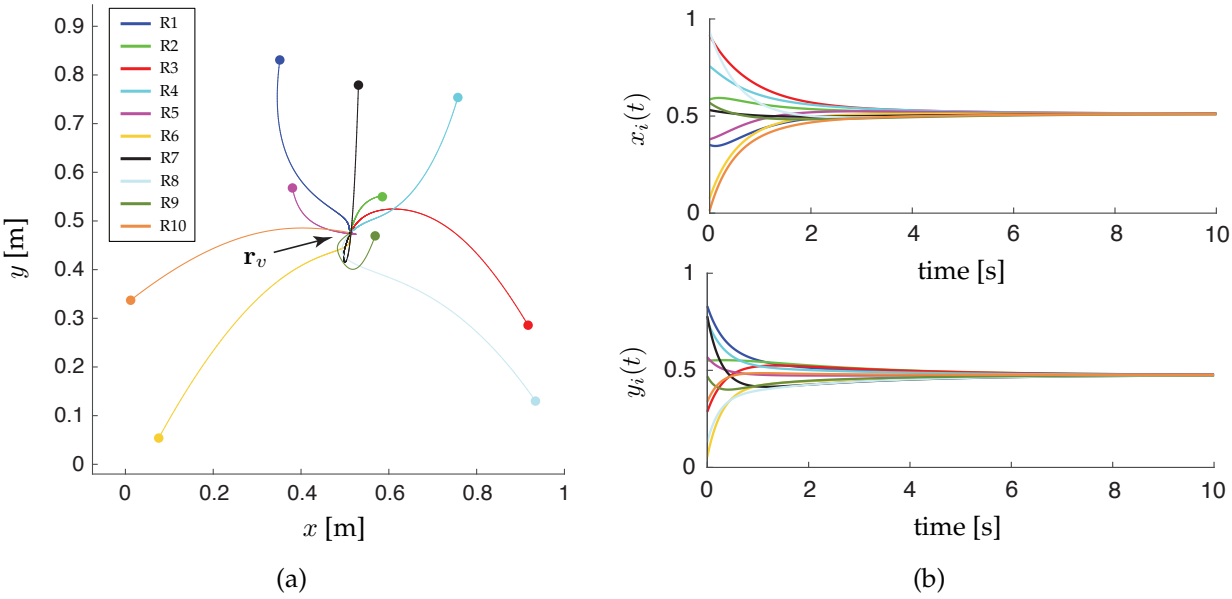
$$\lim_{t \,\to\, \infty} \mathbf{x}(t) \,=\, \frac{1}{n}\mathbb{1}\mathbb{1}^T\mathbf{x}_0 \,=\, \left(\frac{1}{n}\sum_{i=1}^{n} x_i(0)\right)\mathbb{1},$$

where $\mathbf{x}_0 \,=\, [x_1(0), \ldots, x_n(0)]^T$ is the vector of initial states, i.e. *average consensus* is achieved. The convergence factor to average consensus is the second smallest eigenvalue of $\mathbf{L}$, i.e. $\lambda_2(\mathbf{L})$, (Bullo 2022). It easy to verify that $2(1-\cos(\pi/n)) = \lambda_2(\mathbf{L}(P_n)) \leq \lambda_2(\mathbf{L}(\mathcal{G})) \leq \lambda_2(\mathbf{L}(K_n)) = n$. By omitting the time dependence to unburden the notation, system (3.5) can be equivalently written as

$$\dot{\mathbf{x}} \,=\, -\nabla\Phi_{\mathcal{G}}(\mathbf{x}),$$

where $\nabla\Phi_{\mathcal{G}}(\mathbf{x})$ denotes the gradient of the *Laplacian potential*

$$\Phi_{\mathcal{G}}(\mathbf{x}) \,=\, \frac{1}{2}\,\mathbf{x}^T\mathbf{L}\,\mathbf{x} \,\geq\, 0,$$



(a) (b)

Figure 3.3: Rendezvous problem: *(a) Trajectories of the ten single-integrator robots (the initial positions are marked with disks); (b) Time-evolution of the x- and y-coordinates of the ten robots. The communication graph is $\mathcal{G} = C_{10}$ and the algebraic connectivity $\lambda_2(\mathbf{L}(C_{10})) = 2(1 - \cos(\pi/5)) = 0.3820$.*

associated with graph $\mathcal{G}$ (if $\mathcal{G}$ is connected, $\Phi_{\mathcal{G}}(\mathbf{x}) = 0$ if and only if $x_i = x_j, \forall\, i, j$). As an illustration, Fig. 3.3 shows the evolution of dynamical system

$$\dot{\mathbf{x}}(t) \;=\; (-\mathbf{L} \otimes \mathbf{I}_2)\, \mathbf{x}(t),$$

for a cycle graph with ten nodes ($\mathcal{G} = C_{10}$), where "$\otimes$" denotes the Kronecker product and the state vector $\mathbf{x} \in \mathbb{R}^{20}$ stacks the x- and y-coordinates of the ten robots. In particular, Fig. 3.3(a) reports the trajectory of the ten robots converging towards the rendezvous point $\mathbf{r}_v = (0.5113,\, 0.4757)$ from a random initial position, and Fig. 3.3(b) shows the time-evolution of the x- and y-coordinates of the robots.

3.2.2 Discrete time

Similarly to equation (3.5), in *discrete-time*, we obtain the system (Olfati-Saber et al. 2007, Sect. IIC)

$$\mathbf{x}(k+1) \;=\; \mathbf{P}\, \mathbf{x}(k), \quad k \in \{0, 1, 2, \ldots\}, \tag{3.6}$$

where

$$\mathbf{P} \;=\; \mathbf{I}_n - \varepsilon \mathbf{L},$$

is the Perron matrix of graph $\mathcal{G}$ with parameter ε, $\mathbf{x} = [x_1, \ldots, x_n]^T \in \mathbb{R}^n$ is the state vector, and $0 < \varepsilon < 1/d_{\max}(\mathcal{G})$ is the step-size, with $d_{\max}(\mathcal{G}) = \max(\mathbf{A}\mathbb{1})$ the maximum degree of $\mathcal{G}$. System (3.6) is the one-step Euler discretization of the continuous-time Laplacian flow (3.5). The Perron matrix is a symmetric, nonnegative matrix (i.e. all its entries are nonnegative). Moreover, it is row stochastic, i.e. it satisfies $\mathbf{P}\mathbb{1} = \mathbb{1}$, which means that $\mathbb{1}$ is the eigenvector of $\mathbf{P}$ associated with the eigenvalue 1.

As in the continuos-time case, under the previous assumptions, if the graph $\mathcal{G}$ is connected, the state $\mathbf{x}$ of system (3.6) asymptotically converges to $\frac{1}{n}\mathbb{1}\mathbb{1}^T\mathbf{x}_0$. In the discrete-time case, the convergence factor to average consensus is the second largest eigenvalue of $\mathbf{P}$, i.e. $\lambda_{n-1}(\mathbf{P})$, (Bullo 2022). Obviously, $1 - n\varepsilon = \lambda_{n-1}(\mathbf{P}(K_n)) \leq \lambda_{n-1}(\mathbf{P}(\mathcal{G})) \leq \lambda_{n-1}(\mathbf{P}(P_n)) = 1 - 2\varepsilon(1 - \cos(\pi/n))$.

3.3 Modeling of a mobile robot

3.3.1 Linear models

The simplest possible model that we can use to describe a mobile robot is the *single integrator*

$$\dot{\mathbf{x}} = \mathbf{u}, \tag{3.7}$$

where $\mathbf{x} \in \mathbb{R}^m$ denotes the position of the robot in the fixed (world) reference frame $F_{\mathcal{W}}$ and $\mathbf{u} \in \mathbb{R}^m$ its control input, with $m = 2$, if the robot moves in a 2D environment (e.g. a wheeled robot) and $m = 3$, if the robot moves in a 3D environment (e.g. an aerial vehicle). In (3.7), the control input is the velocity vector $\mathbf{u}$, and the vehicle can instantaneously move in any possible direction. Equation (3.7) can thus be used to model omnidirectional robots, in which we can independently act on the m entries of vector $\dot{\mathbf{x}}$.

As an alternative to a velocity input, an acceleration input $\mathbf{u}$ can be contemplated, as in the so-called *double integrator* model

$$\ddot{\mathbf{x}} = \mathbf{u}. \tag{3.8}$$

This model describes the motion of a unit-mass particle subject to a linear acceleration $\mathbf{a} = \ddot{\mathbf{x}}$. If we introduce the auxiliary vector $\mathbf{v} = \dot{\mathbf{x}}$, we can equivalently rewrite (3.8) as a system of two 1st-order ODEs (in state-space form):

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{v}, \\ \dot{\mathbf{v}} = \mathbf{u}, \end{cases}$$

with state vector $\boldsymbol{q} = [\mathbf{x}^T, \mathbf{v}^T]^T \in \mathbb{R}^{2m}$. A drawback of models (3.7) and (3.8), is that the robot is considered as a point without dimensions, and body orientation is not taken into account. The kinematic models that we will present in the next section are more involved but more realistic, and they will allow to overcome this second limitation.

3.3.2 Nonlinear models

In Sect. 3.3.1, we have presented two simple linear models for mobile robots. A third, widely-used (posture) kinematic model, which can be used to describe the behavior of a wheeled robot or a fixed-wing aircraft flying at constant altitude (see Sect. 5.1), for which instantaneous lateral motions are forbidden, is

$$\begin{cases} \dot{x} = v\cos\theta, \\ \dot{y} = v\sin\theta, \\ \dot{\theta} = \omega, \end{cases} \tag{3.9}$$

where $\boldsymbol{q} = [x,\, y,\, \theta]^T \in \mathbb{R}^2 \times \mathbb{S}^1$ denotes the pose (position and orientation) or configuration of the robot in the fixed reference frame $F_{\mathcal{W}}$, and $\mathbf{u} = [v,\, \omega]^T$ is the control input of the robot (its linear velocity and its angular velocity, respectively), see Fig. 3.4(a). The driftless nonlinear system (3.9) can be equivalently rewritten in matrix form as

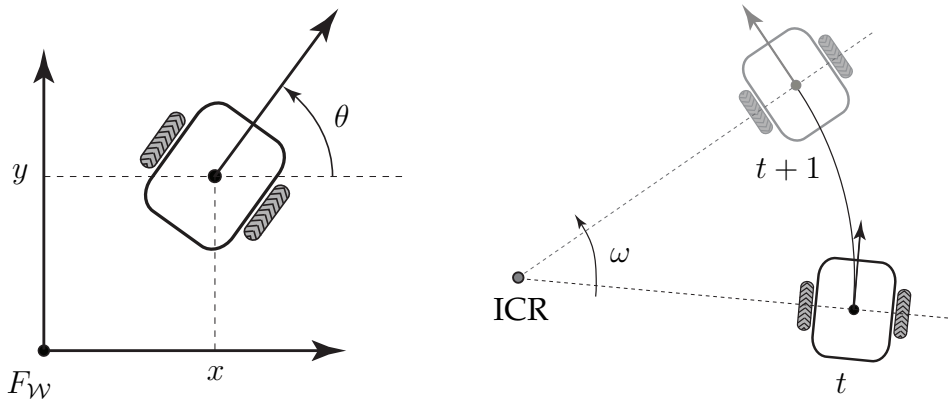
$$\dot{\boldsymbol{q}} = \boldsymbol{f}(\boldsymbol{q})\mathbf{u} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \mathbf{u}.$$

Depending on the interval of definition of inputs v and ω, four different robot models (Bullo et al. 2009, Sect. 3.1) can be identified:

- *Unicycle robot*: $v \in [-1,\, 1]$, $\omega \in [-1,\, 1]$.
- *Differential-drive robot*: We set $v = \frac{1}{2}(\omega_{\mathrm{r}} + \omega_{\mathrm{l}})$, $\omega = \frac{1}{2}(\omega_{\mathrm{r}} - \omega_{\mathrm{l}})$ and we assume that $\omega_{\mathrm{r}}, \omega_{\mathrm{l}} \in [-1,\, 1]$, where ω_{r}, ω_{l} are the angular velocities of the right, left wheel of the robot, respectively.
- *Reeds-Sheep vehicle* (Reeds and Shepp 1990): $v \in \{-1,\, 0,\, 1\}$, $\omega \in [-1,\, 1]$.
- *Dubins vehicle* (Dubins 1957): $v = 1$, $\omega \in [-1,\, 1]$.

The unicycle and differential-drive models are well-suited to describe wheeled robots, while the Dubins vehicle captures the (simplified) dynamics of a fixed-wing aircraft. In fact, since the linear velocity is constant ($v = 1$), the vehicle can only move forward and it can never stop. On the contrary, the unicycle robot, the differential-drive robot and the Reeds-Sheep vehicle can stop and they can move forward and backward.

At each time instant, the motion of a vehicle can be viewed as an instantaneous rotation about the ICR (*Instantaneous Center of Rotation*), whose position with respect to the robot can be time varying (Chung and Iagnemma 2016). Each point of the vehicle body moves instantaneously along a circle of arc with center at ICR. It is easy to verify



(a) (b)

Figure 3.4: *(a) Unicycle robot with generalized coordinates* $\boldsymbol{q} = [x,\, y,\, \theta]^T$ *and control input* $\mathbf{u} = [v,\, \omega]^T$*; (b) Instantaneous Center of Rotation (ICR) of the unicycle robot. Two time instants* (t *and* $t+1$) *are shown in the figure.*

that the ICR of the robot described by system (3.9), lies along the common axle of the two wheels (see Fig. 3.4(b)).

Finally, a *four-wheel* planar vehicle with front and rear axles separated by a distance ℓ, can also be described by system (3.9) with the following differences: $[x,\, y]^T \in \mathbb{R}^2$ is the position of the midpoint of the rear axle, $\theta \in \mathbb{S}^1$ is the orientation of the longitudinal axis of the vehicle, the input v is the linear velocity of the midpoint of the rear axle and it satisfies

$$\omega \;=\; \frac{v}{\ell}\tan\phi,$$

where $\phi \in \mathbb{S}^1$ is the steering angle of the vehicle. The control input of this rear-wheel drive robot is $\mathbf{u} = [v,\, \dot{\phi}]^T$. In robotics, it is generally referred to as *car-like*, see (Siciliano et al. 2009, Ch. 11) for more details.

3.4 Conclusion

In this chapter, we have briefly reviewed some basic notions of graph theory, distributed control and mobile robotics, for later use in Chapters 4 and 5, which form the core of this HDR thesis. We also established the notation used throughout the manuscript (see the Glossary in Sect. 2.4, for the full list of symbols and abbreviations).

Chapter 4

Formation control of mobile robots

Reality is that which, when you stop believing in it, doesn't go away.

"I Hope I Shall Arrive Soon" – Philip K. Dick, 1980

Abstract

This chapter deals with the formation control problem for multiple autonomous robots, a "leitmotif" of my research in the last fifteen years. Different aspects of this problem will be studied. We will start with a new characterization of mobility for distance-bearing formations of unicycle robots, we will proceed with cooperative adaptive cruise control (an example of distance-based leader-follower formation control problem in one dimension) under string-stability constraints, and we will conclude with a presentation of functions of Laplacian matrix and their application to shape-based distributed formation control.

The material of this chapter is adapted from (Morbidi and Bretagne 2018, Morbidi et al. 2013, Morbidi 2022) (cf. [C33], [C24], [J21] in Sect. 1.1.2).

This chapter, entirely dedicated to the formation control problem, is organized as follows. In Sect. 4.1, we study distance-bearing formations of unicycle robots and extend the well-known classification of wheeled robots, by *type*. Sect. 4.2 presents new decentralized optimal strategies for cooperative adaptive cruise control of a car platoon under string-stability constraints. Sect. 4.3 introduces the notion of function of combinatorial Laplacian and investigates its possible application to distributed formation control of single-integrator robots. Finally, in Sect. 4.4, the main contributions of Chapter 4 are briefly summarized.

The work in Sect. 4.2 was carried out during my stay at Johannes Kepler University, Austria (2011-2012), while the results in Sect. 4.1 and Sect. 4.3, were obtained at the University of Picardie Jules Verne.

4.1 Mobility of formations of unicycle robots

In this section, we present a new characterization of mobility for formations of unicycle robots defined by distance-bearing constraints. In fact, by introducing a simple reduction procedure which associates a prescribed formation with a "macro-robot", we extend the classification by *type* proposed by (Campion et al. 1996), to multi-agent systems. To simplify the classification task, which only leverages the nonslip condition for a conventional centered wheel, we assume that the robots are disposed at the vertices of a regular convex polygon. We demonstrate the practical utility of the notion of macro-robot in a trajectory-tracking control problem for a formation of unicycles.

4.1.1 Introduction

Teams of cooperating robots are being increasingly deployed in the real world for complex tasks, such as object transport (Alonso-Mora et al. 2017), warehouse management (Farinelli et al. 2017), and inspection of industrial infrastructures (Caprari et al. 2012), to name but a few relevant examples. However, while considerable efforts have gone into the design of (distributed) algorithms for controlling the geometric shape of multi-robot formations (Belta and Kumar 2004, Oh et al. 2015), relatively few works (Tabuada et al. 2005, Sun and Anderson 2016), so far, have dealt with their *structural properties*. Notably, a general and comprehensive theory which explains how the formation constraints affect the *mobility* of the resulting multi-robot system is, to the best of our knowledge, still missing. A vast literature on distance rigidity (Anderson et al. 2008, Krick et al. 2009) and more recently on bearing and SE(2) rigidity (Zelazo, Robuffo Giordano and Franchi 2015, Zhao and Zelazo 2016, Stacey and Mahony 2018, Sun et al. 2018) of robot formations has emerged, but it has only provided a partial answer to the motion feasibility problem. In fact, simplistic mathematical models (single or double integrators) have been considered for the vehicles therein, and the constraints resulting from wheel-ground interaction have been overlooked.

The characterization of mobility of *wheeled mobile robots* (WMRs) described by kinematic and dynamic models, and their elegant classification into five *types* according to their degrees of mobility and steerability, traces back to the seminal work of Campion *et al.* (Campion et al. 1996) in the 90's. The *nonslip condition*, i.e. the fact that the kinematic constraints imply that the velocity of the center of a conventional wheel is parallel to the wheel plane, plays a crucial role in (Campion et al. 1996).

Our goal here is to revisit and extend the classification by type in (Campion et al. 1996) to formations of *unicycle robots* specified by relative distance and bearing constraints. To simplify our analysis, we decided to turn our attention to robots with conventional centered orientable wheels disposed at the vertices of regular convex poly-

gons. While this might seem a serious restriction, this class of formations is extremely rich and diverse, since a large variety of distance-bearing constraints can be imposed. Moreover, the members of this class fall into few generic categories inherited from the single-vehicle case and independent of robots' number. We avail ourselves of a simple *reduction procedure*, leading to the definition of a "macro-robot" (or m-robot, in short): the "wheels" of the m-robot correspond to the original vehicles in the formation, and the geometry of its "chassis" depends on the nature of the distance-bearing constraints imposed on the robots. The nonslip condition can be defined for each "wheel" of the m-robot, from which the degree of mobility and steerability, and thus the *formation type* can be deduced using the tools developed in (Campion et al. 1996). The taxonomy proposed here is not merely of theoretical interest. In fact, it may inform the design of more efficient formation control strategies for multiple nonholonomic robots and shed some light on the intrinsic limitations of the existing ones (see (Egerstedt and Hu 2001, Dong and Farrell 2008, Consolini et al. 2009, Zhao et al. 2018) and the references therein). In particular, thanks to our reduction procedure, we could easily adapt the trajectory-tracking controller via dynamic feedback originally proposed in (Thuilot et al. 1996) for mobile robots with multiple steering wheels, to distance-bearing formations of unicycles.

It is finally worth pointing out here that an early attempt towards a characterization of the mobility of leader-follower formations of unicycles was made in (Consolini et al. 2012). However, the analysis in (Consolini et al. 2012) is tangential to this work, since it does not leverage the nonslip condition for centered wheels, but the more abstract, system-theoretic notion of internal dynamics.

The rest of this section is organized as follows. In Sect. 4.1.2, we briefly review the classification of WMRs by type proposed in (Campion et al. 1996). In Sect. 4.1.3, this classification is extended to distance-bearing formations of unicycle robots. In Sect. 4.1.4, we discuss the results of numerical simulations featuring a multi-robot formation tracking a reference trajectory, and in Sect. 4.1.5 some possible directions for future research are outlined.

4.1.2 Mobility of wheeled robots

In this section, we will briefly review the standard classification of WMRs by type (Campion et al. 1996, Chung and Iagnemma 2016). We will start by introducing the nonslip condition for a conventional centered wheel, and proceed to define the degree of mobility and steerability of a robot. Let $\{O_I;\, x_I,\, y_I\}$ be an inertial frame, and $\{O_R;\, x_R,\, y_R\}$ the moving frame rigidly attached to a robot (see Fig. 4.1). The pose of the robot with respect to the inertial frame, is described by the posture vector $\mathbf{q} = [x,\, y,\, \theta]^T$, where x and y are the coordinates of point O_R and θ describes the orientation of the moving frame with respect to the inertial frame. The position of the center of wheel A is

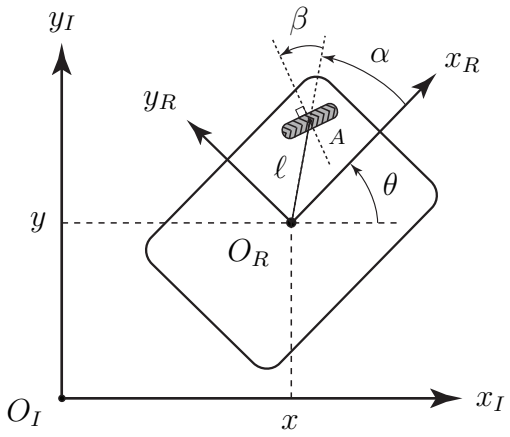


Figure 4.1: *Notation for a conventional centered wheel of a mobile robot.*

expressed in polar coordinates by the distance ℓ and angle α. Finally, the angle between the segment $\overline{O_R A}$ and the axle of the wheel (a.k.a. propulsion axis) is denoted by β. For a conventional wheel, the kinematic constraints imply that the velocity of the center of the wheel is parallel to the wheel plane (*nonslip condition*) and proportional to the wheel spinning velocity (*pure rolling condition*). For a centered (passive or active) wheel, the nonslip condition at the contact point between the wheel and the ground reads,

$$\big[\cos(\alpha+\beta),\ \sin(\alpha+\beta),\ \ell\sin\beta\big]\,\mathbf{R}^T(\theta)\,\dot{\mathbf{q}} \;=\; 0, \tag{4.1}$$

where

$$\mathbf{R}(\theta) \;=\; \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

is the elementary rotation matrix of an angle θ about the z-axis and $\mathbf{R}^T(\theta)\dot{\mathbf{q}}$ is used to transform the motion parameters $\dot{\mathbf{q}}$ expressed in the inertial reference frame, into motion parameters in the local reference frame of the robot. For a fixed wheel, β in (4.1) is constant, while for a steering wheel, $\beta = \beta(t)$, i.e. it varies as a function of time (being the wheel actuated or non-actuated). Let us now consider a robot equipped with n centered wheels. It is expedient to introduce the subscripts "f" and "s" to identify quantities related to *fixed* and *steering* wheels, respectively. The number of wheels of each type are denoted by n_f and n_s, with $n = n_f + n_s$. The nonslip conditions for centered fixed or steering wheels can be rewritten in a compact form as,

$$\mathbf{C}_1^*(\boldsymbol{\beta}_s)\,\mathbf{R}^T(\theta)\,\dot{\mathbf{q}} \;=\; \mathbf{0},$$

where $\mathbf{C}_1^*(\boldsymbol{\beta}_s) = [\mathbf{C}_{1f}^T,\ \mathbf{C}_{1s}^T(\boldsymbol{\beta}_s)]^T$, $\mathbf{C}_{1f} \in \mathbb{R}^{n_f \times 3}$, $\mathbf{C}_{1s}(\boldsymbol{\beta}_s) \in \mathbb{R}^{n_s \times 3}$ and $\boldsymbol{\beta}_s = [\beta_1, \beta_2, \ldots, \beta_{n_s}]^T \in \mathbb{R}^{n_s}$ is the vector of the angles of the steering wheels of the robot. The mobility of a

robot is directly related to the rank of $\mathbf{C}_1^*(\boldsymbol{\beta}_s)$ which depends on the robot design (Chung and Iagnemma 2016). The *degree of mobility* δ_m of a robot is defined as:

$$\delta_m \;=\; \text{nul}(\mathbf{C}_1^*(\boldsymbol{\beta}_s)) \;=\; 3 - \text{rank}(\mathbf{C}_1^*(\boldsymbol{\beta}_s)),$$

which is equal to the number of degrees of freedom that can be directly manipulated from the robot's inputs without reorientation of the steering wheels. On the other hand, the *degree of steerability* δ_s of a robot is defined as:

$$\delta_s \;=\; \text{rank}(\mathbf{C}_{1s}(\boldsymbol{\beta}_s)),$$

which corresponds to the number of steering wheels that can be oriented independently in order to steer the robot. If a robot is equipped with more than δ_s steering wheels, the motion of the extra ones must be coordinated in order to guarantee the existence of the Instantaneous Center of Rotation (ICR) at all times (Clavien et al. 2018). Finally, the degree of maneuverability δ_M of a robot is defined as $\delta_M = \delta_m + \delta_s$. For WMRs of practical interest, the indices δ_m and δ_s must comply with a set of constraints (Chung and Iagnemma 2016). Therefore, only four structures are relevant in practice, corresponding to the following four pairs:

$$\begin{array}{r|cccc} \delta_m & 2 & 2 & 1 & 1 \\ \delta_s & 0 & 1 & 1 & 2 \end{array} \tag{4.2}$$

Each structure is referred to as a *type* $(\delta_m,\, \delta_s)$ *robot*. Note that in (4.2), we omitted type $(3,\, 0)$ robots, since they do not have fixed or centered steering wheels (in fact, "omnimobiles", which have full mobility in the plane, are only equipped with Swedish or off-centered orientable wheels).

4.1. REMARK. *Type* $(2,\, 0)$ *robots have no steering wheels, but either one or several fixed wheels with a common axle: a typical example is the* unicycle robot *(Chung and Iagnemma 2016). Type* $(2,\, 1)$ *robots have no fixed wheels and at least one steering wheel: if there is more than one steering wheel, their orientation must be coordinated in such a way that* $\delta_s = 1$. *Type* $(1,\, 1)$ *robots have one or several fixed wheels on a single common axle, and also one or several steering wheels, under the condition that their centers are not located on the common axle of the fixed wheels and that their orientation is coordinated.* Car-like robots *belong to this class. Finally, type* $(1,\, 2)$ *robots have no fixed wheels, but at least two steering wheels: if there are more than two steering wheels, then their orientation must be coordinated in order to satisfy the condition* $\delta_s = 2$. *A typical example is the* two-steer robot *(Siegwart et al. 2011, Sect. 3.3).* ◇

4.1.3 Mobility of distance-bearing formations of unicycles

In this section, we extend the classification by type presented in Sect. 4.1.2, to formations of unicycle robots defined by relative distance and bearing constraints. A generic

n-robot formation can be encoded in terms of the weighted undirected graph $\mathcal{G}_d = (V, E_d, \mathbf{D})$ and the weighted directed graph $\mathcal{G}_\phi = (V, E_\phi, \mathbf{\Phi})$, whose set of vertices $V = \{1, 2, \ldots, n\}$ is indexed by the robots in the team, and whose set of edges E_d (E_ϕ) contains unordered (ordered) pairs of vertices that represent inter-robot formation specifications. $\mathbf{D} = [d_{ij}]$, $\mathbf{\Phi} = [\phi_{ij}]$ are the $n \times n$ weighted adjacency matrices of distance and bearing constraints, respectively, where,

$$\begin{cases} d_{ij} > 0, & \text{if } \{i, j\} \in E_d, \\ d_{ij} = 0, & \text{otherwise,} \end{cases} \qquad \begin{cases} \phi_{ij} > 0, & \text{if } (i, j) \in E_\phi, \\ \phi_{ij} = 0, & \text{otherwise.} \end{cases}$$

For the sake of exposition, in the following we will make the following assumption:

4.1. ASSUMPTION. *The $n \geq 2$ robots are disposed at the vertices of a* regular convex polygon *of side d, and the origin O_F of the reference frame $\{O_F; x_F, y_F\}$ of the formation is placed at the circumcenter of the polygon, with the x_F-axis pointing towards robot 1.* ◇

Note that while the last choice is arbitrary, it will allow us to easily draw a parallel between the single-robot case studied in Sect. 4.1.2 and the multi-robot case, via the notion of "macro-robot". We also observe that, by definition, the formations studied here are *rigid* (Anderson et al. 2008). In what follows, we shall proceed in stages and analyze formations with an increasing number of robots. We will use the terms "m-robot" and "m-wheel" as shorthands for macro-robot and for a macro-robot's wheel, respectively.

Formation of two unicycles

Fig. 4.2(a) shows a formation of two unicycle robots defined by a single distance constraint (the distance d between their centers O_{R_1} and O_{R_2}). In this case, $V = \{1, 2\}$, $E_d = \{\{1, 2\}\}$, $E_\phi = \emptyset$ (the empty set), and $n_f + n_s = 0 + 2$. The mobility analysis for this formation reduces to that of the m-robot reported in Fig. 4.2(b), whose m-wheels are separated by the distance d. By denoting the orientation of the front (rear) steering m-wheel with respect to the "chassis" of the m-robot by β_1 (β_2), and by noticing that for the front m-wheel $\alpha_1 = 0$ and for the rear m-wheel $\alpha_2 = \pi$ (cf. Fig. 4.1), we have that:

$$\mathbf{C}_1^*(\beta_1, \beta_2) = \begin{bmatrix} \cos\beta_1 & \sin\beta_1 & \frac{1}{2}\, d \sin\beta_1 \\ -\cos\beta_2 & -\sin\beta_2 & \frac{1}{2}\, d \sin\beta_2 \end{bmatrix}.$$

Therefore, $(\delta_m, \delta_s) = (1, 2)$, $\delta_M = 3$, and the formation behaves as a two-steer robot.

Fig. 4.2(c) reports a formation of two unicycles defined by a distance and a bearing constraint. On par with the previous case, $V = \{1, 2\}$ and $E_d = \{\{1, 2\}\}$, but now $E_\phi = \{(2, 1)\}$. By applying the same formation reduction procedure as above, we have

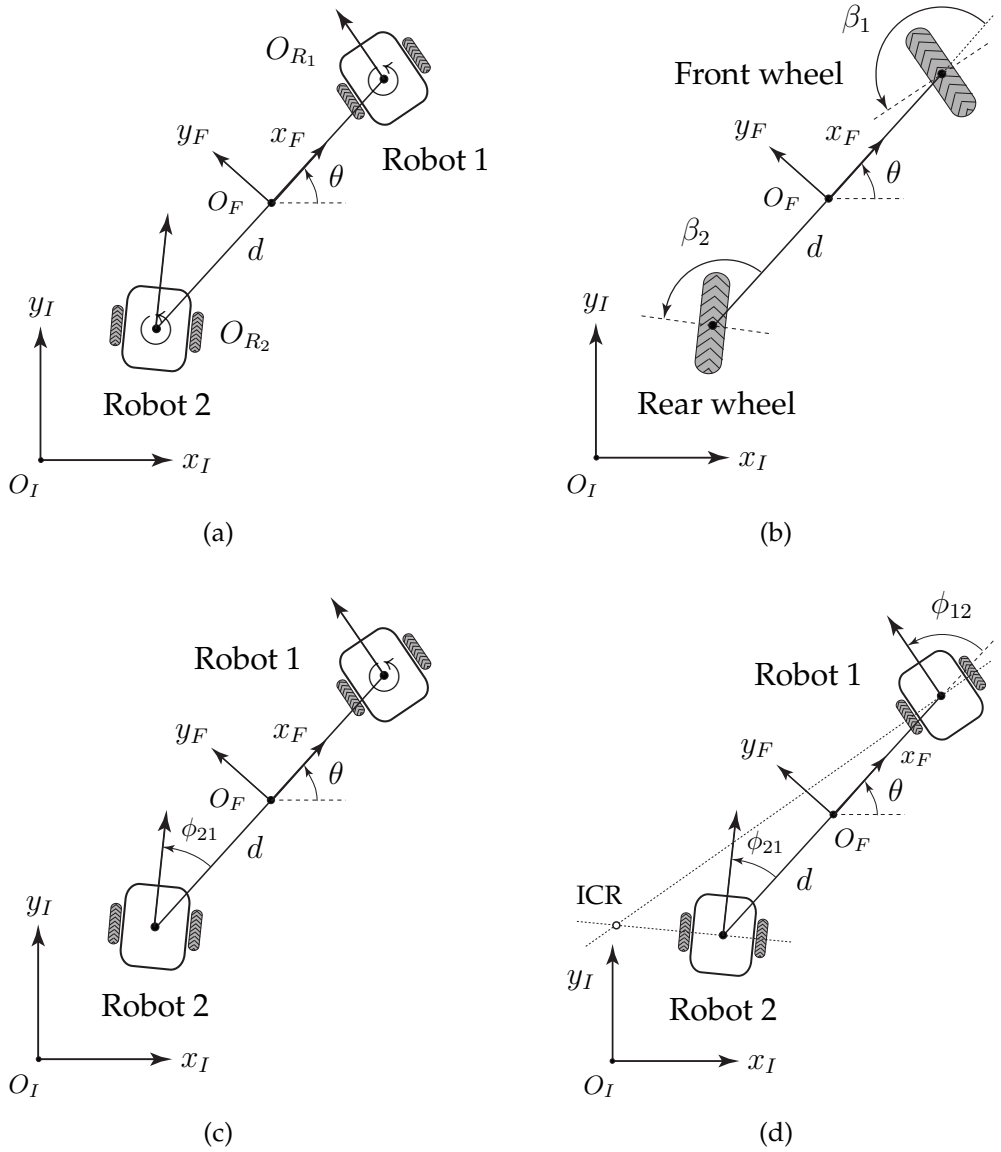


Figure 4.2: Two unicycles. *(a) Formation defined by a single distance constraint, d, and (b) corresponding m-robot after the application of the reduction procedure. The formation in (c) is defined by a distance constraint, d, and a bearing constraint, ϕ_{21}. The formation in (d) is defined by a distance constraint, d, and two bearing constraints, ϕ_{12}, ϕ_{21} with $\phi_{12} \neq \phi_{21}$. The symbol $\circlearrowleft$ indicates that a robot can freely rotate about its vertical axis.*

that the front m-wheel of the associated m-robot is steerable with an angle β_1 and that $\alpha_1 = 0$. On the other hand, the rear m-wheel of the m-robot is now fixed with constant

angle $\beta_2 = \phi_{21} + \pi/2$ and $\alpha_2 = \pi$. This entails that ($n_f + n_s = 1 + 1$):

$$\mathbf{C}_1^*(\beta_1) = \begin{bmatrix} \sin\phi_{21} & -\cos\phi_{21} & \frac{1}{2}\,d\cos\phi_{21} \\ \cos\beta_1 & \sin\beta_1 & \frac{1}{2}\,d\sin\beta_1 \end{bmatrix},$$

and thus $(\delta_m,\, \delta_s) = (1,\, 1)$, meaning that the formation is equivalent to a car-like robot.

Finally, Fig. 4.2(d) shows the case of $E_d = \{\{1,\, 2\}\}$ and $E_\phi = \{(1,\, 2),\, (2,\, 1)\}$. The m-wheels of the corresponding m-robot are both fixed, thus $\delta_s = 0$. Note that if $\phi_{12} \neq \phi_{21}$, this formation is not of practical interest, since the only possible motion is about the fixed ICR of the m-robot. However, if $\phi_{12} = \phi_{21} \neq 0$, the m-wheels of the m-robot are parallel and $(\delta_m,\, \delta_s) = (2,\, 0)$, i.e. the formation behaves as a unicycle robot.

Formation of three unicycles

Fig. 4.3(a) shows a formation defined by three distance constraints: $V = \{1, 2, 3\}$ and $E_d = \{\{1,\, 2\}, \{2,\, 3\}, \{3,\, 1\}\}$. The reduction procedure described in the previous subsection can be applied to this formation as well. By noticing that $\alpha_1 = 0$, $\alpha_2 = 2\pi/3$, $\alpha_3 = -2\pi/3$, and that the circumradius of the equilateral triangle in Fig. 4.3(a) is $\sqrt{3}\,d/3$, we obtain:

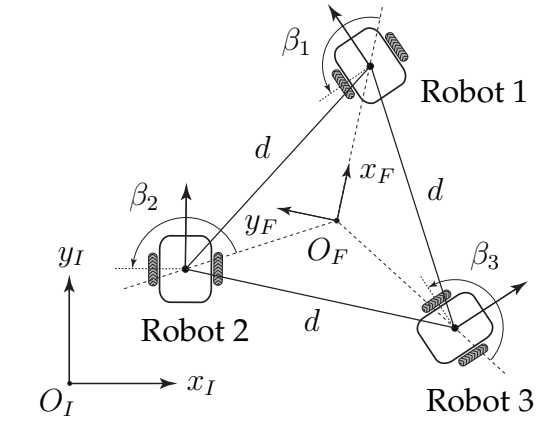
$$\mathbf{C}_1^*(\beta_1, \beta_2, \beta_3) = \begin{bmatrix} \cos\beta_1 & \sin\beta_1 & \frac{\sqrt{3}}{3}\,d\sin\beta_1 \\ \cos(\beta_2 + \frac{2\pi}{3}) & \sin(\beta_2 + \frac{2\pi}{3}) & \frac{\sqrt{3}}{3}\,d\sin\beta_2 \\ \cos(\beta_3 - \frac{2\pi}{3}) & \sin(\beta_3 - \frac{2\pi}{3}) & \frac{\sqrt{3}}{3}\,d\sin\beta_3 \end{bmatrix}.$$

In order to gain some insight into the mobility properties of the formation, let us study the determinant of $\mathbf{C}_1^*(\beta_1, \beta_2, \beta_3)$. By elementary trigonometry, we find that:

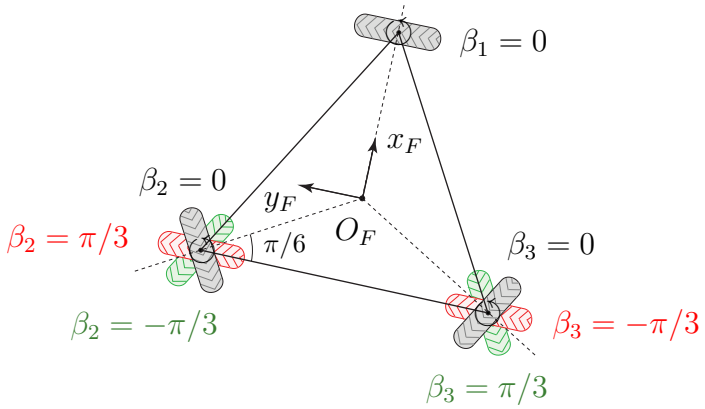
$$\det(\mathbf{C}_1^*(\beta_1, \beta_2, \beta_3)) = \frac{d}{2}\big[\sin(-\beta_1 + \beta_2 + \beta_3) + \sin(\beta_1 - \beta_2 + \beta_3) + \sin(\beta_1 + \beta_2 - \beta_3)\big].$$

Although it is not trivial to provide an exhaustive characterization of the zeros of this transcendental function of β_1, β_2 and β_3, we can easily identify the following special configurations of the m-wheels:

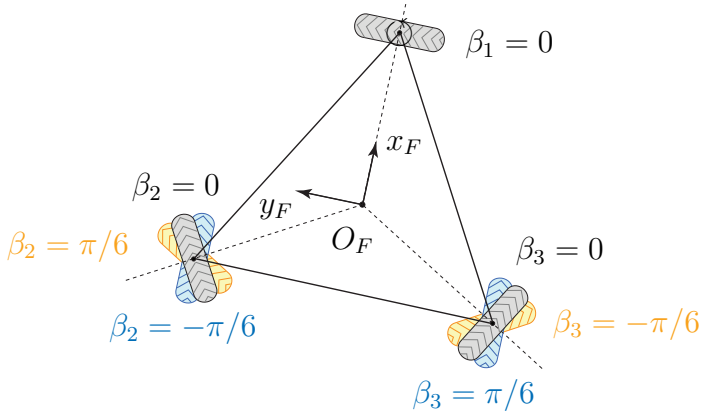
1. Let $\beta_1 = \beta_1(t)$ (free), $\beta_2 = \beta_1 \pm \pi/3$, $\beta_3 = \beta_1 \mp \pi/3$. The orientation of the three m-wheels of the m-robot is coupled (red and green, respectively, in Fig. 4.3(b)), and the formation behaves as a *three-wheel synchro-drive robot* (Clavien et al. 2018). It is well-known that the kinematic model of a synchro-drive robot is equivalent to that of a type $(1,\, 1)$ robot (Chung and Iagnemma 2016),(Siegwart et al. 2011, Sect. 3.3.3). In fact, a synchro-drive robot can only manipulate, in total, two degrees of freedom (the orientation of the chassis cannot be changed).



(a)



(b)



(c)

Figure 4.3: Three unicycles. *(a) Formation defined by three distance constraints; (b), (c) Special orientations of the m-wheels of the m-robot (for illustration, we set $\beta_1 = 0$ in both cases).*

2. Let $\beta_1 = \beta_1(t)$ (free), $\beta_2 = -\pi/6$, $\beta_3 = \pi/6$ (cyan in Fig. 4.3(c)). The m-wheels 2 and 3 are fixed and parallel to the x_F-axis, hence the formation type is $(1,\, 1)$.

3. Let $\beta_1 = \beta_1(t)$ (free), $\beta_2 = \pi/6$ and $\beta_3 = -\pi/6$ (yellow in Fig. 4.3(c)). The axles of m-wheels 2 and 3 concur at the center of m-wheel 1 (the ICR), and the only possible motion of the formation is about this fixed point. This arrangement is called *singular wheels configuration* (SWC) in (Thuilot et al. 1996, Def. 2).

4. Let $\beta_1 = \beta_2 = \beta_3 = 0$. The three m-wheels are tangent to the circle circumscribing the triangular "chassis" of the m-robot (gray in Fig. 4.3(b) or Fig. 4.3(c)). In this degenerate configuration, the only possible motion is a rotation about the center O_F (the ICR) of the formation.

Formation of n unicycles

Fig. 4.4 reports a formation of $n \geq 3$ unicycles defined by n distance constraints: $V = \{1, 2, \ldots, n\}$ and $E_d = \{\{1,\, 2\}, \{2,\, 3\}, \ldots, \{n,\, 1\}\}$. Recall that the central angle of a regular convex polygon with n vertices is $\gamma = 2\pi/n$ and that the circumradius is $r_c = d/(2\sin(\pi/n))$. Hence, $\alpha_i = \gamma(i-1)$, $i \in \{1, 2, \ldots, n\}$. By applying the reduction procedure seen in the previous sections, we obtain:

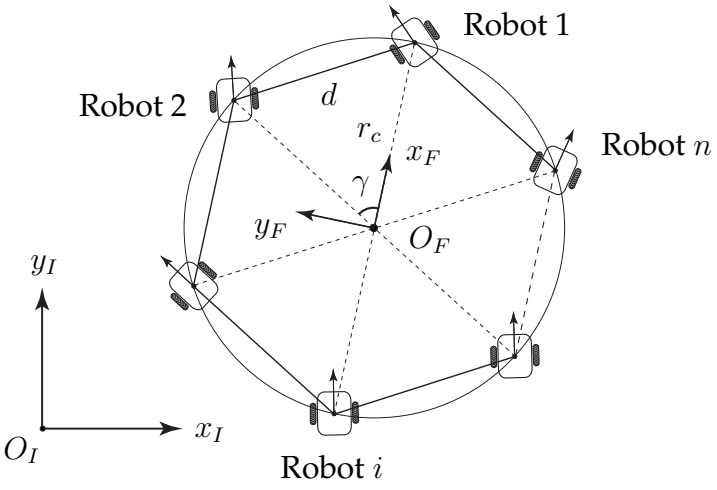


Figure 4.4: n unicycles. *The formation is specified by n distance constraints. The regular convex polygon has side d and circumradius r_c.*

$$\mathbf{C}_1^*(\beta_1,\ldots,\beta_n) = \begin{bmatrix} \cos\beta_1 & \sin\beta_1 & r_c\sin\beta_1 \\ \cos(\beta_2+\gamma) & \sin(\beta_2+\gamma) & r_c\sin\beta_2 \\ \vdots & \vdots & \vdots \\ \cos(\beta_n+\gamma(n-1)) & \sin(\beta_n+\gamma(n-1)) & r_c\sin\beta_n \end{bmatrix}.$$

The next result is instrumental in studying the rank deficiency of $\mathbf{C}_1^*(\beta_1,\ldots,\beta_n) \in \mathbb{R}^{n\times 3}$.

4.1. PROPOSITION. *The $n+1$ points $O_{R_1}, O_{R_2}, \ldots, O_{R_n}, O_F$ obey the classic Descartes' principle of instantaneous motion, i.e., at each instant, their motion coincides either with a pure translation or with a pure rotation about the ICR.*

Proof : *The distance between any two given points in $\{O_{R_1}, O_{R_2}, \ldots, O_{R_n}, O_F\}$ remains constant over time in the horizontal plane defined by $\{O_I;\, x_I,\, y_I\}$. Hence, the proof is the same as that traditionally used for a rigid body in mechanics.* ■

4.1. LEMMA. *The following condition holds true:*

$$1 < \mathit{rank}(\mathbf{C}_1^*(\beta_1,\ldots,\beta_n)) \leq 3. \tag{4.3}$$

Proof : *See (Morbidi and Bretagne 2018).* ■

The following lemma provides an algebraic characterization of the full-rank case in Lemma 4.1.

4.2. LEMMA. *$\mathit{rank}(\mathbf{C}_1^*(\beta_1,\ldots,\beta_n)) = 3$ if and only if there exist $i,j,k \in \{1,2,\ldots,n\}$, $i \neq j \neq k$, such that:*

$$\sin\beta_i\sin(\beta_k-\beta_j+\gamma(k-j))-\sin\beta_j\sin(\beta_k-\beta_i+\gamma(k-i))+\sin\beta_k\sin(\beta_j-\beta_i+\gamma(j-i)) \neq 0. \tag{4.4}$$

Proof : *See (Morbidi and Bretagne 2018).* ■

In the next proposition, we use geometric arguments to show that, actually, only two types are admissible for nontrivial n-robot formations.

4.2. PROPOSITION. *Under the previous assumptions, a formation of $n \geq 3$ unicycles defined by n distance constraints is either of type $(1,\,1)$ or $(1,\,2)$.*

Proof : *See (Morbidi and Bretagne 2018)* ■

To make the link with the special configurations studied in Sect. 4.1.3, we list below their n-robot counterparts.

1. Let $\beta_1 = \beta_1(t)$ (free), $\beta_k = \beta_1 + 2\,\gamma\,(k-1)$, $k \in \{2,3,\ldots,n\}$. The orientation of the n m-wheels is coupled and the formation behaves as an *n-wheel synchro-drive robot* (cf. Case 1 in Sect. 4.1.3, "Formation of three unicycles").

2. Let $\beta_1 = \beta_1(t)$ (free), $\beta_k = \beta_1 - \gamma\,(k-1)$, $k \in \{2, 3, \ldots, n\}$. As in the previous case, the orientation of the n m-wheels of the m-robot is coupled (cf. Case 1 and 2 in Sect. 4.1.3, "Formation of three unicycles").

3. For n even, let $\beta_1 = \beta_1(t)$ (free), $\beta_2 = \gamma(n-2)/4$, …, $\beta_{(n/2)-1} = \gamma$, $\beta_{n/2} = \gamma/2$, $\beta_{(n/2)+1} = \pi$, $\beta_{(n/2)+2} = \pi - \gamma/2$, $\beta_{(n/2)+3} = \pi - \gamma$, …, $\beta_n = \pi - \gamma(n-2)/4$, and for n odd, let $\beta_1 = \beta_1(t)$ (free), $\beta_2 = \gamma(n-2)/4$, …, $\beta_{(n+1)/2-2} = 5\,\gamma/4$, $\beta_{(n+1)/2-1} = 3\,\gamma/4$, $\beta_{(n+1)/2} = \gamma/4$, $\beta_{(n+1)/2+1} = \pi - \gamma/4$, $\beta_{(n+1)/2+2} = \pi - 3\,\gamma/4$, …, $\beta_n = \pi - \gamma(n-2)/4$. The axles of m-wheels $2, 3, \ldots, n$ concur at the center of m-wheel 1 (the ICR), and the formation can only rotate about this fixed point (cf. Case 3 in Sect. 4.1.3, "Formation of three unicycles").

4. Let $\beta_1 = \beta_2 = \ldots = \beta_n = 0$. The m-wheels are tangent to the circle circumscribing the polygonal "chassis" of the m-robot. The only possible motion is a rotation about the center O_F of the formation (cf. Case 4 in Sect. 4.1.3, "Formation of three unicycles").

4.2. REMARK (RECONFIGURABLE FORMATIONS). *With reference to Fig. 4.1, if the polar coordinates* $(\ell,\, \alpha)$ *used to express the position of the center* A *of the wheel with respect to* O_R *are* time-varying, *condition (4.1) becomes:*

$$\big[\cos(\alpha+\beta),\ \sin(\alpha+\beta),\ \ell\sin\beta\big]\,\mathbf{R}^T(\theta)\,\dot{\mathbf{q}} \;+\; \big[\cos\beta,\ \ell\sin\beta\big]\,\dot{\boldsymbol{\tau}} \;=\; 0, \tag{4.5}$$

where $\boldsymbol{\tau} = [\ell,\, \alpha]^T$. *Eq. (4.5) can be obtained by plugging the coordinates of point* A, $(x_A,\, y_A) = (x \,+\, \ell\cos(\theta +\, \alpha),\, y_A = y \,+\, \ell\sin(\theta+\alpha))$, *into the nonslip condition* $\dot{y}_A\cos(\theta+\alpha+\beta+\pi/2) \,-\, \dot{x}_A\sin(\theta+\alpha+\beta+\pi/2) = 0$. *The generalized nonslip condition* (4.5) *can be used to study the mobility of reconfigurable formations, i.e. formations whose shape can vary over time. While formations able to contract/expand are relevant in the applications, the extension of our taxonomy to these structures goes beyond the scope of this study, and it is left to future work.* ⋄

4.1.4 Trajectory tracking of a formation of unicycles

In this section, we adapt the trajectory-tracking control law via dynamic feedback originally introduced in (Thuilot et al. 1996, Sect. III) for mobile robots equipped with multiple steering wheels, to distance-bearing formations of unicycles. This task is made easy by the proposed notion of macro-robot. Note that we are not interested here in a stabilization problem, i.e. we assume that the unicycles satisfy the formation constraints *initially*, and such constraints should be respected at all times.

Following (Thuilot et al. 1996), let us describe the position of the ICR of the formation using the polar coordinates $(1/\sigma,\, \zeta)$ expressed in the frame centered at m-wheel 1

(whose axes are parallel to the longitudinal and lateral axes of the "chassis" of the m-robot, respectively). In particular, $1/\sigma$ is the (signed) instantaneous radius of curvature of the m-robot viewed from the frame of m-wheel 1 and ζ is the angle between the x-axis of the reference frame of m-wheel 1 and its axle $\mathbf{r} = [\cos\zeta,\, \sin\zeta]^T$, namely,

$$\sigma = \begin{cases} 0, & \text{if the ICR is at infinity,} \\ \dfrac{\operatorname{sgn}(\langle \mathbf{r},\, \mathbf{w}_{1,\text{ICR}} \rangle)}{\|\mathbf{w}_{1,\text{ICR}}\|}, & \text{otherwise,} \end{cases}$$

where $\mathbf{w}_{1,\text{ICR}}$ is the vector from the center of m-wheel 1 to the ICR, and $\operatorname{sgn}(\cdot)$ denotes the sign function (see Fig. 4 in (Thuilot et al. 1996)). Let us introduce the state vector $\mathbf{x} = [x,\, y,\, \theta,\, \zeta,\, \sigma]^T$ and the input vector $\mathbf{u} = [\eta_1,\, \eta_2,\, \eta_3]^T$, where $[x,\, y,\, \theta]^T$ now denotes the pose of the frame of m-wheel 1 with respect to $\{O_I;\, x_I,\, y_I\}$, η_1 is the linear velocity of m-wheel 1, $\eta_2 = \dot{\zeta}$ and $\eta_3 = \dot{\sigma}$. We can then write the following state-space model for the m-robot:

$$\dot{\mathbf{x}} = \begin{bmatrix} \sin(\theta+\zeta) & 0 & 0 \\ -\cos(\theta+\zeta) & 0 & 0 \\ \sigma & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{u}\,. \tag{4.6}$$

Note that this model is completely independent of the number of m-wheels and of their disposition in the "chassis" of the m-robot. As shown in (Thuilot et al. 1996, Th. 1), system (4.6) can be fully dynamically linearized if SWCs (system's intrinsic singularities) and rest configurations (singularities of the linearizing control law) are not met. A candidate output vector which provides such a linearization is $\boldsymbol{\varphi} = [x,\, y,\, \theta/\sigma_F]^T$ where $\sigma_F \triangleq \min\{\sigma_2,\, \sigma_3,\, \ldots,\, \sigma_n\}$, being $(1/\sigma_j,\, \zeta_j)$, $j \in \{2, 3, \ldots, n\}$, the polar coordinates of m-wheel j expressed in the frame of m-wheel 1. Note that $\sigma_F > 0$ is a characteristic constant of the formation considered. Using the dynamic extension algorithm, the input η_1 can be delayed. This leads to the following *extended model* for the m-robot (cf. eq. (4.6)),

$$\dot{\mathbf{x}}_e = \begin{bmatrix} \eta_1 \sin(\theta+\zeta) \\ -\eta_1 \cos(\theta+\zeta) \\ \sigma\, \eta_1/\sigma_F \\ \mathbf{0}_{3\times 1} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{3\times 3} \\ \mathbf{I}_{3\times 3} \end{bmatrix} \mathbf{u}_e, \tag{4.7}$$

where $\mathbf{x}_e = [x,\, y,\, \theta/\sigma_F,\, \zeta,\, \sigma,\, \eta_1]^T$ and $\mathbf{u}_e = [\eta_2,\, \eta_3,\, \eta_4]^T$ are the extended state and input vectors, respectively. Let now $\boldsymbol{\varphi}_{\text{ref}}(t) = [x_{\text{ref}}(t),\, y_{\text{ref}}(t),\, \theta_{\text{ref}}(t)/\sigma_F]^T \in \boldsymbol{\mathcal{C}}^2$ be the reference trajectory that the output vector $\boldsymbol{\varphi}(t)$ should follow, and let $\mathbf{H}_1, \mathbf{H}_2 \in \mathbb{R}^{3\times 3}$ be two diagonal positive-definite gain matrices. Then, as long as $\eta_1 \neq 0$ (i.e. with the proviso that singular rest configurations are avoided), trajectory tracking for system (4.7)

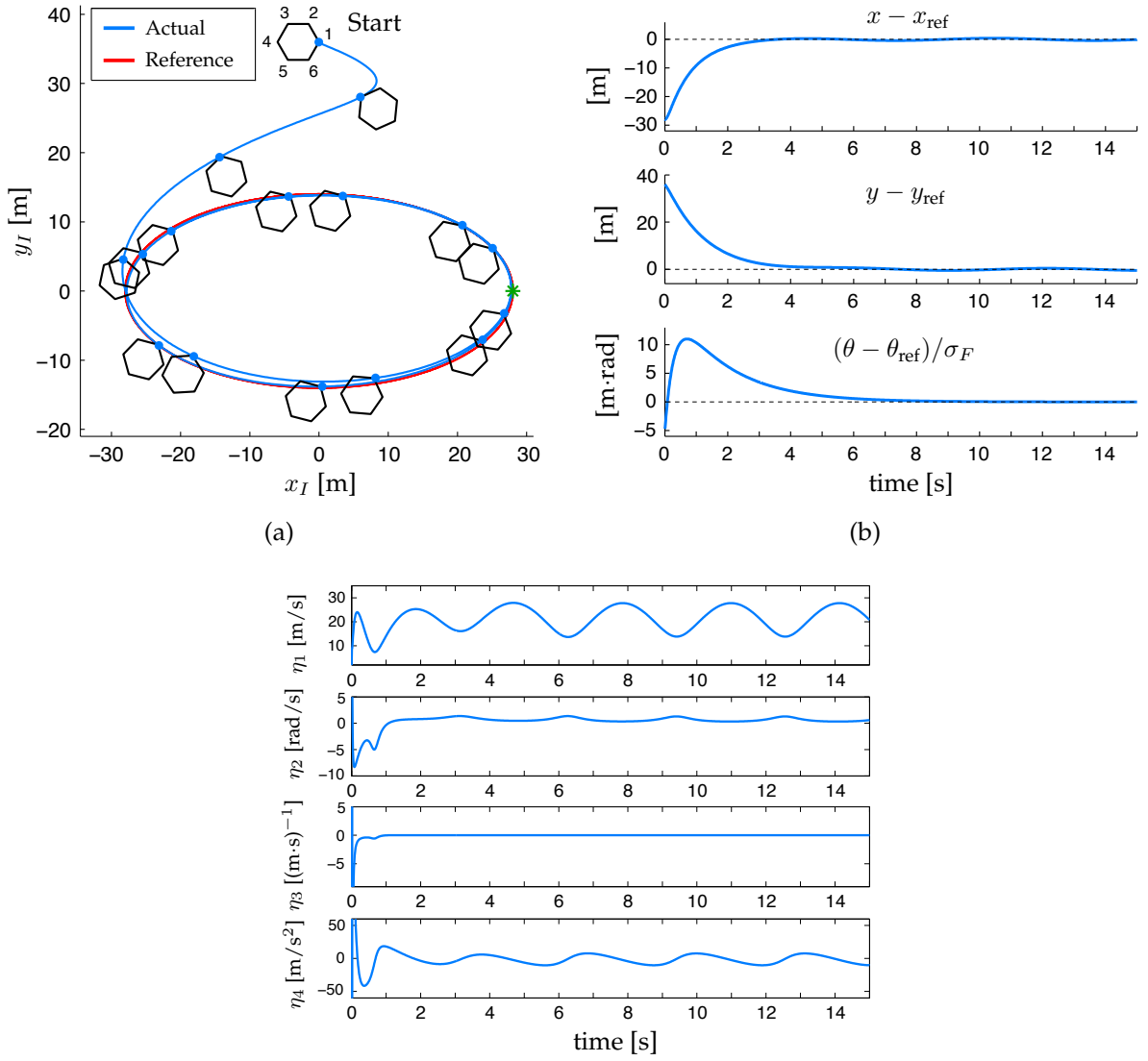


Figure 4.5: *(a) Trajectory of robot 1 of the hexagonal formation of unicycles (blue, actual; red, reference): to provide a time reference, the type (1, 1) formation is depicted every second; (b) Time evolution of the trajectory-tracking error $\boldsymbol{\varphi}(t) - \boldsymbol{\varphi}_{\text{ref}}(t)$; (c) Time history of $\eta_1(t)$ and $\mathbf{u}_e(t)$.*

is achieved via:

$$\mathbf{u}_e \;=\; \boldsymbol{\Delta}^{-1}(\mathbf{x}_e)\Big(\mathbf{u}_a - \begin{bmatrix} \sigma\,\eta_1^2\cos(\theta+\zeta) \\ \sigma\,\eta_1^2\sin(\theta+\zeta) \\ 0 \end{bmatrix}\Big), \tag{4.8}$$

where

$$\mathbf{\Delta}(\mathbf{x}_e) \;=\; \begin{bmatrix} \eta_1 \cos(\theta+\zeta) & 0 & \sin(\theta+\zeta) \\ \eta_1 \sin(\theta+\zeta) & 0 & -\cos(\theta+\zeta) \\ 0 & \eta_1/\sigma_F & \sigma/\sigma_F \end{bmatrix},$$

and the auxiliary control input $\mathbf{u}_a$ is designed as,

$$\mathbf{u}_a \;=\; -\mathbf{H}_1(\boldsymbol{\varphi} - \boldsymbol{\varphi}_{\text{ref}}) \;-\; \mathbf{H}_2(\dot{\boldsymbol{\varphi}} - \dot{\boldsymbol{\varphi}}_{\text{ref}}) \;+\; \ddot{\boldsymbol{\varphi}}_{\text{ref}}\,.$$

Note that the control law (4.8) is invariant to the number of robots in the formation, and given $\sigma_2,\, \sigma_3,\, \ldots, \sigma_n$, no communication exchange between the robots is needed at runtime (in this sense, it can thus be regarded as decentralized).

Case study: hexagonal formation of unicycles

We evaluated the effectiveness of the trajectory tracking control law (4.8) on a type $(1,\, 1)$ formation of six unicycles disposed at the vertices of a regular hexagon of side $d = 3$ m. Since $n = 6$, then $\gamma = \pi/3$ and $r_c = d = 3$ m. Moreover, $[\sigma_2,\, \sigma_3,\, \sigma_4,\, \sigma_5,\, \sigma_6]^T = [1/d,\, 1/(\sqrt{3}\,d),\, 1/(2d),\, 1/(\sqrt{3}\,d),\, 1/d]^T$ and thus $\sigma_F = 1/(2d) = 1/6$ m^{-1}. We chose the initial condition $\mathbf{x}_e(0) = [0,\, 36,\, 0,\, 0,\, 1/2,\, 1]^T$ and selected the following proportional and derivative diagonal gain matrices: $\mathbf{H}_1 = \text{diag}(16, 6, 2)$ and $\mathbf{H}_2 = \text{diag}(14, 8, 4)$. Finally, we defined the following elliptical reference trajectory for $\boldsymbol{\varphi}(t)$,

$$\boldsymbol{\varphi}_{\text{ref}}(t) \;=\; \Big[\mu_{\text{M}} \cos(t),\; \mu_{\text{m}} \sin(t),\; \frac{\pi}{4\,\sigma_F}\Big]^T,\;\; t \in [0,\, 15] \text{ s},$$

where $\mu_{\text{M}} = 28$ m, $\mu_{\text{m}} = 14$ m are the lengths of the semi-major and semi-minor axes of the ellipse, respectively. Note that $\boldsymbol{\varphi}_{\text{ref}}(0) = [28,\, 0,\, 3\pi/2]^T$ (see the green star in Fig. 4.5(a)). Fig. 4.5(a) reports the trajectory of robot 1 (i.e. m-wheel 1) of the formation, Fig. 4.5(b) the time evolution of the trajectory-tracking error $\boldsymbol{\varphi}(t) - \boldsymbol{\varphi}_{\text{ref}}(t)$, and Fig. 4.5(c) the time history of $\eta_1(t)$ and $\mathbf{u}_e(t)$. Note that $\eta_1(t) \geq 1$, $\forall\, t \in [0,\, 15]$ s, hence no singular rest configuration is crossed by the formation. A video illustrating the simulation results is available at the address reported below[1].

4.1.5 Conclusions and future work

In this section, we have studied the mobility of distance-bearing formations of unicycle robots and introduced a new classification, corresponding to two indices $(\delta_m,\, \delta_s)$, inspired by the taxonomy in (Campion et al. 1996).

[1] `https://home.mis.u-picardie.fr/~fabio/Eng/Video/MoBr_IROS18.m4v`

In future works, more general formations lacking dihedral symmetry and other inter-robot constraints (e.g. relative positions and heading angles) will be investigated. More involved wheel's and robot's models will be studied as well: in this respect, conventional off-centered orientable wheels and car-like robots, seem the most natural candidates. Finally, we would like to establish a link between the type of a formation and the (rigidity (Anderson et al. 2008)) properties of the weighted graph which defines the constraints between the robots.

4.2 Cooperative adaptive cruise control

This section presents new decentralized optimal strategies for *Cooperative Adaptive Cruise Control* (CACC) of a car platoon under string-stability constraints. Two related scenarios are explored: in the first one, a linear-quadratic regulator in the presence of measurable disturbances is synthesized, and the string-stability of the platoon is enforced over the controller's feedback and feedforward gains. In the second scenario, H_2- and H_∞-performance criteria, respectively accounting for the desired group behavior and the string-stability of the platoon, are simultaneously achieved using the recently-proposed *compensator blending* method. An analytical study of the impact of actuation/communication delays and uncertain model parameters on the stability of the multi-vehicle system, is also conducted. The theory is illustrated via numerical simulations.

4.2.1 Introduction

Motivation and related work

Traffic congestion has become a serious issue in modern cities' life. In 2010, congestion caused urban Americans to travel 4.8 billion hours more and to purchase an extra 1.9 billion gallons of fuel, for a congestion cost of $101 billion (Schrank et al. 2011). In 2017, the average commuter wasted nearly seven full working days in extra traffic delay, which translated to over $1000 in personal costs (Schrank et al. 2019). Because of such a big impact on productivity, pollution and human welfare, a considerable effort has been devoted in the last decades toward devising innovative systems which may reduce traffic jams and improve driver's safety and comfort. This research activity, together with numerous "intelligent highway" initiatives in the U.S. (e.g., California PATH research program), Japan and Europe, has led to the development of *Adaptive Cruise Control* (ACC) systems, currently available in numerous sedans, and lately to the design of *Cooperative Adaptive Cruise Control* (CACC) systems which extend the functionality of ACC by leveraging the information exchanged via vehicle-to-vehicle and/or vehicle-to-infrastructure wireless communication.

The idea of using *optimization-based* policies for CACC is not new and dates back at least to the end of 90s. In (Sheikholeslam and Desoer 1993) the longitudinal control of each car is computed using a gradient-based descent algorithm, and no communication with the leading vehicle of the platoon is needed. In (Stanković et al. 2000) a decentralized overlapping controller is developed using the inclusion principle: possible extensions to the basic scenario are also discussed, comprising the use of reduced-order observers for estimating the state of the preceding vehicle and the identification of suit-

able stability-preserving conditions. A similar control framework is adopted in (Guo and Yue 2012), where the authors analyze the impact of range-limited sensing, assuming that the lead car broadcasts its state information, i.e. its speed and acceleration information, to all platoon members. Recently, we have witnessed a growing interest in CACC based on *Model Predictive Control* (MPC). In (Naus et al. 2008) an explicit MPC controller for "Stop-&-Go" ACC is synthesized, and its performance is evaluated by distinguishing between comfort of the resulting longitudinal vehicle behavior and behavior due to the traffic constraints. A similar MPC approach is considered in (Naus, Ploeg, de Molengraft, Heemels and Steinbuch 2010), where the tuning of the cruise controller is made simple by the parameterization of multiple performance indices. In (Li et al. 2011), a multi-objective MPC-based CACC strategy is developed for multiple trucks and tested in realistic traffic conditions. An analogous setup is considered in (Marzbanrad and Karimian 2011), where the performance of MPC is compared with that of a PD and a sliding-mode controller, in a real driving cycle. Finally, in (Stanger and del Re 2013), the authors use linear MPC for fuel-efficient CACC.

A significant body of research in the CACC literature has also focused on robustness and stability issues, and notably on the so-called *string stability* of a car platoon. A platoon is said string stable under an assigned control policy, if oscillations are attenuated upstream the traffic flow. In (Liu et al. 2001, Swaroop and Rajagopal 2001), early studies were conducted concerning the effect of communication delays on the string stability. A similar analysis has been recently carried out in (Naus, Vugts, Ploeg, van de Molengraft and Steinbuch 2010) in the frequency domain with heterogeneous vehicles, under a simple PD control. In (Middleton and Braslavsky 2010), sufficient conditions are given that imply a lower bound on the peak of the frequency-response magnitude of the transfer function mapping a disturbance to the leading vehicle to a vehicle in the chain. This bound quantifies the effect of spacing policy, inter-vehicle communication policy, and vehicle settling response performance. Finally, in (Öncü et al. 2011), the problem of regulating inter-vehicle distances in a car platoon is approached from a networked-system perspective. Tradeoffs between CACC performance and network specifications are pointed out, and a study of the impact of network-induced effects on string stability is conducted.

Original contributions

After an introductory section devoted to the modeling of a car platoon adapted from (Li et al. 2011), Sect. 4.2.3 presents original results concerning the *decentralized optimal* CACC of a team of n vehicles under string-stability constraints and a constant-time headway spacing policy. The CACC problem is approached here from two different perspectives. In the first scenario, an infinite-time linear-quadratic regulator in the pres-

ence of measurable disturbances is synthesized and the string-stability of the platoon is enforced over the regulator's feedback and feedforward gains. In the second scenario, we simultaneously achieve H_2- and H_∞-performance criteria, which respectively dictate the desired group behavior and string-stable behavior of the platoon, by using the *compensator blending method* recently proposed in (Blanchini et al. 2008). This method is more intuitive and simpler to implement than the classical recursive approaches to mixed H_2/H_∞ optimal control (Colaneri et al. 1997), lately used in (Maschuw et al. 2011) to design a constant-spacing CACC strategy for a chain of trucks.

As an aside, it is worth pointing out here that unlike the MPC methods described in Sect. 4.2.1, state and input constraints cannot be handled by the strategies described below. However, differently from those methods, the relative simplicity of our control design procedures allowed us to establish insightful *analytical conditions* for the solvability of the *optimal* CACC problem with string stability, both in the "nominal case" and in the presence of actuation/communication delays and, for the first time, uncertain model parameters. In Sect. 4.2.4, the proposed theoretical results are illustrated via numerical simulations, and finally, in Sect. 4.2.5, the original contributions are summarized and possible avenues for future research are outlined.

4.2.2 Modeling of the platoon

Compensation of nonlinear longitudinal dynamics

Consider a platoon of n identical cars moving in one dimension, where vehicle 1 is the *leader* of the platoon and v_1, a_1, ..., v_n, a_n denote the velocity and acceleration of the n cars, respectively. In the following, we will assume that a_1 is an *assigned* acceleration profile.

As it is known, the longitudinal dynamics of a car is nonlinear and its main features include the static nonlinearity of engine torque maps, time-varying gear position and aerodynamic drag force. Following (Li et al. 2011), we will avail ourselves of a hierarchical controller for each vehicle, consisting of a *lower-level* and an *upper-level* controller, as illustrated in Fig. 4.6. The lower-level controller determines the value of the accelerator pedal position ($a_{i,\,\mathrm{accl}}$) and brake pressure ($\mathcal{P}_{i,\,\mathrm{brk}}$) of i-th car, $i \in \{2, \ldots, n\}$, so that the desired acceleration $a_{i,\mathrm{des}}$ is tracked by the actual acceleration a_i. On the other end, the upper level controller determines the desired longitudinal acceleration according to the inter-vehicle and vehicle i's internal variables, which include the engine speed, gear ratio and car's speed and acceleration. We assume that the internal variables are all measured by the on-board car sensors (cf. Fig. 4.6).

The inter-vehicle variables are the relative distance d_i between vehicle $i-1$ and vehicle i and the speed error $\Delta v_i = v_{i-1} - v_i$, which are measured by a radar mounted

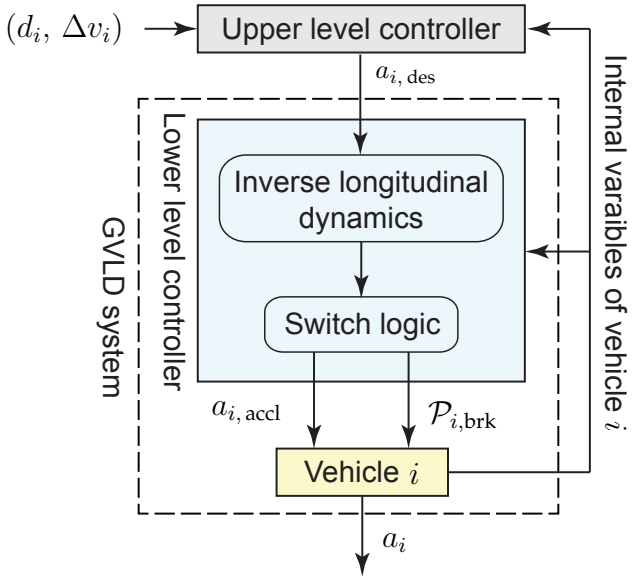


Figure 4.6: *Hierarchical control architecture of vehicle i. In the lower-level controller, a switching logic is adopted to avoid simultaneous actions from the drive train and braking system (see (Li et al. 2011) for more details).*

in front of the car. When designing the lower-level controller, one of the challenges is the presence of several nonlinearities coming from engine, transmission, and aerodynamic drag. To compensate for them, following (Li et al. 2011), the inverse-dynamics control design method is used here. The lower-level controller together with vehicle i, then yield a new plant with input $a_{i,\,\mathrm{des}}$ and output a_i, called *Generalized Vehicle Longitudinal Dynamic* (GVLD) system, described by,

$$a_i(s) \;=\; \frac{K_{\mathrm{L}}}{T_{\mathrm{L}}\, s + 1}\; a_{i,\,\mathrm{des}}(s), \quad i \in \{2, \ldots, n\}, \tag{4.9}$$

where $K_{\mathrm{L}} > 0$ is the system gain (ideally equal to 1), and T_{L} is the time constant of GVLD.

Car-following model

In order to design the *upper-level controller*, a car-following model is built by combining the GVLD system and the inter-vehicular longitudinal dynamics. For the inter-vehicular dynamics, two state variables are of interest: the *clearance error* $\Delta\, d_i(t) = d_i(t) - d_{i,\,\mathrm{des}}(t)$ and the speed error Δv_i, where $d_{i,\,\mathrm{des}}(t)$ denotes driver's *desired inter-vehicle distance*, cf. (Li et al. 2011). Various models for $d_{i,\,\mathrm{des}}$ have been proposed in the literature: we adopt here the popular *constant-time headway* spacing policy

$$d_{i,\,\mathrm{des}}(t) \;=\; \tau_h\, v_i(t) \;+\; d_0,$$

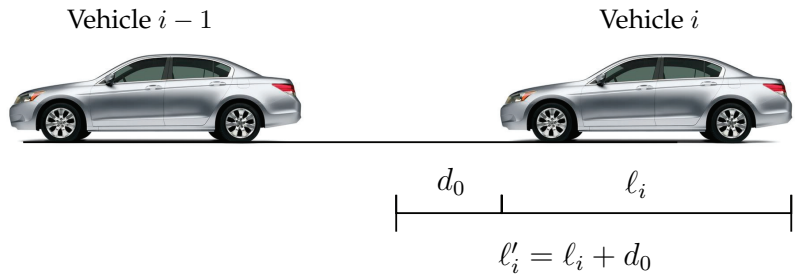


Figure 4.7: *Two vehicles in the platoon: ℓ_i is the actual length of vehicle i, d_0 is the desired distance at standstill, and $\ell_i' = \ell_i + d_0$ is the "extended length" of vehicle i, cf. Sect. 4.2.3.*

where τ_h is the *nominal time headway* and d_0 is the *desired distance at standstill* (Naus, Vugts, Ploeg, van de Molengraft and Steinbuch 2010).

Note that d_0 can be regarded as an extension of the length ℓ_i of vehicle i (cf. (Naus, Vugts, Ploeg, van de Molengraft and Steinbuch 2010) and see Fig. 4.7), and we can redefine the vehicle's length as $\ell_i' = \ell_i + d_0$. Hence, in the following, d_0 will be neglected. By collecting the inter-vehicular dynamics and equation (4.9) together, we end up with the following linear time-invariant system (Li et al. 2011),

$$\dot{\mathbf{x}}_i = \mathbf{A}\,\mathbf{x}_i + \mathbf{B}\,u_i + \mathbf{G}\,z_i, \quad i \in \{2, \ldots, n\}, \tag{4.10}$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & -\tau_h \\ 0 & 0 & -1 \\ 0 & 0 & -1/T_\mathrm{L} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ K_\mathrm{L}/T_\mathrm{L} \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \tag{4.11}$$

$\mathbf{x}_i = [\Delta d_i,\, \Delta v_i,\, a_i]^T \in \mathbb{R}^3$ is the state of the system, $u_i = a_{i,\mathrm{des}} \in \mathbb{R}$ is the control input, and $z_i = a_{i-1} \in \mathbb{R}$ is a measurable disturbance. In the following, we will assume the transmission of the acceleration a_{i-1} from vehicle $i-1$ to vehicle i.

4.2.3 String-stable optimal cooperative adaptive cruise control

In this section, we present two *decentralized* optimal CACC strategies, which preserve the string stability of the car platoon.

LQ regulation with guaranteed string stability

In order to specify the desired behavior of the platoon, let us introduce the following optimal control problem,

$$\begin{aligned} \min_{u_i} \quad & \int_0^\infty \big(\mathbf{x}_i^T \mathbf{Q}\, \mathbf{x}_i \,+\, r\, u_i^2 \big)\, dt, \\ \text{s.t.} \quad & \dot{\mathbf{x}}_i = \mathbf{A}\, \mathbf{x}_i + \mathbf{B}\, u_i + \mathbf{G}\, z_i, \end{aligned} \qquad i \in \{2, \ldots, n\}, \tag{4.12}$$

where $\mathbf{Q} \succeq \mathbf{0}$ and $r > 0$ are suitable weights on the state $\mathbf{x}_i$ and input u_i of system (4.10). This is an infinite-time linear-quadratic (LQ) regulation problem in the presence of the measurable disturbance z_i. If we assume that z_i is constant (cf. (Stanković et al. 2000)), this problem admits the following closed-form solution (Anderson and Moore 1989, Sect. 4.3),

$$u_i^* = -\, r^{-1}\, \mathbf{B}^T (\mathbf{P}\, \mathbf{x}_i \,+\, \mathbf{q}_i), \tag{4.13}$$

where $\mathbf{P} \succeq \mathbf{0}$ is the solution of the algebraic Riccati equation,

$$\mathbf{P}\, \mathbf{A} \,+\, \mathbf{A}^T \mathbf{P} \,-\, r^{-1}\, \mathbf{P}\, \mathbf{B}\, \mathbf{B}^T \mathbf{P} \,+\, \mathbf{Q} \,=\, \mathbf{0},$$

and $\mathbf{q}_i = [(\mathbf{A} - r^{-1}\, \mathbf{B}\, \mathbf{B}^T \mathbf{P})^T]^{-1}\, \mathbf{P}\, \mathbf{G}\, z_i$. Note that the control law (4.13) can be rewritten more compactly as,

$$u_i^* \,=\, \mathbf{k}^T \mathbf{x}_i \,+\, k_F\, z_i, \tag{4.14}$$

where

$$\begin{aligned} \mathbf{k}^T \,&=\, [k_1,\, k_2,\, k_3] \,\triangleq\, -r^{-1}\, \mathbf{B}^T\, \mathbf{P}, \\ k_F \,&\triangleq\, -r^{-1}\, \mathbf{B}^T [(\mathbf{A} - r^{-1}\, \mathbf{B}\, \mathbf{B}^T\, \mathbf{P})^T]^{-1}\, \mathbf{P}\, \mathbf{G}. \end{aligned} \tag{4.15}$$

By substituting equation (4.14) into system (4.10), we finally obtain the following closed-loop dynamics,

$$\dot{\mathbf{x}}_i \,=\, (\mathbf{A} + \mathbf{B}\, \mathbf{k}^T)\, \mathbf{x}_i \,+\, (\mathbf{B}\, k_F + \mathbf{G})\, z_i, \tag{4.16}$$

which is the basis for our forthcoming developments.

The following definition introduces the notion of string stability used in this section (Naus, Vugts, Ploeg, van de Molengraft and Steinbuch 2010).

4.1. DEFINITION (STRING STABILITY). *Consider the following transfer function,*

$$\Lambda_i(s) = \frac{a_i(s)}{a_{i-1}(s)}, \quad i \in \{2, \ldots, n\}, \tag{4.17}$$

where $a_i(s)$ and $a_{i-1}(s)$, denote the Laplace transforms of the acceleration signals $a_i(t)$ and $a_{i-1}(t)$, respectively. A sufficient condition for the string stability *of a platoon of n identical cars is that,*

$$\|\Lambda_i(j\,\omega)\|_\infty \leq 1, \quad i \in \{2, \ldots, n\}, \tag{4.18}$$

where $\|\Lambda_i(j\,\omega)\|_\infty \triangleq \sup_\omega |\Lambda_i(j\,\omega)|$ denotes the H_∞ norm of the transfer function in (4.17). $\diamond$

In other words, the longitudinal dynamics of a platoon is string stable whether oscillations are not amplified upstream the traffic flow. The next proposition provides sufficient conditions on the feedback and feedforward control gains in (4.15), for the string stability of the car platoon. These conditions are successively extended to the case of constant *communication delays* among the vehicles and within the individual GVLD systems.

4.3. PROPOSITION (STRING-STABILITY CONDITIONS). *Consider system (4.16). The car platoon is string stable if the following two inequalities are satisfied:*

$$\begin{aligned} &(K_L k_3 - 1)^2 - 2\,T_L\,K_L(\tau_h k_1 + k_2) - K_L^2\,k_F^2 \geq 0\,, \\ &2k_1(K_L k_3 - 1) + k_1 K_L(\tau_h^2 k_1 + 2\,(\tau_h k_2 + k_F)) \geq 0\,. \end{aligned} \tag{4.19}$$

Proof : *See (Morbidi et al. 2013).* ■

Note that it is generally possible to enforce the conditions in (4.19) by properly tuning the weights $\mathbf{Q}$ and r in (4.12). Next, we will try to repeat the previous analysis in the more challenging scenario in which the signal $z_i = a_{i-1}$ is transmitted between vehicle $i-1$ and vehicle i with a constant delay θ, and that a constant actuator's communication delay ϕ is present in the GVLD system. It is immediate to verify that under these conditions, equation (4.10), for $i \in \{2, \ldots, n\}$, transforms into:

$$\dot{\mathbf{x}}_i(t) \;=\; \mathbf{A}\,\mathbf{x}_i(t) + \mathbf{B}\,u_i(t-\phi) + \mathbf{G}\,z_i(t-\theta). \tag{4.20}$$

Let us now choose a control input of the form,

$$u_i(t) \;=\; \mathbf{k}^T\mathbf{x}_i(t) \;+\; k_F\,z_i(t-\theta). \tag{4.21}$$

Following the same outline of the proof of Proposition 4.3, from (4.20) we obtain

$$\left[s - \Big(\frac{K_{\mathrm{L}}\,k_3\,e^{-\phi\,s} - 1}{T_{\mathrm{L}}}\Big)\right] a_i(s) \;=\; \frac{K_{\mathrm{L}}}{T_{\mathrm{L}}}\,e^{-\phi\,s}\big[k_1\Delta\,d_i(s) + \;k_2\,\Delta\,v_i(s) + k_F\,a_{i-1}(s)\,e^{-\theta\,s}\big],$$

and the transfer function:

$$\Lambda_i(s) = \frac{K_{\mathrm{L}}\,e^{-\phi s}(k_1 + k_2\,s + k_F\,s^2\,e^{-\theta s})}{T_{\mathrm{L}}s^3 + s^2 + K_{\mathrm{L}}\,e^{-\phi s}[-k_3 s^2 + (k_1\tau_h + k_2)s + k_1]}.$$

If we now impose $|\Lambda_i(j\omega)| \leq 1$, $\forall\,\omega > 0$, we obtain the following quasipolynomial inequality in the variable ω:

$$\begin{aligned} &T_{\mathrm{L}}^2\,\omega^4 + 2K_{\mathrm{L}}\,k_3\,T_{\mathrm{L}}\,\sin(\phi\,\omega)\,\omega^3 + [1 + (k_3^2 - k_F^2)K_{\mathrm{L}}^2 \;-\; 2K_{\mathrm{L}}\cos(\phi\,\omega)(k_3 + T_{\mathrm{L}}(k_1\tau_h + k_2))]\,\omega^2 \\ &- 2[k_2\,k_F K_{\mathrm{L}}^2\sin(\theta\,\omega) + K_{\mathrm{L}}\sin(\phi\,\omega)(k_1(\tau_h - T_{\mathrm{L}}) + k_2)]\,\omega \;+\; K_{\mathrm{L}}[-k_2^2\,K_{\mathrm{L}} + 2k_1 K_{\mathrm{L}}\,k_F\cos(\theta\,\omega) \\ &+ 2K_{\mathrm{L}}\,k_1 k_3 \;+\; K_{\mathrm{L}}(k_1\tau_h + k_2)^2 - 2k_1\cos(\phi\,\omega)] \geq 0. \end{aligned} \tag{4.22}$$

The study of the feasibility of (4.22) is made complicated by the presence of the sinusoidal and cosinusoidal terms, and suitable approximations to these functions need to be introduced in order to establish conditions on the gains of controller (4.21), similar to those in (4.19). A simple option, consists of using the following Maclaurin series expansions of the cosine and sine functions $\cos(\alpha\omega) \simeq 1 - (\alpha\omega)^2/2!$, $\sin(\alpha\omega) \simeq \alpha\omega - (\alpha\omega)^3/3!$, $\alpha \in \{\theta, \phi\}$, under the assumption of "small" $\alpha\omega$. Inequality (4.22) can thus be rewritten as:

$$\begin{aligned}
&-\tfrac{1}{3}\,K_{\rm L}k_3T_{\rm L}\phi^3\omega^6 + \left\{T_{\rm L}^2 + 2K_{\rm L}\,k_3\,T_{\rm L}\,\phi + K_{\rm L}[k_3 + T_{\rm L}(k_1\tau_h + k_2)]\,\theta^2\right.\\
&\left.+\tfrac{1}{3}\,[k_2\,k_F\,K_{\rm L}^2\,\theta^3 + K_{\rm L}(k_1(\tau_h - T_{\rm L}) + k_2)\phi^3]\right\}\omega^4\\
&+\left\{(K_{\rm L}\,k_3 - 1)^2 - 2\,T_{\rm L}\,K_{\rm L}(\tau_h\,k_1 + k_2) - K_{\rm L}^2\,k_F^2\right.\\
&\left.-K_{\rm L}^2\,k_F\,\theta\,(2k_2 + \theta k_1) - 2K_{\rm L}\,[k_2 + k_1(\tau_h - T_{\rm L})]\,\phi + K_{\rm L}\,k_1\,\phi^2\right\}\omega^2\\
&+2K_{\rm L}\,k_1(K_{\rm L}k_3 - 1) \;+\; k_1K_{\rm L}^2\,[\tau_h^2\,k_1 + 2(\tau_h k_2 + k_F)] \geq 0.
\end{aligned} \tag{4.23}$$

A sufficient condition for the nonnegativity of the six-order polynomial on the left-hand side of (4.23), is that all its coefficients are nonnegative, from which we deduce the following four inequalities:

$$-\,k_3\,\phi^3 \geq 0,$$

$$T_{\rm L}^2 + 2K_{\rm L}k_3T_{\rm L}\,\phi + K_{\rm L}[k_3 + T_{\rm L}(k_1\tau_h + k_2)]\theta^2 + \tfrac{1}{3}[k_2\,k_F\,K_{\rm L}^2\,\theta^3 + K_{\rm L}(k_1\tau_h + k_2 - k_1T_{\rm L})\phi^3] \geq 0,$$

$$\begin{aligned}
&(K_{\rm L}\,k_3 - 1)^2 - 2\,T_{\rm L}\,K_{\rm L}(\tau_h\,k_1 + k_2) - K_{\rm L}^2\,k_F^2 - K_{\rm L}^2\,k_F\,\theta\,(2k_2 + \theta k_1)\\
&-2K_{\rm L}[k_2 + k_1(\tau_h - T_{\rm L})]\,\phi \;+ K_{\rm L}\,k_1\,\phi^2 \geq 0,
\end{aligned}$$

$$2k_1(K_{\rm L}k_3 - 1) + k_1K_{\rm L}(\tau_h^2\,k_1 + 2(\tau_h k_2 + k_F)) \geq 0.$$

These inequalities are approximate sufficient conditions for the string stability of the car platoon in the presence of the constant communication delays θ and ϕ.

Simultaneous H_2- and H_∞-performance achievement via compensator blending

In this section, we present a decentralized CACC strategy alternative to that considered above. By relying on the *compensator blending* method proposed in (Blanchini et al. 2008), we are interested in jointly solving two problems: minimize the H_2-performance index in (4.12) and achieve an H_∞ criterion (cf. equation (4.18)) accounting for the string-stable behavior of the platoon. We will separately design the H_2 and H_∞ control laws $u_i = \mathbf{k}_2^T\,\mathbf{x}_i$, $u_i = \mathbf{k}_\infty^T\,\mathbf{x}_i$, $i \in \{2, \ldots, n\}$, and obtain a (dynamic) compensator of the form,

$$\boldsymbol{\mathcal{K}}_i : \begin{cases} \dot{\mathbf{z}}_i = \mathbf{A}_{\mathcal{K},i}\,\mathbf{z}_i \;+\; \mathbf{B}_{\mathcal{K},i}\,\mathbf{x}_i, \\ u_i = \mathbf{C}_{\mathcal{K},i}\,\mathbf{z}_i \;+\; \mathbf{D}_{\mathcal{K},i}\,\mathbf{x}_i, \end{cases} \tag{4.24}$$

which *simultaneously* fulfills the H_2 and H_∞ criteria. To this end, let us introduce the following system,

$$\boldsymbol{\mathcal{G}}_i : \begin{cases} \dot{\mathbf{x}}_i = \mathbf{A}\,\mathbf{x}_i \,+\, \mathbf{B}\,u_i \,+\, \mathbf{x}_i(0)\,z_{i,2} \,+\, \mathbf{G}\,z_{i,\infty}, \\ \mathbf{y}_{i,2} = \mathbf{C}_2\,\mathbf{x}_i + \mathbf{D}_2\,u_i, \quad i \in \{2,\ldots,n\}, \\ y_{i,\infty} = \mathbf{C}_\infty\,\mathbf{x}_i, \end{cases}$$

where

$$z_{i,\,2}(t) = \delta(t),\;\; \mathbf{C}_2 = \begin{bmatrix} \mathbf{Q}^{1/2} \\ \mathbf{0}_{1\times 3} \end{bmatrix},\;\; \mathbf{D}_2 = \begin{bmatrix} \mathbf{0}_{3\times 1} \\ r^{1/2} \end{bmatrix},\; z_{i,\,\infty}(t) \,=\, z_i(t),\;\; \mathbf{C}_\infty \,=\, \begin{bmatrix} 0 & 0 & 1 \end{bmatrix},$$

$\mathbf{x}_i(0)$ is the initial state, $\delta(t)$ is the Dirac's delta, $\mathbf{0}_{1\times 3}$ is an 1×3 vector of zeros, and the subscripts "2" and "∞" refer to the H_2- and H_∞-performance indices, respectively.

Note that the compensator blending procedure in (Blanchini et al. 2008), is valid under the assumption of a stabilizable pair $(\mathbf{A},\, \mathbf{B})$ (in our specific case, $(\mathbf{A},\, \mathbf{B})$ is indeed controllable, cf. equation (4.11)), and of full column-rank matrices $[\mathbf{x}_i(0) \quad \mathbf{G}]$, $i \in \{2,\ldots,n\}$. An additional requirement is that $\mathbf{k}_2$ and $\mathbf{k}_\infty$ are *stabilizing*.

Since the regulator $\mathbf{k}_2$ can be easily synthesized, in what follows we will limit ourselves to the design of the more challenging $\mathbf{k}_\infty = [k_{\infty,1},\, k_{\infty,2},\, k_{\infty,3}]^T$ (which we *cannot* straightforwardly calculate using state-of-the-art methods owing to our peculiar choice of the output matrix $\mathbf{C}_\infty$). Note that the characteristic polynomial of matrix,

$$\widehat{\mathbf{A}} \,=\, \mathbf{A} \,+\, \mathbf{B}\,\mathbf{k}_\infty^T = \begin{bmatrix} 0 & 1 & -\tau_h \\ 0 & 0 & -1 \\ \frac{K_L k_{\infty,1}}{T_L} & \frac{K_L k_{\infty,2}}{T_L} & \frac{K_L k_{\infty,3}-1}{T_L} \end{bmatrix}, \tag{4.25}$$

is

$$\det(\lambda \mathbf{I}_3 - \widehat{\mathbf{A}}) \,=\, \lambda^3 - \left(\frac{K_L k_{\infty,3} - 1}{T_L}\right)\lambda^2 \,+\, \frac{K_L}{T_L}\,(k_{\infty,1}\tau_h + k_{\infty,2})\lambda \,+\, \frac{K_L}{T_L} k_{\infty,1}. \tag{4.26}$$

Hence, from the Routh-Hurwitz stability criterion, $\widehat{\mathbf{A}}$ is Hurwitz (and thus $\mathbf{k}_\infty$ is stabilizing), if the following four inequalities are satisfied:

$$\begin{aligned} &K_L\,k_{\infty,3} < 1, \quad k_{\infty,1}\,\tau_h + k_{\infty,2} > 0, \quad k_{\infty,1} > 0, \\ &(K_L\,k_{\infty,3} - 1)(k_{\infty,1}\,\tau_h + k_{\infty,2}) \,+\, k_{\infty,1}\,T_L < 0. \end{aligned} \tag{4.27}$$

Moreover, we have that:

$$\Lambda_i(s) \,=\, \mathbf{C}_\infty(s\,\mathbf{I}_3 - \widehat{\mathbf{A}})^{-1}\,\mathbf{G} \,=\, \frac{K_{\mathrm{L}}(k_{\infty,1} + k_{\infty,2}\,s)}{T_{\mathrm{L}}s^3 - (K_{\mathrm{L}}k_{\infty,3} - 1)s^2 + (\tau_h k_{\infty,1} + k_{\infty,2})K_{\mathrm{L}}s + K_{\mathrm{L}}k_{\infty,1}}.$$

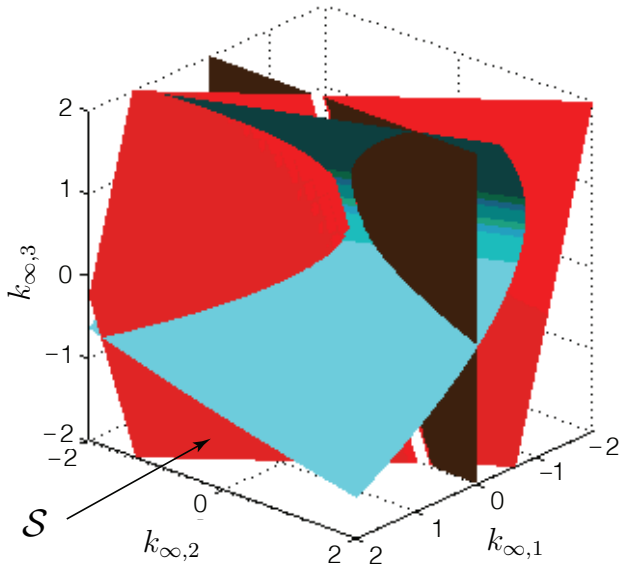


Figure 4.8: *The set $\mathcal{S}$ of all feasible regulators $\mathbf{k}_\infty$ in $[-2,\,2]^3$, for $\tau_h = 2.5$, $K_L = 1$ and $T_L = 0.45$. For this parameters' selection, $\mathcal{S}$ (in the front lower corner of the figure) is completely defined by the two inequalities in (4.28), depicted in cyan and red, respectively.*

If, as in the proof of Proposition 4.3, we now impose that $|\Lambda_i(j\omega)| \leq 1$, $\forall\, \omega > 0$, for string stability, we end up with the following two inequalities (which coincide with those in (4.19) for $k_F = 0$), which add to those in (4.27):

$$\begin{aligned} &(K_{\mathrm{L}}\, k_{\infty,3} - 1)^2 - 2\, T_{\mathrm{L}}\, K_{\mathrm{L}}(\tau_h k_{\infty,1} + k_{\infty,2}) \geq 0, \\ &2k_{\infty,1}(K_{\mathrm{L}}\, k_{\infty,3} - 1) + k_{\infty,1} K_{\mathrm{L}} \tau_h (\tau_h k_{\infty,1} + 2k_{\infty,2}) \geq 0. \end{aligned} \tag{4.28}$$

Note that (4.27)-(4.28) define the set $\mathcal{S} \subset \mathbb{R}^3$ of all feasible regulators $\mathbf{k}_\infty$: as illustrated in the example of Fig. 4.8, $\mathcal{S}$ is a nonconvex set. Since $\mathcal{S}$ contains infinite gain vectors, one needs an optimal criterion to select $\mathbf{k}_\infty$, such as, e.g., minimizing any vector norm. In the numerical simulations in Sect. 4.2.4, we chose the $\mathbf{k}_\infty$ with minimum squared 2-norm.

Given the regulator $\mathbf{k}_2$ and a regulator $\mathbf{k}_\infty \in \mathcal{S}$, by using Procedure 2.1 in (Blanchini et al. 2008), the matrices $\mathbf{A}_{\mathcal{K},i}$, $\mathbf{B}_{\mathcal{K},i}$, $\mathbf{C}_{\mathcal{K},i}$, $\mathbf{D}_{\mathcal{K},i}$ of the compensator in (4.24) can be simply computed as,

$$\begin{bmatrix} \mathbf{D}_{\mathcal{K},i} & \mathbf{C}_{\mathcal{K},i} \\ \mathbf{B}_{\mathcal{K},i} & \mathbf{A}_{\mathcal{K},i} \end{bmatrix} = \begin{bmatrix} \mathbf{k}_2^T & \mathbf{k}_\infty^T \\ \mathbf{V}_{2,i} & \mathbf{V}_\infty \end{bmatrix} \begin{bmatrix} \mathbf{I}_3 & \mathbf{I}_3 \\ \mathbf{Z}_{2,i} & \mathbf{Z}_\infty \end{bmatrix}^{-1}, \tag{4.29}$$

where

$$\mathbf{V}_{2,i} \triangleq \mathbf{Z}_{2,i}(\mathbf{A} + \mathbf{B}\,\mathbf{k}_2^T), \quad \mathbf{V}_\infty \triangleq \mathbf{Z}_\infty(\mathbf{A} + \mathbf{B}\,\mathbf{k}_\infty^T), \tag{4.30}$$

and

$$\mathbf{Z}_{2,i} \triangleq [\mathbf{0}_{3\times 1} \quad \widetilde{\mathbf{Z}}_2]\,[\mathbf{x}_i(0) \quad \widetilde{\mathbf{E}}_{2,i}]^{-1}, \quad \mathbf{Z}_{\infty} \triangleq [\mathbf{0}_{3\times 1} \quad \widetilde{\mathbf{Z}}_{\infty}]\,[\mathbf{G} \quad \widetilde{\mathbf{E}}_{\infty}]^{-1}. \tag{4.31}$$

$\widetilde{\mathbf{E}}_{2,i}$, $\widetilde{\mathbf{E}}_{\infty} \in \mathbb{R}^{3\times 2}$ in (4.31) are such that matrices $[\mathbf{x}_i(0) \quad \widetilde{\mathbf{E}}_{2,i}]$, $i \in \{2,\dots,n\}$, $[\mathbf{G} \quad \widetilde{\mathbf{E}}_{\infty}]$, respectively, are invertible, and $\widetilde{\mathbf{Z}}_2$, $\widetilde{\mathbf{Z}}_{\infty} \in \mathbb{R}^{3\times 2}$ are such that matrix

$$\begin{bmatrix} \mathbf{x}_i(0) & \mathbf{G} & \widetilde{\mathbf{E}}_{2,i} & \widetilde{\mathbf{E}}_{\infty} \\ \mathbf{0}_{3\times 1} & \mathbf{0}_{3\times 1} & \widetilde{\mathbf{Z}}_2 & \widetilde{\mathbf{Z}}_{\infty} \end{bmatrix} \in \mathbb{R}^{6\times 6}, \;\; i \in \{2,\dots,n\},$$

is invertible. We conclude this section with Proposition 4.4, which provides sufficient conditions for $\mathbf{k}_{\infty}$ to be stabilizing when the parameters K_L and T_L are *not exactly* known (e.g., because of an inaccurate identification of the GVLD system).

4.4. PROPOSITION (STABILIZING $\mathbf{k}_{\infty}$ WITH UNCERTAIN K_L, T_L). *Let us suppose that the parameters K_L and T_L of the GVLD system (4.9) are not exactly known, and lie within the intervals $K_{L,m} \le K_L \le K_{L,M}$, $T_{L,m} \le T_L \le T_{L,M}$ where $K_{L,m}$, $K_{L,M}$, $T_{L,m}$ and $T_{L,M}$ are known positive constants. Then, matrix $\widehat{\mathbf{A}}$ in (4.25) is Hurwitz if the following inequalities are satisfied:*

$$\begin{aligned} &K_{L,M}\,k_{\infty,3} < 1, \quad k_{\infty,1}\,\tau_h + k_{\infty,2} > 0, \quad k_{\infty,1} > 0, \\ &(K_{L,m}\,k_{\infty,3} - 1)(k_{\infty,1}\,\tau_h + k_{\infty,2}) + k_{\infty,1}\,T_{L,M} < 0, \\ &(K_{L,M}\,k_{\infty,3} - 1)(k_{\infty,1}\,\tau_h + k_{\infty,2}) + k_{\infty,1}\,T_{L,m} < 0. \end{aligned} \tag{4.32}$$

Proof : *See (Morbidi et al. 2013).* ■

Note that the inequalities in (4.32) reduce to those in (4.27) for $K_L = K_{L,m} = K_{L,M}$ and $T_L = T_{L,m} = T_{L,M}$, as expected.

4.2.4 Numerical experiments

Simulation experiments have been carried out to study the performance of the control strategies described in Sect. 4.2.3. The desired behavior of the platoon is specified in both cases by the following three performance metrics for $i \in \{2,\dots,n\}$, cf. (Li et al. 2011):

1. *Distance and velocity tracking*: $\mathscr{C}_{T,\,i} = r_{\Delta d}\,\Delta\,d_i^2 + r_{\Delta v}\,\Delta\,v_i^2$ where $r_{\Delta d}$, $r_{\Delta v}$ are positive gains.

2. *Driver's comfort*: $\mathscr{C}_{C,\,i} = r_u\,u_i^2$ where r_u is a positive gain.

3. *Driver's car following*: $\mathscr{C}_{D,\,i} = r_a\,(a_{\text{ref},i} - a_i)^2$ where $a_{\text{ref},i}$ is the reference acceleration calculated according to the linear *driver's car-following model* $a_{\text{ref},i} = \kappa_D\,\Delta\,d_i + \kappa_V\,\Delta\,v_i$, and r_a, κ_D, κ_V are positive gains.

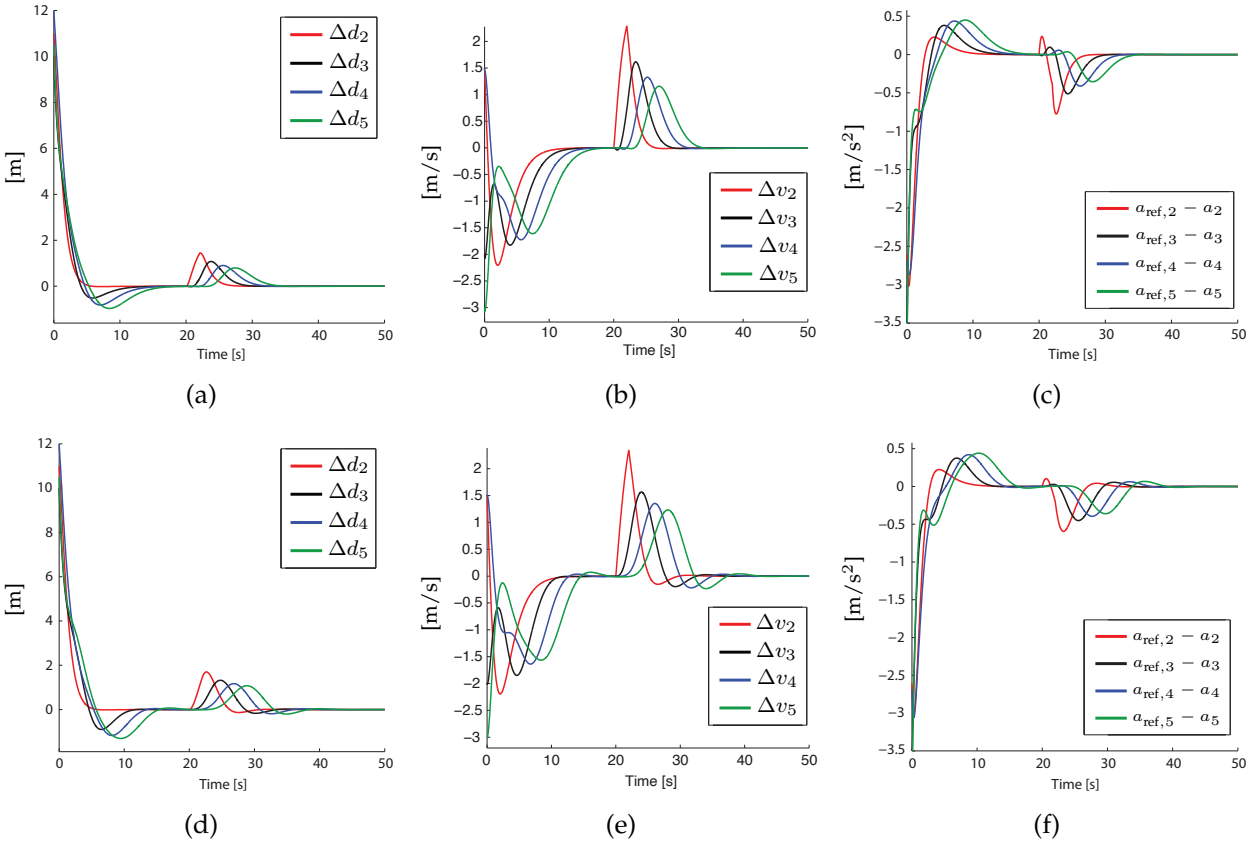


Figure 4.9: First row, *LQ regulation with guaranteed string stability;* second row, *compensator blending method. (a),(d) Time evolution of* $\Delta d_i(t)$*, (b),(e) of* $\Delta v_i(t)$*, and (c),(f) of* $a_{\text{ref},i}(t) - a_i(t)$*,* $i \in \{2, \dots, 5\}$.

The combination of $\mathscr{C}_{T,i}$, $\mathscr{C}_{C,i}$ and $\mathscr{C}_{D,i}$ yields the following weight matrices in the quadratic cost function in (4.12):

$$\mathbf{Q} = \begin{bmatrix} r_{\Delta d} + \kappa_D^2 r_a & \kappa_D \kappa_V r_a & -\kappa_D r_a \\ \kappa_D \kappa_V r_a & r_{\Delta v} + \kappa_V^2 r_a & -\kappa_V r_a \\ -\kappa_D r_a & -\kappa_V r_a & r_a \end{bmatrix}, \quad r = r_u.$$

LQ regulation with guaranteed string stability

Figs. 4.9(a)-(c) show the simulation results relative to the approach described in the first part of Sect. 4.2.3. A platoon of 5 vehicles was simulated for 50 seconds, with

$$a_1(t) = \begin{cases} 1.5 \text{ m/s}^2 & \text{for } t \in [20, 22)\text{s}, \\ 0 & \text{otherwise}, \end{cases}$$

and with initial conditions

$$\mathbf{x}_2(0) = [11,\ 1.5,\ 3.2]^T,\ \ \mathbf{x}_3(0) = [10,\ -2,\ 3.5]^T,$$
$$\mathbf{x}_4(0) = [12,\ 1.5,\ 3.3]^T,\ \ \mathbf{x}_5(0) = [10.5,\ -3,\ 3.5]^T.$$

The other selected parameters, are $\tau_h = 1.8$s, $T_L = 0.5$s, $K_L = 1$, $k_D = 0.02$, $k_V = 0.25$ and $r_{\Delta d} = r_{\Delta v} = 4$, $r_a = 0.1$, $r_u = 18$ (note that in CACC of cars, τ_h is typically in the subsecond time scale in the literature (Naus, Vugts, Ploeg, van de Molengraft and Steinbuch 2010): in our simulations, we selected a slightly larger τ_h for improving the readability of our plots). Using (4.15), we obtained a feedback control gain $\mathbf{k} = [0.4714,\ 0.7182,\ -0.6038]^T$ and a feedforward control gain $k_F = -0.3110$. Figs. 4.9(a)-(c) show the time evolution of $\Delta\, d_i$, $\Delta\, v_i$ and $a_{\text{ref},i} - a_i$, and Fig. 4.10 (top) the time history of u_i for $i \in \{2, \ldots, 5\}$ and of a_1. Note that with our parameters' selection, the inequalities in (4.19) are satisfied and the platoon is string stable. If, instead, we set $r_{\Delta d} = 1$ and keep all the other parameters unchanged, the second condition in (4.19) is not fulfilled anymore, thus possibly leading to a string-unstable behavior.

Compensator blending method

Figs. 4.9(d)-(f) show the simulation results relative to the approach described in the second part of Sect. 4.2.3. To compare the performance of the controller designed with the compensator blending method and the LQ regulator, we repeated the simulation experiment of the previous subsection with the same initial conditions and parameters. We set $\mathbf{k}_2 = \mathbf{k} = [0.4714,\ 0.7182,\ -0.6038]^T$ and determined the H_∞ regulator by numerically solving in Matlab with an interior-point algorithm (the barrier method), the optimization problem $\min_{\mathbf{k}_\infty \in \mathcal{S}} \|\mathbf{k}_\infty\|_2^2$ (the initial condition is $\mathbf{k}_\infty(0) = [1,\ 0,\ 0]^T \in \mathcal{S}$), which

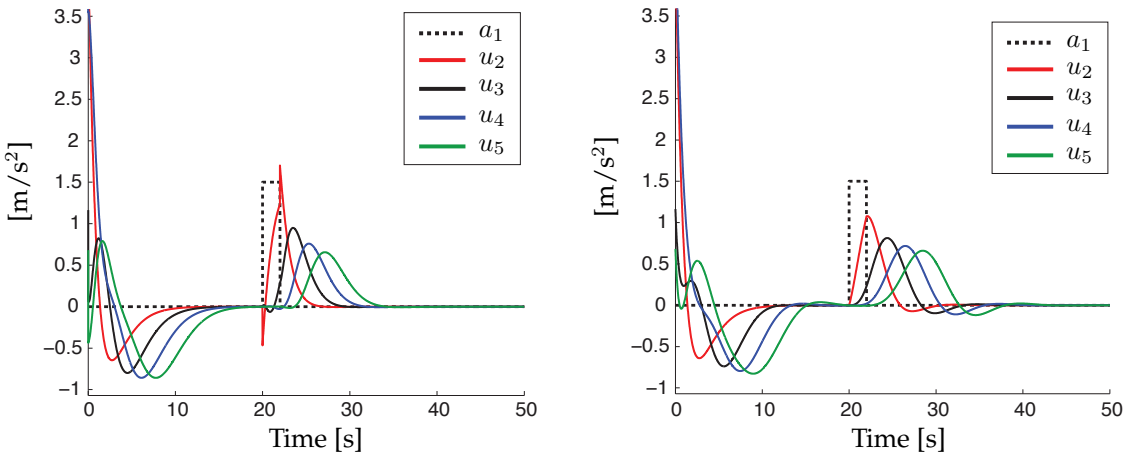


Figure 4.10: *Time evolution of $u_i(t)$, $i \in \{2, \ldots, 5\}$, for: (top) the LQ regulator, (bottom) the regulator based on the compensator blending method.*

yielded $\mathbf{k}_\infty = [0.2360,\ 0.2622,\ 0.1457]^T$. The application of the blending procedure to $\mathbf{k}_2$ and $\mathbf{k}_\infty$, led us to

$$\mathbf{Z}_{2,2} = \begin{bmatrix} 0.1298 & 0.6483 & -0.7502 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},\ \mathbf{Z}_{2,3} = \begin{bmatrix} -0.1855 & 0.5255 & 0.8303 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\mathbf{Z}_{2,4} = \begin{bmatrix} 0.1197 & 0.6648 & -0.7373 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},\ \mathbf{Z}_{2,5} = \begin{bmatrix} -0.2616 & 0.1887 & 0.9466 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},\ \mathbf{Z}_\infty = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

from which the dynamic compensators $\boldsymbol{\mathcal{K}}_i$, $i \in \{2, \ldots, 5\}$, were computed using (4.29) and (4.30). Figs. 4.9(d)-(f) show the time evolution of $\Delta\, d_i$, $\Delta\, v_i$ and $a_{\text{ref},i} - a_i$, and Fig. 4.10 (bottom) the time history of u_i for $i \in \{2, \ldots, 5\}$ and of a_1. From Fig. 4.9 we notice that the two proposed controllers achieve comparable satisfactory performances: however, from Fig. 4.10 (and consistently with our choice of $\mathbf{k}_\infty$), we can notice that the compensator blending method results in a smaller control effort.

4.2.5 Conclusions and future work

In this section, we have presented two novel *decentralized optimal* strategies for *Cooperative Adaptive Cruise Control* (CACC) of a car platoon under string-stability constraints. Some variations to the basic problem setup have also been explored and the results of numerical simulations have been provided to support our theoretical findings.

Note that the feedforward part of controller (4.14) does not include anticipatory characteristics for variable disturbances z_i. Any adjustment to this controller to get improved transient response usually involves lead-lag networks to replace the constant gain k_F (Anderson and Moore 1989): the design of such networks will be considered in future works. In future research, we are also going to verify whether a static controller which optimally switches between $\mathbf{k}_2$ and $\mathbf{k}_\infty$ may possibly outperform the dynamic regulator based on the compensator blending method, we are going to study the case of time-varying communication delays $\theta(t)$ and $\phi(t)$ (Gu et al. 2003), and to test the developed control strategies in more advanced car simulators, e.g. in IPG's CarMaker (*CarMaker, IPG Automotive* 2023).

4.3 Functions of Laplacian matrix: Application to formation control

In this third and last section, we study a class of matrix functions of the combinatorial Laplacian that preserve its structure, i.e. that define matrices which are positive semidefinite, and which have zero row-sum and non-positive off-diagonal entries. This formulation has the merit of presenting different incarnations of the Laplacian matrix appeared in the recent literature, in a unified framework. For the first time, we apply this family of Laplacian functions to consensus theory, and we show that they leave the agreement value unchanged and offer distinctive advantages in terms of performance and design flexibility. The theory is illustrated via worked examples and numerical experiments featuring four representative Laplacian functions in a shape-based distributed formation control strategy for single-integrator robots.

4.3.1 Introduction

The combinatorial Laplacian is ubiquitous in network science (Newman 2018), and over the last decade we have witnessed the emergence of several variants to describe distributed dynamic processes. In particular, the recent advent of graph signal processing (Shuman et al. 2016, Ortega et al. 2018), graph neural networks (Gama et al. 2020, Wu et al. 2021), and the growing popularity of network systems (Mesbahi and Egerstedt 2010, Bullo 2022), has had a catalytic effect on the research in this field.

An interesting variant of combinatorial Laplacian is the *deformed Laplacian*, which has found applications in multi-agent systems theory (Morbidi 2013a), semi-supervised learning (Gong et al. 2015), and in the design of new centrality measures for undirected and directed networks (Grindrod et al. 2018, Arrigo et al. 2018, Arrigo et al. 2019, Arrigo et al. 2020). Along the same lines as (Morbidi 2013a), the *parametric Laplacian* has been introduced in (Morbidi 2014). The *Bethe-Hessian matrix*, the reversal of the deformed Laplacian in the case of undirected graphs (Arrigo et al. 2018), has been used for spectral clustering (Saade et al. 2014) and community detection in sparse heterogeneous networks (Dall'Amico et al. 2019) (see also (Stoll 2020)). A different definition of deformed Laplacian, which encompasses several Laplacian-like matrices available in the literature (*connection*, *magnetic* (Fanuel et al. 2018), *signed* and *dilation*), was proposed in (Fanuel and Suykens 2019). Notably, the dilation Laplacian has been shown to be useful for spectral ranking in directed graphs.

The p*-Laplacian*, a nonlinear generalization of the combinatorial Laplacian, has recently attracted the attention of the machine learning community. The p-Laplacian reduces to the standard Laplacian for $p = 2$, and it has been successfully applied to solve two-class (Bühler and Hein 2009) and multi-class (Luo et al. 2010) clustering problems.

The *exponential* of the negated Laplacian is common in the study of classical transport on lattices and networks, and it is often referred to as the "heat kernel" because of its interpretation as a diffusion process related to the heat equation (Michelitsch et al. 2019). It also appears in discrete-time agreement protocols (Mesbahi and Egerstedt 2010).

Laplacian powers have been widely explored in the literature as well. In particular, the bi-Laplacian (or Laplacian squared) has been considered in (Li and Ng 2019) for image colorization. The bi-Laplacian corresponds to the discretization of the bi-harmonic PDE equation with Neumann boundary condition, which together with its numerical schemes is widely studied and applied in problems of linear elasticity (Henrot 2006, Ch. 11), data interpolation, and computer vision (image inpainting). For non-integer powers in the $(0, 1]$ interval, we obtain the so-called *fractional Laplacian*, which has been used to study random walks on networks, among other applications (Michelitsch et al. 2019, Lischke et al. 2020). In (Bautista et al. 2019), the authors have proposed a generalization of the PageRank algorithm for semi-supervised learning based on a (non-necessary integer) power of the Laplacian matrix. Finally, it has been recently shown that graph filters can be represented by *matrix polynomials* of the combinatorial Laplacian (Segarra, Marques and Ribeiro 2017, Levie et al. 2019).

Original contributions

Following (Michelitsch et al. 2019), we study here *matrix functions of the Laplacian* of an undirected graph, that preserve its main algebraic properties, i.e. that give rise to matrices which are positive semidefinite and which have zero row-sum and non-positive off-diagonal entries. These Laplacian functions promote the emergence of non-local network correlations: in fact, thanks to their ability to reorganize the information exchanges between the nodes of a graph, long-range interactions are enabled. This general formulation allows us to bring together a number of results scattered across different areas and shed new light on them. We also revisit the classical continuous-time consensus protocol by replacing the combinatorial Laplacian with our family of Laplacian functions. These functions leave the agreement value *unchanged* and offer some practical advantages. In fact, in many situations, by simply tuning a scalar parameter, the user can seamlessly modify the dynamic behavior of the multi-agent system, for example, to guarantee faster convergence towards consensus or to adapt to variable external conditions. This generalizes existing work in the literature, where integer powers of the Laplacian have been considered for fast consensus seeking (by introducing a "multi-hop relay" interpretation (Jin and Murray 2006)), and for dynamic consensus over wireless sensor networks (Manfredi 2013). The price to pay for the increased design flexibility, is that the interaction graph associated with our Laplacian functions is generally more dense than that of the original Laplacian. A simple yet effective approximation

of the Laplacian functions is thus proposed to ensure a distributed implementation of the new coordination protocols. The theoretical findings are illustrated via worked examples and validated via numerical simulations in which single-integrator robots run a consensus-based formation control algorithm.

The rest of this section is organized as follows. In Sect. 4.3.2, we briefly recall the definition of two matrix functions, and some basic notions of algebraic graph theory. In Sect. 4.3.3, we study a family of functions of the combinatorial Laplacian that retain its special structure, and in Sect. 4.3.4 we use them in a novel shape-based distributed formation control protocol, which is validated via extensive numerical experiments. Finally, in Sect. 4.3.5, the original contributions are summarized and some possible avenues for future research are outlined.

4.3.2 Preliminaries

We briefly recall below the definitions of matrix logarithm and matrix pth root, for later reference.

Matrix logarithm

A logarithm of $\mathbf{A} \in \mathbb{C}^{n\times n}$ is any matrix $\mathbf{X}$ such that $e^{\mathbf{X}} = \mathbf{A}$. Any nonsingular matrix $\mathbf{A}$ has infinitely many logarithms (Higham 2008a, Th. 1.27). Hereafter, we assume that $\mathbf{A} \in \mathbb{C}^{n\times n}$ has no eigenvalues on the closed negative real axis, and $\log(\cdot)$ always denotes the *principal logarithm*, which is the unique logarithm all of whose eigenvalues lie in the strip $\{z \in \mathbb{C} : -\pi < \mathrm{Im}(z) < \pi\}$ (Higham 2008a, Th. 1.31). If $\mathbf{A}$ is real, then its principal logarithm is real.

Matrix pth root

Let $p \geq 2$ be an integer. Matrix $\mathbf{X}$ is a pth root of $\mathbf{A} \in \mathbb{C}^{n\times n}$ if $\mathbf{X}^p = \mathbf{A}$. We recall the following result on existence of pth roots (Higham 2008a, Th. 7.2).

4.1. THEOREM (PRINCIPAL pTH ROOT). *Let $\mathbf{A} \in \mathbb{C}^{n\times n}$ have no eigenvalues on the closed negative real axis. There is a unique pth root $\mathbf{X}$ of $\mathbf{A}$ all of whose eigenvalues lie in the sector $\{z \in \mathbb{C} : -\pi/p < \arg(z) < \pi/p\}$, and it is a primary matrix function[2] of $\mathbf{A}$. We refer to $\mathbf{X}$ as the principal pth root of $\mathbf{A}$ and write $\mathbf{X} = \mathbf{A}^{1/p}$. If $\mathbf{A}$ is real, then $\mathbf{A}^{1/p}$ is real.* ■

For $p = 2$, we have the following extension of Theorem 4.1 which allows $\mathbf{A}$ to be *singular* (see Problem 1.27 in (Higham 2008a)).

[2]For more details on *primary* and *nonprimary matrix functions*, see (Higham 2008a, Sect. 1.4).

4.5. PROPOSITION (SQUARE ROOT OF A SINGULAR MATRIX). *Let $\mathbf{A} \in \mathbb{C}^{n\times n}$ have no eigenvalues on the closed negative real axis, except possibly for a semisimple[3] zero eigenvalue. There is a unique square root $\mathbf{X}$ of $\mathbf{A}$ that is a primary matrix function of $\mathbf{A}$ and whose nonzero eigenvalues lie in the open right half-plane. If $\mathbf{A}$ is real, then $\mathbf{X}$ is real.* ■

There exist four main numerical algorithms for the computation of the matrix pth root: Schur, Newton, and Schur-Newton algorithms, and matrix sign method (Higham 2008a, Sect. 7). The Schur algorithm assumes that $\mathbf{A} \in \mathbb{R}^{n\times n}$ is nonsingular, whereas the other methods assume that $\mathbf{A}$ has no eigenvalues on the closed negative real axis.

4.3.3 Functions of Laplacian matrix

In what follows, we study matrix functions $f(\mathbf{L})$ of the combinatorial Laplacian $\mathbf{L}$. These functions define matrices which incorporate the network topology information and which promote non-local interactions between the nodes of graph $\mathcal{G}$. We will exploit them in Sect. 4.3.4, to design a new distributed formation control strategy for single-integrator robots. Our exposition here follows the outline of (Michelitsch et al. 2019, Ch. 1): however, several new examples and results are provided, and the notation has been modified to conform to that conventionally used in systems theory. For the sake of simplicity, we henceforth use $f_{ij}(\mathbf{L})$ to denote the $(i,\, j)$ entry of matrix $f(\mathbf{L})$.

Given a well-defined function $f(x) : \mathbb{R} \to \mathbb{R}$, the matrix $f(\mathbf{L})$ can be obtained using the series expansion $f(x) = \sum_{\ell=0}^{\infty} c_\ell\, x^\ell$ where c_ℓ is the real coefficient of the ℓth term, or using the spectral decomposition of $\mathbf{L}$. The latter option gives

$$f(\mathbf{L}) \;=\; \sum_{i=1}^{n} f(\lambda_i)\, \mathbf{v}_i \mathbf{v}_i^T. \tag{4.33}$$

Hence, to find $f(\mathbf{L})$, one should calculate the spectrum $\{\lambda_1, \lambda_2, \ldots, \lambda_n\}$ of $\mathbf{L}$ and then compute $\{f(\lambda_1), f(\lambda_2), \ldots, f(\lambda_n)\}$: the eigenvectors $f(\mathbf{L})$ remain the same as those of $\mathbf{L}$. From (4.33), we also notice that matrix $f(\mathbf{L})$ is symmetric by construction. For further details on the theory of functions of matrices, see (Higham 2008a) and (Lancaster and Tismenetsky 1985, Ch. 9).

Note that while equation (4.33) allows to calculate general functions of $\mathbf{L}$, we are interested here in functions which *preserve* the special structure of the Laplacian matrix (as reported in Sect. 3.1). In order to retain these desirable properties, the matrix $f(\mathbf{L})$ should satisfy the following three conditions:

- **Condition I**: $f(\mathbf{L})$ is positive semidefinite, i.e. the eigenvalues of $f(\mathbf{L})$ are restricted to be positive or zero.

[3] An eigenvalue is called *semisimple* if its algebraic multiplicity equals its geometric multiplicity.

- **Condition II**: the entries $f_{ij}(\mathbf{L})$ satisfy $\sum_{j=1}^{n} f_{ij}(\mathbf{L}) = 0$ for $i \in \{1, 2, \ldots, n\}$, or equivalently $f(\mathbf{L})\mathbb{1} = \mathbf{0}$, i.e. each row-sum is equal to zero.
- **Condition III**: the off-diagonal entries of $f(\mathbf{L})$ are non-positive and they are not allowed to be all simultaneously zero. Hence, owing to Condition II, the diagonal elements of $f(\mathbf{L})$ are all strictly positive.

Condition I is guaranteed if $f(x) \geq 0$ for $x \geq 0$. Then $f(\lambda_i) \geq 0$ for all i, and the eigenvalues of $f(\mathbf{L})$ can be arranged in increasing order as those of $\mathbf{L}$. On the other hand, by using equation (4.33), it is easy to verify that Condition II is fulfilled if $f(0) = 0$. However, these two conditions on the function f do not guarantee that the off-diagonal entries of $f(\mathbf{L})$ are non-positive, as required by Condition III. For example, if we take $\mathcal{G} = S_n$ (the star graph with n nodes), we have that

$$\mathbf{L}(\mathcal{G}) = \begin{bmatrix} n-1 & -1 & -1 & \cdots & -1 \\ -1 & 1 & 0 & \cdots & 0 \\ -1 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -1 & 0 & 0 & \cdots & 1 \end{bmatrix}.$$

The function $f(x) = x^2$ is non-negative for $x \geq 0$ and $f(0) = 0$. Thereby, $f(\mathbf{L}) = \mathbf{L}^2$, a.k.a. bi-Laplacian (Li and Ng 2019), satisfies Conditions I and II, but the structure imposed by Condition III is not preserved: in fact, $\mathbf{L}^2$ has strictly positive (unitary) off-diagonal entries. On the contrary, the function $f(x) = \log(x+1)$ is non-negative for $x \geq 0$, $f(0) = 0$, and $f(\mathbf{L}) = \log(\mathbf{L} + \mathbf{I}_n)$ has negative off-diagonal entries. Hence, $\log(\mathbf{L} + \mathbf{I}_n)$ is an admissible Laplacian function.

The following notion of *completely monotonic function* (Miller and Samko 2001) plays an important role in the characterization of the admissible functions $f(\mathbf{L})$.

4.2. DEFINITION (COMPLETELY MONOTONIC FUNCTION). *A function $g(x)$ defined on $0 < x < \infty$ is said to be completely monotonic, if it possesses derivatives $g^{(m)}(x) = \frac{d^m}{dx^m} g(x)$ for all $m \in \mathbb{Z}_{\geq 0}$ and if $(-1)^m g^{(m)}(x) \geq 0$ for all $x > 0$.* ◇

In (Michelitsch et al. 2019, Sect. 1.4.2), the authors have proved that the functions $f(\mathbf{L})$ that satisfy the necessary Condition III, can be constructed via the scalar functions

$$f(x) = H(0) - H(-x), \quad 0 \leq x < \infty, \tag{4.34}$$

where $H(x)$ denotes a primitive of the auxiliary scalar function $\mathrm{h}(x)$, i.e. $\mathrm{h}(x) = \frac{d}{dx} H(x)$. The function $f(x)$ can be expressed in terms of a function $g(x) = \mathrm{h}(-x)$ defined on $0 \leq x < \infty$, which satisfies the following conditions:

$$\frac{d}{dx} f(x) = g(x) > 0, \quad 0 \leq x < \infty, \tag{4.35}$$

and

$$(-1)^m\, g^{(m)}(x) \geq 0, \quad 0 \leq x < \infty, \quad m \in \mathbb{Z}_{>0}. \tag{4.36}$$

According to Definition 4.2, a function $g(x)$ that fulfills conditions (4.35) and (4.36) is a completely monotonic function. By (4.35), $g(x)$ is strictly positive for $x \geq 0$, including, in particular, the entire spectral interval $0 \leq x \leq \lambda_n$ of the Laplacian $\mathbf{L}$, and the function $f(x)$ monotonically increases on its interval of definition $0 \leq x < \infty$. This monotonicity property implies that the algebraic multiplicity of the eigenvalues of $\mathbf{L}$ is maintained in $f(\mathbf{L})$ (in fact, $f(\lambda_i) > f(\lambda_j)$ for $\lambda_i > \lambda_j$). The function $f(x)$ is also given by the following integral

$$f(x) \;=\; \int_0^x g(y)\, dy, \quad 0 \leq x < \infty, \tag{4.37}$$

i.e. it is the primitive of $g(x)$ with $f(0) = 0$ and with $f(x) > 0$ for $x > 0$. In (Michelitsch et al. 2019, Sect. 1.4.3), the authors have shown that relations (4.34)-(4.37) are indeed *sufficient* to generate scalar $\mathcal{C}^\infty$ functions $f(x)$ which define admissible Laplacian functions $f(\mathbf{L})$ satisfying Conditions I-III. Note that $f(x)$ in (4.37) has the general structure given in equation (4.34): hence, the admissible Laplacian functions can be represented as

$$f(\mathbf{L}) \;=\; H(0)\mathbf{I}_n - H(-\mathbf{L}) = \sum_{i=2}^{n} (H(0) - H(-\lambda_i))\, \mathbf{v}_i \mathbf{v}_i^T.$$

There is no term $\mathbf{v}_1 \mathbf{v}_1^T$ since $f(\lambda_1) = 0$ (recall equation (4.33)). The eigenvector of $f(\mathbf{L})$ associated with the zero eigenvalue is $\mathbb{1}/\sqrt{n}$, since $\mathbf{L}$ and $f(\mathbf{L})$ have the same set of eigenvectors.

Examples of $f(\mathbf{L})$

There exist several classes of completely monotonic functions which in conjunction with equation (4.37) allow to construct admissible Laplacian functions satisfying Conditions I-III. We focus here on six examples (half of which are not reported in (Michelitsch et al. 2019)), that are of special interest.

1. The function $g(x) = (\lambda - x)^m$ for $0 \leq x < \lambda$ with $m \in \mathbb{Z}_{\geq 0}$ and $0 < \lambda_n < \lambda$, satisfies conditions (4.35) and (4.36). From equation (4.37), we obtain

$$f(x) \;=\; \int_0^x (\lambda - y)^m\, dy \;=\; \frac{1}{m+1}\big(\lambda^{m+1} - (\lambda - x)^{m+1}\big).$$

This yields the matrix function:

$$f(\mathbf{L}) \;=\; \frac{1}{m+1}\big(\lambda^{m+1}\, \mathbf{I}_n - (\lambda \mathbf{I}_n - \mathbf{L})^{m+1}\big) \;=\; \sum_{i=1}^{n} \frac{1}{m+1}\big(\lambda^{m+1} - (\lambda - \lambda_i)^{m+1}\big)\, \mathbf{v}_i \mathbf{v}_i^T.$$

Note that in the trivial case of $m = 0$, the function $g(x) = 1$, and $f(x)$ is the identity function, which yields $f(\mathbf{L}) = \mathbf{L}$.

2. Consider the function

$$g(x) \;=\; \frac{a}{(ax+b)^m},\;\; a>0,\; b\geq 0,\; m\in\mathbb{Z}_{>0}.$$

Let us start with the case of $m = 1$. From equation (4.37), we obtain $f(x) = \log(ax+b)$ for $x \geq 0$. However, the constraint $f(0) = 0$, requires $f(x) = \log(ax + b) - \log b$. Hence, we have

$$f(\mathbf{L}) = \log(a\mathbf{L} + b\mathbf{I}_n) - \log(b\mathbf{I}_n).$$

Let us now consider the general case of $m \in \{2, 3, \ldots\}$. Following the same procedure as above, we obtain:

$$f(x) \;=\; -\frac{1}{(m-1)(ax+b)^{m-1}},$$

which we modify into

$$f(x) \;=\; \frac{1}{m-1}\left[-\frac{1}{(ax+b)^{m-1}} + \frac{1}{b^{m-1}}\right],$$

to guarantee that $f(0) = 0$. Hence, we end up with

$$f(\mathbf{L}) \;=\; \frac{1}{m-1}\left[-(a\mathbf{L} \,+\, b\mathbf{I}_n)^{1-m} \,+\, b^{1-m}\,\mathbf{I}_n\right].$$

3. The *exponential* $g(x) = ae^{-ax}$ with $a > 0$, is a completely monotonic function. The corresponding function that retains the Laplacian structure is $f(x) = 1 - e^{-ax}$ and thus $f(\mathbf{L}) = \mathbf{I}_n - e^{-a\mathbf{L}}$. Matrix $e^{-a\mathbf{L}}$ is doubly stochastic and it has appeared under different forms in consensus theory (Mesbahi and Egerstedt 2010). Functions like $e^{-a\mathbf{L}}$ and the regularized Laplacian $(\mathbf{I}_n + a\mathbf{L})^{-1}$, have been also used as kernels to compute similarities between the nodes of an undirected graph (Fouss et al. 2016).

4. The function

$$g(x) \;=\; \frac{1}{b \,-\, ce^{-ax}},\;\; a>0,\; b>c>0,$$

defined on $0 < x < \infty$, is completely monotonic (Miller and Samko 2001). From (4.37), we obtain

$$f(x) \;=\; \frac{1}{ab}\big[\log(b - ce^{-ax}) + ax\big].$$

To guarantee $f(0) = 0$, we set

$$f(x) \;=\; \frac{1}{ab}\big[\log(b - ce^{-ax}) + ax - \log(b-c)\big],$$

yielding

$$f(\mathbf{L}) = \frac{1}{ab}\big[\log(b\mathbf{I}_n - ce^{-a\mathbf{L}}) + a\mathbf{L} - \log(b-c)\mathbf{I}_n\big].$$

5. The function $g(x) = (\beta+1)\,x^{\beta}$ with $\beta \leq 0$ is a completely monotonic function. From equation (4.37), we obtain $f(x) = x^{\beta+1}$. The condition $f(0) = 0$, requires that $-1 < \beta \leq 0$. Therefore, the function $f(x) = x^{\gamma}$ with $\gamma \in \mathbb{R}$ such that $0 < \gamma \leq 1$, preserves the Laplacian structure imposed by Conditions I-III. We thus obtain the so-called *fractional Laplacian*, $f(\mathbf{L}) = \mathbf{L}^{\gamma}$, which has been studied in the theory of random walks, and in diffusive and quantum transport on networks (Michelitsch et al. 2019, Ch. 2). Note that integer powers $\gamma \in \{2, 3, \ldots\}$, fail to satisfy Condition III.

6. Node-invariant graph filters are linear graph-signal operators of the form (Segarra, Marques and Ribeiro 2017, Segarra, Marques, Mateos and Ribeiro 2017):

$$f(\mathbf{L}) = \sum_{j=0}^{m-1} b_j \mathbf{L}^j, \tag{4.38}$$

where $b_0, b_1, \ldots, b_{m-1}$ are real coefficients, i.e. graph filters are *polynomials* of the Laplacian $\mathbf{L}$ (or of any other $n \times n$ matrix, such as the adjacency matrix $\mathbf{A}$, whose sparsity pattern captures the local structure of graph $\mathcal{G}$). A graph filter can be equivalently defined as

$$f(\mathbf{L}) = a_0 \prod_{j=1}^{m-1} (\mathbf{L} - a_j \mathbf{I}_n),$$

where $a_0, a_1, \ldots, a_{m-1}$ are real coefficients, which also gives rise to a polynomial on $\mathbf{L}$ of degree $m-1$. From (4.38), we can see that Conditions I and II are fulfilled if $b_0 = 0$ and $b_j \geq 0$ for $j \in \{1, 2, \ldots, m-1\}$, but Condition III is not satisfied, in general. Graph filters based on Chebyshev and Cayley polynomials have also been recently introduced to process graph-structured data (Levie et al. 2019, Wu et al. 2021).

General properties of $f(\mathbf{L})$

In the previous section, we have identified the family of functions that preserve the structure of the combinatorial Laplacian. We now briefly discuss some general properties of the matrix $f(\mathbf{L})$, which will be used in Sect. 4.3.4.

Generalized degree: By construction, the diagonal entries of $f(\mathbf{L})$ are positive, and analogously to the Laplacian $\mathbf{L}$, we can refer to $d_i = f_{ii}(\mathbf{L})$, $i \in \{1, 2, \ldots, n\}$, as the

generalized degree associated with the function f (Michelitsch et al. 2019, Sect. 1.5.1). The average of the generalized degree is defined as

$$\frac{1}{n}\sum_{i=1}^{n} d_i \;=\; \frac{1}{n}\,\mathrm{tr}(f(\mathbf{L})) \;=\; \frac{1}{n}\sum_{i=1}^{n} f(\lambda_i),$$

where $\mathrm{tr}(f(\mathbf{L}))$ denotes the trace of $f(\mathbf{L})$. In the case of the fractional Laplacian, it is called the *average fractional degree*. Note that d_i does not only capture the local information (on its nearest neighbors) of node i, but also incorporates knowledge at the level of the whole network. This "non-locality" property is further explored below.

Laplacian functions for regular graphs: To gain some insight into the structure of the Laplacian functions $f(\mathbf{L})$, it is worth focusing on the special case of *regular graphs*. In a regular graph, each node has the same degree k, and the Laplacian takes the simple form $\mathbf{L} = k\mathbf{I}_n - \mathbf{A}$. By using the series expansion presented at the beginning of Sect. 4.3.3, we can express $f(\mathbf{L})$ as follows (Michelitsch et al. 2019, Sect. 1.5.2):

$$f(\mathbf{L}) \;=\; \sum_{\ell=1}^{\infty} c_\ell (k\mathbf{I}_n - \mathbf{A})^\ell \;=\; \sum_{\ell=1}^{\infty}\sum_{h=0}^{\ell} c_\ell \binom{\ell}{h} k^{\ell-h}(-1)^h \mathbf{A}^h. \tag{4.39}$$

Equation (4.39) unveils a connection between $f(\mathbf{L})$ and the integer powers of the adjacency matrix $\mathbf{A}$. Recall that the $(i,\, j)$ entry of $\mathbf{A}^h$ for $h \in \mathbb{Z}_{>0}$, is the number of all the possible paths connecting node i to node j with h edges, whereas the diagonal entry $(i,\, i)$ of $\mathbf{A}^h$ is the number of closed paths with h edges which start and end at the same node i (Godsil and Royle 2001). Therefore, equation (4.39) shows how the function f changes the local character of the Laplacian $\mathbf{L}$ and makes it a *long-range operator*: matrix $f(\mathbf{L})$ is thus well suited to define dynamical processes with non-local interactions on networks.

Exponential decay of functions of banded matrices: A *band matrix* is a sparse matrix whose nonzero entries are limited to a diagonal band, including the main diagonal and (zero or) more diagonals on either side. Given a matrix $\mathbf{B} \in \mathbb{C}^{n\times n}$, if all entries of $\mathbf{B}$ are zero outside a diagonally bordered band whose range is determined by $r_1, r_2 \in \mathbb{Z}_{\geq 0}$ (i.e. $[\mathbf{B}]_{ij} = 0$ if $j < i - r_1$ or $i + r_2 < j$), then r_1 and r_2 are called the *lower bandwidth* and *upper bandwidth* of $\mathbf{B}$, respectively. The *bandwidth* of $\mathbf{B}$ is the maximum of r_1 and r_2, i.e. it is the integer r such that $[\mathbf{B}]_{ij} = 0$ if $|i - j| > r$. For example, a band matrix with $r_1 = r_2 = 0$ ($r_1 = r_2 = 1$) is a diagonal (tridiagonal) matrix.

For the entries of functions of Hermitian band matrices, we have the following exponentially decaying bound (further decay results applying to general matrices, are available in (Benzi and Razouk 2007)):

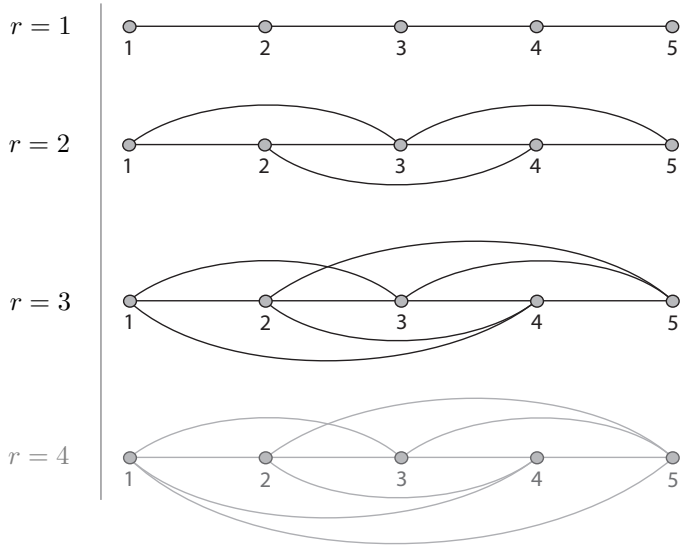


Figure 4.11: *Graphs $\mathcal{B}_5^r$ for $r \in \{1, 2, 3\}$. Note that $\mathcal{B}_5^4$ (light gray) is the complete graph with 5 nodes.*

4.2. THEOREM (BENZI & GOLUB (BENZI AND GOLUB 1999)). *Let $\mathbf{B} \in \mathbb{C}^{n\times n}$ be Hermitian and of bandwidth r, and let the function f be analytic in an ellipse containing the spectrum of $\mathbf{B}$. Then, $f(\mathbf{B})$ satisfies $|f_{ij}(\mathbf{B})| \leq M\varrho^{|i-j|}$, where M is a constant and $\varrho = \mu^{1/r}$, where $\mu \in (0,\, 1)$ depends only on f.* ■

Theorem 4.2 shows that the entries of $f(\mathbf{B})$ are bounded in an exponential decay manner away from the diagonal (the rate of decay depending on f), with the bound decreasing as the bandwidth r decreases. Note that this does not necessarily mean that "decay to zero" is observed in practice (Higham 2008a, Sect. 14.2).

If we turn our attention again to the Laplacian functions $f(\mathbf{L})$, it is easy to verify that the following family of undirected graphs with n nodes, admits a Laplacian $\mathbf{L}$ of bandwidth $r \in \{1, 2, \ldots, n-2\}$: $\mathcal{B}_n^r = (V,\, E)$ where $\{i,\, j\} \in E$ if and only if $j = i + s$ with $i \in \{1, 2, \ldots, n-s\}$ and $s \in \{1, 2, \ldots, r\}$. Graph $\mathcal{B}_n^1$ coincides with P_n (the path graph with n nodes), whose Laplacian is a tridiagonal matrix. We excluded the case of $r = n - 1$, since it corresponds to the complete graph with n nodes, whose Laplacian has zero bandwidth (see the example in Fig. 4.11). Note that this family of graphs is not unique. In fact, the entries on the diagonal band of $\mathbf{L}(\mathcal{B}_n^r)$ are all different from zero, but the definition of band matrix does not preclude null values.

4.3.4 Application to shape-based formation control

In this section, we leverage the Laplacian functions $f(\mathbf{L})$ studied in Sect. 4.3.3, to design a new class of continuous-time consensus protocols. In particular, we will focus

on the following *generalization* of the shape-based formation control strategy presented in (Mesbahi and Egerstedt 2010, Ch. 6.3):

$$\dot{\mathbf{x}}(t) = (-f(\mathbf{L}) \otimes \mathbf{I}_2)(\mathbf{x}(t) - \boldsymbol{\xi}), \quad \mathbf{x}(0) = \mathbf{x}_0, \tag{4.40}$$

where the state vector $\mathbf{x}(t) = [\mathrm{x}_1(t),\, \mathrm{y}_1(t), \ldots, \mathrm{x}_n(t),\, \mathrm{y}_n(t)]^T \in \mathbb{R}^{2n}$ contains the x-and y-coordinates of the positions of n single-integrator robots at time $t \geq 0$, and "$\otimes$" denotes the Kronecker product. This strategy allows to drive the n robots to a translationally-invariant formation encoded through the formation graph $\mathcal{G}_{\mathrm{F}} = (V,\, E_{\mathrm{F}})$ and the associated constant vector of target locations $\boldsymbol{\xi} \in \mathbb{R}^{2n}$. For the sake of simplicity, in the following, we will assume that $\mathcal{G}_{\mathrm{F}}$ coincides with the interaction (or communication) graph $\mathcal{G} = (V,\, E)$ of the robots (thus, the inclusion $E_{\mathrm{F}} \subseteq E$ is trivially satisfied, cf. (Mesbahi and Egerstedt 2010, Th. 6.12)).

4.3. REMARK (INVARIANCE OF THE AGREEMENT VALUE). *By virtue of the spectral properties of the admissible Laplacian functions discussed in Sect. 4.3.3, it is easy to prove that the state vector* $\mathbf{x}(t)$ *of system* (4.40) *asymptotically converges, for any initial condition* $\mathbf{x}_0 \in \mathbb{R}^{2n}$*, to a constant vector whose value does not depend on the* $f(\mathbf{L})$ *chosen (in other words, the agreement value is* f-invariant*).* ◇

4.4. REMARK (DISTRIBUTED IMPLEMENTATION OF SYSTEM (4.40)). *While some admissible functions* f *conserve, at least in part, the sparsity pattern of the Laplacian* $\mathbf{L}$*, the emergence of long-range interactions between the nodes of the graph typically translates into dense matrix functions of the Laplacian (i.e. the weighted interaction graph associated with* $f(\mathbf{L})$ *tends to be fully connected). This means that to implement* (4.40)*, the robots should adopt an all-to-all communication pattern, which is undesirable, in practice. However, in many instances (cf. Theorem 4.2), a large percentage of the off-diagonal entries of* $f(\mathbf{L})$ *is very close to zero, which correspond to network connections which bring negligible information to the nodes. This calls for an operator which most nearly transforms* $f(\mathbf{L})$ *into a new matrix* $f^q(\mathbf{L})$ *which is sparser than* $f(\mathbf{L})$*. A simple solution is to define* $f^q(\mathbf{L})$ *as follows:*

$$f^q_{ij}(\mathbf{L}) = \begin{cases} 0 & \text{if } |f_{ij}(\mathbf{L})| < q,\ i \neq j, \\ f_{ij}(\mathbf{L}) & \text{otherwise}, \end{cases}$$

where q *is a small positive constant (a threshold). This transformation does not break the symmetry of* $f(\mathbf{L})$ *and the off-diagonal entries* $f^q(\mathbf{L})$ *remain non-positive as required by Condition III. Moreover, the Gershgorin circle theorem ensures that* $f^q(\mathbf{L})$*, as* $f(\mathbf{L})$*, is positive semidefinite. In conclusion, while* $f^q(\mathbf{L})$ *still remains less sparse than* $\mathbf{L}$ *in general, it makes protocol* (4.40) *amenable to a distributed implementation (see Sect. 4.3.4 for more details).* ◇

Case studies

For later use in Sect. 4.3.4, we recall here some elementary properties of four of the Laplacian functions examined in Sect. 4.3.3.

1. *Logarithmic function*: Let $f(x) = \log(ax+1)$ with $a > 0$. Note that
$$\lim_{a \to 0^+} f(\mathbf{L}) = \log(a\mathbf{L} + \mathbf{I}_n) = \mathbf{0}_n.$$
Moreover, $f(\mathbf{L})$ grows unbounded, as $a \to \infty$.

2. *Exponential function*: Let $f(x) = 1 - e^{-ax}$ with $a > 0$. Matrix $f(\mathbf{L}) = \mathbf{I}_n - e^{-a\mathbf{L}}$ ranges between $\mathbf{0}_n$ (as $a \to 0^+$) and $\mathbf{I}_n - \frac{1}{n}\mathbb{1}\mathbb{1}^T$ (as $a \to \infty$). In fact
$$\lim_{a \to \infty} e^{-a\mathbf{L}} = \mathbf{I}_n - \mathbf{L}\mathbf{L}^{\#} = \frac{1}{n}\mathbb{1}\mathbb{1}^T,$$
where $\mathbf{L}^{\#}$ denotes the group generalized inverse of $\mathbf{L}$ (Bernstein 2009, Prop. 11.8.2). Finally, if we consider the Loewner ordering (i.e. the partial ordering "$\preceq$" defined by the convex cone of positive semidefinite matrices), the following property holds true for $f(\mathbf{L}) = \mathbf{I}_n - e^{-a\mathbf{L}}$:
$$\mathbf{0}_n \preceq f(\mathbf{L}) \preceq \mathbf{I}_n - \frac{1}{n}\mathbb{1}\mathbb{1}^T.$$

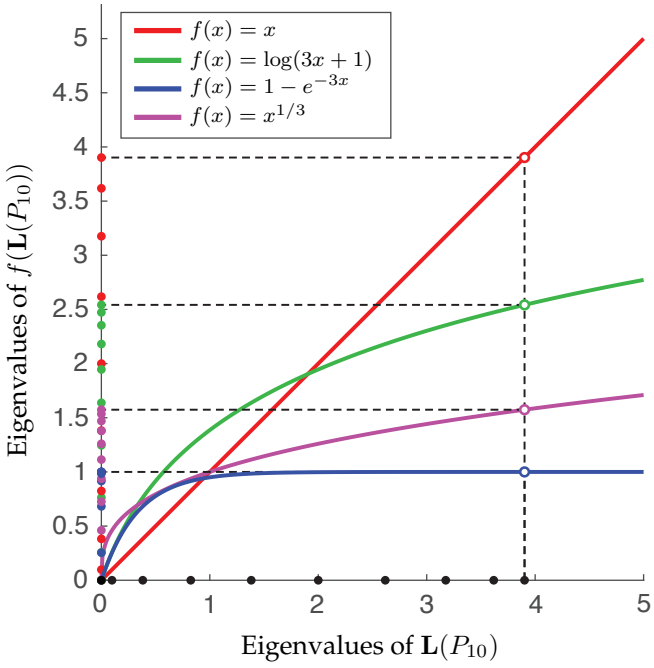


Figure 4.12: *Mapping of the spectrum of $\mathbf{L}(P_{10})$ via four admissible functions $f(x)$. The 10 eigenvalues of $\mathbf{L}(P_{10})$ are represented as black dots on the horizontal axis and those of $f(\mathbf{L}(P_{10}))$ as colored dots on the vertical axis.*

3. *Quadratic function*: Let $g(x) = \lambda - x$ for $0 \leq x < \lambda$, which is positive over the spectrum of $\mathbf{L}$, if $\lambda > \lambda_n$ with $2 - 2\cos((n-1)\pi/n) \leq \lambda_n \leq n$. By choosing such a λ, from (4.37), we have that $f(x) = \frac{1}{2}[\lambda^2 - (\lambda - x)^2] = \frac{1}{2}x(2\lambda - x)$, which yields the quadratic matrix function $f(\mathbf{L}) = -\frac{1}{2}\mathbf{L}^2 + \lambda\mathbf{L}$.

4. *Fractional power*: Let $f(x) = x^\gamma$ with $0 < \gamma \leq 1$. Matrix $f(\mathbf{L}) = \mathbf{L}^\gamma$ ranges between $\mathbf{I}_n$ (as $\gamma \to 0^+$) and $\mathbf{L}$ (for $\gamma = 1$).

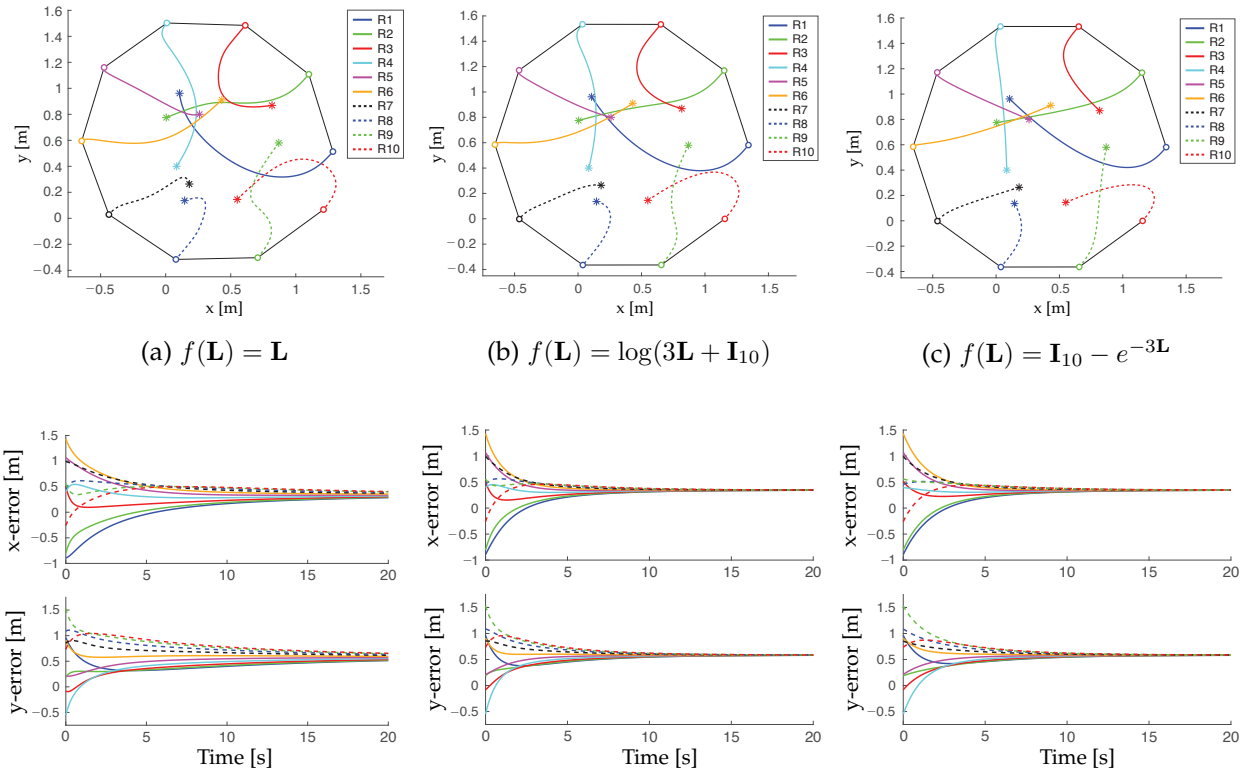
For the sake of illustration, Fig. 4.12 shows how the functions $f(x) = x$ (red), $f(x) = \log(3x+1)$ (green), $f(x) = 1 - e^{-3x}$ (blue), and $f(x) = x^{1/3}$ (magenta), map the 10 eigenvalues $2 - 2\cos(k\pi/10)$, $k \in \{0, 1, \ldots, 9\}$, of the Laplacian $\mathbf{L}$ of the path graph P_{10}. Note that the exponential function maps the spectrum of $\mathbf{L}(P_{10})$ into the $[0,\ 1)$ interval: in fact $1 - e^{-3\lambda_n} = 1 - 8 \times 10^{-6}$.

Numerical results

In our numerical experiments, we simulated system (4.40) for 10 robots, using different Laplacian functions, interaction graphs and vectors of target locations. We ran 20-second simulations using Matlab `ode45` solver with a variable step (max step size 0.01 s).

We used Matlab built-in commands `logm` and `expm` for calculating the matrix logarithm and matrix exponential, respectively. The computation of the fractional Laplacian being more delicate, we opted for the `rootpm_real` routine of *The Matrix Function Toolbox* (Higham 2008b) (see also (Higham 2008a, Appendix D)). This routine computes the pth root of a real matrix via the real Schur form: similar results were obtained with `rootpm_sign` which computes the pth root via the matrix sign function. Both routines handle singular matrices ($\mathbf{L}$ has a zero eigenvalue), and return non-principal pth roots.

In our **first example**, the target formation is a regular decagon of unit radius centered at the origin, i.e. $\boldsymbol{\xi} = [\cos(0),\ \sin(0),\ \cos(2\pi/10),\ \sin(2\pi/10),\ \ldots,\ \cos(9\pi/10), \sin(9\pi/10)]^T \in \mathbb{R}^{20}$. The interaction graph is $\mathcal{G} = P_{10}$, and the Laplacian functions considered are $f(\mathbf{L}) = \mathbf{L}$, $f(\mathbf{L}) = \log(3\mathbf{L} + \mathbf{I}_{10})$ and $f(\mathbf{L}) = \mathbf{I}_{10} - e^{-3\mathbf{L}}$. The initial positions $\mathbf{x}_0$ of the 10 robots have been randomly generated by drawing the x- and y-coordinates from the standard uniform distribution on the open interval $(0,\ 1)$. The first row of Fig. 4.13 reports the trajectory of the robots for each of the three functions (the edges of the formation graph are solid black, and the initial positions of the robots are marked with stars). The second row of Fig. 4.13 shows the corresponding time-evolution of the formation error $\boldsymbol{e}(t) = \mathbf{x}(t) - \boldsymbol{\xi}$ (x-coordinates, top; y-coordinates, bottom). In the three cases, $\boldsymbol{e}(t) \to [\tau_\mathrm{x}, \tau_\mathrm{y}, \ldots, \tau_\mathrm{x}, \tau_\mathrm{y}]^T$ as $t \to \infty$, where $\tau_\mathrm{x} \simeq 0.3451$ and $\tau_\mathrm{y} \simeq 0.5841$ are constant offsets. The average of the generalized degree associated with the three Laplacian functions is 1.8, 1.54 and 0.7833, respectively (recall Sect. 4.3.3), and their second smallest eigenvalues are $\lambda_2 = 2 - 2\cos(\pi/10) \simeq 0.0979$, $\log(3\lambda_2 + 1) \simeq 0.2575$ and



(a) $f(\mathbf{L}) = \mathbf{L}$ (b) $f(\mathbf{L}) = \log(3\mathbf{L} + \mathbf{I}_{10})$ (c) $f(\mathbf{L}) = \mathbf{I}_{10} - e^{-3\mathbf{L}}$

(d) $f(\mathbf{L}) = \mathbf{L}$ (e) $f(\mathbf{L}) = \log(3\mathbf{L} + \mathbf{I}_{10})$ (f) $f(\mathbf{L}) = \mathbf{I}_{10} - e^{-3\mathbf{L}}$

Figure 4.13: First example, $\boldsymbol{\xi}$ is the regular decagon and $\mathcal{G} = P_{10}$: *(1st row) Trajectory of the 10 robots. The initial and final positions are marked with a star "$*$" and a hollow circle "$\circ$", respectively, and the edges of the formation graph are solid black; (2nd row) Time evolution of the formation error* $\boldsymbol{e}(t) = \mathbf{x}(t) - \boldsymbol{\xi}$ *of the 10 robots (for color coding, see the legends in the 1st row). (a),(d)* $f(\mathbf{L}) = \mathbf{L}$*, (b),(e)* $f(\mathbf{L}) = \log(3\mathbf{L} + \mathbf{I}_{10})$*, and (c),(f)* $f(\mathbf{L}) = \mathbf{I}_{10} - e^{-3\mathbf{L}}$.

$1 - e^{-3\lambda_2} \simeq 0.2545$, respectively. Therefore, the logarithmic and exponential functions accelerate convergence towards the desired formation, and the convergence speed can be adjusted by tuning a single real parameter (the positive scalar a). However, both functions keep the agreement value unchanged.

To study the impact of the approximation introduced in Remark 4.4, in Fig. 4.14 we considered the same interaction graph, target formation, and initial conditions as in Figs. 4.13(b),(e), but we replaced $f(\mathbf{L}) = \log(3\mathbf{L} + \mathbf{I}_{10})$ with $f^q(\mathbf{L})$, where the threshold $q = 0.01$. Note that the exponential decay property of Theorem 4.2 holds for $f(\mathbf{L}) = \log(3\mathbf{L}+\mathbf{I}_{10})$ with $r = 1$. Figs. 4.14(a),(b) report the trajectory of the 10 robots and the time-evolution of the formation error $\boldsymbol{e}(t)$, respectively, and they qualitatively show that the proposed approximation has a negligible effect on the dynamic behavior of sys-

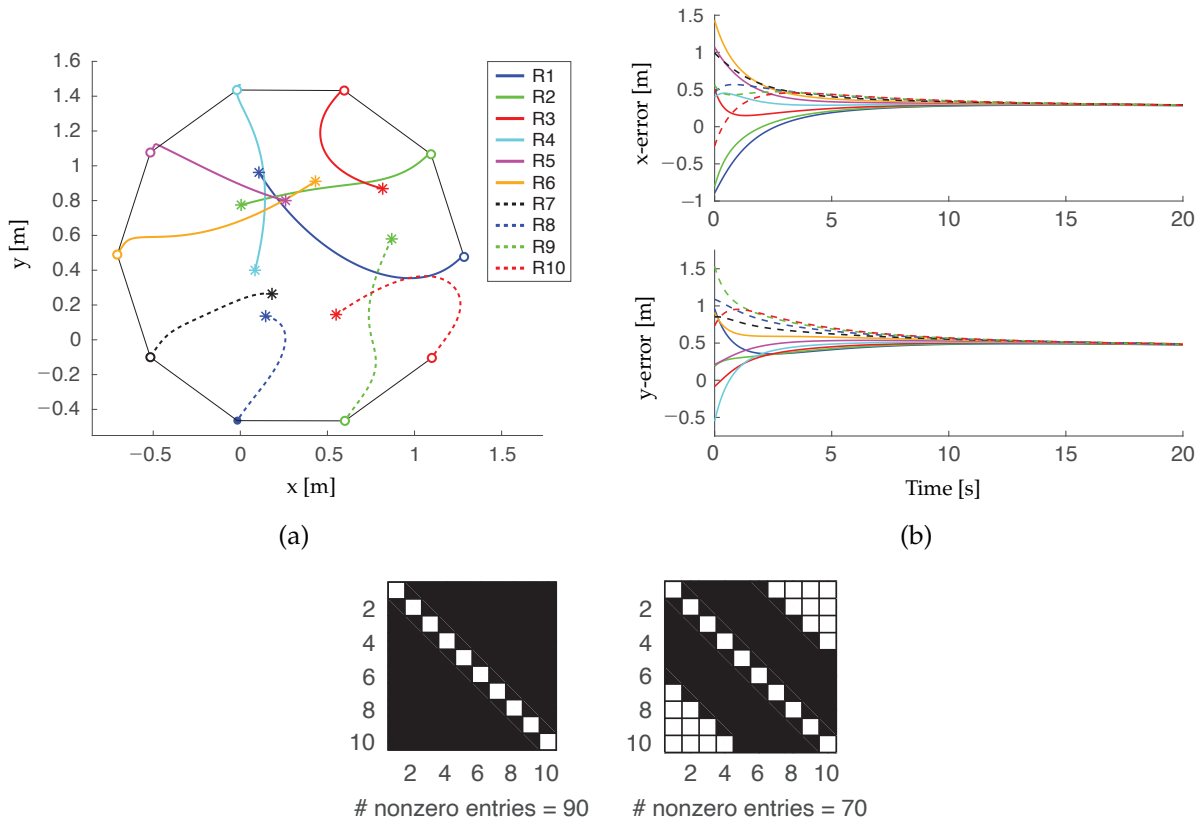


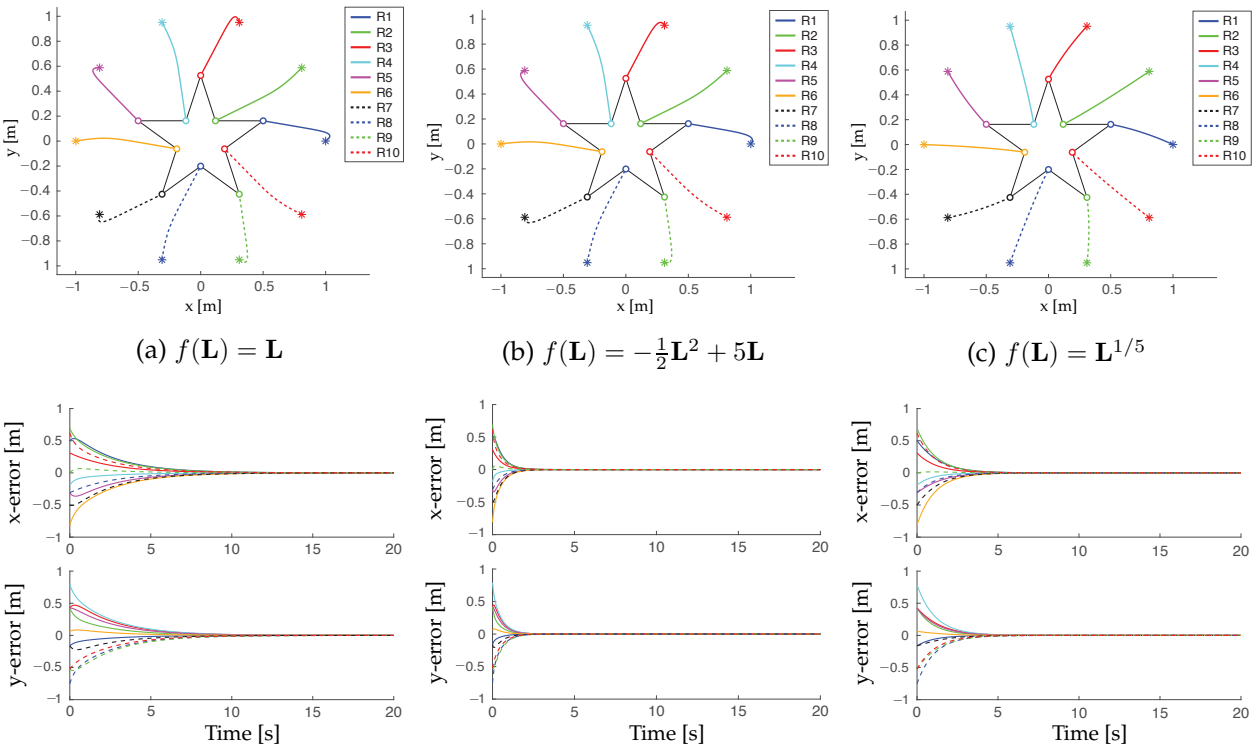
Figure 4.14: First example, $\boldsymbol{\xi}$ is the regular decagon and $\mathcal{G} = P_{10}$: *(a) Trajectory of the 10 robots obtained by considering the approximation* $f^q(\mathbf{L})$ *of* $f(\mathbf{L}) = \log(3\mathbf{L}+\mathbf{I}_{10})$ *with* $q = 0.01$*; (b) Time evolution of the formation error* $\boldsymbol{e}(t) = \mathbf{x}(t) - \boldsymbol{\xi}$ *of the 10 robots; (c) Sparsity pattern (white squares, zero entries; black squares, nonzero entries) of the adjacency matrices associated with* $f(\mathbf{L})$ *(left) and* $f^q(\mathbf{L})$ *(right).*

tem (4.40). More specifically, at the end-time, $t = 20$ s, we have that $\|\boldsymbol{e}(t)\| \simeq 2.1453$ for $f(\mathbf{L})$ and $\|\boldsymbol{e}(t)\| \simeq 1.7851$ for $f^q(\mathbf{L})$. However, the weighted interaction graph associated with $f^q(\mathbf{L})$ is sparser than that associated with $f(\mathbf{L})$.

This is evident in Fig. 4.14(c), which shows the sparsity pattern of adjacency matrices $\mathrm{diag}(f_{1,1}(\mathbf{L}), \ldots, f_{10,10}(\mathbf{L})) - f(\mathbf{L})$ (left) and $\mathrm{diag}(f^q_{1,1}(\mathbf{L}), \ldots, f^q_{10,10}(\mathbf{L})) - f^q(\mathbf{L})$ (right), where the zero entries are white and the nonzero entries are black.

In our **second example**, we chose the interaction graph $\mathcal{G} = C_{10}$, the cycle graph with 10 nodes, and the desired formation is the pentagram (or five-pointed star). Let

$$R = \sqrt{\frac{5-\sqrt{5}}{10}}, \quad \rho = \sqrt{\frac{25-11\sqrt{5}}{10}},$$



(a) $f(\mathbf{L}) = \mathbf{L}$ (b) $f(\mathbf{L}) = -\frac{1}{2}\mathbf{L}^2 + 5\mathbf{L}$ (c) $f(\mathbf{L}) = \mathbf{L}^{1/5}$

(d) $f(\mathbf{L}) = \mathbf{L}$ (e) $f(\mathbf{L}) = -\frac{1}{2}\mathbf{L}^2 + 5\mathbf{L}$ (f) $f(\mathbf{L}) = \mathbf{L}^{1/5}$

Figure 4.15: Second example, $\boldsymbol{\xi}$ is the pentagram and $\mathcal{G} = C_{10}$: *(1st row) Trajectory of the 10 robots. The initial and final positions are marked with a star "$*$" and a hollow circle "$\circ$", respectively, and the edges of the formation graph are solid black; (2nd row) Time evolution of the formation error* $\boldsymbol{e}(t) = \mathbf{x}(t) - \boldsymbol{\xi}$ *of the 10 robots (for color coding, see the legends in the 1st row). (a),(d)* $f(\mathbf{L}) = \mathbf{L}$*, (b),(e)* $f(\mathbf{L}) = -\frac{1}{2}\mathbf{L}^2 + 5\mathbf{L}$*, and (c),(f)* $f(\mathbf{L}) = \mathbf{L}^{1/5}$.

be the circumradius of the pentagram and the circumradius of its inner pentagon, respectively. Then, the target locations of the 10 robots can be expressed in polar coordinates as

$$\begin{bmatrix} [\boldsymbol{\xi}]_j \\ [\boldsymbol{\xi}]_{j+1} \end{bmatrix} = \begin{cases} \rho\big[\cos(\frac{j\pi}{10}),\ \sin(\frac{j\pi}{10})\big]^T \text{ if } j \in \{1, 5, 9, \ldots\}, \\ R\big[\cos(\frac{j\pi}{10}),\ \sin(\frac{j\pi}{10})\big]^T \text{ if } j \in \{3, 7, 11, \ldots\}, \end{cases}$$

where $j \in \{1, 3, 5, \ldots, 19\}$. The Laplacian functions considered in this second example, are $f(\mathbf{L}) = \mathbf{L}$, $f(\mathbf{L}) = -\frac{1}{2}\mathbf{L}^2 + 5\mathbf{L}$ and $f(\mathbf{L}) = \mathbf{L}^{1/5}$. In the quadratic function, we set $\lambda = 5 > \lambda_{10} = 4$, to satisfy the condition discussed in the "Case studies" section. The vector of initial positions $\mathbf{x}_0$, is not random this time: the 10 robots are initially placed at the vertices of the same regular decagon of unit radius considered as target formation in the first example. As in Fig. 4.13, the first row of Fig. 4.15 reports the trajectory

of the 10 robots for the three Laplacian functions, and the second row, the corresponding time-evolution of the formation error $\boldsymbol{e}(t) = \mathbf{x}(t) - \boldsymbol{\xi}$. Since the regular decagon is centered at the origin, we have that $\boldsymbol{e}(t) \to \mathbf{0}$ as $t \to \infty$, in all cases. Finally, note that the average fractional degree of $\mathbf{L}^{1/5}$ is 1.0115, and that the user can take advantage of the non-integer power γ to modulate the convergence speed of the 10 robots towards the desired formation. In fact, the second smallest eigenvalues of $\mathbf{L}$ and $\mathbf{L}^{1/5}$, are $\lambda_2 = 2 - 2\cos(\pi/5) \simeq 0.3820$ and $\lambda_2^{1/5} \simeq 0.8249$, respectively, which explains the faster convergence rate observed in Figs. 4.15(c),(f).

4.3.5 Conclusions and future work

In this last section of Chapter 4, we have explored a general class of matrix functions of the combinatorial Laplacian, that retain its structural properties. This allowed us to present under a common framework, several variants of the Laplacian scattered across different domains. For the first time, this family of Laplacian functions has been utilized in a consensus-based formation control protocol, revealing some attractive features. In fact, the selected functions do not alter the agreement value, and offer greater design flexibility. For example, our numerical experiments show that in many situations, a single scalar parameter is sufficient to adjust the convergence speed towards consensus without drastically increasing the number of communication exchanges between the agents.

This work lays the ground for further research and experimentation. One line of future research will be to relax Condition III and consider a broader class of admissible Laplacian functions. As far as the consensus protocol is concerned, we see potential for extension to weighted directed interaction graphs, and we plan to perform hardware experiments with mobile robots. Finally, in (Michelitsch et al. 2019), the authors have shown a connection between the fractional Laplacian of a graph and the operators in fractional calculus. There might then exist a link between the agreement protocol driven by the fractional Laplacian, and the coordination algorithms for fractional-order systems studied in (Cao et al. 2010).

4.4 Conclusion

In this chapter, we have presented original insights and results pertaining to the formation control problem. Extensive numerical examples and simulation results have supported the proposed theory. In particular, in Sect. 4.1, we have studied distance-bearing formations of unicycle robots and proposed a natural extension to the well-known classification of wheeled robots by *type*. In Sect. 4.2, we have presented decentralized optimal strategies for cooperative adaptive cruise control of a car platoon (a special 1D leader-follower formation control problem). Moreover, as an original contribution, we showed how to explicitly incorporate the string-stability constraints of the platoon into the optimization-based formulation. In Sect. 4.3, we have revisited the notion of function of Laplacian matrix originally introduced in (Michelitsch et al. 2019), and explored its possible application to shape-based formation control of single-integrator robots.

As mentioned in the partial conclusions of the three sections, the results presented in this chapter provide fertile ground for further research. To fill the gap between theory and practice, the validation of the proposed control strategies with real robots and instrumented cars is important subject of future work.

In the next chapter, three tasks that need the coordinated action of multiple autonomous robots will be analyzed and two further extensions of the classical consensus protocol will be discussed.

Chapter 5

Coordinated control of multi-robot systems

If an elderly but distinguished scientist says that something is possible, he is almost certainly right; but if he says that it is impossible, he is very probably wrong.

Arthur C. Clarke

Abstract

This chapter is divided into two parts. In the first part, we deal with three tasks (environmental monitoring, active target tracking, 3D reconstruction), which require the coordinated action of of a group of aerial robots and the interaction with the surrounding environment. For these tasks, the goal will be to design local *and* decentralized *control strategies, which are generally preferable to global and centralized ones, because of their inherent scalability and robustness. In the second part of the chapter, we present two extensions of the conventional consensus protocol, that we have already encountered in Chapters 3 and 4: we first introduce a generalization of combinatorial Laplacian which guarantees enhanced flexibility in the execution of coordinated tasks (rendezvous of single-integrator robots will be considered as a benchmark problem). Second, we propose a general geometric formulation to easily handle the state constraints in the discrete-time consensus protocol without affecting the consensus value, and show relevant applications (obstacle avoidance).*

The material of this chapter is drawn from (Morbidi, Freeman and Lynch 2011, Morbidi and Mariottini 2013, Hardouin et al. 2023) and (Morbidi 2013a, Morbidi 2020) (cf. [C19], [J12], [J24] and [J13], [C35] in Sect. 1.1.2).

In Chapter 4, we have seen two examples of distributed algorithms for the formation control problem (cf. Sect. 4.2 and Sect. 4.3). In this chapter, we develop this aspect further and consider other relevant tasks which can be performed by a team of mobile robots in a distributed fashion. In all cases, the consensus protocol, a fundamental building block for the design of distributed control laws, will play a central role. In fact, this protocol and its dynamic extensions (dynamic average consensus estimators (Kia et al. 2019)) allow, under certain assumptions, to transform a centralized controller or estimator into a decentralized one. The plan of this chapter is the following. In Sect. 5.1,

Sect. 5.2 and Sect. 5.3, we study three related problems pertaining to the coordination of multiple UAVs in the 3D world (cooperative environmental monitoring based on geometric moments, cooperative active target tracking, and cooperative 3D reconstruction of unknown environments). In Sect. 5.4 and Sect. 5.5, we consider two variants of the traditional consensus protocol by introducing a new GSO and a new principled approach based on subspace projectors which allows to seamlessly incorporate state constraints. Finally, Sect. 5.6 recapitulates the major contributions of the chapter.

The work in Sect. 5.1, Sect. 5.2 and Sect. 5.4 was carried out during my stays, as a post-doc, at Northwestern University, University of Texas at Arlington, USA, and Joannes Kepler University, Austria, respectively (2009-2012). The material in Sect. 5.3 is part of the Ph.D. thesis of G. Hardouin, which I co-supervised with E. Mouaddib and J. Moras (ScanBot project, 2018-2022, see Sect. 1.1.5). Finally, the results presented in Sect. 5.3 and Sect. 5.5 were obtained at ONERA DTIS and at the University of Picardie Jules Verne.

Chapter 5

Part I: Cooperative tasks with UAVs

5.1 Cooperative environmental monitoring

This section presents a distributed estimation and control strategy for cooperative monitoring by swarms of *unmanned aerial vehicles* (UAVs) modeled as constant-speed unicycles. The *geometric moments*, encoding an abstraction of the swarm, are controlled via a nonlinear gradient descent to match those of a discrete set of particles describing the occurrence of some event of interest to be monitored. Because of its limited sensing capabilities, each agent can measure the position of only a subset of the overall particles, from which it locally estimates the desired moments of the swarm running a *proportional-integral* (PI) *average consensus estimator*. The closed-loop stability of the system arising from the combination of the gradient-descent controllers and the consensus estimators is studied and simulation results are provided to illustrate the theory.

5.1.1 Introduction

Recent years have witnessed an acceleration in research efforts aimed at designing environmental monitoring algorithms for mobile sensor networks (Leonard et al. 2007, Smith et al. 2010). In fact, as known, *mobile sensors* offer distinctive advantages over *static* ones, in terms of quality of sensing and estimation, area coverage, adaptability to changing conditions and robustness against failures.

Our goal is to monitor a set of moving particles describing the occurrence of some event of interest in a 2-D environment, with a team of *unmanned aerial vehicles* (UAVs) flying at fixed altitude. With the term *particle*, we refer to any discrete entity belonging to a given ensemble, whose position in the plane has to be tracked over time: examples include animals in a group, people in a crowd, smoke particles in a plume, multiple wildfire spots, droplets in an oil spill. The "shape" of the UAV swarm and of the ensemble of particles is synthetically described in terms of their *geometric moments*. Because of its limited sensing capabilities, each agent, modeled as a *unicycle* with constant positive forward velocity, can only measure the position of a subset of the overall particles. The objective is to design a distributed estimation and control strategy to match the moments of the swarm with those of the particles: this in turn guarantees the UAVs to properly cover the region of interest.

Literature review

Two literature domains are relevant to this work. In the first one, the goal is to design *distributed algorithms for multiple agents to detect and track the boundary of a region of interest*. In (Marthaler and Bertozzi 2004), a "snake algorithm" is adopted to identify and track the boundary of harmful algae blooms using a team of agents equipped with

chemical sensors. In (Clark and Fierro 2007), a random coverage controller is used to detect and surround oil spills, and in (Casbeer et al. 2006) an algorithm is described to allow multiple UAVs to cooperatively monitor and track the propagation of large forest fires. Recently, in (Susca et al. 2008), a method has been proposed to optimally approximate an environmental boundary with a polygon. The mobile agents rely only on sensed local information to position some interpolation points and define the approximating polygon whose vertices are uniformly distributed along the boundary of the target region.

In the second literature domain of interest for this work, the goal is to synthesize *decentralized simultaneous estimation and control strategies for multiple agents*. In (Yang et al. 2008), a general framework to design collective behaviors for groups of mobile robots has been proposed: each agent communicates with its neighbors and estimates the global performance properties of the robotic network needed to make a local control decision. In (Lynch et al. 2008), a decentralized strategy for modeling of environmental parameters is presented and a gradient control is used to move the agents in order to maximize their sensory information relative to the current uncertainty in the model. A distributed learning and cooperative control strategy has been proposed in (Choi et al. 2009). Each agent recursively estimates an unknown field of interest from noisy measurements and moves towards its peaks using the gradient of its estimated field, while maintaining network-wide connectivity. Other distributed approaches to controlled sampling and modeling of deterministic or stochastic scalar fields, include (Ögren et al. 2004, Zhang and Leonard 2010, Martínez 2010). In (Ögren et al. 2004), the agents move in a fixed platoon along an estimated gradient, while in (Zhang and Leonard 2010) they are controlled to track a level set of the field. Recently, in (Martínez 2010), a procedure to adapt local interpolations to represent spatial fields as they are measured by a mobile sensor network, has been presented.

Original contributions

We propose here a distributed estimation and control strategy for cooperative monitoring by swarms of UAVs. Differently from the first literature domain above, our focus is not on the boundary of the region spanned by the particles (that may be faint or fuzzy in real settings, and thus hard to detect and track), but on controlling the (first- and second-order) *geometric moments* encoding an abstraction of the swarm (Belta and Kumar 2004, Michael and Kumar 2009), to match the moments of the ensemble of particles observed by the UAVs. Although full aircraft dynamics are quite complex, the essential components for level cruise flight can be captured by the model of a planar *constant-speed unicycle*. This model is challenging to control because the vehicle cannot stop (nor move directly sideways), and so it is not small-time locally controllable (Lalish

et al. 2007). A new nonlinear gradient-descent angular control is proposed to steer our team of unicycle-type vehicles.

Differently from (Cortés et al. 2004, Kwok and Martínez 2010) and related works in the coverage literature, we assume here that the agents have *not* access to a *distribution density function*, providing an *a priori global* measure of information or probability that some event takes place over the region of interest. On the contrary, similarly to (Schwager et al. 2009) where sensor measurements are used to learn the distribution of sensory information in the environment, each agent is equipped with a limited-footprint sensor (e.g., a camera pointing downward), which allows it to detect only a fraction of the overall particles. We will abstract from any actual sensor model, and assume that each agent processes only the particles lying within the Voronoi cell that it generates (and ignores those possibly located outside): from them each UAV locally estimates the desired moments of the swarm running a *proportional-integral* (PI) *average consensus estimator* (Lynch et al. 2008). As known, if the inputs of the PI estimators as well as the topology of the connected network are slowly varying (this is true in our setting since the dynamics of the environment is assumed to be significantly slower than that of the swarm, cf. Theorem 5.2), small estimation errors are achieved. The closed-loop stability of the system arising from the combination of the gradient-descent angular controllers and the PI estimators has been studied and simulation experiments have been performed to illustrate the proposed theory.

The rest of the section is organized as follows. In Sect. 5.1.2 we introduce the gradient-based control, assuming that the desired moments of the swarm are *a priori known* to each agent. In Sect. 5.1.3 and Sect. 5.1.4 we deal with the distributed estimation problem and the closed-loop stability analysis, respectively. Finally, in Sect. 5.1.5 simulation results are presented and in Sect. 5.1.6 the main contributions are summarized and possible avenues for future research are highlighted.

5.1.2 Control design

Consider a swarm of n unmanned aerial vehicles (hereafter, simply *agents* or *vehicles*) flying at fixed altitude, with the following unicycle model (cf. Sect. 3.3.2),

$$\begin{cases} \dot{p}_{ix}(t) = v_i(t)\cos(\theta_i(t)), \\ \dot{p}_{iy}(t) = v_i(t)\sin(\theta_i(t)), \quad i \in \{1,\dots,n\}, \\ \dot{\theta}_i(t) = \omega_i(t), \end{cases} \tag{5.1}$$

where $\mathbf{p}_i(t) = [p_{ix}(t),\, p_{iy}(t)]^T \in \mathbb{R}^2$ denotes the position of agent i at time t in the plane of motion, $\theta_i(t) \in [-\pi,\, \pi)$ its heading and $[v_i(t),\, \omega_i(t)]^T \in [v_{\min}, +\infty) \times \mathbb{R}$, $v_{\min} > 0$, its forward and angular velocities. Let $\mathbf{p} = [\mathbf{p}_1^T, \dots, \mathbf{p}_n^T]^T \in (\mathbb{R}^2)^n$. The configuration of

the agents is described by using a *swarm moment function* $\mathbf{f} : (\mathbb{R}^2)^n \to \mathbb{R}^\ell$ that we will assume to be of the form:

$$\mathbf{f}(\mathbf{p}) = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\phi}(\mathbf{p}_i),$$

where the *moment-generating function* $\boldsymbol{\phi} : \mathbb{R}^2 \to \mathbb{R}^\ell$ is defined as,

$$\boldsymbol{\phi}(\mathbf{p}_i) \triangleq [\, p_{ix},\ p_{iy},\ p_{ix}^2,\ p_{iy}^2,\ p_{ix}p_{iy},\ p_{ix}^3,\ p_{iy}^3, \dots]^T. \tag{5.2}$$

Note that $\ell = \frac{1}{2}(r+1)(r+2) - 1$ where $r \in \mathbb{Z}_{>0}$ is the maximum order of the moments appearing in (5.2), and that if ℓ moment constraints are specified on n agents then there is in general a $(2n-\ell)$-dimensional algebraic set of swarm configurations that satisfy them. The primary objective of the agents is to move so that their final arrangement minimizes the error $\mathbf{f}(\mathbf{p}) - \mathbf{f}^\star$, where the *goal vector* $\mathbf{f}^\star \in \mathrm{im}(\mathbf{f})$ defines the desired shape of the formation. For the sake of simplicity, thorough this section we will assume that $\mathbf{f}^\star$ is *constant* and *a priori known* to each agent: we will relax this hypothesis in Sect. 5.1.3, where each agent independently estimates the goal vector from the environmental data. Our control strategy relies on the gradient of the *potential function* $\Pi : (\mathbb{R}^2)^n \to \mathbb{R}_{\geq 0}$,

$$\Pi(\mathbf{p}) = (\mathbf{f}(\mathbf{p}) - \mathbf{f}^\star)^T\, \boldsymbol{\Gamma}\, (\mathbf{f}(\mathbf{p}) - \mathbf{f}^\star), \tag{5.3}$$

where $\boldsymbol{\Gamma} \in \mathbb{R}^{\ell\times\ell}$ is an assigned symmetric positive-definite gain matrix. Let $\mathrm{Crit}(\Pi) \triangleq \{\mathbf{p} \in (\mathbb{R}^2)^n : \nabla_{\mathbf{p}}\, \Pi(\mathbf{p}) = \mathbf{0}\}$ denote the set of critical points of (5.3) and classify such points as *good critical points* where $\mathbf{f}(\mathbf{p}) = \mathbf{f}^\star$ (these are the global minima of Π) and *bad critical points* where $\mathbf{f}(\mathbf{p}) \neq \mathbf{f}^\star$. Given a closed set of swarm configurations $\mathcal{P} \subset (\mathbb{R}^2)^n$ and a goal vector $\mathbf{f}^\star \in \mathbf{f}(\mathcal{P})$, let $\mathcal{G}(\mathbf{f}^\star, \mathcal{P})$ be the convex cone of all symmetric positive-definite matrices $\boldsymbol{\Gamma}$ such that no bad critical points of Π in $\mathcal{P}$ are local minima of Π. To reduce the risk of the swarm "getting stuck" at bad critical points of Π, we would ideally choose a gain matrix $\boldsymbol{\Gamma}$ belonging to $\mathcal{G}(\mathbf{f}^\star, \mathcal{P})$ for a large set $\mathcal{P}$. Actually, for $r = 2$ one can always compute members of $\mathcal{G}(\mathbf{f}^\star, \mathcal{P})$ when $\mathcal{P}$ contains all possible configurations of at least 3 agents. This idea is made precise in the following theorem, readapted from (Yang et al. 2008, Th. 2):

5.1. THEOREM. *Let $\mathcal{P} = (\mathbb{R}^2)^n$ with $n \geq 3$ and $\mathbf{f}^\star \in \mathbf{f}(\mathcal{P})$. Then there exists a symmetric positive-definite matrix $\boldsymbol{\Gamma}$ such that for every bad critical point $\mathbf{p} \in \mathcal{P}$ of Π, the Hessian matrix $\mathcal{H}(\Pi(\mathbf{p}))$ has at least one strictly negative eigenvalue (hence, $\mathbf{p}$ cannot be a local minimum of Π). In particular, $\boldsymbol{\Gamma} \in \mathcal{G}(\mathbf{f}^\star, \mathcal{P})$.* ■

In the rest of this section, we will then restrict to the case of $r = 2$, (i.e., $\ell = 5$), i.e., only the *first- and second-order moment statistics* will be considered. To provide concise statements in the sequel, we introduce the function $\mathrm{proj}(\cdot)$, which maps the angle $\alpha \in \mathbb{R}$ into the interval $[-\pi, \pi)$,

$$\mathrm{proj}(\alpha) \triangleq ((\alpha + \pi) \bmod 2\pi) - \pi, \tag{5.4}$$

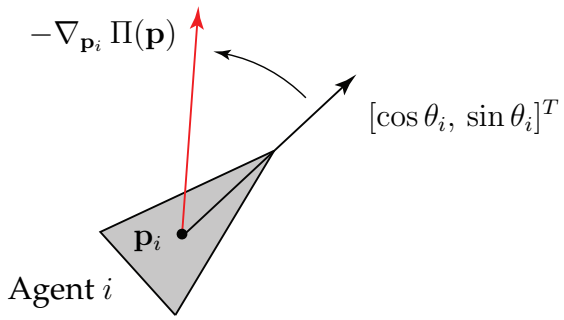


Figure 5.1: *The angular control of agent i forces its heading direction $[\cos\theta_i,\ \sin\theta_i]^T$ to align with the negative gradient of $\Pi(\mathbf{p})$.*

where "mod" stands for the modulo operator which returns the remainder after division. The symbol $\mathbf{A}[i,j]$ will be used to denote the (i,j)-th component of a matrix $\mathbf{A}$.

The geometric intuition behind our control strategy (see equation (5.5) below) is simple: in fact, the forward velocity of the agents is set to the same positive constant value and the angular velocities are chosen so that the heading direction of each vehicle is forced to align with the negative gradient of the potential function $\Pi(\mathbf{p})$ (see Fig. 5.1). Let $n \geq 3$ and choose $\mathbf{\Gamma}$ and $\mathcal{P}$ as in Theorem 5.1. Let us define the vector function $\mathbf{g}_i : \mathbb{R}_{\geq 0} \to \mathbb{R}^2$, $i \in \{1,\ldots,n\}$,

$$\mathbf{g}_i(t) \triangleq -\nabla_{\mathbf{p}_i}\Pi(\mathbf{p}(t)) = -(\mathcal{J}\boldsymbol{\phi}(\mathbf{p}_i(t)))^T\,\mathbf{\Gamma}\,(\mathbf{f}(\mathbf{p}(t)) - \mathbf{f}^\star),$$

where $\mathcal{J}\boldsymbol{\phi}(\cdot) \in \mathbb{R}^{\ell\times 2}$ is the Jacobian matrix of $\boldsymbol{\phi}(\cdot)$ and set $\alpha_i(t) \triangleq \text{proj}(\arg(\mathbf{g}_i(t)) - \theta_i(t))$ where $\arg : \mathbb{R}^2 \to [-\pi,\ \pi)$. Consider the following control input for agent i,

$$v_i(t) = v, \qquad \omega_i(t) = \rho\,\alpha_i(t), \tag{5.5}$$

where $v \geq v_{\min}$ is a positive constant and ρ is a positive gain.

We observe the following two properties, for almost every initial configuration of the agents:

a) For any $\epsilon > 0$, there exists a sufficiently large gain ρ such that the error on the desired swarm configuration is uniformly ultimately bounded with an ultimate bound ϵ, i.e., for every $\zeta > 0$ there is a positive constant $t_0 = t_0(\zeta)$ such that:

$$\Pi(\mathbf{p}(0)) < \zeta \ \Rightarrow\ \Pi(\mathbf{p}(t)) \leq \epsilon,\ \forall\, t \geq t_0.$$

b) Let $d^\theta_{ij}(t) \triangleq \text{proj}(\theta_i(t) - \theta_j(t))$ and $d^\omega_{ij}(t) \triangleq \omega_i(t) - \omega_j(t)$, $i,j \in \{1,\ldots,n\}$, $i \neq j$, be the functions measuring the disagreement between the heading directions and angular velocities of agents i and j at time t, respectively. Then, for any $\epsilon_\theta,\ \epsilon_\omega > 0$ there exists a sufficiently large constant $\mu \in \mathbb{R}_{>0}$ satisfying,

$$\mathbf{\Gamma}[1,1], \mathbf{\Gamma}[2,2] \geq \mu\big|\mathbf{\Gamma}[h,l]\big|, \quad h,l \in \{1,\ldots,\ell\},\ (h,l) \neq \{(1,1),\ (2,2)\},$$

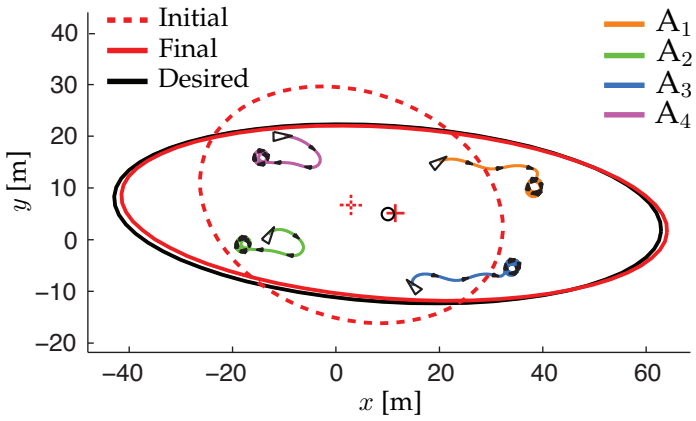


Figure 5.2: Example*: Trajectory of $n = 4$ agents using control (5.5). The red dashed ellipse graphically represents the* initial *moments of the swarm, the red solid ellipse the moments of the swarm at the* final time instant *($t = 80$ sec.), and the black ellipse the* desired *moments of the swarm.*

such that $|d_{ij}^{\theta}(t)|$, $|d_{ij}^{\omega}(t)|$ are uniformly ultimately bounded with ultimate bounds ϵ_θ, ϵ_ω, respectively.

Point b) states that $d_{ij}^{\theta}(t)$ becomes bounded as $t \to +\infty$: this behavior is typically referred to as *phase locking* in the literature of coupled oscillators (Strogatz 2000).

5.1. REMARK. *Note that the angular control in (5.5) presents discontinuities: in fact, the function proj$(\cdot)$ in (5.4) is discontinuous at $(2m+1)\pi$, $m \in \mathbb{Z}$, and* arg$(\cdot)$ *is not defined at the origin. It is possible to modify the control in order to make it smooth: however, in order to simplify our subsequent analysis we will not pursue this direction herein.* ◇

5.2. REMARK. *To compute the angular control in (5.5), agent i needs to know the position of* all *the other agents (i.e., the vector $\mathbf{p}$) at each time instant. This means that control (5.5) is* not *implementable in a distributed fashion. We will sidestep this issue in Sect. 5.1.3 where we will introduce distributed estimators of the swarm moment function $\mathbf{f}(\mathbf{p})$ (as well as of the goal vector, using the environmental data) for each agent.* ◇

For the sake of illustration, Fig. 5.2 shows the trajectory of $n = 4$ agents implementing control (5.5) with $v = 1$ m/s, $\rho = 0.5$, $\mathbf{f}^\star = [10,\ 5,\ 800,\ 100,\ 10]^T$ and $\mathbf{\Gamma} = \text{diag}(1000,\ 1000, 0.1, 0.1, 0.1)$. The red dashed ellipse graphically represents the *initial* moments of swarm (i.e., the uniform-density ellipse has the same mass and the same first- and second-order moments as the swarm at the initial time), the red solid ellipse the moments of the swarm at the *final time instant*, and the black ellipse the *desired* moments of the swarm. Note that differently from (Yang et al. 2008), where double-integrator

agents are considered, the red solid and black ellipses are not perfectly superimposed at steady state and the vehicles hover about four points corresponding to a minimum of $\Pi(\mathbf{p})$. In other words, the swarm converges to a configuration that satisfies the desired moment statistics up to an inherent error, consequence of the inability of the agents to stop moving forward. Actually, with a control of the form (5.5), spinning around the equilibrium turns out to be the "best strategy" for the vehicles to reach the goal.

5.1.3 Distributed estimation

In this section we present a distributed algorithm to locally estimate the swarm moment function $\mathbf{f}(\mathbf{p})$ and the vector of desired geometric moments using the environmental data. The following notion of *Voronoi partition* (Okabe et al. 2000) is essential for the forthcoming developments.

5.1. DEFINITION (VORONOI PARTITION). *Given a set $\mathcal{Q} \subset \mathbb{R}^2$ and n distinct points $\{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n\}$ in $\mathcal{Q}$, the* Voronoi partition *of $\mathcal{Q}$ generated by $\{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n\}$ is the collection of sets $\{V_1, \ldots, V_n\}$ defined by, for each $i \in \{1, \ldots, n\}$,*

$$V_i \triangleq \big\{\boldsymbol{z} \in \mathcal{Q} \mid \|\boldsymbol{z} - \boldsymbol{p}_i\| \leq \|\boldsymbol{z} - \boldsymbol{p}_j\|,\ \forall\, j \neq i\big\},$$

where $\|\cdot\|$ denotes the standard Euclidean norm. We will refer to V_i as the Voronoi cell *of $\boldsymbol{p}_i$.* $\diamond$

Given a set $\mathcal{Q} \subset \mathbb{R}^2$, let $\mathbf{q}_k = [q_{kx},\, q_{ky}]^T$, $k \in \{1, \ldots, N\}$ be the k-th of N *particles* describing the occurrence of some event of interest in $\mathcal{Q}$ and evolving over time according to,

$$\dot{\mathbf{q}} = \boldsymbol{\Upsilon}(\mathbf{q},\, t), \tag{5.6}$$

where $\mathbf{q} = [\mathbf{q}_1^T, \ldots, \mathbf{q}_N^T]^T$ and $\boldsymbol{\Upsilon} = [\boldsymbol{\Upsilon}_1^T, \ldots, \boldsymbol{\Upsilon}_N^T]^T : \mathcal{Q}^N \times \mathbb{R}_{\geq 0} \to (\mathbb{R}^2)^N$ is a vector field *unknown* to the agents. Since each vehicle is assumed to be equipped with a limited-footprint sensor (e.g., a camera pointing downward), it will be able to measure the x-, y-coordinates of only a subset of the N particles. For the sake of simplicity, we will abstract from any actual sensor model and assume that agent i processes only the particles lying within the Voronoi cell V_i that it generates, while ignoring those possibly located outside (see Fig. 5.3): in other words, agent i is only responsible for the particles over its "dominance region" V_i. Note that since the sides $V_i \cap V_j$, $i \neq j$, of the Voronoi cells are of measure zero, each particle will be assigned exactly to one agent, thus avoiding possible double counting. We will denote by N_i, $0 < N_i < N$, the number of particles lying within V_i (note that by construction $\sum_{i=1}^n N_i = N$), and we will assume that agent i computes the following vector from the N_i particles:

$$\mathbf{h}_i \ = \sum_{\mathbf{q}_k :\, \mathbf{q}_k \in V_i} \boldsymbol{\phi}(\mathbf{q}_k), \quad i \in \{1, \ldots, n\}.$$

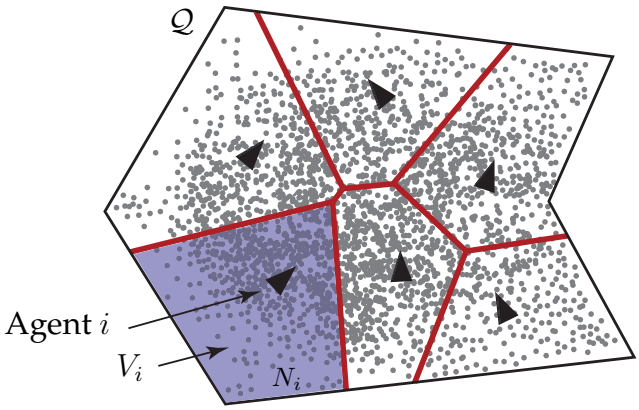


Figure 5.3: *Voronoi partition of the set $\mathcal{Q}$ generated by six agents (black triangles). Agent i processes only N_i of the overall N particles (gray dots): these are the particles lying within the Voronoi cell V_i of $\mathbf{p}_i$ (purple).*

Note that the Voronoi cells $V_1, \ldots, V_n$ can be *locally* computed and maintained by the agents using the distributed asynchronous algorithms presented in (Cortés et al. 2004, Sect. IVB).

We recall here that our ultimate goal is to match the geometric moments of the swarm with those of the ensemble of particles: this in turn guarantees a suitable coverage of the region spanned by the particles. In order to obtain local estimates of the function $\mathbf{f}(\mathbf{p})$, necessary for a distributed implementation of control (5.5) (recall Remark 5.2) and of the *environmental goal vector*,

$$\mathbf{f}^\star_{\text{env}} \triangleq \frac{1}{N}\sum_{k=1}^{N} \boldsymbol{\phi}(\mathbf{q}_k) = \frac{1}{N}\sum_{i=1}^{n} \mathbf{h}_i,$$

obtained from the *overall* particles and thus unknown to the vehicles, we will suppose that agent i runs the following *proportional-integral (PI) average consensus estimator* (Yang et al. 2008, Lynch et al. 2008):

$$\dot{\boldsymbol{\xi}}_i = -\gamma\,\boldsymbol{\xi}_i - \sum_{j\neq i}\sigma(\mathbf{p}_i,\mathbf{p}_j)\,(\boldsymbol{\xi}_i - \boldsymbol{\xi}_j) + \sum_{j\neq i}\tau(\mathbf{p}_i,\mathbf{p}_j)\,(\boldsymbol{\eta}_i - \boldsymbol{\eta}_j) + \gamma\,[\boldsymbol{\phi}(\mathbf{p}_i)^T,\, \mathbf{h}_i^T,\, N_i]^T, \tag{5.7}$$

$$\dot{\boldsymbol{\eta}}_i = -\sum_{j\neq i}\tau(\mathbf{p}_i,\mathbf{p}_j)\,(\boldsymbol{\xi}_i - \boldsymbol{\xi}_j),$$
$$\boldsymbol{\chi}_i = \boldsymbol{\xi}_i[1:\ell] - \frac{\boldsymbol{\xi}_i[\ell+1:2\ell]}{\boldsymbol{\xi}_i[2\ell+1]}, \tag{5.8}$$

where $[\boldsymbol{\phi}(\mathbf{p}_i)^T,\, \mathbf{h}_i^T,\, N_i]^T \in \mathbb{R}^{2\ell} \times\ \mathbb{Z}_{>0}$, $i \in \{1,\ldots,n\}$, is agent i's vector input, $\boldsymbol{\xi}_i$ is agent i's estimate of the average of all the agents' input, $\boldsymbol{\eta}_i$ is the internal estimator state, $\gamma > 0$

is a global *forgetting factor* governing the rate at which new information replaces old information in the dynamic averaging process, and $\sigma,\, \tau : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}_{\geq 0}$ are $\mathcal{C}^1$ bounded symmetric gain functions (i.e., $\sigma(\mathbf{p}_i,\, \mathbf{p}_j) = \sigma(\mathbf{p}_j,\, \mathbf{p}_i)$ and $\tau(\mathbf{p}_i,\, \mathbf{p}_j) = \tau(\mathbf{p}_j,\, \mathbf{p}_i)$, $\forall\ \mathbf{p}_i,\, \mathbf{p}_j$ with $i \neq j$), such that $\sigma(\mathbf{p}_i, \mathbf{p}_j)$ and $\tau(\mathbf{p}_i, \mathbf{p}_j)$ are different from zero only if agents i and j can communicate with each other. We also suppose that $\tau(\cdot, \cdot)$ has bounded first-order partial derivatives. Vector $\boldsymbol{\chi}_i \in \mathbb{R}^\ell$ is the *output* of the PI estimator and $\boldsymbol{\xi}_i[1 : \ell]$ denotes the vector consisting of the first ℓ components of $\boldsymbol{\xi}_i$. Note that the last entry of the input vector of the PI estimator is necessary for a correct estimation of $\mathbf{f}^\star_{\text{env}}$: in fact the *overall* number of particles N is unknown to the agents (in fact, agent i has only knowledge of the number N_i of particles that it processes).

We henceforth assume that each agent is able to measure its pose $[\mathbf{p}_i^T,\, \theta_i]^T$, and that agents i and j can communicate with each other if and only if $\|\mathbf{p}_i - \mathbf{p}_j\| \leq R$, where $R > 0$ represents a fixed communication radius (agent j is then said a *neighbor* of agent i). Each configuration $\mathbf{p} \in (\mathbb{R}^2)^n$ then defines the graph of an underlying communication network and we will use $\mathfrak{C} \subset (\mathbb{R}^2)^n$ to denote the set of all such configurations for which this graph is connected. As the agents move with time, the topology of the network can change, but we will assume that $\mathbf{p}(t) \in \mathfrak{C}$, i.e., that the network remains connected in forward time. Each agent transmits its estimate $\boldsymbol{\xi}_i$ and its internal estimator state $\boldsymbol{\eta}_i$ to its neighbors in the network. Each $\boldsymbol{\xi}_i$ will approximately track the true average of the inputs $[\boldsymbol{\phi}(\mathbf{p}_i)^T,\, \mathbf{h}_i^T,\, N_i]^T$, $i \in \{1, \ldots, n\}$. If the input and the topology of the (connected) network were *ideally constant*, each $\boldsymbol{\xi}_i$ would exactly converge to $\frac{1}{n}\sum_{i=1}^n [\boldsymbol{\phi}(\mathbf{p}_i)^T,\, \mathbf{h}_i^T,\, N_i]^T = [\mathbf{f}(\mathbf{p})^T,\, \frac{1}{n}\sum_{k=1}^N \boldsymbol{\phi}(\mathbf{q}_k)^T,\, N/n]^T$ and the output $\boldsymbol{\chi}_i$ of the PI estimator would converge exactly to $\mathbf{f}(\mathbf{p}) - \mathbf{f}^\star_{\text{env}}$. However, it has been shown in (Lynch et al. 2008, Th. 3) that the network of n PI estimators (5.7) is *input-to-state stable*: hence, even if the inputs and the topology of the network are arbitrary fast time-varying, a bound on the norm of the input implies a bound on the norm of the estimation error.

We define the *proportional Laplacian* $\mathbf{L}_P(\mathbf{p}) \in \mathbb{R}^{n \times n}$ to be the symmetric matrix whose off-diagonal elements in row i, column j are equal to $-\sigma(\mathbf{p}_i, \mathbf{p}_j)$ and whose diagonal elements are such that $\mathbf{L}_P(\mathbf{p})\mathbb{1} = \mathbf{0}_{n\times 1}$, where $\mathbb{1}$ denotes the vector of n ones and $\mathbf{0}_{n\times 1}$ the vector of n zeros. The *integral Laplacian* $\mathbf{L}_I(\mathbf{p}) \in \mathbb{R}^{n \times n}$ is defined in an analogous way, but using function $\tau(\cdot, \cdot)$ instead of $\sigma(\cdot, \cdot)$. Let $\text{Orth}(\mathbb{1})$ denote the collection of $n \times (n-1)$ matrices $\mathbf{S}$ such that $\mathbf{S}^T\mathbf{S} = \mathbf{I}_{n-1}$, where $\mathbf{I}_{n-1}$ is the $(n-1) \times (n-1)$ identity matrix, and $\mathbf{S}^T\mathbb{1} = \mathbf{0}_{n-1\times 1}$. Then, by orthogonal decomposition,

$$\mathbf{I}_n = \mathbf{S}\,\mathbf{S}^T + \frac{1}{n}\,\mathbb{1}\mathbb{1}^T, \tag{5.9}$$

and thus $\|\mathbf{AS}\|_F \leq \|\mathbf{A}\|_F$ for any n-column real matrix $\mathbf{A}$, where $\|\cdot\|_F$ denotes the Frobenius norm. Fixing some $\mathbf{S} \in \text{Orth}(\mathbb{1})$, we finally define the *reduced proportional* and *integral Laplacians* to be the $(n-1) \times (n-1)$ symmetric matrices $\mathbf{L}^\star_P(\mathbf{p}) \triangleq \mathbf{S}^T\mathbf{L}_P(\mathbf{p})\mathbf{S}$ and

$\mathbf{L}_I^\star(\mathbf{p}) \triangleq \mathbf{S}^T\mathbf{L}_I(\mathbf{p})\mathbf{S}$, respectively. We finally assume that there exist constants $\varrho > -\gamma$ and $\vartheta > 0$, such that,

$$\varrho\,\mathbf{I}_{n-1} \leq \mathbf{L}_P^\star(\mathbf{p}) \leq \bar{\varrho}\,\mathbf{I}_{n-1}, \quad \vartheta\,\mathbf{I}_{n-1} \leq \mathbf{L}_I^\star(\mathbf{p}) \leq \bar{\vartheta}\,\mathbf{I}_{n-1}, \tag{5.10}$$

along trajectories in forward time (this implies a connected network $\mathbf{p}(t) \in \mathfrak{C}$). The constants $\bar{\varrho}$, $\bar{\vartheta} > 0$ represent upper bounds on the reduced Laplacians, which exist since the functions $\sigma(\cdot,\cdot)$ and $\tau(\cdot,\cdot)$ are bounded.

5.1.4 Closed-loop stability analysis

In this section we study the closed-loop behavior of the system arising from the combination of the gradient-descent controllers (5.5) and the PI estimators (5.7)-(5.8). Following (Yang et al. 2008), we will assume that the maximum diameter of a connected swarm of n agents,

$$\mathfrak{d}(n) \triangleq \sup_{\mathbf{p}\,\in\,\mathfrak{C}\,\cap\,(\mathbb{R}^2)^n} \max_{i,j\,\in\,\{1,\ldots,n\}} \|\mathbf{p}_i - \mathbf{p}_j\|, \tag{5.11}$$

is finite for every n. It follows from (5.11) that there exists a class-$\mathcal{K}$ function $\mathfrak{a}$ and a C^1 function $\varpi : \mathbb{R}^2 \to \mathbb{R}_{\geq 0}$ such that

$$\|[\boldsymbol{\phi}(\mathbf{p}_i)^T, \mathbf{h}_i^T, N_i]^T - [\boldsymbol{\phi}(\mathbf{p}_j)^T, \mathbf{h}_j^T, N_j]^T\|^2 \leq \mathfrak{a}(\mathfrak{d}(n))\varpi(\mathbf{p}_i), \tag{5.12}$$

for every $\mathbf{p} \in \mathfrak{C}$ and every $i, j \in \{1,\ldots,n\}$. Let us also assume that the following inequality holds,

$$\lambda_{\max}(\text{blkdiag}(\boldsymbol{\Gamma}, \mathbf{B})) < 2\,\delta_1, \tag{5.13}$$

where $\lambda_{\max}(\text{blkdiag}(\boldsymbol{\Gamma}, \mathbf{B}))$ denotes the largest eigenvalue of the block diagonal matrix $\text{blkdiag}(\boldsymbol{\Gamma}, \mathbf{B})$, and $\mathbf{B}$ is a certain $(\ell+1)\times(\ell+1)$ invertible symmetric matrix (see the proof of Theorem 5.2 for more details). $\delta_1 > 0$ is a scalar constant depending on n, ϱ, $\bar{\varrho}$, ϑ, $\bar{\vartheta}$, γ and the bounds on the partial derivatives of $\tau(\cdot,\cdot)$ (the exact dependencies are given in the proof). Note that inequality (5.13) represents a *small-gain condition*: in fact, for given estimator gains which determine δ_1, the control gain $\boldsymbol{\Gamma}$ (as well as the matrix $\mathbf{B}$) should be sufficiently small. Given a closed set of swarm configurations $\mathcal{P} \subset (\mathbb{R}^2)^n$ and $\mathbf{f}_{\text{env}}^\star(t) \in \mathbf{f}(\mathcal{P})$, $\forall\, t \geq 0$, let $\bigcap_{t\geq 0} \mathcal{G}(\mathbf{f}_{\text{env}}^\star(t), \mathcal{P})$ denote the convex cone of all symmetric positive-definite matrices $\boldsymbol{\Gamma}$ such that no bad critical points of the potential function,

$$\Pi_{\text{env}}(\mathbf{p}) \triangleq (\mathbf{f}(\mathbf{p}) - \mathbf{f}_{\text{env}}^\star)^T\,\boldsymbol{\Gamma}\,(\mathbf{f}(\mathbf{p}) - \mathbf{f}_{\text{env}}^\star),$$

in $\mathcal{P}$ are local minima of Π_{env}.

5.2. THEOREM (CLOSED-LOOP STABILITY). *Let us suppose there exists a closed set $\mathcal{P} \subset (\mathbb{R}^2)^n$ such that $\boldsymbol{\Gamma} \in \bigcap_{t\geq 0} \mathcal{G}(\mathbf{f}_{env}^\star(t), \mathcal{P})$ and $\mathbf{p}(t) \in \mathcal{P}$, $\forall\, t \geq 0$. Suppose that $n \geq 3$ is fixed, that (5.10)*

holds for some $\varrho > -\gamma$ and $\vartheta > 0$ (with $\gamma > 0$), and that (5.13) is satisfied. Let the control input of agent i be of the form (5.5), with $\alpha_i(t) \triangleq proj(\arg(\mathbf{g}_i(t)) - \theta_i(t))$ and

$$\mathbf{g}_i(t) = -(\mathcal{J}\boldsymbol{\phi}(\mathbf{p}_i(t)))^T \, \boldsymbol{\Gamma} \, \boldsymbol{\chi}_i(t). \tag{5.14}$$

Finally, let us suppose that the evolution of the N particles is governed by equation (5.6) and that $\|\boldsymbol{\Upsilon}_k(\mathbf{q}, t)\|$, $\forall\, k \in \{1, \ldots, N\}$, is sufficiently smaller than v. Then, for almost every initial configuration of the agents, each trajectory of the system (5.1), (5.5), (5.7)-(5.8), (5.14) is bounded in forward time. Moreover, for any $\epsilon > 0$, there exists a sufficiently large gain ρ on the angular control such that the error on the desired swarm configuration is uniformly ultimately bounded with an ultimate bound ϵ.

Proof : *See (Morbidi, Freeman and Lynch 2011).* ■

5.1.5 Numerical experiments

Numerical simulations have been conducted to illustrate the proposed theory. Fig. 5.4 shows the closed-loop behavior of the gradient-descent controllers and PI estimators for a swarm of $n = 4$ agents. The plots were generated using $v = 1$ m/s, $\rho = 3$ and $\boldsymbol{\Gamma} = \text{diag}(100, 100, 0.1, 0.1, 0.1)$. As far as the PI estimators are concerned, we set $\gamma = 7$ and chose the gain functions according to an equal weighting scheme with a communication radius $R = 27$ m: $\sigma(\mathbf{p}_i, \mathbf{p}_j) = 25$ and $\tau(\mathbf{p}_i, \mathbf{p}_j) = 0.8$ when $\|\mathbf{p}_i - \mathbf{p}_j\| \leq R$ and $\sigma(\mathbf{p}_i, \mathbf{p}_j) = \tau(\mathbf{p}_i, \mathbf{p}_j) = 0$, otherwise. The PI estimators have been initialized with $\boldsymbol{\xi}_i(0) = [0,\ 0,\ 80,\ 80, 0, 0, 0, 0, 0, 0, 50]^T$ and $\boldsymbol{\eta}_i(0) = \mathbf{0}$, for all i. The particles $\mathbf{q}_k$, $k \in \{1, \ldots, N\}$, $N = 200$, have been drawn from a bivariate normal distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with mean $\boldsymbol{\mu} = [10,\ 5]^T$ and variance $\boldsymbol{\Sigma} = \left[\begin{smallmatrix} 70 & 1 \\ 1 & 70 \end{smallmatrix}\right]$, and lie within a rectangular domain $\mathcal{Q}$ with vertices $(-30,\ -22)$, $(32,\ -22)$, $(32,\ 30)$, $(-30,\ 30)$.

Fig. 5.4(a) shows the cluster of particles and the corresponding ellipse of desired geometric moments of the swarm. Fig. 5.4(b) reports the initial random pose of the four agents and the corresponding ellipse of geometric moments (red). The initial Voronoi partition $\{V_1,\ V_2,\ V_3, V_4\}$ of $\mathcal{Q}$ and the graph of the underlying communication network are represented with solid and dashed lines in the figure. Fig. 5.4(c) shows the trajectory of the four agents and the ellipse of geometric moments of the swarm at the final time instant: note that the red and black ellipses are almost exactly superimposed and the agents rotate around four points corresponding to a minimum of $\Pi_{\text{env}}(\mathbf{p})$. Finally, Fig. 5.4(d) shows the time history of $\log_{10} \Pi_{\text{env}}(\mathbf{p}(t))$, and Figs. 5.4(e) and 5.4(f) the time evolution of $\mathbf{f}^{\star}_{\text{env}}(t)$ (dashed) and $\mathbf{f}(\mathbf{p}(t))$ (solid), respectively: the symbols CMx, CMy and Ixx, Iyy, Ixy refer to the first- and second-order moment statistics, respectively.

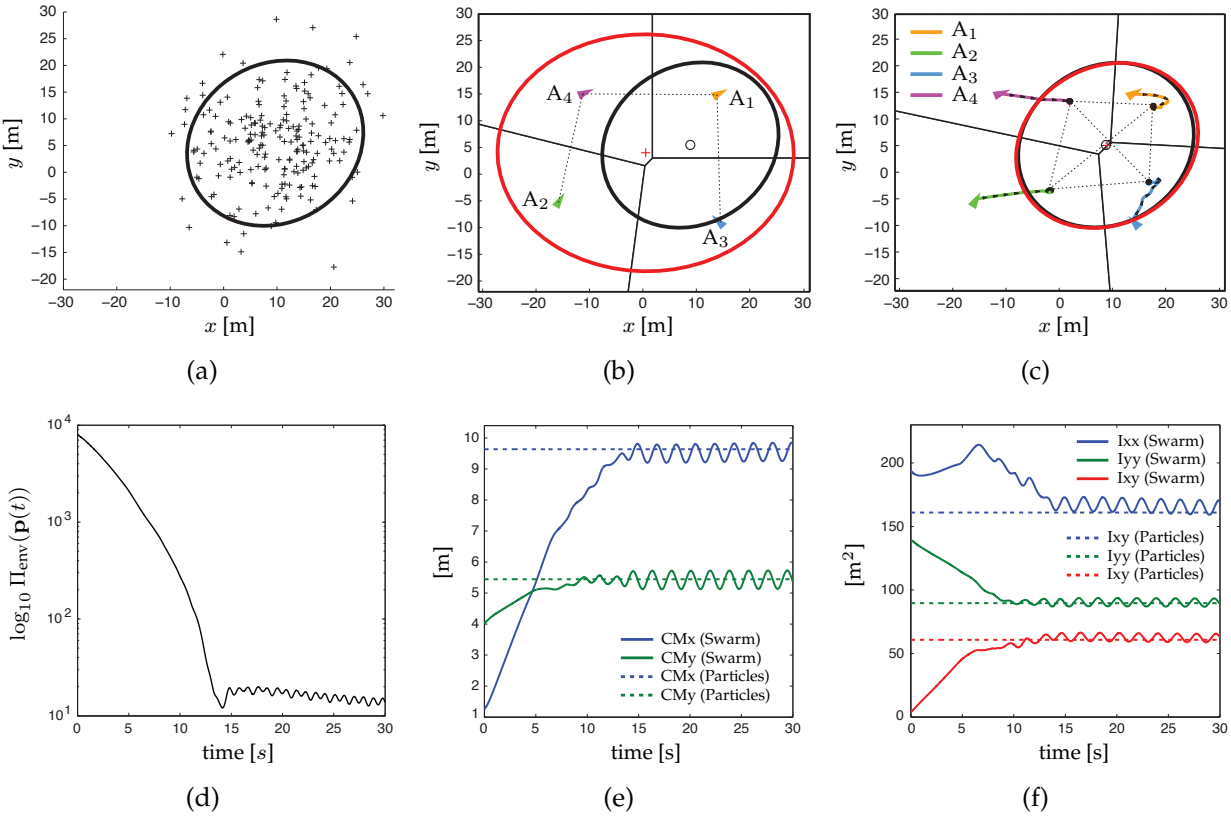


Figure 5.4: Simulation results*: (a) Cluster of $N = 200$ particles and corresponding ellipse of desired geometric moments of the swarm (black); (b) Initial pose of the four agents and corresponding ellipse of geometric moments (red). The initial Voronoi partition of $\mathcal{Q}$ and the graph of underlying communication network are represented with solid and dashed lines; (c) Trajectory of the four agents and ellipse of geometric moments at the final time instant (red); (d) Time history of* $\log_{10} \Pi_{env}(\mathbf{p}(t))$*; (e) First-order moments and (f) second-order moments of* $\mathbf{f}^{\star}_{env}(t)$ *(dashed, "Particles") and* $\mathbf{f}(\mathbf{p}(t))$ *(solid, "Swarm").*

5.1.6 Conclusions and future work

In this section we have proposed a new estimation and control strategy for distributed monitoring tasks. The geometric moments of a team of UAVs modeled as constant-speed unicycles are controlled via a nonlinear gradient descent to match those of an ensemble of discrete particles describing the occurrence of some event of interest to be monitored. A PI average consensus estimator is run by each agent to locally estimate the desired moments of the swarm from the environmental data. The closed-loop stability of the system has been studied and simulation results have been presented to support the theoretical analysis.

In future investigations, we would like to extend our strategy to $\mathrm{SE}(3)$ and to vehicles with *non-constant* positive forward velocity. we also aim to use second-order *central moments* in order to have a translation-invariant description of the desired swarm configuration, and we plan to test our estimation and control algorithm on field data (e.g., on recorded trajectories of marine oil spills).

5.2 Cooperative active target tracking

In this section, we study the active target-tracking problem for a team of double-integrator aerial vehicles equipped with 3-D range-finding sensors. A gradient-based control strategy that encompasses the three major optimum experimental design criteria is proposed, and a Kalman filter is used for estimating the target's position in a cooperative setting. The control strategy is *active* because it moves the vehicles along paths that minimize the uncertainty about the location of the target. Simulation experiments illustrate the proposed theory.

5.2.1 Introduction

Problem statement and related work

We are interested here in the optimal deployment of a team of *unmanned aerial vehicles* in the 3-D space, in order to maximize the accuracy of the position estimate of one or multiple moving targets (*active target-tracking problem*). This problem has relevant applications in surveillance, military and environmental monitoring tasks, and offers different levels of complexity depending on whether the position of the agents is perfectly known or needs to be estimated together with that of the targets (*localization problem*).

An essential capability that each autonomous robot should possess is that of being able to efficiently measure the surrounding environment and to promptly respond to stimuli coming from other robots, humans, or moving targets. Extensive research has been done in the literature on the subject of tracking targets with *static sensors* (Bar-Shalom et al. 2001, Taylor et al. 2006), or on the optimal placement of fixed sensors (Uciński 2005). *Mobile sensor networks* are known to offer distinctive advantages over static sensor arrays in terms of quality of sensing and estimation, area coverage, adaptability to changing conditions (in the environment as well as in the target's behavior), and robustness against failures (Robin and Lacroix 2016). *Cooperative active sensing* leverages the mobility of a robotic sensor network in order to enhance the target tracking performances (Chung et al. 2006, Yang et al. 2007, Olfati-Saber 2007, Olfati-Saber and Jalalkamali 2012), (Zhou and Roumeliotis 2008, Zhou and Roumeliotis 2011, Stroupe and Balch 2005, Martínez and Bullo 2006, Zhao et al. 2002, Singh et al. 2009), (Grocholsky et al. 2006, Stump et al. 2009): in the simplest instance of this problem, n sensors have to fuse their local measurements and move in order to attain the best position estimate of a moving target. This mechanism is sometimes referred to as "information-driven mobility" in the literature, and it amounts, in practice, to minimize a certain scalar function of the covariance matrix of the position estimates: in *optimum experimental design theory* (Walter and Pronzato 1994, Uciński 2005), this function is typically the determi-

nant or the trace of the covariance matrix (D- and A-optimality criteria). In the cooperative active target-tracking literature, a large body of research has focused on agents equipped with sensors providing *range-only* or *range-bearing* measurements to the target, while relatively fewer works have dealt with the more challenging bearing-only case (cf. (Passerieux and Cappel 1998)). Moreover, most of the existing papers have limited themselves to *single targets* moving in a *2-D environment*.

In (Chung et al. 2006), a motion-planning algorithm has been presented for solving the cooperative 2-D target-tracking problem using range-bearing measurements. The control law proposed by the authors is based on the gradient of the determinant of the covariance matrix of the target's position estimate with respect to each of the robot's coordinates. The multiple-target case is briefly treated as a possible extension, and the role of imperfect communication between agents is also investigated, thus elucidating the trade-offs in performance between sensing and communication. The results in (Chung et al. 2006) have been extended in two directions in (Yang et al. 2007). First, by using dynamic average consensus estimators, the controller proposed in (Chung et al. 2006) can be implemented in a distributed fashion. Second, a new control design procedure based on the distributed Kalman filter (Olfati-Saber 2005) for estimation fusion is presented, and the case of range-only sensors is addressed. Pursuing this line of research, the same authors have lately proposed decentralized estimation algorithms alternative to the Kalman filter (H_∞ filters, set-valued estimators), and studied the impact of "intelligent" evasive targets on the tracking performances. Embracing an approach similar to (Yang et al. 2007), the collaborative active-sensing problem has been studied in (Olfati-Saber 2007) for a group of double-integrator *ground robots* equipped with range-bearing sensors. The network-connectivity issue is addressed using a flocking-based mobility model (which also accounts for collision avoidance), and a modified version of the distributed Kalman filter in (Olfati-Saber 2005) for estimating the target's state is presented, along with information-driven decentralized control laws for the agents. More recently, in (Olfati-Saber and Jalalkamali 2012), a formal closed-loop stability analysis of an enhanced version of the coupled estimation-and-control algorithm described in (Olfati-Saber 2007), has been proposed by the same author. In (Zhou and Roumeliotis 2008), the authors focused on discrete-time target-tracking for a team of *unicycle robots* with bounds on the positive forward velocity, and shown that the associated optimization problem is NP-hard in general. Non-trivial relaxations to this problem are then proposed for determining the set of feasible locations that each robot should move to, in order to collect the most informative distance measurements. Note that differently from (Chung et al. 2006), where the nonlinearities inherent to the range-bearing model are "absorbed" into the non-stationary sensor noise, the approach in (Zhou and Roumeliotis 2008) takes the nonlinear nature of the distance-only measurements explicitly into account. In (Zhou and Roumeliotis 2011), the results

in (Zhou and Roumeliotis 2008) have been extended to the case of ground vehicles processing mixtures of relative measurements (i.e., range and/or bearing) and a closed-form global optimal solution for the active-sensing problem is derived in the case of a *single vehicle* for arbitrary target-motion models. In addition, constraints on the maximum speed of the robots and on the minimum distance between the agents and the target have been incorporated into the problem formulation. However, the mathematical tools used by the authors in (Zhou and Roumeliotis 2008, Zhou and Roumeliotis 2011), appear much more involved than those in (Chung et al. 2006), thus making their approach difficult to extend to different scenarios or to implement in a distributed fashion. Moreover, differently from (Chung et al. 2006, Yang et al. 2007, Olfati-Saber 2007), inter-agent *cooperation* does not seem fully exploited in (Zhou and Roumeliotis 2008, Zhou and Roumeliotis 2011). In (Stroupe and Balch 2005) an *approximate* tracking algorithm is proposed, where the agents try to minimize the target's location uncertainty using range-bearing sensors. The optimization problem (minimize the determinant of the target-position estimates' covariance matrix), is solved separately by each agent by a greedy search over the discretized set of candidate headings. In (Martínez and Bullo 2006) the authors have studied the optimal placement of range-only sensors for non-random *static target* position estimation. In this specific scenario, the determinant of the Fisher information matrix (the inverse of the covariance matrix) can be computed in closed-form and its critical points easily characterized. An optimal configuration, then turns out to be one in which the sensors are uniformly placed in a circular fashion around the target.

Related work on cooperative active target tracking has been done in (Triplett et al. 2009), where *constant-speed unicycle robots* are controlled using a distributed behavior-based approach, the communication takes place on a broadcast network and estimation is achieved by an unscented Kalman filter. However, differently from (Martínez and Bullo 2006, Zhou and Roumeliotis 2008) and (Zhou and Roumeliotis 2011), this paper does not provide an accurate theoretical characterization of the properties of the proposed coordination algorithm. Finally, in (Quintero et al. 2010), the authors have studied the optimal routing of two camera-equipped fixed-wing aircraft cruising at *fixed altitude*, cooperatively tracking a single ground target. A perspective transformation relating the image-plane measurements to the ground allows to derive the geolocation error covariance matrix, and dynamic programming is used by the authors to compute optimal coordinated control policies which minimize the fused target localization error covariance.

Original contributions

This work builds upon (Chung et al. 2006, Yang et al. 2007) and extend them in two new directions. First, by using a spherical coordinates representation, we present a gen-

eralization for a team of unmanned aerial vehicles modeled as double integrators and equipped with 3-D range-finding sensors. Second, we introduce a gradient-based control design procedure that encompasses the three major optimum experimental design criteria: a Kalman filter is used for the fusion process.

It is also worth emphasizing herein that we are not specifically concerned with the problem of designing *distributed control laws* for the aerial vehicles: in fact, one can straightforwardly move in this direction, by adapting the distributed strategies proposed in (Olfati-Saber 2007, Yang et al. 2007) to our specific setup. Moreover, in order to keep our presentation accessible and concise, we decided not to incorporate *collision-avoidance* or *maximum-speed constraints* in our problem formulation. These issues will be addressed in future, more experiment-oriented works.

The rest of this section is organized as follows. Sect. 5.2.2 presents the measurement model and estimation-and-control strategies for the aerial vehicles. In Sect. 5.2.3 the results of extensive simulation experiments are discussed, and in Sect. 5.2.4 the main contributions are summarized and possible avenues of future research are outlined.

5.2.2 Problem formulation and main results

Measurement model

Consider a team of n aerial vehicles (hereafter, also simply "agents") with positions $\mathbf{p}_1, \ldots, \mathbf{p}_n \in \mathbb{R}^3$ expressed with respect to a common global reference frame, and a target moving in 3-D according to the following model,

$$\dot{\mathbf{x}}(t) = \mathbf{F}\,\mathbf{x}(t) + \mathbf{G}\,\mathbf{u}_{\mathrm{T}}(t) + \mathbf{w}(t)\,, \tag{5.15}$$

where $\mathbf{x}(t) \in \mathbb{R}^3$ denotes the position of the target at time t, $\mathbf{u}_{\mathrm{T}}(t) \in \mathbb{R}^3$ is an exogenous input and $\mathbf{w}(t) \in \mathbb{R}^3$ is a continuous-time white Gaussian noise with zero mean and covariance matrix $\mathbf{Q} \in \mathbb{R}^{3\times 3}$. The observation $\mathbf{z}_i(t) \in \mathbb{R}^3$ of the target's position made by the i-th vehicle at time t, is assumed to be given by the following measurement model,

$$\mathbf{z}_i(t) = \mathbf{H}_i\,\mathbf{x}(t) + \mathbf{v}_i(t)\,,$$

where $\mathbf{v}_i(t)$ is a continuous-time zero-mean white Gaussian noise. In the following, we will assume that the measurement noises of the vehicles are independent and that the position $\mathbf{p}_i$ of vehicle i is perfectly known, e.g., from accurate GPS measurements. In addition, we will suppose that each agent is equipped with a magnetic compass (or an equivalent sensor), which provides its absolute orientation with respect to the fixed global reference frame: in this way, without loss of generality, we can assume that the reference frames attached to the vehicles are all aligned.

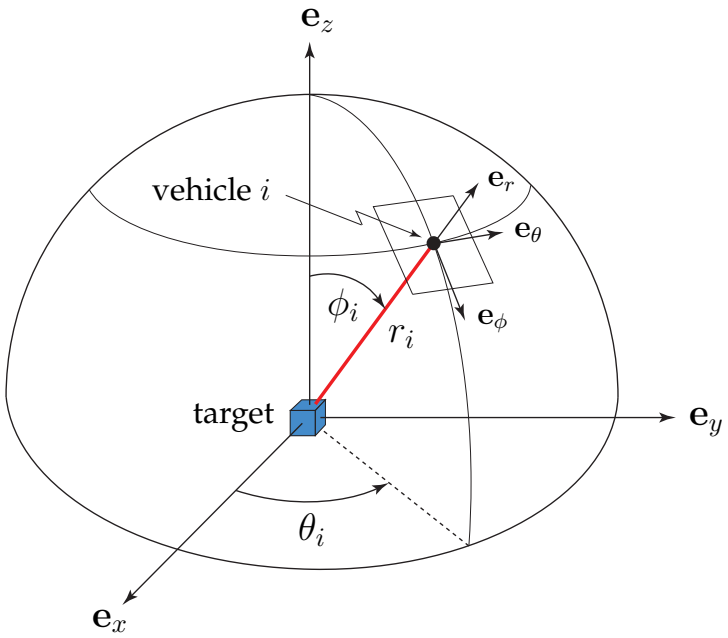


Figure 5.5: *Measurement model for the 3-D range-finding sensor of vehicle* i. *The spherical coordinates* $(r_i,\, \theta_i,\, \phi_i)$ *have been used, where* $r_i \in (0,\, +\infty)$ *is the* distance *from vehicle* i *to the target,* $\theta_i \in (0,\, 2\pi) \setminus \{\pi\}$ *is the* bearing angle *and* $\phi_i \in (0,\, \pi)$ *is the* polar angle. $\{\mathbf{e}_x,\, \mathbf{e}_y,\, \mathbf{e}_z\}$, $\{\mathbf{e}_\phi,\, \mathbf{e}_\theta,\, \mathbf{e}_r\}$ *represent the Cartesian and spherical orthonormal bases, respectively.*

By adopting a 3-D range-finding sensor model (Ramachandra 2000, Sect. 4.3.2), we have that $\mathbf{H}_i = \mathbf{I}_3$ and that the covariance matrix $\mathbf{R}_i^{\text{Car}}(t) \in \mathbb{R}^{3\times 3}$ of $\mathbf{v}_i(t) \in \mathbb{R}^3$ assumes the form,

$$\mathbf{R}_i^{\text{Car}}(t) \triangleq \mathbf{T}_i(t)\, \mathbf{R}_i(t)\, \mathbf{T}_i^T(t)\,, \tag{5.16}$$

where the rotation matrix $\mathbf{T}_i$ (dropping the time index t), is given by,

$$\mathbf{T}_i = \mathcal{R}_z(\theta_i)\mathcal{R}_y(\phi_i) = \begin{bmatrix} \cos\theta_i \cos\phi_i & -\sin\theta_i & \cos\theta_i \sin\phi_i \\ \sin\theta_i \cos\phi_i & \cos\theta_i & \sin\theta_i \sin\phi_i \\ -\sin\phi_i & 0 & \cos\phi_i \end{bmatrix}, \tag{5.17}$$

and $\mathcal{R}_z(\theta_i)$, $\mathcal{R}_y(\phi_i)$ denote the basic 3×3 rotation matrices about the z- and y-axes of an angle θ_i and ϕ_i, respectively (see Fig. 5.5). $\mathbf{R}_i$ in (5.16) is the covariance matrix of the measurement noise in the range-bearing-polar frame of vehicle i, and it has the following diagonal structure,

$$\mathbf{R}_i = \text{diag}\big(\sigma_{\phi_i}^2,\, \sigma_{\theta_i}^2,\, \sigma_{r_i}^2\big). \tag{5.18}$$

The variance of the range-measurement noise $\sigma_{r_i}^2$ is typically represented by a function $f_r(r_i)$ of the Euclidean distance $r_i \triangleq \|\mathbf{p}_i - \mathbf{x}\|_2$ from agent i to the target (Chung

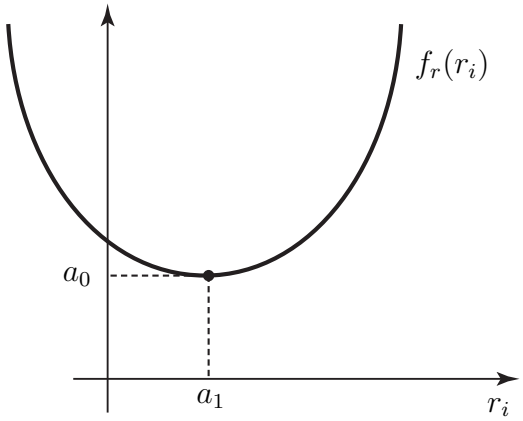


Figure 5.6: *The uncertainty in measurements is minimal at a distance a_1 from the target. The locus of points at distance a_1 from the target is called the "*sweet spot*", cf. (Chung et al. 2006).*

et al. 2006): the polar and bearing measurement-noise variances $\sigma^2_{\phi_i}$ and $\sigma^2_{\theta_i}$ can also be modeled as dependent on the range r_i through the functions $f_\phi(r_i)$ and $f_\theta(r_i)$, respectively. We will consider the following simple form for these functions,

$$\begin{aligned} f_r(r_i) &\triangleq a_2(r_i - a_1)^2 + a_0, \\ f_\theta(r_i) &\triangleq \alpha_\theta\, f_r(r_i), \\ f_\phi(r_i) &\triangleq \alpha_\phi\, f_r(r_i), \end{aligned}$$

where a_0, a_1, a_2, α_θ and α_ϕ are strictly positive parameters (however, note that our subsequent results can be straightforwardly extended to the case of general differentiable positive-valued convex functions $f_r(r_i)$, $f_\theta(r_i)$, $f_\phi(r_i)$). This model assumes the existence of a "*sweet spot*" located at a distance a_1 from the target, where uncertainty in measurements is minimal (see Fig. 5.6). Parameter a_1 depends on the physics of the sensing device, and it may vary from several tens of centimeters for infrared sensors, to few meters for time-of-flight sensors (e.g. ultrasonic sensors). Although environmental factors, such as temperature, humidity or background noise, might affect the sweet spot, a_1 is typically assumed to be a constant parameter in the literature. It is finally worth observing that a_1, together with the other parameters a_0, a_2, α_θ, α_ϕ, can be estimated from real data collected by the range-finding sensor, using standard least-squares methods (cf. (Gowal et al. 2011)).

5.3. REMARK. *Note that the transformation (5.16) from spherical coordinates, $\mathbf{R}_i$, to Cartesian coordinates*[1]*, $\mathbf{R}_i^{\text{Car}}$, is valid only for small θ_i and ϕ_i (cf. (Bar-Shalom et al. 2001, Sect. 10.4.3)).*

[1]The spherical coordinates $(r,\, \theta,\, \phi)$ are related to the Cartesian coordinates $(x,\, y,\, z)$ by the *nonlinear transformation* $r = \sqrt{x^2 + y^2 + z^2}$, $\theta = \text{atan2}(y, x)$, $\phi = \arccos(z/r)$, where $\text{atan2}(\cdot, \cdot)$ stands for the four-quadrant inverse tangent.

However, our simulation experiments (see Sect. 5.2.3), have shown that a violation of this "small-angle condition", does not lead to a significant performance degradation of our target-tracking algorithm. ◇

Cooperative control

The target-position measurements $\mathbf{z}_i$ and covariances $\mathbf{R}_i^{\text{Car}}$ from the n agents will be fused together to obtain a global target position estimate $\widehat{\mathbf{x}}_{\text{fus}}$ and global position-error covariance $\mathbf{P}_{\text{fus}}$. In this section, we will present a cooperative control strategy that moves the aerial vehicles in order to minimize a scalar function of the covariance $\mathbf{P}_{\text{fus}}$, thus reducing the target's localization error, cf. (Yang et al. 2007).

In order to show the generality of our approach, for the design of the gradient-based control laws we will simultaneously consider the cost functions for the three most popular optimum experimental design criteria (Uciński 2005, Sect. 2.3),

$$J = \ln \det(\mathbf{P}_{\text{fus}}), \tag{5.19}$$

for the D-*optimality* (determinant) criterion,

$$J = \text{tr}(\mathbf{P}_{\text{fus}}), \tag{5.20}$$

for the A-*optimality* (trace) criterion, and

$$J = \lambda_{\max}(\mathbf{P}_{\text{fus}}) = \|\mathbf{P}_{\text{fus}}\|_2\,, \tag{5.21}$$

for the E-*optimality* criterion, where $\lambda_{\max}(\mathbf{P}_{\text{fus}})$ denotes the maximum eigenvalue of $\mathbf{P}_{\text{fus}}$ and $\|\mathbf{P}_{\text{fus}}\|_2$ is the spectral norm of $\mathbf{P}_{\text{fus}}$. The D-optimum design (the most widely used criterion), minimizes the volume of the uncertainty ellipsoid for the estimates, the E-optimum design minimizes the length of the largest axis of the same ellipsoid, and the A-optimum design suppresses the average variance of the estimates. Note that although the cost functions (5.19)-(5.21) are convex with respect to $\mathbf{P}_{\text{fus}}^{-1}$ (assuming that $\text{tr}(\mathbf{P}_{\text{fus}}) = \infty$ and $\lambda_{\max}(\mathbf{P}_{\text{fus}}) = \infty$, if $\det(\mathbf{P}_{\text{fus}}^{-1}) = 0$, cf. (Uciński 2005, Sect B.5)), they are *not* convex with respect to the relative position and orientation of the vehicles and target, and they are thus prone to local minima (Chung et al. 2004).

Since the cost function (5.21) is *not differentiable* in general (in fact, the gradient of $\lambda_{\max}(\mathbf{P}_{\text{fus}})$ does not exist when the maximum eigenvalue of $\mathbf{P}_{\text{fus}}$ is not simple), in the following we will consider its *smoothed version* (Chen et al. 2004):

$$\Phi_\varepsilon(\mathbf{P}_{\text{fus}}) \;=\; \varepsilon \ln \Big(\sum_{i=1}^{3} \exp\big(\lambda_i(\mathbf{P}_{\text{fus}})/\varepsilon\big)\Big),$$

where $\varepsilon > 0$ is an assigned smoothing parameter and $\lambda_i(\mathbf{P}_{\text{fus}})$ denotes the i-th eigenvalue of $\mathbf{P}_{\text{fus}}$. Note that function $\Phi_\varepsilon(\mathbf{P}_{\text{fus}})$ is of class $\boldsymbol{\mathcal{C}}^\infty$ and it possesses the following uniform approximation property to $\lambda_{\max}(\mathbf{P}_{\text{fus}})$,

$$0 \leq \Phi_\varepsilon(\mathbf{P}_{\text{fus}}) - \lambda_{\max}(\mathbf{P}_{\text{fus}}) \leq \varepsilon \ln(3), \quad \forall\, \varepsilon > 0,$$

from which it follows that $\lim_{\varepsilon \downarrow 0} \Phi_\varepsilon(\mathbf{P}_{\text{fus}}) = \lambda_{\max}(\mathbf{P}_{\text{fus}})$. As indicated in (Polak 1997, p. 249), it turns out that when $\varepsilon \in [10^{-6}, 10^{-4}]$, $\Phi_\varepsilon(\mathbf{P}_{\text{fus}})$ yields an excellent approximation to $\lambda_{\max}(\mathbf{P}_{\text{fus}})$. The following *double-integrator model* will be used for the aerial vehicles (cf. Sect. 3.3.1),

$$\dot{\mathbf{p}}_i = \mathbf{q}_i, \quad \dot{\mathbf{q}}_i = \mathbf{u}_i, \quad i \in \{1, \ldots, n\}, \tag{5.22}$$

where $\mathbf{q}_i \in \mathbb{R}^3$ denotes the velocity of vehicle i and $\mathbf{u}_i \in \mathbb{R}^3$ its control input. The gradient controller of vehicle i will be then of the form,

$$\mathbf{u}_i = -\mathbf{B}\,\mathbf{q}_i - \boldsymbol{\Gamma}\,\mathbf{T}_i \left[\frac{1}{r_i \sin\theta_i} \frac{\partial J}{\partial \phi_i}, \; \frac{1}{r_i} \frac{\partial J}{\partial \theta_i}, \; \frac{\partial J}{\partial r_i} \right]^T, \tag{5.23}$$

where $\mathbf{B} \in \mathbb{R}^{3\times 3}$, $\mathbf{B} \succ \mathbf{0}$, is a damping matrix and $\boldsymbol{\Gamma} \in \mathbb{R}^{3\times 3}$, $\boldsymbol{\Gamma} \succ \mathbf{0}$, is a gain matrix[2]. In order to simplify the forthcoming derivations, let us introduce the following 3×3 matrices, whose explicit expression will be given in Sect 5.2.2,

$$\mathbf{P}_i^r \triangleq \frac{\partial \mathbf{P}_{\text{fus}}}{\partial r_i}, \quad \mathbf{P}_i^\theta \triangleq \frac{\partial \mathbf{P}_{\text{fus}}}{\partial \theta_i}, \quad \mathbf{P}_i^\phi \triangleq \frac{\partial \mathbf{P}_{\text{fus}}}{\partial \phi_i}. \tag{5.24}$$

By using (Uciński 2005, Th. B.17), we can explicitly compute the partial derivatives appearing on the right-hand side of (5.23) for the D-optimality criterion,

$$\frac{\partial J}{\partial r_i} = \operatorname{tr}(\mathbf{P}_{\text{fus}}^{-1}\mathbf{P}_i^r), \quad \frac{\partial J}{\partial \theta_i} = \operatorname{tr}(\mathbf{P}_{\text{fus}}^{-1}\mathbf{P}_i^\theta), \quad \frac{\partial J}{\partial \phi_i} = \operatorname{tr}(\mathbf{P}_{\text{fus}}^{-1}\mathbf{P}_i^\phi),$$

for the A-optimality criterion,

$$\frac{\partial J}{\partial r_i} = \operatorname{tr}(\mathbf{P}_i^r), \quad \frac{\partial J}{\partial \theta_i} = \operatorname{tr}(\mathbf{P}_i^\theta), \quad \frac{\partial J}{\partial \phi_i} = \operatorname{tr}(\mathbf{P}_i^\phi),$$

and for the E-optimality criterion,

$$\begin{aligned}
\frac{\partial J}{\partial r_i} &= \operatorname{tr}\big(\mathbf{U}^T \operatorname{diag}(\nabla_{\boldsymbol{\lambda}}\, \Phi_\varepsilon(\mathbf{P}_{\text{fus}}))\, \mathbf{U}\, \mathbf{P}_i^r\big), \\
\frac{\partial J}{\partial \theta_i} &= \operatorname{tr}\big(\mathbf{U}^T \operatorname{diag}(\nabla_{\boldsymbol{\lambda}}\, \Phi_\varepsilon(\mathbf{P}_{\text{fus}}))\, \mathbf{U}\, \mathbf{P}_i^\theta\big), \\
\frac{\partial J}{\partial \phi_i} &= \operatorname{tr}\big(\mathbf{U}^T \operatorname{diag}(\nabla_{\boldsymbol{\lambda}}\, \Phi_\varepsilon(\mathbf{P}_{\text{fus}}))\, \mathbf{U}\, \mathbf{P}_i^\phi\big).
\end{aligned} \tag{5.25}$$

[2]The symbols "$\succ$", "$\succeq$", denote the matrix inequality in the positive definite and positive semidefinite sense, respectively.

The l-th component of the gradient vector $\nabla_{\boldsymbol{\lambda}}\,\Phi_\varepsilon(\mathbf{P}_{\text{fus}})$ in (5.25) is,

$$[\nabla_{\boldsymbol{\lambda}}\,\Phi_\varepsilon(\mathbf{P}_{\text{fus}})]_l \;=\; \frac{\exp\bigl(\lambda_l(\mathbf{P}_{\text{fus}})/\varepsilon\bigr)}{\sum_{j=1}^{3}\exp\bigl(\lambda_j(\mathbf{P}_{\text{fus}})/\varepsilon\bigr)},\;\; l\in\{1,2,3\},$$

and the orthogonal matrix $\mathbf{U}$ satisfies $\mathbf{P}_{\text{fus}} = \mathbf{U}^T\text{diag}(\boldsymbol{\lambda}(\mathbf{P}_{\text{fus}}))\,\mathbf{U}$, where $\text{diag}(\boldsymbol{\lambda}(\mathbf{P}_{\text{fus}}))$ is a diagonal matrix with the elements of the vector $\boldsymbol{\lambda}(\mathbf{P}_{\text{fus}}) \triangleq [\lambda_1(\mathbf{P}_{\text{fus}}),\,\lambda_2(\mathbf{P}_{\text{fus}}),\,\lambda_3(\mathbf{P}_{\text{fus}})]^T$, $\lambda_1(\mathbf{P}_{\text{fus}}) \geq \lambda_2(\mathbf{P}_{\text{fus}}) \geq \lambda_3(\mathbf{P}_{\text{fus}})$, put on its main diagonal.

5.4. REMARK. *Note that the controller (5.23) is* centralized *since $\mathbf{P}_{fus}$, $\mathbf{P}_i^r$, $\mathbf{P}_i^\theta$ and $\mathbf{P}_i^\phi$ contain information from* all *the agents. However, the control law (5.23) can be implemented in a distributed fashion by replacing any unavailable global quantity with local estimates, as done, e.g., in (Yang et al. 2007, Olfati-Saber 2007).* ◇

Kalman-filter fusion

As anticipated, we describe here a method inspired by (Yang et al. 2007), to fuse the local target position measurements and error covariance matrices: in this way, we can determine the three matrices in (5.24), necessary for the implementation of controller (5.23). The method defines $\widehat{\mathbf{x}}_{\text{fus}}$ and $\mathbf{P}_{\text{fus}}$ by means of a Kalman-Bucy filter (Stengel 1994, Sect. 4.5):

$$\dot{\mathbf{P}}_{\text{fus}} = \mathbf{F}\,\mathbf{P}_{\text{fus}} + \mathbf{P}_{\text{fus}}\,\mathbf{F}^T + \mathbf{Q} - \mathbf{P}_{\text{fus}}\,\mathbf{C}\,\mathbf{P}_{\text{fus}}, \tag{5.26}$$

$$\dot{\widehat{\mathbf{x}}}_{\text{fus}} = \mathbf{F}\,\widehat{\mathbf{x}}_{\text{fus}} + \mathbf{G}\,\mathbf{u}_{\text{T}} + \mathbf{P}_{\text{fus}}\,(\mathbf{y} - \mathbf{C}\,\widehat{\mathbf{x}}_{\text{fus}}), \tag{5.27}$$

where

$$\mathbf{C} \triangleq \sum_{i=1}^{n}(\mathbf{R}_i^{\text{Car}})^{-1}, \quad \mathbf{y} \triangleq \sum_{i=1}^{n}(\mathbf{R}_i^{\text{Car}})^{-1}\,\mathbf{z}_i,$$

and (5.26) and (5.27) are respectively initialized with,

$$\mathbf{P}_{\text{fus}}(t_0) \;=\; \left[\sum_{i=1}^{n}(\mathbf{R}_i^{\text{Car}}(t_0))^{-1}\right]^{-1}, \tag{5.28}$$

$$\widehat{\mathbf{x}}_{\text{fus}}(t_0) \;=\; \mathbf{P}_{\text{fus}}(t_0)\sum_{i=1}^{n}(\mathbf{R}_i^{\text{Car}}(t_0))^{-1}\,\mathbf{z}_i(t_0), \tag{5.29}$$

where t_0 denotes the initial time instant. Note that (5.29) is a weighted least-squares estimate for the position of the target, and that (5.28) is the covariance matrix of the weighted least-squares estimator. By taking the partial derivatives with respect to r_i, θ_i

and ϕ_i on both sides of the *Riccati differential equation* (RDE) (5.26), we get the following three *Lyapunov differential equations*,

$$\begin{aligned}
\dot{\mathbf{P}}_i^r &= \big(\mathbf{F} - \mathbf{P}_{\text{fus}}\,\mathbf{C}\big)\mathbf{P}_i^r + \mathbf{P}_i^r\big(\mathbf{F} - \mathbf{P}_{\text{fus}}\,\mathbf{C}\big)^T + 2\,a_2(r_i - a_1)\,\mathbf{P}_{\text{fus}}\,\mathbf{T}_i\,\mathbf{R}_i^{-2}\,\text{diag}(\alpha_\phi,\,\alpha_\theta,\,1)\,\mathbf{T}_i^T\,\mathbf{P}_{\text{fus}},\\
\dot{\mathbf{P}}_i^\theta &= \big(\mathbf{F} - \mathbf{P}_{\text{fus}}\,\mathbf{C}\big)\mathbf{P}_i^\theta + \mathbf{P}_i^\theta\big(\mathbf{F} - \mathbf{P}_{\text{fus}}\,\mathbf{C}\big)^T - \mathbf{P}_{\text{fus}}\big(\mathbf{A}_{\theta_i} + \mathbf{A}_{\theta_i}^T\big)\mathbf{P}_{\text{fus}}\,,\quad \mathbf{A}_{\theta_i} = \boldsymbol{\Theta}\,\mathbf{T}_i\,\mathbf{R}_i^{-1}\,\mathbf{T}_i^T,\\
\dot{\mathbf{P}}_i^\phi &= \big(\mathbf{F} - \mathbf{P}_{\text{fus}}\,\mathbf{C}\big)\mathbf{P}_i^\phi + \mathbf{P}_i^\phi\big(\mathbf{F} - \mathbf{P}_{\text{fus}}\,\mathbf{C}\big)^T - \mathbf{P}_{\text{fus}}\big(\mathbf{A}_{\phi_i} + \mathbf{A}_{\phi_i}^T\big)\mathbf{P}_{\text{fus}}\,,\quad \mathbf{A}_{\phi_i} = \mathbf{T}_i\boldsymbol{\Upsilon}\,\mathbf{R}_i^{-1}\,\mathbf{T}_i^T,
\end{aligned} \tag{5.30}$$

where

$$\boldsymbol{\Theta} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad \boldsymbol{\Upsilon} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}.$$

The equations in (5.30) are respectively initialized with,

$$\begin{aligned}
\mathbf{P}_i^r(t_0) &= 2\,a_2(r_i(t_0) - a_1)\,\mathbf{P}_{\text{fus}}(t_0)\,\mathbf{T}_i(t_0)\,\mathbf{R}_i^{-2}(t_0)\,\text{diag}(\alpha_\phi, \alpha_\theta, 1)\,\mathbf{T}_i^T(t_0)\,\mathbf{P}_{\text{fus}}(t_0),\\
\mathbf{P}_i^\theta(t_0) &= -\,\mathbf{P}_{\text{fus}}(t_0)\big(\mathbf{A}_{\theta_i}(t_0) + \mathbf{A}_{\theta_i}^T(t_0)\big)\mathbf{P}_{\text{fus}}(t_0),\\
\mathbf{P}_i^\phi(t_0) &= -\,\mathbf{P}_{\text{fus}}(t_0)\big(\mathbf{A}_{\phi_i}(t_0) + \mathbf{A}_{\phi_i}^T(t_0)\big)\mathbf{P}_{\text{fus}}(t_0).
\end{aligned}$$

5.2.3 Numerical experiments

Simulation experiments have been performed to illustrate the theory presented in the previous section: in all our tests we used the cost function for the D-optimality criterion to design the gradient controller, because of its invariance under any nonsingular reparametrization (Walter and Pronzato 1994, Sect. 6.1). In the simulations, the initial velocity $\mathbf{q}_i(t_0) = \mathbf{q}_i(0)$ of the aerial vehicles is set to zero.

Fig. 5.7(a) shows the trajectory of 4 aerial vehicles cooperatively tracking a target whose motion is governed by system (5.15), with $\mathbf{F} = -5\,\text{diag}(10^{-3},\,10^{-4},\,10^{-2})$, $\mathbf{G} = \mathbf{I}_3$, $\mathbf{u}_{\text{T}}(t) = [0,\,-10^{-2},\,0]^T$ and $\mathbf{Q} = 0.07\,\mathbf{I}_3$. The initial positions of the agents and target are, in meters, $\mathbf{p}_1(0) = [-3,\,15,\,0]^T$, $\mathbf{p}_2(0) = [3,\,15,\,0]^T$, $\mathbf{p}_3(0) = [0,\,15,\,5]^T$, $\mathbf{p}_4(0) = [0,\,15,\,-5]^T$, $\mathbf{x}(0) = [0,\,12,\,0]^T$, and are respectively marked with circles and a square in Fig. 5.7(a). The parameters of the measurement model are $a_0 = a_2 = 0.2$, $a_1 = 3$, $\alpha_\theta = \alpha_\phi = 0.8$, and the gain and damping matrices are $\boldsymbol{\Gamma} = 0.5\,\mathbf{I}_3$ and $\mathbf{B} = 2\,\mathbf{I}_3$, respectively. Fig. 5.7(b) reports the time history of $J = \ln\det(\mathbf{P}_{\text{fus}})$ and Fig. 5.7(c) the target's estimation error $\mathbf{x} - \widehat{\mathbf{x}}_{\text{fus}}$. Fig. 5.7(d) displays the time evolution of r_i, $i \in \{1, \ldots, 4\}$ (colored): note that after about 150 seconds all the vehicles reach their sweet spot located at a distance $a_1 = 3\,\text{m}$ from the target (black, dashed line). It is also worth mentioning that although the "small-angle condition" is violated here (recall Remark 5.3), in

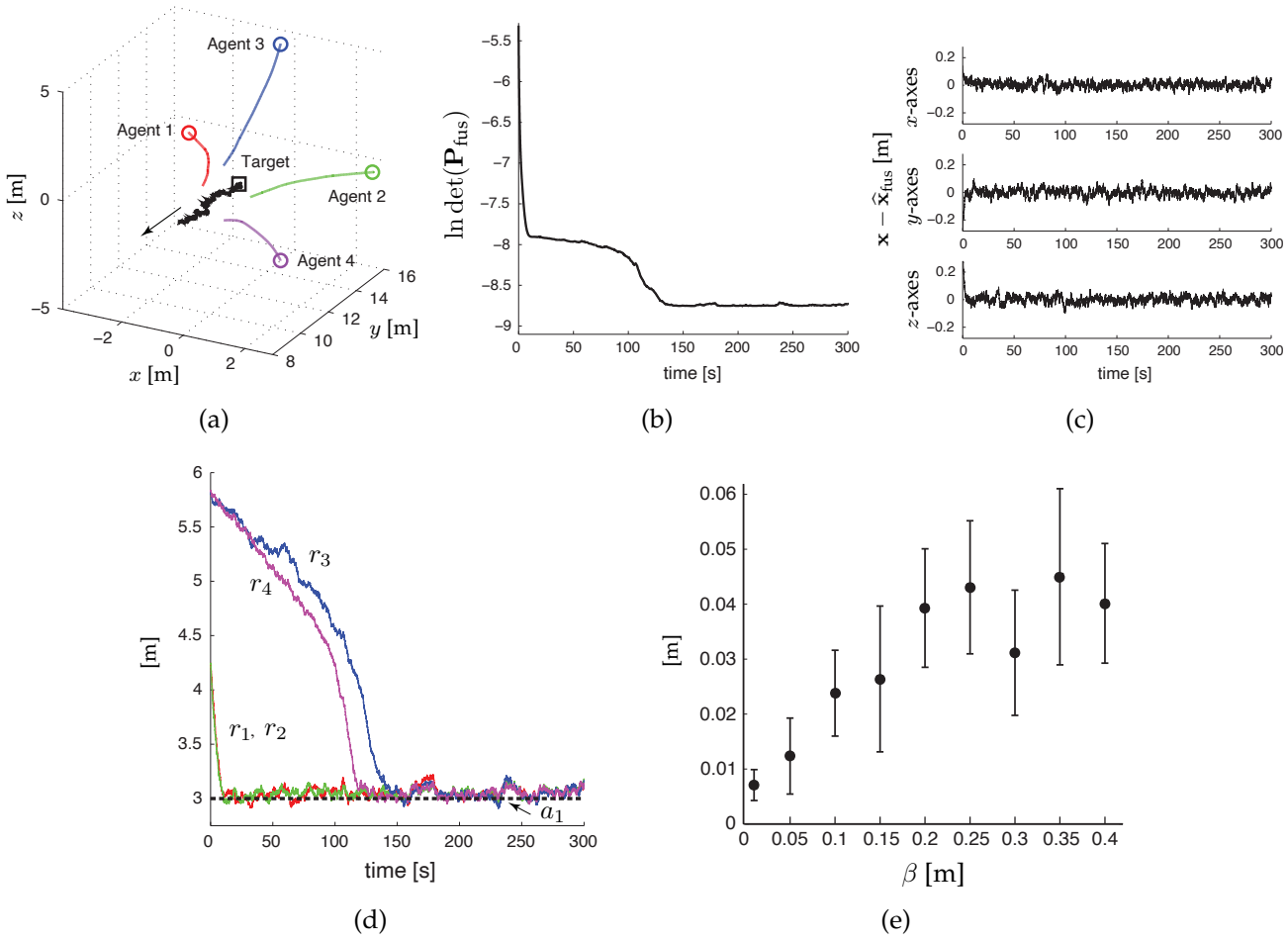


Figure 5.7: *(a) Trajectory of the 4 agents (colored) and of the target (black): the initial positions are marked with circles and a square, respectively, and the direction of motion of the target is indicated with an arrow; (b) Time history of the cost function* $J = \ln\det(\mathbf{P}_{\text{fus}})$*; (c) Time evolution of the estimation error* $\mathbf{x} - \widehat{\mathbf{x}}_{\text{fus}}$*; (d) Time history of* r_i*,* $i \in \{1, \ldots, 4\}$ *(colored) and* $a_1 = 3$ *m (black, dashed); (e) Mean and standard deviation of* $\|\mathbf{x}(T_{fin}) - \widehat{\mathbf{x}}_{\text{fus}}(T_{fin})\|_2$*,* $T_{fin} = 300$ *sec., over 10 trials for* $\mathbf{Q} = \beta^2\,\mathbf{I}_3$*,* $\beta \in \{0.01,\ 0.05,\ 0.1,\ 0.15,\ 0.2,\ \ldots,\ 0.4\}$*.*

this as well as in many other simulation experiments, we never experienced a significant performance degradation of our algorithm. Finally, Fig. 5.7(e) shows the influence of the process noise $\mathbf{w}(t)$ on the steady-state target's estimation error. In particular, in the figure we have reported the mean and standard deviation of $\|\mathbf{x}(T_{\text{fin}}) - \widehat{\mathbf{x}}_{\text{fus}}(T_{\text{fin}})\|_2$, ($T_{\text{fin}} = 300$ sec. is the final simulation time), over 10 trials for increasing process-noise power: $\mathbf{Q} = \beta^2\,\mathbf{I}_3$, $\beta \in \{0.01,\ 0.05,\ 0.1,\ 0.15,\ 0.2,\ \ldots,\ 0.4\}$. For consistency, in all the trials we used the same initialization for the Kalman filter: $\widehat{\mathbf{x}}_{\text{fus}}(0) = \mathbf{x}^T(0) + [1,\ -3,\ -2]^T$, $\mathbf{P}_{\text{fus}}(0) = \mathbf{I}_3$.

5.2.4 Conclusions and future work

In this section we have proposed an active target-tracking strategy to deploy a team of unmanned aerial vehicles along paths that minimize the uncertainty about the position of a moving target. The proposed theoretical results have been illustrated via numerical simulations.

There is a series of interesting open issues that have not been addressed in the present study. First, in future research, we are going to extend the proposed results to heterogeneous teams of ground and aerial robots (e.g., modeled as more realistic nonholonomic vehicles, cf. (Morbidi, Ray and Mariottini 2011)). We also like to study the impact of team's topology on the target estimation accuracy in the case of limited inter-vehicle communication (Mourikis and Roumeliotis 2006), and to test the proposed approach on real flying platforms (e.g. quadrotors). Finally, in future works we will investigate convex approximations to the cost functions (5.19)-(5.21), by exploiting, e.g., the concavity property of the Riccati differential equation of the Kalman-Bucy filter.

5.3 Cooperative 3D reconstruction

In this section, we are interested in the problem of surface reconstruction of large-scale unknown environments, with multiple cooperative robots. As they progress through the 3D environment, the robots rely on volumetric maps obtained via a TSDF representation to extract discrete Incomplete Surface Elements (ISEs), and a list of candidate viewpoints is generated to cover them. A Next-Best-View planning approach, which approximately solves a Traveling Salesman Problem (TSP) via greedy allocation, is then used to iteratively assign these viewpoints to the robots. Two multi-agent architectures, a centralized one (TSP-Greedy Allocation or TSGA) and a distributed one (dist-TSGA), in which the robots locally compute their maps and share them, are developed and compared. Extensive numerical and real-world experiments with multiple aerial and ground robots in challenging 3D environments, show the flexibility and effectiveness of our surface representation of a volumetric map. The experiments also shed light on the nexus between reconstruction accuracy and surface completeness, and between total distance traveled and execution time.

5.3.1 Introduction

In this section, we present a novel 3D reconstruction method for multiple cooperative robots that addresses the problem of surface inspection of unknown environments, via a Next-Best-View (NBV) frontier-based planner. A single objective function, which takes the *surface representation* explicitly into account, is used to plan collision-free paths for the robots within a *centralized* and a *distributed architecture*. Numerical simulations and real-world experiments show that the proposed architectures are robust against uncertain measurements, and provide accurate, complete and time-efficient 3D reconstructions.

We can summarize our original contributions, as follows:

- Differently from the conventional NBV exploration approaches, which select viewpoints among a large set of randomly-sampled poses (Yoder and Scherer 2016, Schmid et al. 2020), we directly consider a 3D representation of the incompleteness of the surface to generate a roadmap of viewpoints and to carry out the reconstruction with a team of mobile robots,

- We introduce multi-agent NBV planners to route robots to viewpoint configurations, and achieve surface reconstruction. More specifically, clusters of configurations are greedily allocated to the robots by successively (approximately) solving a Traveling Salesman Problem (TSP),

- To validate our approach, which admits both a centralized and a distributed implementation, we undertook a large campaign of numerical simulations with UAVs and real-world experiments with ground robots.

The rest of this section is organized as follows. Sect. 5.3.2 reviews and catalogues the related work, while Sect. 5.3.3 is devoted to the problem formulation. Sect. 5.3.4 provides a general overview of the proposed approach, and Sect. 5.3.5 and Sect. 5.3.6 deal with the perception and planning problems, respectively. The centralized and distributed multi-robot architectures are tested via numerical and hardware experiments in Sect. 5.3.7 and Sect. 5.3.8, respectively. Finally, Sect. 5.3.9 concludes the section and discusses possible areas of improvement.

5.3.2 Related work

Planning informative paths for real-time 3D modeling of unknown environments has been the subject of intense research in recent years. Volumetric-mapping methods represent the free and occupied space, and provide accurate surface estimations for reconstruction, both for single and multiple robots. In tandem with these emerging mapping methods, volume exploration planners have been developed to rapidly explore unknown volumes and provide coarse 3D reconstructions. The main focus of early work on surface-based reconstruction has been on accuracy, and the optimized cost function incorporates one or multiple surface criteria. Recently, the volumetric and surface-based approaches have been combined to take advantage of their unique properties, but only the single-robot case has been studied.

Volumetric mapping

Volumetric mapping consists in discretizing the 3D space into small cells: the cells can be unknown, empty or occupied, and they can be used to represent a 3D scene of interest. A popular online 3D modeling approach, *OctoMap*, was introduced in (Hornung et al. 2013). It relies on a 3D occupancy grid with an internal octree data structure. This representation adapts the level of detail of the map to the environment, which reduces memory usage, enabling real-time processing on CPU. In (Newcombe et al. 2011), the authors proposed *KinectFusion*, which uses measurements from an RGB-D camera to calculate the Truncated Signed Distance Function (TSDF) (Curless and Levoy 1996) over a grid, leading to an implicit surface representation. Successive improvements of the method resulted in reduced memory usage (Nießner et al. 2013), and recently the open source library CHISEL (Klingensmith et al. 2015) made it available for 3D reconstruction on mobile devices. In (Oleynikova et al. 2017), this mapping method has been revisited by considering Euclidean Signed Distance Fields (ESDF), which improves the ac-

curacy of the distance map. The concept of manifold mapping (Howard 2004) has been incorporated within a TSDF framework, to mitigate the mapping error due to localization drift (Millane et al. 2018), and the traditional monolithic map has been replaced by a collection of multiple local sub-maps (or patches). In (Duhautbout et al. 2019), the authors have extended this concept to distributed mapping, for multiple ground robots.

Single-robot planning

Given a volumetric map, a robot can try to explore an unknown volume containing objects of interest. *Volumetric-exploration methods* usually leverage a volumetric representation (e.g. an OctoMap), to identify known, unknown, or occupied areas. *Sampling-based planners*, such as Rapidly-exploring Random Trees, RRT (LaValle and Kuffner Jr 2001), RRT* (Karaman and Frazzoli 2011), are used for trajectory generation by incrementally expanding a tree constituted of randomly-sampled sensor poses in the free space. On the other hand, Probabilistic Roadmap planners, PRM (Kavraki et al. 1996), Lazy PRM* (Hauser 2015), extract the trajectory/path from a graph formed by randomly-sampled poses between a start and a goal configuration, according to a given objective function. In an NBV-planning framework, volumetric exploration methods are usually related to *informative path planning*, which consists in expanding trees and in selecting the NBV trajectory that guarantees the maximum coverage of the volume, see (Bircher et al. 2016, Papachristos et al. 2019, Selin et al. 2019, Batinovic et al. 2021, Respall et al. 2021) for RRT-based methods, and (Xu et al. 2021) for a PRM-based method. Pose coverage along the trajectory is evaluated via ray tracing (Bresenham 1965), and as the robot follows the assigned path, the volume is automatically explored. These fast and efficient methods rely on a coarse volumetric map of the environment for navigation purposes, but they do not explicitly account for the completeness and accuracy of the reconstructed surface.

Surface-inspection methods use the current surface to generate a roadmap of candidate viewpoints which ensure a complete and accurate 3D reconstruction. NBV methods determine the next best viewpoints to visit, depending on the mission of the robots. Among them, *frontier-based methods* yield viewpoint configurations pointing towards the frontier of the known surface, represented as a mesh, according to a given orientation, position, or sensing constraint. By visiting these configurations, the robot-sensor gathers new surface information with some overlap, for ensuring continuity (Connolly 1985, Vasquez-Gomez et al. 2014, Border et al. 2018). Most of these approaches deal with small objects reconstructed by cameras mounted on the end-effector of robot manipulators, and make strong assumptions on the navigable free space. In the last five years, NBV inspection methods have been extended to mobile robots by using volumetric representations (Vasquez-Gomez et al. 2014, Yoder and Scherer 2016) and the

Method	Mapping	View planning
Ours (2020)	**TSDF**	**Roadmap**
(Bircher et al. 2018)	TSDF	Random sampling
(Newcombe et al. 2011)	TSDF, Point cloud	Random sampling
(Schmid et al. 2020)	TSDF	Random sampling
(Kompis et al. 2021)	TSDF, ESDF	Roadmap
(Mannucci et al. 2017)	OctoMap	Random sampling
(Corah and Michael 2021)	General occupancy grid	Random sampling
(Lauri et al. 2020)	General occupancy grid	Offline sampling

Method	Criterion	Path computation	Multi-robot
Ours (2020)	**Surface**	**Lazy PRM***	✔
(Bircher et al. 2018)	Volumetric	RRT	✘
(Newcombe et al. 2011)	Volumetric	RRT*	✘
(Schmid et al. 2020)	Surface	RRT*	✘
(Kompis et al. 2021)	Surface	A*	✘
(Mannucci et al. 2017)	Volumetric	RRT*	✔
(Corah and Michael 2021)	Volumetric	MCTS	✔
(Lauri et al. 2020)	Volumetric	Offline	Multi-sensor

Table 5.1: *Main features of some representative state-of-the-art NBV reconstruction methods, compared to ours.*

TSDF (Hardouin, Morbidi, Moras, Marzat and Mouaddib 2020, Monica and Aleotti 2018, Schmid et al. 2020).

Recently, *hybrid* or *mixed methods*, which benefit from the main advantages of surface-inspection and volumetric-exploration approaches, have emerged for improved reconstruction accuracy and faster coverage (see Table 5.1). In (Bircher et al. 2018), the reconstruction process includes two steps: first, a coarse TSDF map of the environment is generated, and then, in a second modeling step, the surface is refined. Reasoning on the occupancy map, the algorithm in (Song and Jo 2017) allows the robot to cover the whole surface model. The authors have extended their approach in (Song and Jo 2018), where the RRT* path is further refined by taking the completeness of the environment surface, into account. More recently, in (Song et al. 2022), the method has been improved and validated via numerical and real-world experiments. These approaches rely on both volumetric and surface representations, which are costly to generate, and only a few of them solve the surface-inspection problem directly. A RRT-based planning method with a volumetric representation of the surface, has been presented in (Schmid

et al. 2020). However, the poses are still sampled randomly rather than planned in advance, which may lead to unnecessary maneuvers. In (Hardouin, Morbidi, Moras, Marzat and Mouaddib 2020), we proposed to extract the viewpoints directly from the knowledge of the map. A planning roadmap is created and used for 3D reconstruction, and the magnitude of unknown surface determines the stopping criterion. Finally, in (Kompis et al. 2021), the authors introduced a method reminiscent of ours, based on ESDF mapping and A* path search.

Multi-robot planning

As alluded to in the previous sections, the existing research on incremental online reconstruction generally targets a single robot, and multi-robot cooperative systems are still relatively rare in the literature (Amigoni and Gallo 2005). By considering robots equipped with laser scanners in a 2D environment, the authors in (Burgard et al. 2005) were the first to propose a method for computing frontier cells, and to define the trade-off between utility and distance traveled in a multi-agent exploration task. In (Juliá et al. 2012), the authors compared multiple volumetric objective functions in terms of reconstruction time and volume completeness, by considering pose measurement errors in 2D environments. In (Mannucci et al. 2017), the authors considered a team of aerial robots in an uncluttered outdoor environment, and proposed one of the first cooperative frontier-based methods, which relies on 3D space modeling for multi-robot exploration. The centralized OctoMap and the (RRT*-based) coordinated motion planning of the aerial vehicles are computed on a base station, and the algorithm is evaluated via realistic numerical experiments with emulated stereo sensors in ROS/Gazebo. Exploration methods based on probabilistic occupancy maps with entropy reduction, such as decMCTS (Best et al. 2016) or SGA (Atanasov et al. 2015), stem from the notion of mutual information of range sensors (Charrow et al. 2015). The authors in (Corah et al. 2019, Corah and Michael 2019) have proposed a finite-horizon decentralized planner, called DSGA (Distributed Sequential Greedy Assignment), which relies on sampling-based Monte-Carlo Tree Search (MCTS) (Chaslot 2010). The paths are allocated to the robots by solving a submodular maximization problem over matroid constraints with greedy assignment heuristics. More recently, in (Corah and Michael 2021), the authors have established connections between information-theoretic and volumetric coverage objectives in terms of expected coverage, for teams of mobile robots. Finally, a similar matroid-constrained submodular maximization problem has been considered in (Lauri et al. 2020) for multi-sensor NBV planning, and real-world experiments have been conducted with two KUKA robot arms. One NBV per sensor and per iteration is computed, and viewpoint sampling and trajectory generation are performed offline, by assuming a partial prior knowledge of the environment (location of the target object).

Based on this literature review, we can notice that a large body of research has leveraged volumetric exploration for 3D reconstruction, but without taking the problems of surface completeness and occlusion explicitly into account. Moreover, even though the surface inspection problem with mobile robots is becoming increasingly popular, to the best of our knowledge, no multi-robot formulation exists.

We propose here a generic NBV planning strategy inspired by the mixed approaches, to solve the surface inspection problem and cooperatively reconstruct large-scale environments with a team of mobile robots. Our frontier-based method relies on a volumetric representation of the surface, which allows to identify areas of interest to be scanned. Candidate viewpoint configurations are generated from these areas in compliance with the sensing and dynamic constraints of the robots, and they are clustered according to their location in space. In order to find the best path for each robot, we evaluate the interest of visiting a specific cluster. By successively solving this assignment problem, we find the paths which allow to explore the unknown environment and maximize the completeness of reconstructed surface, while ensuring short travel distances and execution times. The proposed strategy, called TSGA (TSP-Greedy Allocation), has been validated via extensive experiments with multiple quadrotor UAVs and wheeled robots, by using a centralized architecture. A distributed variant, referred to as dist-TSGA, is also introduced and tested via numerical and real-world experiments. With dist-TSGA, it is possible to perform cluster assignment in a decentralized fashion, and to keep track of the map under construction.

5.3.3 Problem formulation

Let us consider a team of N cooperative mobile agents[3]. Let $\mathbf{q}^i \in \mathrm{SE}(m)$ be the pose of agent i, and $\mathbf{q}_0^i$ its initial configuration, $i \in \{1, 2, \ldots, N\}$: $m = 2$ in the case of ground vehicles, and $m = 3$ in the case of aerial vehicles. We assume that all agents are equipped with an accurate localization system which allows to estimate their pose with respect to a global reference frame, and that a robust low-level trajectory tracking algorithm is available. Each agent is equipped with a forward-facing depth sensor with limited field-of-view (FoV) and sensing range, extrinsically calibrated with respect to its body frame. The agents should cooperatively scan an unknown 3D environment (for instance, a building), characterized by its surface. A mapping algorithm is required to build a representation of the reconstructed surface and identify the free space for navigation. We consider a volumetric mapping, which allows to build a map M as a collection of discretized 3D space elements. These elements, referred to as *voxels* $\mathbf{v} \in M$, represent *unknown*, *occupied*, or *empty* space. Let $X \subset M$ be the set of unknown voxels, and $A \subset M$ the set of known voxels such that $X \cap A = \emptyset$. Moreover, let $O \subset A$ be

[3]We will use the terms "agent" and "robot", and "path" and "trajectory" interchangeably.

the set of occupied voxels, and $E \subset A$ the set of empty voxels. The goal of the scanning process is to discover unknown voxels: in particular, a voxel is said to be *scanned* when it becomes known, once at least one agent has detected it. Similarly to (Yoder and Scherer 2016, Monica and Aleotti 2018, Kompis et al. 2021), we aim at identifying the incomplete surface within a given volume. Based on (Monica and Aleotti 2018, equ. (2)), we propose the following general definition of incompleteness of a surface model:

5.2. Definition (Incomplete surface element). *We call* Incomplete Surface Element*, or ISE, for short, a voxel* $\mathbf{v} \in M$ *lying on the surface at a frontier, near both the unknown and empty space. Let* C *be the set of all ISEs. A voxel* $\mathbf{v} \in C$ *if and only if*

$$\begin{aligned} &a)\ \mathbf{v} \in E, && (\textit{empty}) \\ &b)\ \exists\, \mathbf{u} \in \mathcal{N}_{\mathbf{v}}^{6}\ \textit{s.t.}\ \mathbf{u} \in X, && (\textit{unknown}) \\ &c)\ \exists\, \mathbf{o} \in \mathcal{N}_{\mathbf{v}}^{18}\ \textit{s.t.}\ \mathbf{o} \in O, && (\textit{occupied}) \end{aligned}$$

where $\mathcal{N}_{\mathbf{v}}^{6}$ *and* $\mathcal{N}_{\mathbf{v}}^{18}$ *denote the 6- and 18-connected voxel neighborhoods of* $\mathbf{v}$*, respectively.* ◇

5.3. Definition (Remaining incomplete surface). *Let* Q *be the set of all collision-free configurations of an agent, and let* $Q_c \subseteq Q$ *be the set of all configurations from which an ISE* $\mathbf{v} \in C$ *can be scanned. The* remaining incomplete surface *is then defined as*

$$C_{\text{rem}} = \bigcup_{\mathbf{v} \in C} \{\mathbf{v} \mid Q_c = \emptyset\}.$$

◇

We will use the function $\mathbf{p}_{j,k}^{i}(s) : [0,\, 1] \to \mathrm{SE}(m)$, $m \in \{2,\, 3\}$, to define the path of agent i from configuration j to configuration k, where $\mathbf{p}_{j,k}^{i}(0) = \mathbf{q}_j^i$ and $\mathbf{p}_{j,k}^{i}(1) = \mathbf{q}_k^i$, $i \in \{1, 2, \ldots, N\}$. We assume that $\mathbf{p}_{j,k}^{i}(s)$ is collision-free and feasible for agent i (i.e. the kinematic/dynamic constraints of the robot are satisfied along the path). The problem studied of interest can then be formally stated as follows.

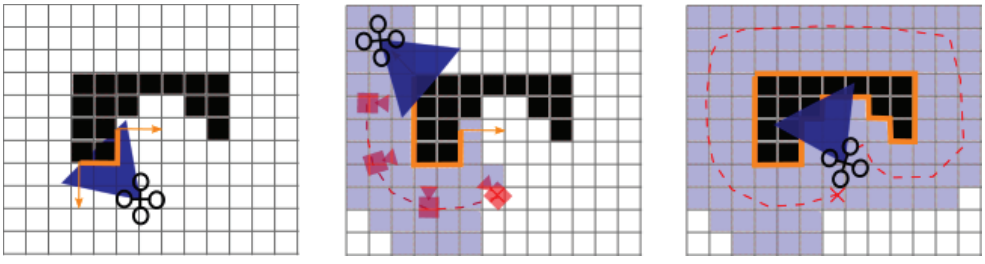


Figure 5.8: *2D illustration (from left to right) of NBV planning for surface inspection via volumetric mapping. White, blue, and black voxels represent unknown, empty, and occupied voxels, respectively. The surface is depicted in orange.*

5.1. PROBLEM (MULTI-AGENT INSPECTION PROBLEM). *Consider a team of N agents with initial configurations $\mathbf{q}_0^i \in Q,\ i \in \{1, 2, \ldots, N\}$. The multi-agent inspection problem asks to find collision-free paths $\mathbf{p}_{0,f}^i(s)$ visiting the poses $\mathbf{q}_k^i,\ k \in \{0, 1, \ldots, f\}$, which allow the agents to scan the set $C_{\text{ins}} = C \setminus C_{\text{rem}}$ of all ISEs contained in the current reconstructed map M.* $\diamond$

By progressing along their paths $\mathbf{p}_{0,f}^i(s),\ i \in \{1, 2, \ldots, N\}$, the agents are able to disclose the unknown space, discover new ISEs, and iteratively solve the inspection problem until $C_{\text{ins}} = \emptyset$. Fig. 5.8 graphically illustrates this idea for a planar quadrotor.

Based on these premises, in the next section, we will provide a general overview of the approach proposed to solve Problem 5.1.

5.3.4 Overview of the proposed approach

In what follows, we present a generic multi-agent system for 3D surface reconstruction of unknown environments, which admits both a centralized and a distributed implementation. The centralized and distributed architectures depicted in Fig. 5.9, are well suited to accommodate multiple ground or aerial robots, or a combination thereof (i.e. heterogeneous fleets). The architectures include two distinct modules: a *perception module* (green block in Fig. 5.9), which extracts the ISEs (cf. Sect. 5.3.3) from a volumetric map estimated online, and a *planning module* (blue and orange blocks in Fig. 5.9), that is in charge of computing the paths of the robots.

A mapping algorithm acts as the front-end of the perception module, which takes as input the depth maps generated by the on-board sensors (RGB-D cameras, stereo-rig, etc.) together with their associated poses, and integrates them into a 3D volumetric map used for reconstruction (i.e. extraction of ISEs) and for navigation (i.e. collision-free path planning and tracking in the free space). The map is then processed for the duration of the overall mission in order to extract new ISEs, which are the centerpieces of the planning module. We chose the TSDF representation for its attractive properties, and in particular for its ability to implicitly represent surfaces. In fact, it allows to generate configurations in the free space that efficiently cover the ISEs, while taking the specificities of the environment and depth sensors (range, resolution, etc.) explicitly into account (see Sect. 5.3.5). However, note that other volumetric mapping methods could be used as well, with minor modifications. The *centralized architecture* incrementally integrates all input data (depth maps and poses from all agents) into a unique map on a single base station, where all the ISEs are generated. In the *distributed architecture*, instead, each agent computes its own local map, henceforth referred to as "*patch*" and identified by a unique ID, based on its own sensing and localization information. The patches are exchanged via a distributed algorithm based on manifold mapping (Howard 2004).

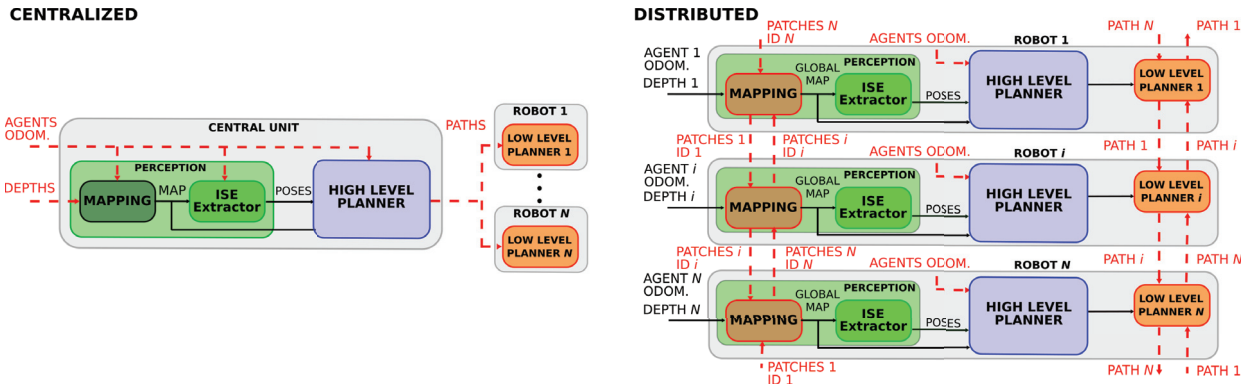


Figure 5.9: *General flow chart of our multi-agent surface-reconstruction architectures: [left] centralized, [right] distributed. The internal structure of the perception module is shown inside the green shaded box. The intra- and inter-block connections are represented with black solid and red dashed lines, respectively.*

The planning module guarantees that the agents complete the surface reconstruction. The scanning process stops when no ISEs are left. Given the list of poses provided by the perception module, one can schedule the visit of each configuration via an appropriate TSP-based path finder. To this end, the TSGA planner (TSP-Greedy Allocation) clusters sets of configurations according to their location in space, in order to identify and rank areas of interest in the incomplete map. It then generates a directed graph which represents the travel utility of visiting a cluster, depending on the capabilities of each robot (terrestrial or aerial). In order to maximize the cumulative utility function at the team level, collision-free paths are extracted from the digraph and broadcast to the robots. The high-level paths are sent to the low-level planners, which generate sampling-based trajectories for the agents and gather the odometric and path-allocation information for collision avoidance (see Sect. 5.3.6). In the distributed architecture, global-map inconsistencies, due, e.g., to patch losses or communication delays, may result in clusters assigned to multiple robots. To overcome this problem, the low-level planners exchange their current paths with the robots, check the consistency between the individual and team-wise allocation, and wait for a re-assignment, if needed. Finally, trajectory tracking is performed with standard controllers (e.g. PID or Model Predictive Control).

5.3.5 Perception

Surface-based mapping

A volumetric map M, here based on a TSDF representation, is used to detect non-reconstructed areas, as defined by the extraction of ISEs. The TSDF map (Curless and Levoy 1996) consists of a voxel grid, where each voxel contains a truncated signed distance value $\phi \in \mathbb{R}$ and a positive weight w. It implicitly represents surfaces, which correspond to the zero level set of the distance field: hence, the TSDF volume is a *volumetric representation of a surface*. Algorithms such as MarchingCubes (Lorensen and Cline 1987) can be used to extract a triangular mesh, which is an explicit representation of those surfaces, e.g. for visualization. The map is built in an incremental fashion by sequentially integrating depth measurements. In order to keep the map consistent, the pose of the sensor on-board the robot must be used to relocate depth measurements with respect to the map frame. We assume that this pose is provided by a localization system which relies, for example, on a visual SLAM algorithm (cf. (Engel et al. 2015, Mur-Artal and Tardós 2017)), and that the pose estimates are sufficiently accurate. At each time step and for each voxel $\mathbf{v}$, the integration is performed by recursively computing a weighted mean of the distance. In order to take into account uncertainty due to sensors (Nguyen et al. 2012, Oleynikova et al. 2020), the new measurements are weighted by an inverse-squared distance increment $1/z_{\mathbf{q}}^2(\mathbf{v})$, where $z_{\mathbf{q}}(\mathbf{v})$ is the distance between voxel $\mathbf{v}$ and the current pose $\mathbf{q}$ of an agent[4]. The state of a voxel $\mathbf{v}$ is set to *known* (either *occupied* or *empty*), if $w(\mathbf{v}) \geq W_{\text{th}}$ and to *unknown* if $w(\mathbf{v}) < W_{\text{th}}$, where the positive threshold W_{th} depends on the sensing range of the depth sensor.

ISE extractor and viewpoint generation

Similarly to (Monica and Aleotti 2018), a voxel $\mathbf{v} \in M$ is considered as an ISE, i.e. $\mathbf{v} \in C$, as stated in Definition 5.2, if it verifies the following conditions:

$$
\begin{aligned}
&a)\ w(\mathbf{v}) \geq W_{\text{th}} \wedge \phi(\mathbf{v}) > 0, && \text{(empty)}\\
&b)\ \exists\, \mathbf{u} \in \mathcal{N}_{\mathbf{v}}^{6} \text{ s.t. } w(\mathbf{u}) < W_{\text{th}}, && \text{(unknown)}\\
&c)\ \exists\, \mathbf{o} \in \mathcal{N}_{\mathbf{v}}^{18} \text{ s.t. } w(\mathbf{o}) \geq W_{\text{th}} \wedge \phi(\mathbf{o}) \leq 0. && \text{(occupied)}
\end{aligned}
$$

5.4. DEFINITION (SCANNED ELEMENT). *A voxel* $\mathbf{v} \in M$ *which satisfies,* $w(\mathbf{u}) \geq W_{\text{th}}$, $\forall\, \mathbf{u} \in \mathcal{N}_{\mathbf{v}}^{6}$, *is called a* scanned element. ◇

[4]In the interest of clarity, in the remainder of this section, we will drop the superscript i, and we will simple write $\mathbf{q}$ instead of $\mathbf{q}^i$.

The direction $\overline{\mathbf{n}}_{\mathbf{v}}$ to observe the ISE $\mathbf{v}$, is determined from the gradient of the weight function $\nabla w(x, y, z)$, which can be computed as

$$\mathbf{n}_{\mathbf{v}} = \sum_{\mathbf{c} \in \mathcal{N}_{\mathbf{v}}^{26}} w'(\mathbf{c}) \frac{\mathbf{c} - \mathbf{v}}{\|\mathbf{c} - \mathbf{v}\|}, \quad \overline{\mathbf{n}}_{\mathbf{v}} = \frac{\mathbf{n}_{\mathbf{v}}}{\|\mathbf{n}_{\mathbf{v}}\|},$$

where $\mathcal{N}_{\mathbf{v}}^{26}$ is the 26-connected neighborhood of $\mathbf{v}$ and the weight function

$$w'(\mathbf{c}) = \begin{cases} -W_{\text{th}} & \text{if voxel } \mathbf{c} \text{ is occupied,} \\ W_{\text{th}} & \text{otherwise.} \end{cases}$$

Note that the last definition differs from (Monica and Aleotti 2018, equ. (6)), since $w'(\mathbf{c})$ takes the value W_{th} if the voxel is unknown or empty (in our experiments, we empirically observed that this variant is more robust against noisy data). A sensor configuration is generated along the direction $\overline{\mathbf{n}}_{\mathbf{v}}$ at a distance δ_{pose} from the corresponding voxel $\mathbf{v}$ (see Fig. 5.10). The sensor is oriented towards $\mathbf{v}$ along $-\overline{\mathbf{n}}_{\mathbf{v}}$, and the value of δ_{pose} depends on the sensing range of the depth sensor.

If two poses $\mathbf{q}_j, \mathbf{q}_k \in Q$ generated from the ISEs $\mathbf{v}_j, \mathbf{v}_k \in C$, are located within a short distance (i.e. $\text{dist}(\mathbf{q}_j, \mathbf{q}_k) < \epsilon$, for a small $\epsilon > 0$), and their viewing directions $\overline{\mathbf{n}}_{\mathbf{v}_j}$, $\overline{\mathbf{n}}_{\mathbf{v}_k}$ are almost parallel (i.e. $|\overline{\mathbf{n}}_{\mathbf{v}_j} \cdot \overline{\mathbf{n}}_{\mathbf{v}_k}| \simeq 1$, where "$\cdot$" denotes the dot product), then these configurations are merged into a single viewpoint, by averaging their positions and orientations. This allows to reduce the overall number of poses, without missing key information. Even after the merging step, a large number of candidate poses pointing towards the ISEs is typically generated in large-scale environments, which is not compatible with the planning objective. To avoid this problem, neighboring viewpoints are

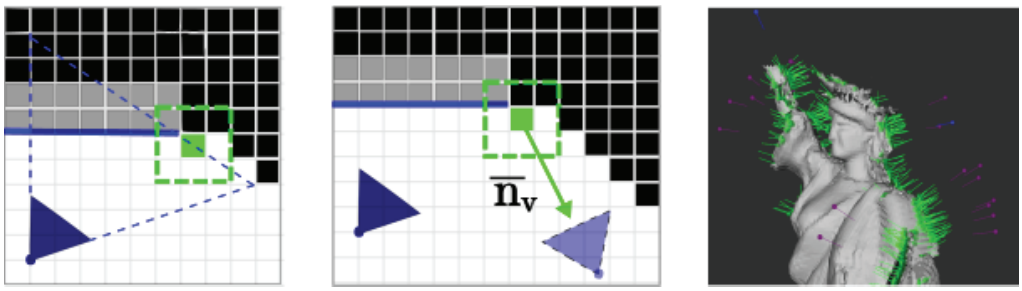


Figure 5.10: *[left] Two-dimensional example of ISE* $\mathbf{v}$ *(filled green square). Its 2D neighborhood is represented by a dashed green square. Unknown voxels are black, occupied are gray, and empty voxels are white. The reconstructed surface is depicted as a blue segment, and the sensor configuration and its frustum as a dark blue triangle. [center] Direction from the contour,* $\overline{\mathbf{n}}_{\mathbf{v}}$*, and corresponding viewpoint configuration at distance* δ_{pose} *from* $\mathbf{v}$ *(light blue triangle). [right] Snapshot of a simulated reconstruction in progress, with ISEs and their directions from the contour (green arrows).*

grouped into N_c *clusters* U_j, $j \in \{1, 2, \ldots, N_c\}$ (the idea, as will become apparent later, is to assign each cluster to an agent, in order to improve efficiency). The set of all clusters is denoted by $\mathcal{U} = \{U_1, U_2, \ldots, U_{N_c}\}$. A configuration $\mathbf{q}_l$ belongs to a generic cluster U if $\exists\, \mathbf{q}_j \in U$ such that $\mathrm{d}(\tau_l^j) < d_\nu$, where $\mathrm{d}(\tau_l^j)$ denotes the length of the path τ_l^j between $\mathbf{q}_l$ and $\mathbf{q}_j$ on a directed graph (to be defined in Sect. 5.3.6), and d_ν is an upper bound on the distance. If no neighbors are found, d_ν is increased up to a maximum value $d_\nu^{\max}$. Once the clusters have been defined, their respective level of informativeness needs to be quantified, since each cluster does not necessarily contain the same number of viewpoints. To evaluate a configuration, we use the ray-tracing method (Bresenham 1965) from a frontier-based perspective, i.e. the ISEs that can be seen, are counted. Let $C_\mathbf{q}$ be the set of all ISEs seen from viewpoint $\mathbf{q}$ and let $C_U = \bigcup_{\mathbf{q} \in U} C_\mathbf{q}$. The *gain* $g(U)$ of cluster U is then defined as

$$g(U) = \frac{|C_U|}{|C|}, \tag{5.31}$$

where $|C_U|$ denotes the cardinality of the set C_U.

Centralized vs distributed mapping

In the centralized architecture, all the computations are performed on the base station. The agents send their poses and depth maps to it, and a GPU-based algorithm[5] fuses them to create the global TSDF volume. The ISE extractor then computes the ISEs for the whole team. The correct operation of the base station is crucial to the centralized architecture: in fact, if it crashes, a system breakdown occurs. If the communication with a robot is temporary broken, the robot freezes, and a degradation of planning performances is experienced. On the other hand, in the distributed architecture, a CPU-based distributed manifold mapping founded upon (Duhautbout et al. 2019), enables each agent to compute TSDF (sub-)maps on its own embedded computer. The general block diagram of the mapping module is depicted in Fig. 5.11. To synchronize the map among the agents, it is subdivided into different patches. Each patch is a local TSDF with a unique ID. It is associated with a local frame (i.e. the frame of the first depth map processed), which is used to integrate the depth maps until a certain user-defined event is triggered. Originally, in (Duhautbout et al. 2019), a distance-traveled criterion was considered, for simultaneous localization and mapping (SLAM). Here, we assume that a new patch is created, when a certain number of depth maps has been integrated into the current map. Once this occurs, the current patch is locally stored into the agent's *private map* and shared with the others, and a new "current" patch is initialized. Moreover, for each new patch stored, the list of patch IDs is updated and broadcast to the other

[5]When the depth maps sent by all the agents are fused, a GPU implementation is needed to ensure real-time performance.

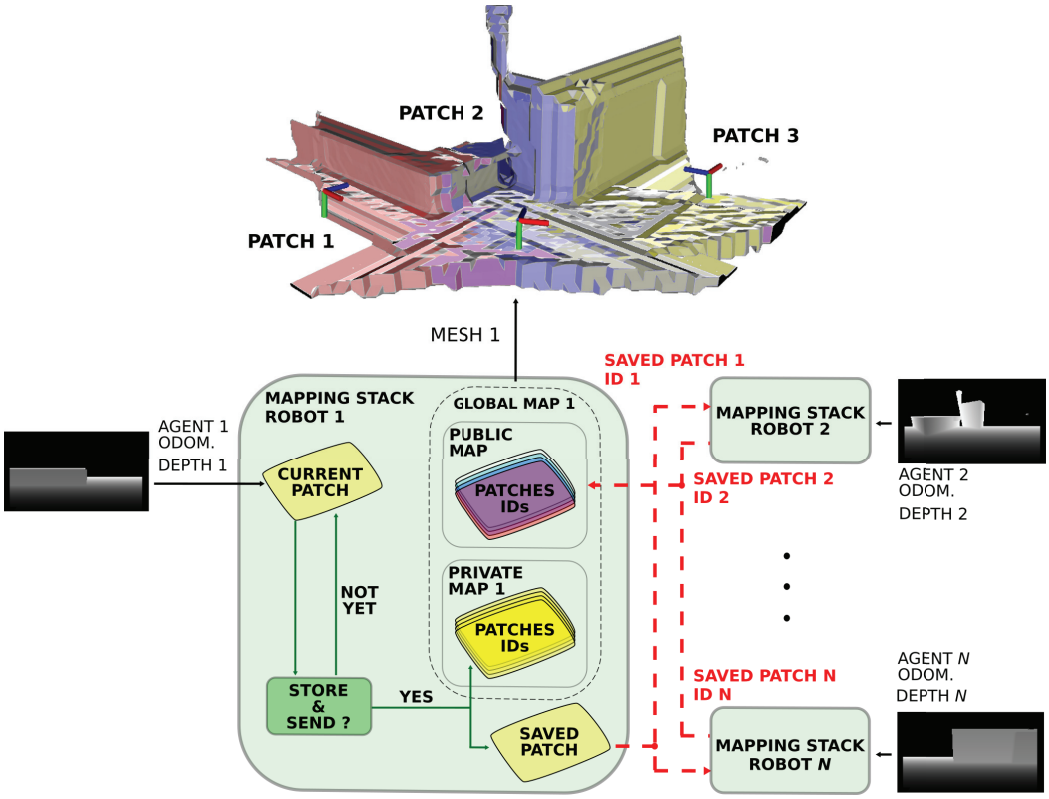


Figure 5.11: *Distributed mapping: inner structure of the module. The intra- and inter-block connections are represented with green solid and red dashed lines.*

agents. Thanks to this ID list, a client-server protocol allows the other agents to request a missing patch which might have been lost during the first communication attempt. The request is processed by the closest agent which owns it. Finally, when a robot receives a patch sent by another agent, it stores it locally into its *public map*.

An approximation of the global map, denoted by $\widehat{M}_i$, can thus be constructed by agent i, as the union of its current patches, its private map, and its public map. We improved the method proposed in (Duhautbout et al. 2019), in order to rebuild the TSDF map from the collection of patches. In fact, the volumes are aggregated, and the overlapping regions between the patches are fused together by summing up the weights and computing the weighted average of distance values for each TSDF voxel. To not overload the communication network, the current map of an agent is not accessible to the others until its completion. Therefore, even excluding the communication losses, the agents do not have access to the full map M, simultaneously. However, it is worth pointing out that our list-and-request mechanism to synchronize the older patches between the agents, ensures that the majority of the global map M is available to the agents, except for the most recent patches which are currently being built by each agent. Follow-

ing the procedure described in the previous subsection, the ISEs are extracted by agent i from $\widehat{M}_i$, and $\widehat{\mathcal{U}}_i = \{U_1^i, U_2^i, \ldots, U_{N_C}^i\}$ is used to denote the set of clusters that should be observed to complete the surface reconstruction process. Finally, the gains are computed with equation (5.31), as in the centralized case. Fault tolerance is an asset to our architecture: in fact, in the event of a communication failure, each agent can evolve independently using the last map exchanged. New patches are locally stored as the agent reconstructs the environment on its own, and if communication is re-established, they can be requested by the team, once the IDs list has been updated.

5.3.6 Planning

Next-Best-View planning

The high-level planner allocates clusters to the agents and schedules their visit according to a given common TSDF map (or its best approximation, in the distributed architecture). To formulate our optimization problem, we introduce the weighted directed graph $\mathcal{G} = (\mathcal{U}, \mathcal{E}, \{a_{UV}\}_{(U,V)\in\mathcal{E}})$, where $\mathcal{U}$ is the set of clusters, $\mathcal{E}$ is the set of edges, and $\{a_{UV}\}_{(U,V)\in\mathcal{E}}$ is the collection of weights associated to the edges. Each directed edge $e_{UV} \in \mathcal{E}$ connects cluster U to cluster V, with $U, V \in \mathcal{U}$. It is assumed that the initial configuration of agent i belongs to one of the clusters of $\mathcal{G}$, i.e. $\mathbf{q}_0^i \in \mathcal{U}$. Let $\mathbf{q}_k^i$ be a configuration in cluster U, and $\mathbf{q}_l^i$, $\mathbf{q}_m^i$ two configurations in cluster V. Then, the weight a_{UV} between cluster U and V is the 6-tuple

$$a_{UV} = \left\{\tau_k^l,\, \tau_l^m,\, g(V),\, \mathrm{d}(\tau_k^l),\, \mathrm{d}(\tau_l^m),\, f_{UV}\right\}, \tag{5.32}$$

where

- τ_k^l denotes the path from $\mathbf{q}_k^i \in U$ to $\mathbf{q}_l^i \in V$, i.e. the path between cluster U and cluster V,
- τ_l^m denotes the shortest Hamiltonian path (Godsil and Royle 2001) including configurations of V, which starts at $\mathbf{q}_l^i$ and ends at $\mathbf{q}_m^i$,
- $g(V)$ is the gain of cluster V, as defined in (5.31),
- $\mathrm{d}(\tau_k^l)$ is the cost associated with the inter-cluster path τ_k^l, i.e. the length of τ_k^l,
- $\mathrm{d}(\tau_l^m)$ is the cost associated with the intra-cluster path τ_l^m, i.e. the length of τ_l^m,
- f_{UV} is the *utility function* defined as

$$f_{UV} \;=\; g(V)\exp\big(-\lambda_{\text{tc}}\,\mathrm{d}(\tau_k^l) - \lambda_{\text{ic}}\,\mathrm{d}(\tau_l^m)\big), \tag{5.33}$$

where λ_{tc} and λ_{ic} are positive penalty terms for the inter-cluster and intra-cluster costs, respectively, which can be used to promote the visit of clusters far apart or large clusters. Their value depends on the motion capabilities of the agents (i.e. ground vs aerial robots). A similar utility function was originally proposed in (González-Banos and Latombe 2002).

The weights on the directed graph $\mathcal{G}$ in (5.32), quantify the potential benefit of choosing a certain path, to pursue the 3D reconstruction: in fact, the higher the value of the function f_{UV}, the more convenient is the path. Note that $f_{UV} > 0$, since $g(V) > 0$.

Our inspection problem can be stated as a *maximum Asymmetric Traveling Salesman Problem* (maxATSP), i.e. as the problem of finding a maximum-utility Hamiltonian path $\mathbf{p}$ on $\mathcal{G}$, see (Hardouin, Moras, Morbidi, Marzat and Mouaddib 2020). In a graph, a Hamiltonian path is an undirected or directed path that visits each vertex exactly once. In what follows, we will denote by maxATSP($\mathcal{U}$), the set function that takes the set of clusters $\mathcal{U}$ as input and outputs its utility value p, from which the path $\mathbf{p}$ can be computed. A linear programming formulation of maxATSP is

$$
\begin{aligned}
\max \quad & \sum_{U\in\mathcal{U}} \sum_{V\in\mathcal{U}} && f_{UV}\, x_{UV} \\
\text{s.t.} \quad & && \\
& x_{UV} \in \{0,1\},\, U,V \in \mathcal{U},\, U \neq V, && \\
& \sum_{U\in\mathcal{U},\, U\neq V} && x_{UV} = 1, \qquad && V \in \mathcal{U}\setminus\{\mathbf{q}_0\}, \\
& \sum_{V\in\mathcal{U}\setminus\{\mathbf{q}_0\},\, V\neq U} && x_{UV} \leq 1, \qquad && U \in \mathcal{U}, \\
& \sum_{V\in\mathcal{U}\setminus\{\mathbf{q}_0\},} && x_{\{\mathbf{q}_0\}V} = 1, \qquad && \{\mathbf{q}_0\} \in \mathcal{U}, \\
& \sum_{U\in\mathcal{S}} \sum_{V\in\mathcal{S},\mathcal{V}\neq\mathcal{U}} && x_{UV} \leq |\mathcal{S}| - 1, \qquad && \forall\, \mathcal{S} \subsetneq \mathcal{U},\, |\mathcal{S}| > 2,
\end{aligned}
$$

where $\{\mathbf{q}_0\}$ denotes the cluster which only contains the initial configuration, $\mathcal{S}$ is a proper subset of $\mathcal{U}$, and $x_{UV} = 1$, if the edge belongs to the optimal path, and $x_{UV} = 0$, otherwise. maxATSP is solved by converting it into a symmetric TSP (i.e. a standard TSP) and then by using the classical Lin-Kernighan heuristic (Helsgaun 2000).

Let $\mathcal{U}^i$ be the set of clusters assigned to agent $i \in \{1, 2, \ldots, N\}$, such that $\bigcup_{i=1}^{N} \mathcal{U}^i = \mathcal{U}$. Then, the assignment problem can be stated as follows

$$
\max_{\mathcal{U}^1,\ldots,\mathcal{U}^N \subset \mathcal{U}} \Big\{ \sum_{i=1}^{N} \text{maxATSP}(\mathcal{U}^i) \,\Big|\, \mathcal{U}^i \cap \mathcal{U}^\ell = \emptyset,\; i \neq \ell,\; \bigcup_{i=1}^{N} \mathcal{U}^i = \mathcal{U} \Big\}, \tag{5.34}
$$

where $\sum_{i=1}^{N} \text{maxATSP}(\mathcal{U}^i)$ is a non-decreasing set function, and the space of feasible paths has the structure of a simple partition matroid (Wilson 1973). Problem (5.34) can

Algorithm 1: *TSP-Greedy Allocation (TSGA)*

Set $\mathcal{U}^i = \emptyset$ and $p^i = 0$ for each agent $i \in \{1, 2, \ldots, N\}$;
foreach *cluster* $V \in \mathcal{U}$ **do**
  $i \leftarrow \arg\max_{k \in \{1,2,\ldots,N\}} \{\text{maxATSP}(\mathcal{U}^k \cup V) - p^k\}$;
  $\mathcal{U}^i \leftarrow \mathcal{U}^i \cup V$;
  $p^i \leftarrow \text{maxATSP}(\mathcal{U}^i)$;
  $\mathbf{p}^i_{\mathcal{U}^i} \leftarrow \{p^i, \mathcal{U}^i\}$;
Send paths $\mathbf{p}^1_{\mathcal{U}^1}, \ldots, \mathbf{p}^N_{\mathcal{U}^N}$ to the agents (low-level planner);

be approximately solved via local greedy heuristics (cf. (Bian et al. 2017, Nemhauser et al. 1978, Fisher et al. 1978)), which seek for the local maximum utility, based on an initial ranking of the items to assign. The centralized TSP-Greedy Allocation (TSGA) procedure (Hardouin, Moras, Morbidi, Marzat and Mouaddib 2020) is reported in **Algorithm 1** and its distributed version (dist-TSGA) in **Algorithm 2**. Note that the single-agent algorithm is a special case of the centralized multi-agent algorithm with $N = 1$.

At each ISE extraction, the clusters are formed, and the shortest Euclidean distance to each agent is computed. These distances are then arranged in ascending order for the greedy heuristic. The TSGA planners greedily assign each cluster to an agent. More specifically, a cluster is assigned, when it locally maximizes the overall utility for the team. The path of agent i, e.g. the viewpoint sequence which results from the allocated clusters $\mathcal{U}^i$, is denoted by $\mathbf{p}^i_{\mathcal{U}^i}$, and the associated utility value by p^i. Once $\mathbf{p}^i_{\mathcal{U}^i}$ is computed, it is sent to the low-level planner. Unlike the classical insertion methods, in which a cluster is added to the path of a robot path (Jawaid and Smith 2015), maxATSP is solved for the extended cluster set $\mathcal{U}^i \cup V$ with $V \in \mathcal{U} \setminus \mathcal{U}^i$. This strategy maximizes the individual utility of the agents over disjoint sets, so as to maximize, in turn, team-wise utility. Moreover, it is amenable to a distributed implementation, since only local information is used (e.g. local free space, ISEs, $\mathcal{U}^i$ related to the map of agent i). On the long run, the maximization of the utility function pushes the agents towards the most valuable areas, in terms of completeness. For instance, this might prompt an agent to visit areas at the frontier between a known and an unknown surface containing multiple ISEs, and scan them all (cf. equation (5.31)).

Low-level planner

The low-level planner computes the path of an agent, using Lazy PRM* from the Open Motion Planning Library (OMPL) (Şucan et al. 2012). It leverages the path found by the TSGA planner (**Algorithm 1** or **2**) and the TSDF-map updates. Given a start and an end pose, it computes the path of a robot in the free space, given by the TSDF volume

Algorithm 2: *dist-TSGA*

Set $\widehat{\mathcal{U}}_i^i = \emptyset$ and $p_i^i = 0$ for agent $i \in \{1, 2, \ldots, N\}$;
foreach *cluster* $V \in \widehat{\mathcal{U}}_i$ **do**
  $\ell \leftarrow \arg\max_{k \in \{1,2,\ldots,N\}} \{\text{maxATSP}(\widehat{\mathcal{U}}_i^k \cup V) - p_i^k\}$;
  if $\ell = i$ **then**
    $\widehat{\mathcal{U}}_i^i \leftarrow \widehat{\mathcal{U}}_i^i \cup V$;
    $p_i^i \leftarrow \text{maxATSP}(\widehat{\mathcal{U}}_i^i)$;
    $\mathbf{p}_{\widehat{\mathcal{U}}_i^i}^i \leftarrow \{p_i^i, \widehat{\mathcal{U}}_i^i\}$;
  Send path $\mathbf{p}_{\widehat{\mathcal{U}}_i^i}^i$ to the low-level planner;

(on the plane for ground vehicles, and in the 3D space for aerial vehicles). By gathering all odometric information, each agent knows the position and orientation of the others, and it is then able to detect when another robot is near, when it faces it, or when it will cross its path. A *TrafficPolicy* function takes care of collision avoidance: for example, a robot might be asked to step aside to avoid a frontal impact, or to temporarily stop and wait until a teammate crossing its path, is outside its FoV. The function takes the robot's speed into account, and relies on a safety distance threshold for collision avoidance. On the other hand, the *ObstacleCheck* function triggers an emergency stop via a distance threshold to the surface (obtained from the TSDF), if a new obstacle is detected along the path.

Note that an ISE may be potentially scanned before the planned visit of an agent. To avoid unnecessary back-and-forth motions, the *UtilityCheck* function computes the remaining ISEs along the paths since the last planning iteration, and it waits for a possible update. This ensures a reactive visit of uncovered ISEs, as the map grows over time.

The high-level planner may generate paths of various length: hence, the agents may finish their tours at different time instants. When an agent has completed its current path, it continues the reconstruction asynchronously, with the latest path provided by the high-level planner.

Centralized vs distributed planning

In the centralized architecture, the map is directly generated from the depth maps sent by the agents. The centralized map is used for the extraction of the ISEs and the determination of the configuration clusters. A resolution of problem (5.34) on the base station allows to compute the paths $\mathbf{p}_{\mathcal{U}^1}^1, \ldots, \mathbf{p}_{\mathcal{U}^N}^N$, which are broadcast to all the agents (cf. **Algorithm 1**).

In the distributed architecture, due to material constraints (such as, saturation of

communication network or packet losses), a newly-created patch may not be received by all the agents. Hence, in practice, the agents do not have the same knowledge of the full map. Nevertheless, provided that the patch request is frequent enough, agent i may have a rapid access to a full public map: it can thus update *its own local version of the global map* $\widehat{M}_i$, and synchronize it with the map of the other agents (see Sect. 5.3.5 and Fig. 5.11). Recalling that $\widehat{\mathcal{U}}_i$ denotes the set of all clusters generated from the ISEs extracted by agent i from its current global map $\widehat{M}_i$, let $\widehat{\mathcal{U}}_i^i \subset \widehat{\mathcal{U}}_i$ be the set of clusters assigned to agent i, such that $\bigcup_{j=1}^{N} \widehat{\mathcal{U}}_i^j = \widehat{\mathcal{U}}_i$. Based on the known set of clusters $\widehat{\mathcal{U}}_i$, each agent performs its own cluster allocation in a distributed way, to compute the best path. Once the assignment step is done successfully, agent i sends its path $\mathbf{p}_{\widehat{\mathcal{U}}_i^i}^i$ to the low-level planner (cf. **Algorithm 2**).

Because of possible inconsistencies in the global maps, multiple agents might be assigned to the same viewpoints of a cluster. To avoid such a scenario, the low-level planner broadcasts the path that is currently followed by an agent, to the others. In case of redundancy, the *RedundancyCheck* function of each low-level planner evaluates each agent's progress along the current path. The priority is given to the first agent which can reach the viewpoint, by taking its ranking in the list of visits and its current location, into account. The agents which were not granted priority, keep on moving until their last maneuver before the redundant cluster, and then wait for a new high-level path.

5.3.7 Numerical experiments

In this section, the centralized and distributed multi-robot systems are validated via extensive simulations with synthetic data. As a complement to our preliminary results in (Hardouin, Morbidi, Moras, Marzat and Mouaddib 2020), the single-robot architecture is also compared with (Schmid et al. 2020), using the simulation environment of the authors. Our baseline multi-agent planner, to test TSGA and dist-TSGA, is the Nearest Neighbor (NNB) greedy algorithm. In NNB, only one cluster is allocated to each robot by locally computing $\max_{V \in \mathcal{U}} f_{UV}$ for the updated map (cf. equation (5.33)). Replanning is thus very fast compared to TSGA's, but only one cluster at a time is set to be visited. The different methods are evaluated in terms of map accuracy, surface completeness, total path length, and execution time.

Robots: RotorS simulator (Furrer et al. 2016) has been used to model quadrotor UAVs equipped with a stereo camera, in the ROS-Gazebo environment. We considered 3 and 5 UAVs in our tests, and report the single-robot case previously studied in (Hardouin, Morbidi, Moras, Marzat and Mouaddib 2020), for the sake of completeness. Each UAV has 4 degrees of freedom: its 3D position $[x,\, y,\, z]^T$ and its yaw angle ψ.

Simulation setup: We chose an industrial plant benchmark, which is well-known in the volumetric exploration literature (Song and Jo 2018). The **Powerplant** model[6] (see Fig. 5.12-[top-left]) was scaled to fit in a $65 \times 42 \times 15\,\mathrm{m}^3$ box (as a consequence, the five flues have the same height). To study the impact of the two penalty terms in the utility function (5.33), on the reconstruction accuracy/completeness, we also considered the monumental Statue of Liberty[7] (**SoL**), of size $20 \times 20 \times 60\,\mathrm{m}^3$, see Fig. 5.12-[top-left]. The simulation parameters used in the two scenarios are reported in the first and second column of Table 5.2.

To represent the depth-map uncertainty, we considered a Gaussian noise model. The standard deviation associated with a pixel, corresponds to the depth-value sensing error of the corresponding point located at a distance z, i.e. $\sigma(z) = \frac{|e_d|}{fB} z^2$, where $|e_d|$ is the magnitude of the disparity error, f is the focal length in pixels, and B is the baseline of the stereo camera on the UAVs, in meters. Following (Nguyen et al. 2012, Keselman et al. 2017), the raw depth map was blurred out by using a 3×3 kernel. In the single-UAV and centralized architecture, the TSDF volume was generated with the algorithm proposed in (Zeng et al. 2017), that we adapted to multi-robot case. Our GPU-based algorithm allows to rapidly build and update (at about 1 Hz) the TSDF volume, as new depth maps sensed by robots are sent to the base station. Surface mesh re-

[6] `http://models.gazebosim.org`
[7] `https://free3D.com`

Table 5.2: *Parameters used in the numerical experiments.*

Parameter	**Powerplant**	**SoL**	**CB**
Voxel resolution $r_{\mathbf{v}}$ [m]	0.3	0.15	0.2
Threshold W_{th}	0.3	0.3	0.3
$e_{\max}$ [m]	0.2598	0.1299	0.1732
Camera range [m]	[1.6, 8]	[1, 5]	[1.5, 6]
Camera FoV [deg.] (H, V)	90×60	90×60	90×60
e_d [pixels]	0.1	0.1	0.1
f [pixels]	376	376	376
B [m]	0.11	0.11	0.11
Collision radius [m]	1	1	1
UAV nominal speed [m/s]	0.5	0.5	0.5
δ_{pose} [m]	4.7	3.6	2.5
d_ν [m]	2.0	2.5	1.5
$d_\nu^{\max}$ [m]	5	5	5
Penalty term λ_{tc}	0.3	0.17	0.25
Penalty term λ_{ic}	0.03	0.15	0.08

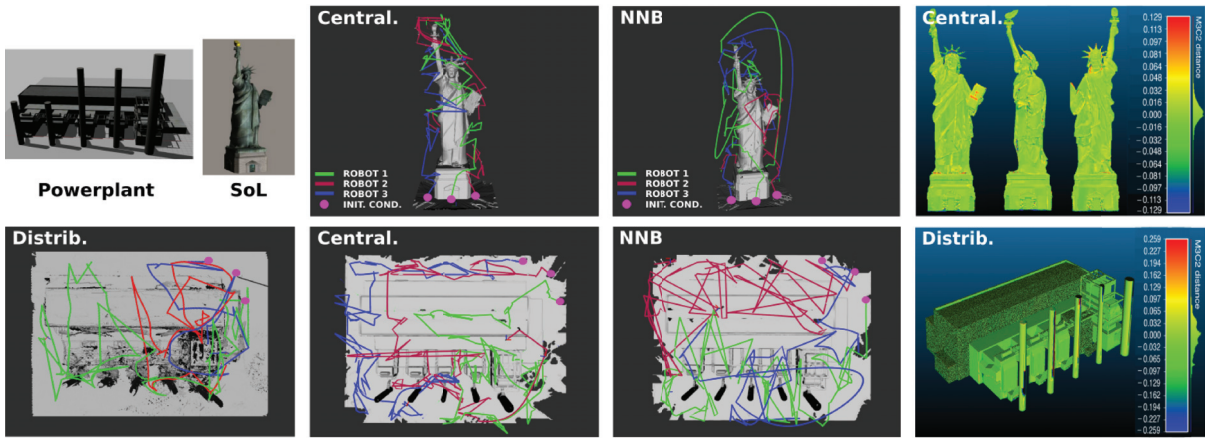


Figure 5.12: Numerical experiments*: [top-left]* ***Powerplant*** *and* ***SoL*** *models. Reconstructed meshes and 3D exploration paths* $\mathbf{p}^1_{0,f}$, $\mathbf{p}^2_{0,f}$, $\mathbf{p}^3_{0,f}$ *(green, red, blue) of the 3 UAVs for the two models, obtained with: [bottom-left] distributed architecture; [2nd column] centralized architecture; [3rd column] centralized architecture with NNB planner. The initial locations of the UAVs are marked with magenta dots. [4th column] Signed distance error: the color bar shows the error in meters with respect to the ground truth, computed with CloudCompare's M3C2 plugin.*

construction is performed with MarchingCubes (Lorensen and Cline 1987), and the TSDF weight increment has been modified to be the inverse of squared distance, as reported in Sect. 5.3.5. Distributed mapping is performed with the method proposed in (Duhautbout et al. 2019), which allows each robot to compute its own local volume, and send it to other robots, so that a global map can be obtained. The event that triggers the integration and broadcast of a new patch to the other robots, is that 5 depth maps have been processed. Unlike centralized mapping, distributed mapping is CPU-based, and it can be run on an embedded computer with limited resources. Lazy PRM* from OMPL (Şucan et al. 2012) is used by the low-level planner (see Sect. 5.3.6) to compute collision-free paths for the UAVs (the collision radius is set to 1 m). The UAVs track the generated paths using Model Predictive Control (Kamel et al. 2017), with a reference translational velocity fixed to 0.5 m/s.

The UAVs are initially located in the same area, around the base station (magenta dots in Fig. 5.12-[Columns 1 through 3]). The quantitative results of our numerical experiments are reported in Table 5.3. The single-UAV architecture with perfect and noisy depth measurements (denoted by "H", (Hardouin, Morbidi, Moras, Marzat and Mouaddib 2020), and "H*", respectively), has been used as a baseline, and compared with the centralized multi-robot architectures (with NNB and TSGA planners), and the distributed architecture (with the dist-TSGA planner), for a fleet of 3 and 5 UAVs. The

Table 5.3: *Results of the numerical experiments with **Powerplant**, **SoL** and **CB** (statistics over 10 trials).*

	Powerplant							
Number of UAVs	1		3			5		
Algorithm	H	H*	NNB	Centr.	Distr.	NNB	Centr.	Distr.
Path length [m]	780	785	1038	790	781	1113	879	872
Compl. time [min.]	32	33′10″	11′09″	10′20″	9′56″	6′51″	6′22″	6′09″
Time gain [%] w.r.t. H*	–	–	66.4	68.8	70.0	79.4	80.8	81.5
Surface coverage [%]	91.5	90.4	90.0	91.0	95.9	90.5	90.6	95.6
M3C2 avg. error [cm]	0.14	−0.13	−0.15	−0.26	0.62	−0.11	−0.3	−0.73
M3C2 std. error [cm]	5.85	7.51	7.52	7.54	3.33	7.50	7.52	3.43
RMSE [cm]	5.86	7.51	7.52	7.55	3.39	7.50	7.52	3.51

	SoL						**CB**	
Number of UAVs	1		3		5		1	
Algorithm	H	H*	NNB	Centr.	NNB	Centr.	S	H
Path length [m]	547.0	550.0	733.0	721.0	580.0	574.2	641	632
Compl. time [min.]	36′	37′	13′10″	10′18″	7′30″	6′45″	27′38″	25′20″
Time gain [%] w.r.t. H*	–	–	64.4	72.2	79.7	81.8	–	–
Surface coverage [%]	92.3	91.2	91.1	91.0	90.9	91.1	94.6	95.7
M3C2 avg. error [cm]	0.29	−0.02	−0.80	−0.03	0.06	−0.01	−1.06	0.28
M3C2 std. error [cm]	3.41	3.67	3.61	3.69	3.65	3.66	7.22	6.34
RMSE [cm]	3.43	3.67	3.69	3.69	3.65	3.66	7.29	6.34

last architecture has been only tested with **Powerplant**. To obtain statistically-significant values, 10 trials per architecture and per team of robots, were performed. For more details about the hardware platforms used in the simulations, see (Hardouin 2022).

Metrics: The architectures have been evaluated in terms of cumulative path length and completion time (to fully cover the 3D environments). This includes travel time and sensing time (e.g., one depth map integration and map update). The reconstructed 3D surface has been evaluated with CloudCompare[8] using the M3C2 (Multiscale Model to Model Cloud Comparison) algorithm (Lague et al. 2013). To assess how well the surface is covered, the reconstructed mesh is compared with a dense point cloud sampled on the ground truth (GT) mesh. The deviation is quantified via a cloud-to-mesh comparison (see Fig. 5.12-[4th column]). For a fair evaluation, all the invisible surfaces of the GT

[8] `https://danielgm.net/cc`

mesh were pruned beforehand (e.g., the interior floor and walls), and the analysis was restricted to the exterior surface mesh only. A point belonging to the GT point cloud is considered to be covered, if the shortest distance to this point along a normal to a mesh facet, is less than the length of the half diagonal of a voxel, i.e. $e_{\max} = r_{\mathbf{v}}\sqrt{3}/2$, where $r_{\mathbf{v}}$ is the voxel resolution. As a result, the more points are accurate, the better the coverage is. The quality of the recovered surface is evaluated in Table 5.3 (average and standard deviation of the signed distance error with respect to the GT point cloud and Root-Mean-Square Error).

Choice of penalty terms: The selection of penalty terms λ_{tc} and λ_{ic} in the utility function (5.33), depends on the nature of the 3D environment to explore. To find the combination of parameters which guarantees the shortest distance traveled, multiple reconstructions of **Powerplant** and **SoL** have been carried out with a single robot and different values of λ_{tc} and λ_{ic}. The results are compiled in Table 5.4, and indicate that the shape of the environment has a clear impact on the tuning of the penalty terms. In particular, in wide box-like environments as **Powerplant**, the majority of ISEs are uncovered near sharp edges or occluded regions, and tend to appear in groups separated by large layers of known space. To minimize the total path length, inter-cluster utility should then take priority over intra-cluster utility, i.e. $\lambda_{\text{tc}} \gg \lambda_{\text{ic}}$. On the other hand, the pedestal of the statue excluded, **SoL** predominantly consists of round surfaces and the average distance between two clusters is much smaller than in **Powerplant**. Hence, similar penalty terms are preferable (i.e. $\lambda_{\text{tc}} \simeq \lambda_{\text{ic}}$).

Results: From an inspection of Table 5.3, we can see that for a single UAV, the presence of noise has an impact on mesh accuracy, but that the navigation performance remains

Table 5.4: *Penalty terms and path lengths for **Powerplant** and **SoL**.*

	Powerplant				
λ_{tc}	0.35	0.3	0.15	0.03	0.01
λ_{ic}	0.01	0.03	0.15	0.3	0.35
Path length [m]	787	780	795	814	822
	SoL				
λ_{tc}	0.35	0.3	0.17	0.03	0.01
λ_{ic}	0.01	0.03	0.15	0.3	0.35
Path length [m]	578	559	550	587	601

Table 5.5: *Comparison between the centralized TSGA and NNB planners.*

	Powerplant		**SoL**	
Number of UAVs	3	5	3	5
Path length gain [%]	23.90	21.00	1.64	1.00
Completion time gain [%]	7.32	7.06	21.80	10.00

largely unaffected. We compared our single-robot algorithm with (Schmid et al. 2020) (denoted by "S", for short), by adapting our code to the simulator developed by the authors. We kept their default parameters and configurations, and selected the City-Building (**CB**) environment (see the third column of Table 5.2). As we can notice in Table 5.3, our method works slightly better than (Schmid et al. 2020) in terms of completion time, path length, surface coverage, and accuracy. The algorithms described in (Song and Jo 2017, Song and Jo 2018) exhibit a similar completion time with **Powerplant**. The deviation is more pronounced with **SoL**: in fact, the algorithm in (Song and Jo 2017) takes twice as long to finish the exploration. As the number of robots grows, the execution time decreases: in fact, with **Powerplant** (see the 4th row of Table 5.3), the distributed (centralized) architecture with 3 UAVs is 70.0% (68.8%) faster, compared to the single-robot case. With 5 UAVs, the distributed (centralized) architecture is 81.5% (80.8%) faster, compared to a single quadrotor. Similarly, with **SoL**, TSGA achieves the task 72.2% (81.8%) faster with a fleet of 3 UAVs (5 UAVs), compared to the single-robot case. The distance traveled per UAV is shorter than that of a single UAV, but the total path length is larger, for any team of aerial vehicles.

The centralized TSGA also guarantees a shorter completion time and shorter distances compared to the classical NNB planner (see Table 5.5), even with **SoL**. In fact, the profile of the statue and the presence of numerous contiguous ISEs should be more favorable, in principle, to fast local planners (for more details, the reader is referred to (Hardouin, Moras, Morbidi, Marzat and Mouaddib 2020, Sect. V)). With **Powerplant**, the centralized architecture ensures that all the incomplete reachable regions are ultimately covered (surface coverage ranges between 90.4% and 91.5%). The overall reconstruction accuracy is better with the distributed mapping algorithm. Coverage improves as well, reaching 95.9% with 3 robots and 95.6% with 5 robots. Since the performance of MarchingCubes is dictated by the voxel size, the reconstructed mesh is more accurate, if the resolution is high. However, if the environment to explore is large, a high resolution entails resource-intensive mapping and ISE-extraction steps, which ultimately make the whole reconstruction process prohibitively expensive. Therefore, a balance between computational efficiency and reconstruction accuracy, should be found.

In summary, our numerical experiments show that the two multi-robot architectures are successful in scanning the 3D environments, covering upwards of 90% of their sur-

face, even in the presence of noise in the depth measurements. In the next section, we will extend our analysis and study the accuracy and robustness of the centralized and distributed architectures deployed on mobile robots in real-life conditions.

5.3.8 Real-world experiments

In this section, which is organized as Sect. 5.3.7, the results of our hardware experiments are presented and discussed. To cope with variable environmental conditions (e.g. lighting change during the day), and slight differences in robot configuration (camera calibration, level of charge of the battery, etc.), a statistical analysis over multiple trials has been carried out.

Robots: The experiments have been conducted with a team of 4 identical Wifibots[9]. Each robot is equipped with an Intel NUC7i7BNH computer, a stereo rig with two IDS UI-1241-LE cameras (baseline $B = 26$ cm), and a Wi-Fi system to communicate with a ground station. An HQ camera (IDS UI-3252-LE) is also installed on each robot to generate a GT map (for more details on the software/hardware specifications, see (Hardouin 2022)). The other parameters used in our experiments are reported in Table 5.6. Only minor changes have been made to adapt our ROS-based system to the real sensors and physical constraints of the Wifibots. In particular, the code that generated the emulated depth maps and odometry, has been replaced with validated software modules: the depth maps are computed with the ELAS algorithm (Geiger et al. 2010), and the pose of the robots is estimated with the vision-based eVO algorithm (Sanfourche et al. 2013).

[9] `https://wifibot.com`

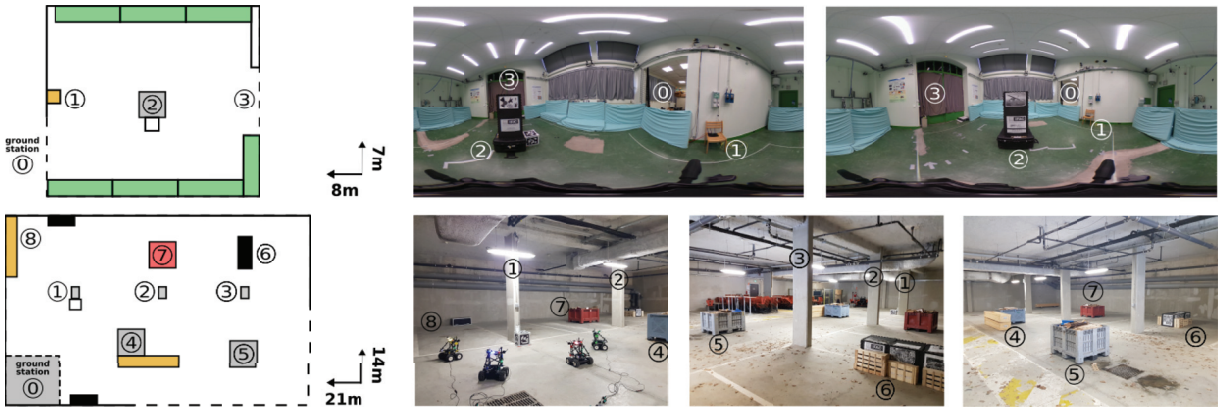


Figure 5.13: Real-world experiments: *[top]* ***Test arena****, [bottom]* ***Parking lot****. [left] 2D maps, and [center, right] photos of the two environments, including two panoramic views of* ***Test arena****. The circled numbers indicate obstacles or areas of interest.*

This latter algorithm does not address the loop closure problem: hence, a localization drift, proportional to the distance traveled may occur. In the centralized architecture, the map is updated upon receipt of a new depth map (at around 1 Hz). Instead, in the distributed architecture, a new patch is stored and broadcast to the robots, every time that 5 depth maps have been integrated into the current TSDF map. The mapping and planning modules of the centralized and distributed architectures are identical to those presented in Sect. 5.3.7.

Environments: Two different indoor environments have been considered in our experiments. An $8 \times 7 \times 2$ m^3 **Test arena**, consisting of a central obstacle surrounded by four walls covered by mattresses (green in Fig. 5.13-[top-left]), and an underground $21 \times 14 \times 2$ m^3 **Parking lot**, containing several obstacles at ground level. The 2D maps and photos of these environments are shown in Fig. 5.13.

Metrics: The same metrics as in the numerical experiments, have been considered (please refer to Sect. 5.3.7). Data exchange has been monitored during the experiments: in the single-robot and centralized multi-robot algorithms, it pertains to depth maps, odometry and path messages transmitted between the ground station and the vehicles, whereas in the distributed algorithm, to patches, odometry, and paths exchanged by the robots.

Table 5.6: *Parameters used in the real-world experiments.*

Parameter	**Test arena**	**Parking lot**
Voxel resolution $r_{\mathbf{v}}$ [m]	0.1	0.2
Threshold W_{th}	0.3	0.3
$e_{\max}$ [m]	0.0866	0.1732
Camera range [m]	[0.3, 5]	[0.3, 5]
Camera FoV [deg.] (H, V)	90×60	90×60
Collision radius [m]	0.55	0.55
Robot nominal speed [m/s]	0.5	0.5
δ_{pose} [m]	1.3	1.3
d_ν [m]	2.0	2.0
$d_\nu^{\max}$ [m]	3.5	3.5
Penalty term λ_{tc}	0.5	0.7
Penalty term λ_{ic}	0.01	0.01

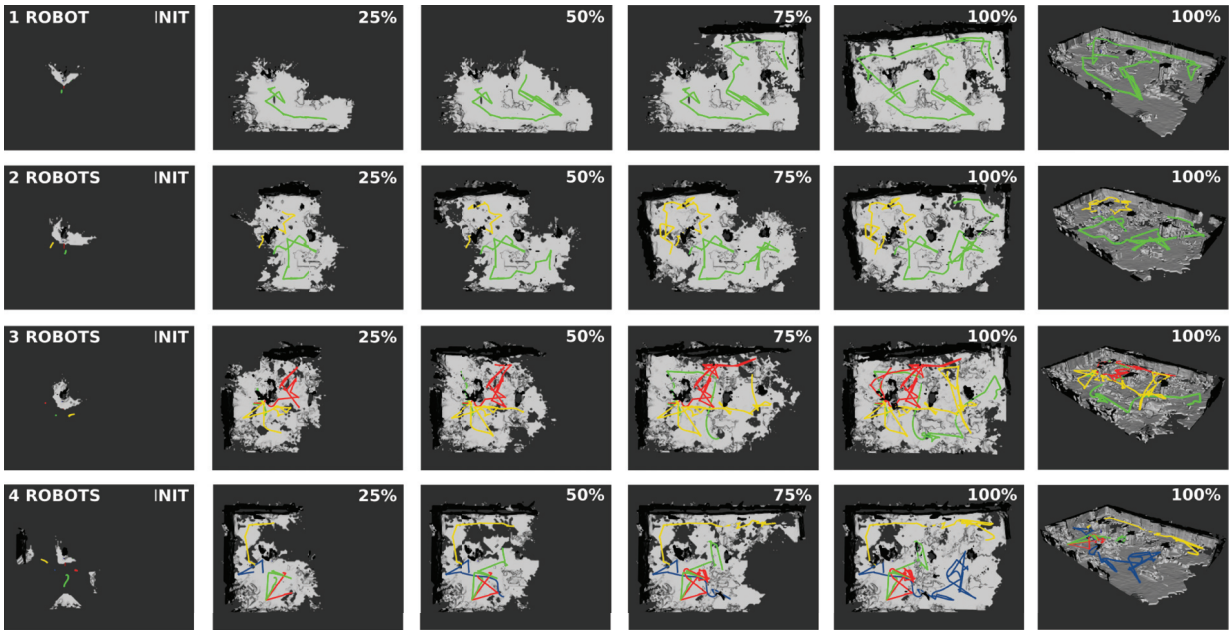


Figure 5.14: Real-world experiments: ***Parking lot***, *centralized architecture. From left to right, time progression expressed in percentage of final 3D reconstruction (top view). In the rows, the number of robots varies between 1 and 4. The last column reports an isometric view of reconstructed mesh and the exploration paths of the robots. The ceiling of the parking lot has been removed to provide visibility of the interior.*

Results: Table 5.7 summarizes our experimental results. For **Test arena**, the reconstruction has been performed with a single robot, and a team of 2 robots for the centralized and distributed architectures. On the other hand, for **Parking lot**, we considered a single robot and teams of 2, 3 and 4 robots for the centralized architecture, and a team of 2 robots for the distributed architecture. For each environment/team, we carried out 5 trials with identical initial positions and orientations for the robots. Fig. 5.14 shows different snapshots of the 3D reconstruction of **Parking lot**, obtained with the centralized architecture. The GT, reconstructed mesh, and signed distance errors for the distributed architecture, are reported in Fig. 5.15 (left, center, and right, respectively). The single-robot case is considered as a reference, in both environments. From Table 5.7, we can see that as the number of robots grows, the completion time decreases while the cumulative path length (at the team level), increases. Nevertheless, taken individually, the distance traveled by each robot, decreases. For example, in **Parking lot**, a team of 4 robots allows to reduce the completion time by 31.1%, compared to the single-robot case. Doubling the number of robots, the gain in completion time is 6.25%, in **Test arena**. The distributed architecture works just as well as the centralized one in

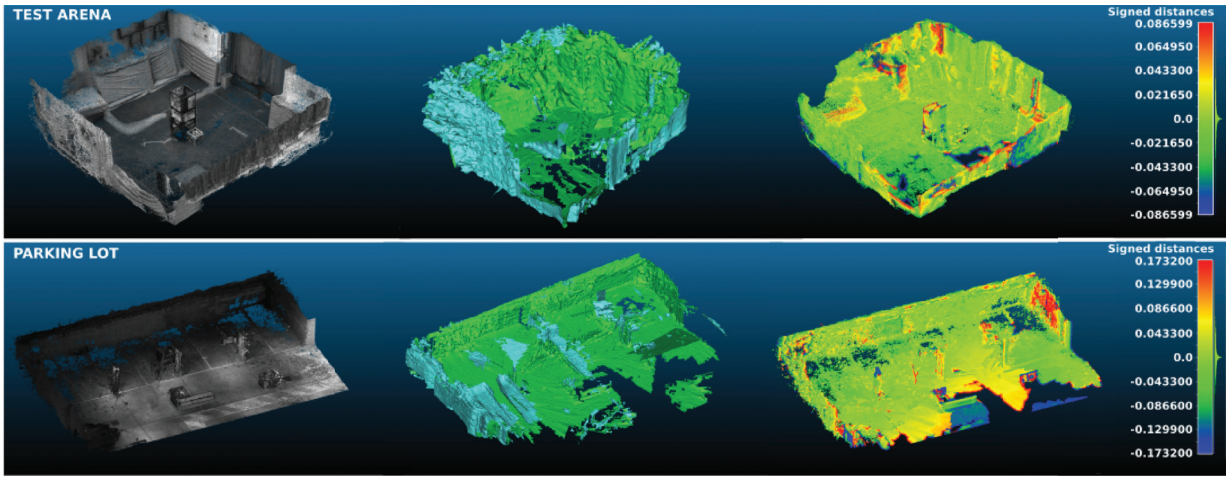


Figure 5.15: Real-world experiments*: [top]* ***Test arena****, [bottom]* ***Parking lot****. [left] GT point cloud. [center] Reconstructed mesh using the distributed architecture with 2 robots. [right] Signed distance error: the color bar shows the error in meters with respect to the GT, computed with CloudCompare's M3C2 plugin.*

terms of distance traveled and time to completion. However, the volume of data exchanged by the robots using the distributed architecture, in "nominal operation", is smaller (3.031 GB vs 6.029 GB for the centralized case). In fact, during our experiments, additional information (mainly meshes) was transmitted on the communication network, to monitor the progression of the robots. This resulted in an 87.5% increase in the volume of data exchanged (24.111 GB), which tended to saturate the network. To circumvent this problem, simpler spatial representations (such as, TSDF maps) could be used for visualization, which is a priority area for future research. As an indication on the scalability of the proposed algorithms, the last row of Table 5.7, also reports the average bandwidth usage.

In the two environments, the robots were left free to cover the entire accessible area. In the single-robot case, the reconstructed mesh covered 89.1% of the surface of **Parking lot** for a voxel resolution $r_{\mathbf{v}} = 20$ cm and an admissible error $e_{\max} = 17.32$ cm. For **Test arena**, instead, the surface coverage was 85.6% with $r_{\mathbf{v}} = 10$ cm and $e_{\max} = 8.66$ cm. Differently from the numerical experiments (cf. Sect. 5.3.7), as the number of robots increases, surface coverage decreases, until reaching the 73% level with 4 robots (**Parking lot**), and the 80.3% level with 2 robots (**Test arena**). In fact, a depth map depends on the pose estimated by a visual odometry algorithm, which is prone to drift. The estimation error due to the drift, has an impact on the TSDF volume when the depth map is integrated. Hence, accumulation of errors is experienced, as the number of robots increases: the overall mesh accuracy decreases, more outliers need to be pruned, and

Table 5.7: *Results of the real-world experiments (statistics over 5 trials).*

	Test arena			**Parking lot**				
Number of robots	1	2		1	2		3	4
Architecture	–	Centr.	Distr.	–	Centr.	Distr.	Centr.	Centr.
Cum. path length [m]	20.6	27.0	24.2	106.0	113.6	107.1	134.2	161.0
Comp. time [min.]	2′53″	2′45″	2′37″	13′56″	11′06″	10′41″	10′14″	9′36″
Time gain [%] w.r.t. 1 robot	–	6.25	9.25	–	20.30	23.30	26.60	31.10
Surface coverage [%]	85.6	80.3	91.0	89.1	81.1	88.4	76.4	73.0
M3C2 avg. error [cm]	0.27	0.40	0.12	0.01	−0.48	0.07	−0.18	0.61
M3C2 std. error [cm]	3.99	4.16	3.33	8.37	8.29	5.85	8.93	8.97
RMSE [cm]	4.00	4.18	3.33	8.37	8.30	5.85	8.94	8.99
Data exchanged [GB]	0.750	0.888	0.501	4.849	6.029	3.031	10.715	13.271
Bandwidth [Mb/s]	35.51	44.09	26.14	47.52	74.16	38.74	142.96	188.74

less free space is covered. The RMSE ranges between 8.37 cm and 8.99 cm for **Parking lot**, and between 4 cm and 4.18 cm for **Test arena**. The distributed mapping outperform the centralized mapping in terms of surface coverage (88.4% vs 80.3%). This depends on the superior accuracy of the distributed mapping algorithm compared to the centralized one, for a given resolution. It is also worth mentioning that the fusion policy in the distributed case (cf. Sect. 5.3.5), superimposes the TSDF patches with an integration rule which prioritizes those which have maximum weights, thus mitigating the impact of depth noise. Fig. 5.14 and Fig. 5.15 show that the centralized and distributed algorithms provide accurate 3D reconstruction, with a decent surface coverage despite the odometry drift.

Finally, the difference in speed-up observed in the numerical and real-world experiments with an increasing number of robots, can be ascribed to the different specifications of the robotic platforms and set-ups considered. In fact, the motion of the UAVs in the 3D space, is far less constrained than that of the Wifibots on the ground.

5.3.9 Conclusions and future work

In this section, we have introduced a new multi-robot system for surface inspection of large-scale unknown 3D environments. The proposed approach relies on Next-Best-View planning to address the coordinated inspection problem, and directly exploits the 3D surface representation. Centralized and distributed architectures (TSGA and dist-TSGA) have been developed and analyzed in detail. To illustrate and validate our al-

gorithms, we performed extensive numerical simulations with quadrotor UAVs using two challenging ROS-Gazebo 3D models, and real-world experiments with up to four wheeled robots in two indoor environments. The simulation results indicate that our solution is competitive with the state-of-the-art in terms of navigation and reconstruction accuracy, and that it can be easily tailored to different software stacks. The experimental results complement the tests with synthetic data, and confirm that our approach is versatile with different types of robots and environments, and effective in generating accurate meshes in real-time.

The experiments also provide evidence that odometry drift, via uncertainty propagation, has a non-negligible impact on the overall reconstruction process. To circumvent this problem, one could envisage a hybrid approach which takes advantage of a SLAM landmark map in the planning module. This would endow the robots with loop-closure capabilities, thus minimizing the effect of drift and ultimately boosting the map accuracy. In future works, we also place a premium on a more efficient mechanism to monitor the progression of the robots, and we plan to perform real-world experiments with larger teams of heterogeneous agents, in obstacle-rich dynamic environments.

Chapter 5

Part II: Extensions of the consensus protocol

5.4 Deformed consensus protocol

In this section, we study a generalization of the standard continuous-time consensus protocol, obtained by replacing the Laplacian matrix of the communication graph with the so-called *deformed Laplacian*. The deformed Laplacian is a second-degree matrix polynomial in the real variable s which reduces to the standard Laplacian for s equal to unity. The stability properties of the ensuing *deformed consensus protocol* are studied in terms of parameter s for some special families of undirected and directed graphs, and for arbitrary graph topologies by leveraging the spectral theory of quadratic eigenvalue problems. Examples and simulation results are provided to illustrate our theoretical findings.

5.4.1 Introduction

Consensus theory originated from the work of (Tsitsiklis 1984), (Jadbabaie et al. 2003) and (Olfati-Saber and Murray 2004), in which the consensus problem was formulated for the first time in system-theoretic terms (cf. Sect. 3.2). A very rich literature emanated from these seminal contributions in recent years. In particular, numerous extensions to the prototypal consensus protocol in (Olfati-Saber and Murray 2004) have been proposed: among them, we mention here the case of time-varying networks (Ren and Beard 2005), networks with delayed (Olfati-Saber and Murray 2004) or quantized/noisy communication and link failure (Frasca et al. 2009, Kar and Moura 2009), random networks (Tahbaz-Salehi and Jadbabaie 2008), and more recently, the emergence of dynamic average consensus estimators (Kia et al. 2019) and open systems (Franceschelli and Frasca 2020). We follow here this active line of research and propose an original extension to the basic continuous-time consensus protocol in (Olfati-Saber and Murray 2004), that exhibits a rich variety of behaviors and whose flexibility makes it ideal for a broad range of mobile robotic applications (e.g., for target enclosing or formation control). The new protocol, termed *deformed consensus protocol*, relies on the so-called *deformed Laplacian matrix*, a second-degree matrix polynomial in the real variable s, which extends the standard Laplacian matrix and reduces to it for s equal to unity. Parameter s has a dramatic effect on the stability properties of the deformed consensus protocol, and it can be potentially used by a supervisor to dynamically modify the behavior of the network and trigger different desired agents' responses according to time-varying external events. The stability properties of the proposed protocol are studied in terms of parameter s for some special families of undirected and directed graphs for which the eigenvalues and eigenvectors of the deformed Laplacian can be computed in closed form. The analysis is extended to arbitrary graph topologies by exploiting the spectral theory of quadratic eigenvalue problems (Tisseur and Meerbergen 2001).

Beside the aforementioned promising applications, we believe that the study of the proposed protocol is of value for shedding new light on known results (Olfati-Saber et al. 2007), and for gaining a more general perspective on consensus algorithms.

The rest of this section is organized as follows. In Sect. 5.4.2, the problem formulation and the main results are presented. In Sect. 5.4.3, the theory is illustrated via numerical simulations, and finally, in Sect. 5.4.4 the main contributions are summarized and possible directions for future research are outlined.

5.4.2 Main results

Problem formulation

As seen in Sect. 3.2, if the static undirected communication graph $\mathcal{G}$ is connected, each component of the state vector $\mathbf{x} = [x_1, \ldots, x_n]^T \in \mathbb{R}^n$ of the linear time-invariant system,

$$\dot{\mathbf{x}}(t) \;=\; -\mathbf{L}\,\mathbf{x}(t), \tag{5.35}$$

asymptotically converges to the average of the initial states $x_1(0), \ldots, x_n(0)$,

$$\lim_{t\to\infty}\; x_i(t) = \frac{1}{n}\sum_{i\,=\,1}^{n} x_i(0) = \frac{1}{n}\mathbf{x}_0^T\mathbb{1},$$

where $\mathbf{x}_0 = [x_1(0),\, \ldots,\, x_n(0)]^T$, i.e., *average consensus* is achieved. Let us now consider the following generalization of the Laplacian $\mathbf{L}$ (cf. Sect. 3.1):

5.5. DEFINITION. *The* deformed Laplacian *of the graph $\mathcal{G}$ is an $n \times n$ matrix defined as,*

$$\mathbf{\Delta}(s) \;=\; (\mathbf{D} - \mathbf{I}_n)\, s^2 \;-\; \mathbf{A}\, s \;+\; \mathbf{I}_n,$$

where s is a real parameter. ◇

Inspired by (5.35), we will study the stability properties of linear system,

$$\dot{\mathbf{x}}(t) = -\mathbf{\Delta}(s)\,\mathbf{x}(t), \tag{5.36}$$

in terms of the real parameter s, assuming that the graph $\mathcal{G}$ is *connected*. We will refer to (5.36), as the *deformed consensus protocol*.

5.5. REMARK. *Note that parameter s in the deformed Laplacian $\mathbf{\Delta}(s)$ can be regarded as a control input and it can be exploited to* dynamically *modify the behavior of system (5.36). This may be useful when the vertices of the graph are mobile robots and a human supervisor is interested in changing the collective behavior of the team over time, e.g., by switching from a marginally- to an asymptotically-stable equilibrium point of system (5.36), or between two*

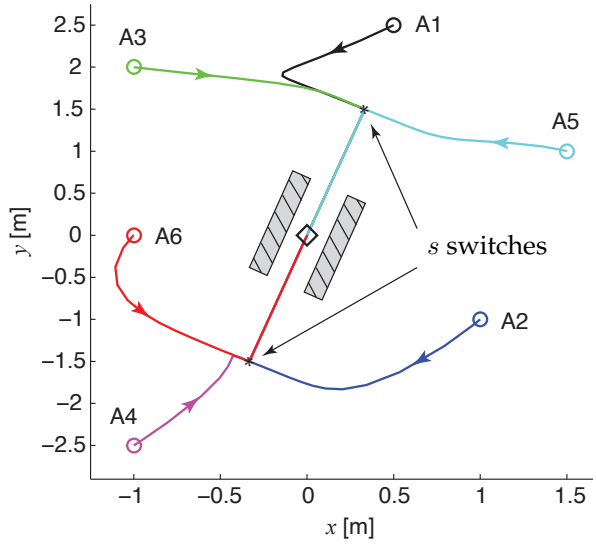


Figure 5.16: Illustrative example*: The 6 agents rendezvous at the origin while avoiding the two obstacles (gray rectangles): this is made possible by switching s from -1 to 0 (the initial position of the vehicles is marked with a circle and the final position with a diamond). The communication graph is the path graph P_6.*

marginally-stable equilibria. The former case is illustrated in the example in Fig. 5.16, where the communication graph is the path graph P_6: in order to make the 6 single-integrator agents rendezvous at the origin while avoiding the two gray obstacles, the supervisor can initially set $s = -1$ and then switch to $s = 0$ (cf. Proposition 5.1 below). ◇

Note that since $\boldsymbol{\Delta}(1) = \mathbf{L}$, we will always achieve average consensus for $s = 1$. Moreover, since $\boldsymbol{\Delta}(s)$ is real symmetric, all the eigenvalues of $\boldsymbol{\Delta}(s)$ are *real*, and the deformed Laplacian admits the spectral decomposition $\boldsymbol{\Delta}(s) = \mathbf{U}(s)\boldsymbol{\Lambda}(s)\mathbf{U}^T(s)$, where $\mathbf{U}(s) = [\mathbf{u}_1(s), \mathbf{u}_2(s), \ldots, \mathbf{u}_n(s)] \in \mathbb{R}^{n\times n}$ is the matrix consisting of normalized and mutually orthogonal eigenvectors of $\boldsymbol{\Delta}(s)$ and $\boldsymbol{\Lambda}(s) = \mathrm{diag}(\lambda_1(\boldsymbol{\Delta}(s)), \ldots, \lambda_n(\boldsymbol{\Delta}(s)))$. The solution of (5.36), can thus be written as,

$$\mathbf{x}(t) = \sum_{i=1}^{n} e^{-\lambda_i(\boldsymbol{\Delta}(s))t} \left(\mathbf{u}_i^T(s)\,\mathbf{x}_0\right) \mathbf{u}_i(s). \tag{5.37}$$

In the next subsection, we will focus on some special families of undirected graphs for which the eigenvalues and eigenvectors of $\boldsymbol{\Delta}(s)$ can be computed in closed form, and thus the stability properties of system (5.36) can be easily deduced from (5.37). We will then address the more challenging case of undirected graphs with arbitrary topology. Finally, we will discuss the case of directed communication graphs.

Stability conditions for special families of graphs

This section presents a sequence of eight propositions which provide stability conditions for system (5.36), in the case of path, cycle, wheel, m-cube, Petersen, complete, complete bipartite and star graphs (see Sect. 3.1 and refer to (Godsil and Royle 2001) for a precise definition of these graphs).The proofs are omitted and they can be found in (Morbidi 2013b). In the following, $\mathbb{k} \triangleq [-1,\, 1, -1,\, 1, \ldots, (-1)^{n-1}, (-1)^n]^T \in \mathbb{R}^n$ and $\lfloor \cdot \rfloor$ will denote the floor function

5.1. PROPOSITION (PATH GRAPH P_n). *For the path graph P_n with $n \geq 2$ vertices, we have:*

- *For $|s| < 1$, system (5.36) is asymptotically stable.*
- *For $|s| > 1$, system (5.36) is unstable.*
- *For $s = -1$, system (5.36) is marginally stable. In this case, it is possible to identify two groups of $n/2$ vertices (if n is even), or one group of $\lfloor n/2 \rfloor$ vertices and one of $\lfloor n/2 \rfloor + 1$ vertices (if n is odd). The states associated to the vertices in one group asymptotically converge to $\frac{1}{n}\,\mathbf{x}_0^T\mathbb{k}$ and the states associated to the vertices in the other group converge to $-\frac{1}{n}\,\mathbf{x}_0^T\mathbb{k}$.* ■

5.2. PROPOSITION (CYCLE GRAPH C_n). *For the cycle graph C_n with $n > 2$ vertices, we have that:*

- *If n is even:*
  - *For all $s \in \mathbb{R} \setminus \{-1,\, 1\}$, system (5.36) is asymptotically stable.*
  - *For $s = -1$, system (5.36) is marginally stable. In this case, the states associated to $n/2$ vertices asymptotically converge to $\frac{1}{n}\,\mathbf{x}_0^T\,\mathbb{k}$ and the states associated to the other $n/2$ vertices converge to $-\frac{1}{n}\,\mathbf{x}_0^T\,\mathbb{k}$.*
- *If n is odd, system (5.36) is asymptotically stable for all $s \in \mathbb{R} \setminus \{\,1\}$.* ■

5.3. PROPOSITION (WHEEL GRAPH W_n). *Consider a wheel graph W_n with $n > 3$ vertices where vertex 1 is the center of the wheel, and let μ be the non-unitary root of,*

$$-\frac{n}{2}\,s^2 \,+\, s \,+\, \frac{\sqrt{((n-4)\,s+2)^2+4(n-1)}}{2}\,s \,-\, 1.$$

μ monotonically decreases from $1/2$ (for $n = 4$) to 0 (for $n = \infty$). We have that:

- *For $s > 1$ or $s < \mu$, system (5.36) is asymptotically stable.*
- *For $s \in (\mu,\, 1)$, system (5.36) is unstable.*

- *For $s = \mu$, system (5.36) is marginally stable. If $n = 4$ average consensus is achieved. If $n > 4$ the state associated to vertex 1 asymptotically converges to $\mathbf{x}_0^T[\alpha^2,\, \alpha\beta,\, \ldots,\, \alpha\beta]^T$, and the states associated to the other $n-1$ vertices converge to $\mathbf{x}_0^T[\alpha\beta,\, \beta^2,\, \ldots,\, \beta^2]^T$, where $[\alpha,\, \beta,\, \ldots,\, \beta]^T$, $\alpha,\, \beta \in \mathbb{R}$, is the unit-norm eigenvector associated to the zero eigenvalue of $-\boldsymbol{\Delta}(\mu)$.* ■

5.4. PROPOSITION (m-CUBE Q_m). *For the m-cube graph Q_m with $n = 2^m > 4$ vertices, we have that:*

- *For $|s| > 1$ or $|s| < \frac{1}{m-1}$, system (5.36) is asymptotically stable.*
- *For $s \in \big(-1,\, -\frac{1}{m-1}\big)$ or $s \in \big(\frac{1}{m-1},\, 1\big)$, system (5.36) is unstable.*
- *For $s = \frac{1}{m-1}$, average consensus is achieved. The convergence rate to $\frac{1}{n}\,\mathbf{x}_0^T\mathbb{1}$ is slower for $s = \frac{1}{m-1}$ than for $s = 1$.*
- *For $s \in \{-1,\, -\frac{1}{m-1}\}$, system (5.36) is marginally stable. In this case, the states associated to $n/2$ vertices asymptotically converge to $\frac{1}{\sqrt{n}}\,\mathbf{x}_0^T\mathbf{u}_1$, while the states associated to the other $n/2$ vertices converge to $-\frac{1}{\sqrt{n}}\,\mathbf{x}_0^T\mathbf{u}_1$, where $\mathbf{u}_1$ is the unit-norm eigenvector associated to the zero eigenvalue of $-\boldsymbol{\Delta}(-1)$ or $-\boldsymbol{\Delta}(-\frac{1}{m-1})$.* ■

5.5. PROPOSITION (PETERSEN GRAPH). *For the Petersen graph, we have that:*

- *For $s > 1$ or $s < 1/2$, system (5.36) is asymptotically stable.*
- *For $s \in (1/2,\, 1)$, system (5.36) is unstable.*
- *For $s = 1/2$, average consensus is achieved. The convergence rate to $\frac{1}{10}\,\mathbf{x}_0^T\mathbb{1}_{10}$ is slower for $s = 1/2$ than for $s = 1$.* ■

5.6. PROPOSITION (COMPLETE GRAPH K_n). *For the complete graph K_n with $n > 2$ vertices, we have that:*

- *For $s > 1$ or $s < \frac{1}{n-2}$, system (5.36) is asymptotically stable.*
- *For $s \in \big(\frac{1}{n-2},\, 1\big)$, system (5.36) is unstable.*
- *For $s = \frac{1}{n-2}$, average consensus is achieved. The convergence rate to $\frac{1}{n}\,\mathbf{x}_0^T\mathbb{1}$ is slower for $s = \frac{1}{n-2}$ than for $s = 1$.* ■

5.7. PROPOSITION (COMPLETE BIPARTITE GRAPH $K_{m,n}$). *For the complete bipartite graph $K_{m,n} = (V_1 \cup V_2, E)$, where $|V_1| = m$, $|V_2| = n$ with $m, n \geq 2$, we have that:*

- *For $|s| > 1$ or $|s| < \frac{1}{\sqrt{(m-1)(n-1)}}$, system (5.36) is asymptotically stable.*

- *For* $s \in \big(-1, -\frac{1}{\sqrt{(m-1)(n-1)}}\big)$ *or* $s \in \big(\frac{1}{\sqrt{(m-1)(n-1)}}, 1\big)$*, system (5.36) is unstable.*
- *For* $s \in \big\{-1, \pm\frac{1}{\sqrt{(m-1)(n-1)}}\big\}$*, system (5.36) is marginally stable. In particular, given* $\mathbf{x}_0 \in \mathbb{R}^{m+n}$*:*
  - *If* $m \neq n$*: for* $s = -1$*, the states associated to the vertices in* V_1 *asymptotically converge to* $\frac{1}{m+n}\,\mathbf{x}_0^T[\mathbb{1}_m^T, -\mathbb{1}_n^T]^T$ *and the states associated to the vertices in* V_2 *converge to* $-\frac{1}{m+n}\,\mathbf{x}_0^T[\mathbb{1}_m^T, -\mathbb{1}_n^T]^T$*.*
    For $s = \pm\frac{1}{\sqrt{(m-1)(n-1)}}$*, the states associated to the vertices in* V_1 *and* V_2 *asymptotically converge to one of the two different values taken by the components of the vector* $(\mathbf{u}_1\,\mathbf{u}_1^T)\,\mathbf{x}_0$*, where* $\mathbf{u}_1$ *is the unit-norm eigenvector associated to the zero eigenvalue of* $-\boldsymbol{\Delta}\big(\pm\frac{1}{\sqrt{(m-1)(n-1)}}\big)$*.*
  - *If* $m = n$*: for* $s = \frac{1}{n-1}$ *average consensus is achieved, and the convergence rate to* $\frac{1}{2n}\,\mathbf{x}_0^T\mathbb{1}_{2n}$ *is slower for* $s = \frac{1}{n-1}$ *than for* $s = 1$*. For* $s \in \big\{-1, -\frac{1}{n-1}\big\}$ *the states associated to the vertices in* V_1 *asymptotically converge to* $\frac{1}{2n}\,\mathbf{x}_0^T[\mathbb{1}_n^T, -\mathbb{1}_n^T]^T$ *and the states associated to the vertices in* V_2 *converge to* $-\frac{1}{2n}\,\mathbf{x}_0^T[\mathbb{1}_n^T, -\mathbb{1}_n^T]^T$*.* ■

From Proposition 5.7, we deduce the following result (note that the star graph is a complete bipartite graph with $m = 1$):

5.8. PROPOSITION (STAR GRAPH $K_{1,n}$)*. For the star graph* $K_{1,n}$ *with* $n \geq 3$*, where vertex 1 is the center of the star, we have that:*

- *For* $|s| < 1$*, system (5.36) is asymptotically stable.*
- *For* $|s| > 1$*, system (5.36) is unstable.*
- *For* $s = -1$*, system (5.36) is marginally stable. In this case, the state associated to vertex 1 asymptotically converges to* $\frac{1}{n+1}\,\mathbf{x}_0^T[1, -\mathbb{1}_n^T]^T$ *and the states associated to the other* n *vertices converge to* $-\frac{1}{n+1}\,\mathbf{x}_0^T[1, -\mathbb{1}_n^T]^T$*.* ■

For the reader's convenience, all the results found in this section are summarized in Table 5.8.

Stability conditions for graphs of arbitrary topology

In order to extend the analysis of the previous subsection to arbitrary undirected graphs, we briefly review here the spectral theory of *quadratic eigenvalues problems* (QEPs) (Tisseur and Meerbergen 2001, Sect. 3). Let $\mathbf{P}(\lambda) = \mathbf{B}_2\,\lambda^2 + \mathbf{B}_1\,\lambda + \mathbf{B}_0$ be an $n \times n$ *matrix polynomial* of degree 2, where $\mathbf{B}_2$, $\mathbf{B}_1$ and $\mathbf{B}_0$ are $n \times n$ complex matrices.

Graph name	**Asymptotic stability for:**
Path graph, P_n, $n \geq 2$	$\lvert s\rvert < 1$
Cycle graph, C_n, $n > 2$, n even	$\forall\, s \in \mathbb{R} \setminus \{-1,\, 1\}$
Cycle graph, C_n, $n > 2$, n odd	$\forall\, s \in \mathbb{R} \setminus \{1\}$
Wheel graph, W_n, $n > 3$	$s > 1$ or $s < \mu$
Cube graph, Q_m, $n = 2^m > 4$	$\lvert s\rvert > 1$ or $\lvert s\rvert < \frac{1}{m-1}$
Petersen graph	$s > 1$ or $s < \frac{1}{2}$
Complete graph, K_n, $n > 2$	$s > 1$ or $s < \frac{1}{n-2}$
Complete bipartite graph, $K_{m,n}$, $m, n \geq 2$	$\lvert s\rvert > 1$ or $\lvert s\rvert < \frac{1}{\sqrt{(m-1)(n-1)}}$
Star graph, $K_{1,n}$, $n \geq 3$	$\lvert s\rvert < 1$

Graph name	**Marginal stability for:**
Path graph, P_n, $n \geq 2$	$s = -1$ (2 groups of vertices)
Cycle graph, C_n, $n > 2$, n even	$s = -1$ (2 groups of vertices)
Cycle graph, C_n, $n > 2$, n odd	
Wheel graph, W_n, $n > 3$	$s = \mu$ (2 groups of vertices for $n > 4$)
Cube graph, Q_m, $n = 2^m > 4$	$s \in \{-1, -\frac{1}{m-1}\}$ (2 groups of vertices) $s = \frac{1}{m-1}$ (average consensus)
Petersen graph	$s = \frac{1}{2}$ (average consensus)
Complete graph, K_n, $n > 2$	$s = \frac{1}{n-2}$ (average consensus)
Complete bipartite graph, $K_{m,n}$, $m, n \geq 2$	$s = \frac{1}{\sqrt{(m-1)(n-1)}}$ (if $m = n$, average consensus) (if $m \neq n$, 2 groups of vertices)
Star graph, $K_{1,n}$, $n \geq 3$	$s = -1$ (2 groups of vertices)

Table 5.8: *Summary of the stability properties of the deformed consensus protocol (5.36), for some special families of undirected graphs. Average consensus is achieved in all cases for* $s = 1$.

5.6. DEFINITION (SPECTRUM OF $\mathbf{P}(\lambda)$). *The* spectrum *of* $\mathbf{P}(\lambda)$*, denoted by* $\Sigma(\mathbf{P})$*, is defined as* $\Sigma(\mathbf{P}) = \{\lambda \in \mathbb{C} : \det(\mathbf{P}(\lambda)) = 0\}$*, i.e., it is the set of eigenvalues of* $\mathbf{P}(\lambda)$. ◇

5.7. DEFINITION (REGULAR $\mathbf{P}(\lambda)$). *The matrix $\mathbf{P}(\lambda)$ is called* regular *when $\det(\mathbf{P}(\lambda))$ is not identically zero for all values of λ, and* nonregular *otherwise.* ◇

Note that $\det(\mathbf{P}(\lambda)) = \det(\mathbf{B}_2)\lambda^{2n}+$ lower-order terms, so when $\mathbf{B}_2$ is nonsingular, $\mathbf{P}(\lambda)$ is regular and has $2n$ *finite* eigenvalues. When $\mathbf{B}_2$ is singular, the degree of $\det(\mathbf{P}(\lambda))$ is $r < 2n$ and $\mathbf{P}(\lambda)$ has r finite eigenvalues and $2n - r$ infinite eigenvalues. The *algebraic multiplicity* of an eigenvalue λ_0 is the order of the corresponding zero in $\det(\mathbf{P}(\lambda))$, while the *geometric multiplicity* of λ_0 is the dimension of $\ker(\mathbf{P}(\lambda_0))$.

5.2. PROBLEM (QUADRATIC EIGENVALUE PROBLEM, QEP). *The QEP consists of finding scalars λ and nonzero vectors $\mathbf{z}$, $\mathbf{y}$, satisfying $\mathbf{P}(\lambda)\,\mathbf{z} = \mathbf{0}$, $\mathbf{y}^*\,\mathbf{P}(\lambda) = \mathbf{0}$ where $\mathbf{z}$, $\mathbf{y} \in \mathbb{C}^n$ are respectively the right and left eigenvector corresponding to the eigenvalue $\lambda \in \mathbb{C}$, and $\mathbf{y}^*$ is the conjugate transpose of $\mathbf{y}$.* ◇

A QEP has $2n$ eigenvalues (finite or infinite) with up to $2n$ right and $2n$ left eigenvectors. If a regular $\mathbf{P}(\lambda)$ has $2n$ distinct eigenvalues, then there exists a set of n linearly independent eigenvectors.

5.1. PROPERTY (SPECTRAL PROPERTY OF $\mathbf{P}(\lambda)$). *If matrices $\mathbf{B}_2$, $\mathbf{B}_1$, $\mathbf{B}_0$ are* real symmetric, *the eigenvalues of $\mathbf{P}(\lambda)$ are either real or occur in complex-conjugate pairs, and the sets of left and right eigenvectors coincide.* ◇

By leveraging the previous facts (note that according to Def. 5.7, $\mathbf{\Delta}(\lambda)$ is a regular matrix since we always have $\det(\mathbf{\Delta}(\lambda)) \neq 0$ for $\lambda = 0$), we deduce the following property of the deformed consensus protocol (5.36).

5.9. PROPOSITION. *The* finite real eigenvalues *λ of the QEP,*

$$((\mathbf{I}_n - \mathbf{D})\,\lambda^2 + \mathbf{A}\,\lambda - \mathbf{I}_n)\,\mathbf{z} \;=\; \mathbf{0}, \tag{5.38}$$

are the values of s for which system (5.36) is marginally stable. Moreover, if λ is one of these eigenvalues with geometric multiplicity one, and $\overline{\mathbf{z}} = \mathbf{z}/\|\mathbf{z}\|$ is the associated unit-norm eigenvector, we have that:

$$\lim_{t\to\infty} \mathbf{x}(t) \;=\; (\overline{\mathbf{z}}\;\overline{\mathbf{z}}^T)\,\mathbf{x}_0.$$

■

5.6. REMARK (COMPUTATION OF THE EIGENVALUES). *The eigenvalues of the QEP (5.38) can be easily computed by converting it to a standard* generalized eigenvalue problem[10] *of size $2n$, by defining the new vector $\mathbf{w} = \lambda\,\mathbf{z}$. In terms of $\mathbf{z}$ and $\mathbf{w}$, problem (5.38) becomes:*

$$\begin{bmatrix} \mathbf{0} & \mathbf{I}_n \\ \mathbf{I}_n & -\mathbf{A} \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{w} \end{bmatrix} \;=\; \lambda \begin{bmatrix} \mathbf{I}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_n - \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{w} \end{bmatrix}.$$

[10]This construction is called "linearization" in the literature and it is not unique in general.

Matlab's function "`polyeig`" uses a linearization procedure similar to the one just described, to numerically solve generic polynomial eigenvalue problems. ◇

The following proposition shows a connection existing between the topology of the communication graph $\mathcal{G}$ and the properties of the QEP (5.38).

5.10. PROPOSITION. *If the graph $\mathcal{G}$ has a vertex with degree equal to one, $\mathbf{I}_n - \mathbf{D}$ is singular and the QEP (5.38) admits at least two infinite eigenvalues.* ■

For example, with the path graph, $\text{rank}(\mathbf{I}_n - \mathbf{D}) = n - 2$, with the star graph, $\text{rank}(\mathbf{I}_n - \mathbf{D}) = 1$, and in both cases, $\det((\mathbf{I}_n - \mathbf{D})\lambda^2 + \mathbf{A}\,\lambda - \mathbf{I}_n) = (-1)^{n-1}(\lambda^2 - 1)$. Hence, for these two graphs the QEP (5.38) admits $2n-2$ infinite eigenvalues. The next proposition shows how to determine the s-stability interval of the deformed consensus protocol, for graphs with arbitrary topology.

5.11. PROPOSITION (STABILITY INTERVAL FOR ARBITRARY $\mathcal{G}$). *Let $q(s) \triangleq \det((\mathbf{I}_n - \mathbf{D})s^2 + \mathbf{A}s - \mathbf{I}_n)$, then:*

- *If n is* even, *system (5.36) is asymptotically stable for all s such that $q(s) > 0$, and unstable for all s such that $q(s) < 0$.*
- *If n is* odd, *system (5.36) is asymptotically stable for all s such that $q(s) < 0$, and unstable for all s such that $q(s) > 0$.* ■

The following example illustrates the rich variety of behaviors exhibited by the deformed consensus protocol on four "generic" graphs with five vertices.

5.1. EXAMPLE. *Consider the four graphs reported in Fig. 5.17. By leveraging Proposition 5.9 and Proposition 5.11, we have that:*

- *With the graph in Fig. 5.17(a), system (5.36) is asymptotically stable $\forall\, s \in \mathbb{R} \setminus \{1\}$.*
- *With the graph in Fig. 5.17(b), system (5.36) is asymptotically stable for $s < 0.7022$ or $s > 1$. For $s = 0.7022$, the system is marginally stable and three groups of vertices can be identified: $\{1\}$, $\{2, 5\}$, $\{3, 4\}$ (different shapes are used in Fig. 5.17(b) to indicate these groups).*
- *With the graph in Fig. 5.17(c), system (5.36) is asymptotically stable for $s < 0.4396$ or $s > 1$. For $s = 0.4396$, the system is marginally stable and three groups of vertices can be identified: $\{1\}$, $\{2, 5\}$, $\{3, 4\}$.*
- *With the graph in Fig. 5.17(d), system (5.36) is asymptotically stable for $s < 0.3804$ or $s > 1$. For $s = 0.3804$, the system is marginally stable and two groups of vertices can be identified: $\{1, 3\}$, $\{2, 4, 5\}$.*

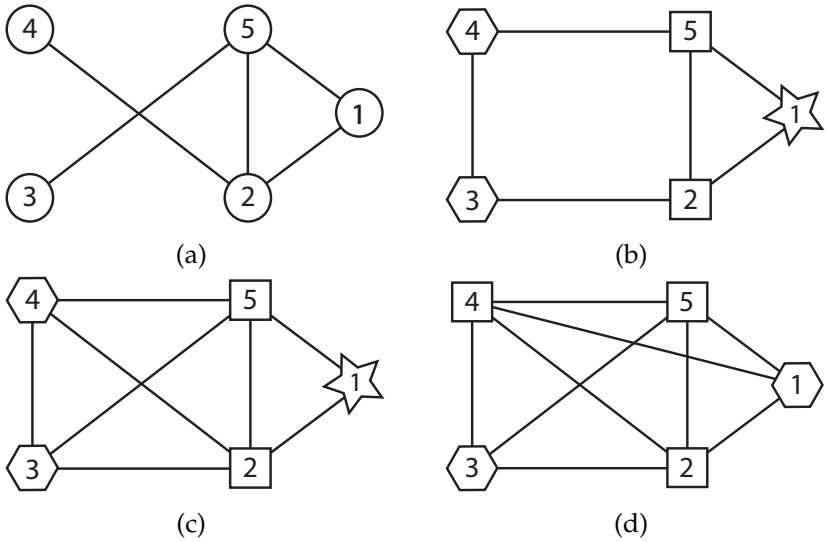


Figure 5.17: Example 1*: In (b)-(d), different shapes are used to identify distinct groups of vertices: the states associated to the vertices in these groups asymptotically converge to the same value when system (5.36) is marginally stable.*

From Figs. 5.17(b)-(d), we notice that vertices in the same group tend to have the same edge degree, and that an increase in the algebraic connectivity of the graph leads to a shrinkage of the s-stability interval of the deformed consensus protocol. In future works, we aim at providing a general explanation for the peculiar grouping behavior exhibited by the vertices of the four graphs considered in this example. ◇

Extension to directed graphs

In this section we assume that the communication graph is directed and contains a rooted out-branching, and we will study the stability properties of the following linear system,

$$\dot{\mathbf{x}}(t) = -\boldsymbol{\Delta}(\mathcal{D}(s))\,\mathbf{x}(t), \tag{5.39}$$

in terms of the real parameter s, where the symbol "$\mathcal{D}(s)$" indicates that the deformed Laplacian is now relative to a directed communication topology. Note that similarly to the undirected case, we have that $\boldsymbol{\Delta}(\mathcal{D}(1)) = \mathbf{L}(\mathcal{D})$ and $\boldsymbol{\Delta}(\mathcal{D}(-1)) = \mathbf{Q}(\mathcal{D})$. In what follows, we first analyze the stability properties of system (5.39) when $\mathcal{D}$ is the directed cycle graph, and then briefly explore the case of digraphs of arbitrary topology with the help of few examples. The proof of the next proposition can be found in (Morbidi 2013b).

5.12. PROPOSITION (DIRECTED CYCLE D_n). *For the directed cycle graph D_n with $n > 2$ vertices, we have that:*

- *For $s = 1$ and for all $n > 2$, average consensus is achieved.*
- *If n is even:*
  - *For $|s| < 1$, system (5.39) is asymptotically stable.*
  - *For $|s| > 1$, system (5.39) is unstable.*
  - *For $s = -1$, system (5.39) is marginally stable. In this case, the states associated to $n/2$ vertices asymptotically converge to $\frac{1}{n}\,\mathbf{x}_0^T\,\mathbb{k}$ and the states associated to the other $n/2$ vertices converge to $-\frac{1}{n}\,\mathbf{x}_0^T\,\mathbb{k}$.*
- *If n is odd:*
  - *System (5.39) is asymptotically stable for $s \in (\vartheta(n),\, 1)$, where,*
$$\vartheta(n) \;=\; \frac{1}{\cos\Big(\frac{n(n-2)+1}{n}\,\pi\Big)}\,.$$
  - *For $s < \vartheta(n)$ or $s > 1$, system (5.39) is unstable.*
  - *For $s = \vartheta(n)$, system (5.39) is marginally stable. At steady-state, we have that the i-th state of system (5.39), $i \in \{1, \ldots, n\}$, obeys,*
$$x_i(t) \;=\; \mathrm{A}\sin\big(2\pi\, f(n)\, t \;+\; \phi_i(n) \;+\; \phi_\circ\big),$$
  where A and $\phi_\circ$ are positive constants, the frequency $f(n) = \frac{1}{2\pi}\tan\big(\frac{n(n-2)+1}{n}\,\pi\big)$, and for $i \in \{1, \ldots, n\}$ the phase $\phi_i(n) = \frac{2\pi\,(i-1)}{n\tan\left(\frac{n(n-2)+1}{n}\,\pi\right)}$. ■

5.2. EXAMPLE. *Consider the four digraphs in Fig. 5.18.*

- *With the digraph in Fig. 5.18(a), we have that system (5.39) is asymptotically stable for $s \in (1/\cos(16\pi/5),\, 1)$ (cf. Proposition 5.12). For $s = 1/\cos(16\pi/5)$, system (5.39) is marginally stable: at steady-state, its states oscillate with the same frequency and amplitude, and the phases are evenly spaced. For $s = 1$, average consensus is achieved (see the simulations in Sect. 5.4.3).*
- *With the digraph in Fig. 5.18(b), system (5.39) is asymptotically stable for $s \in (-1.6889,\, 1)$. For $s = -1.6889$, system (5.39) is marginally stable. At steady-state, its states oscillate with the same frequency but have different amplitudes: the phases are regularly spaced. For $s = 1$, consensus (but not average consensus, since edge $(1,\, 3)$ breaks the balancedness of digraph D_5) is achieved (see the simulations in Sect. 5.4.3).*
- *With the digraph in Fig. 5.18(c), we have the same qualitative behavior as with the digraph in Fig. 5.18(b). The only difference is represented by the stability threshold, that is now $s = -1.9441$ instead of $s = -1.6889$.*

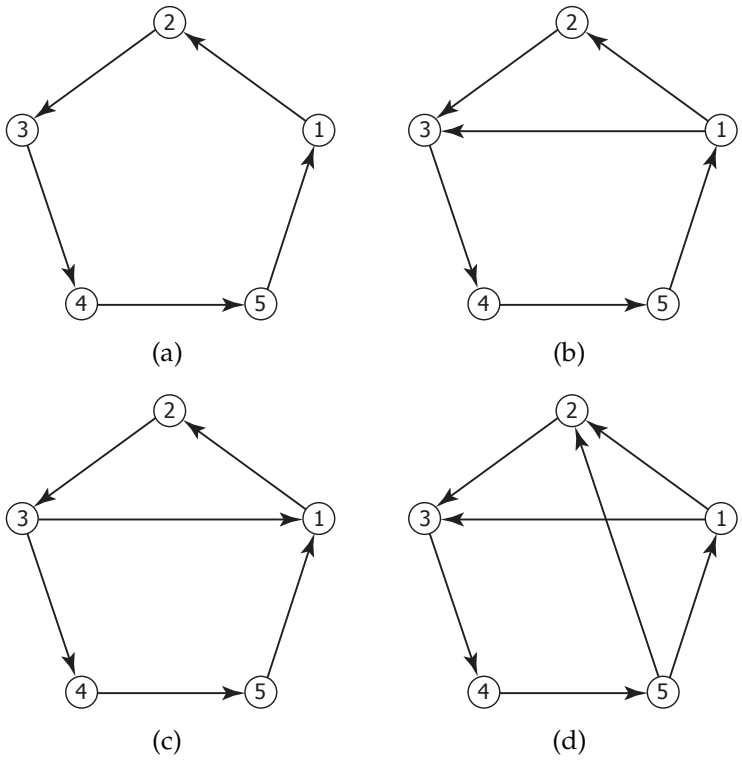


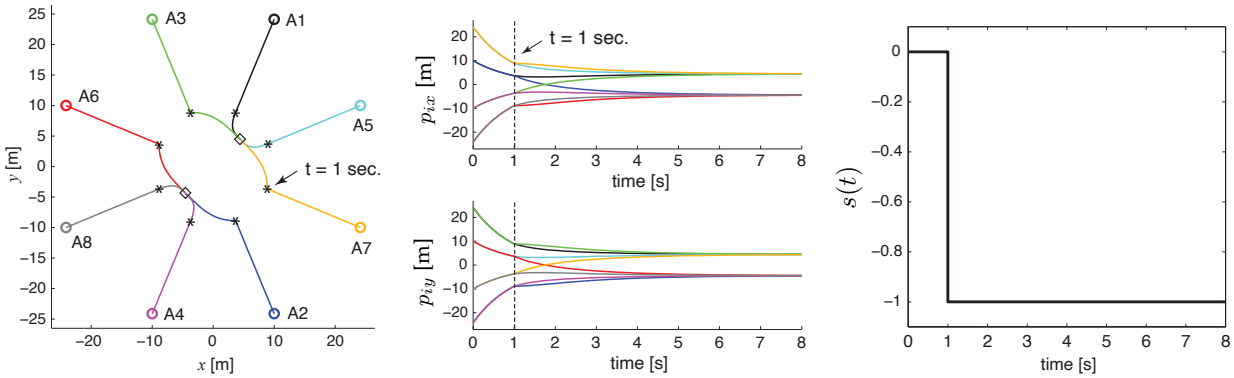
Figure 5.18: Example 2: *Variations on the directed cycle D_5.*

- *With the digraph in Fig. 5.18(d), system (5.39) is asymptotically stable for $s \in (-1.3326,\, 1)$. For $s = -1.3326$, system (5.39) is marginally stable and its* non-oscillating *states asymptotically converge, in general, to five different values. For $s = 1$, consensus (but, again, not average consensus) is achieved.*

Note that we cannot leverage Proposition 5.11 to determine the s-stability interval of system (5.39) when it admits stable periodic solutions (the threshold values for the digraphs in Figs. 5.18(b) and 5.18(c), have been determined on a trial-and-error basis): however, Proposition 5.11 appears to provide the correct stability intervals in all the other cases (e.g., for the digraph in Fig. 5.18(d)). The analytical determination of the stability thresholds for digraphs of arbitrary topology is the subject of on-going research. ◇

5.4.3 Numerical experiments

Numerical simulations have been carried out to illustrate the theory in Sect. 5.4.2. We will separately deal with undirected and directed graphs.



(a) (b) (c)

Figure 5.19: Simulation results – undirected graph*: (a) Trajectory of the 8 vehicles: the communication graph is the cycle C_8 (the initial position is marked with a circle and the final position with a diamond; a star indicates the switching time); (b) Time evolution of the x-, y-coordinates of the vehicles (top and bottom, respectively): the same color convention as in (a) is adopted here; (c) Time history of parameter s.*

Undirected communication graph

Consider a team of n single-integrator agents $\dot{\mathbf{p}}_i(t) = \boldsymbol{\nu}_i(t)$, $i \in \{1, \dots, n\}$, where $\mathbf{p}_i(t) = [p_{ix}(t), p_{iy}(t)]^T \in \mathbb{R}^2$ and $\boldsymbol{\nu}_i(t) \in \mathbb{R}^2$ denote respectively the position and the input of vehicle i at time t. Let the control input of agent i be of the form,

$$\boldsymbol{\nu}_i(t) \;=\; (s^2 - 1)\,\mathbf{p}_i(t) \;+\; s \sum_{j \in \mathcal{N}(i)} (\mathbf{p}_j(t) - s\,\mathbf{p}_i(t)), \tag{5.40}$$

where $\mathcal{N}(i)$ denotes the set of vertices adjacent to vertex i in the communication graph. The collective dynamics of the group of vehicles adopting control (5.40), can then be written as

$$\dot{\mathbf{p}}(t) \;=\; (-\boldsymbol{\Delta}(s) \otimes \mathbf{I}_2)\,\mathbf{p}(t),$$

where $\mathbf{p} = [\mathbf{p}_1^T, \dots, \mathbf{p}_n^T]^T \in \mathbb{R}^{2n}$. Fig. 5.19(a) shows the trajectory of 8 vehicles implementing the control law (5.40), when the communication graph is the cycle C_8 (the vehicles are initially on the vertices of a regular octagon centered at the origin, and their position is marked with a circle in the figure). For the sake of illustration, in our simulation we selected $s(t) = 0$ for $t \in [0,\, 1)$ seconds, and $s(t) = -1$ for $t \in [1,\, 8]$ seconds (see Fig. 5.19(c)). The time evolution of the x-, y-coordinates of the agents is reported in Fig. 5.19(b). As it is evident in Figs. 5.19(a) and 5.19(b), the vehicles first converge towards the origin by maintaining equal inter-distances, and then the even and odd agents cluster in two distinct groups (recall Proposition 5.2).

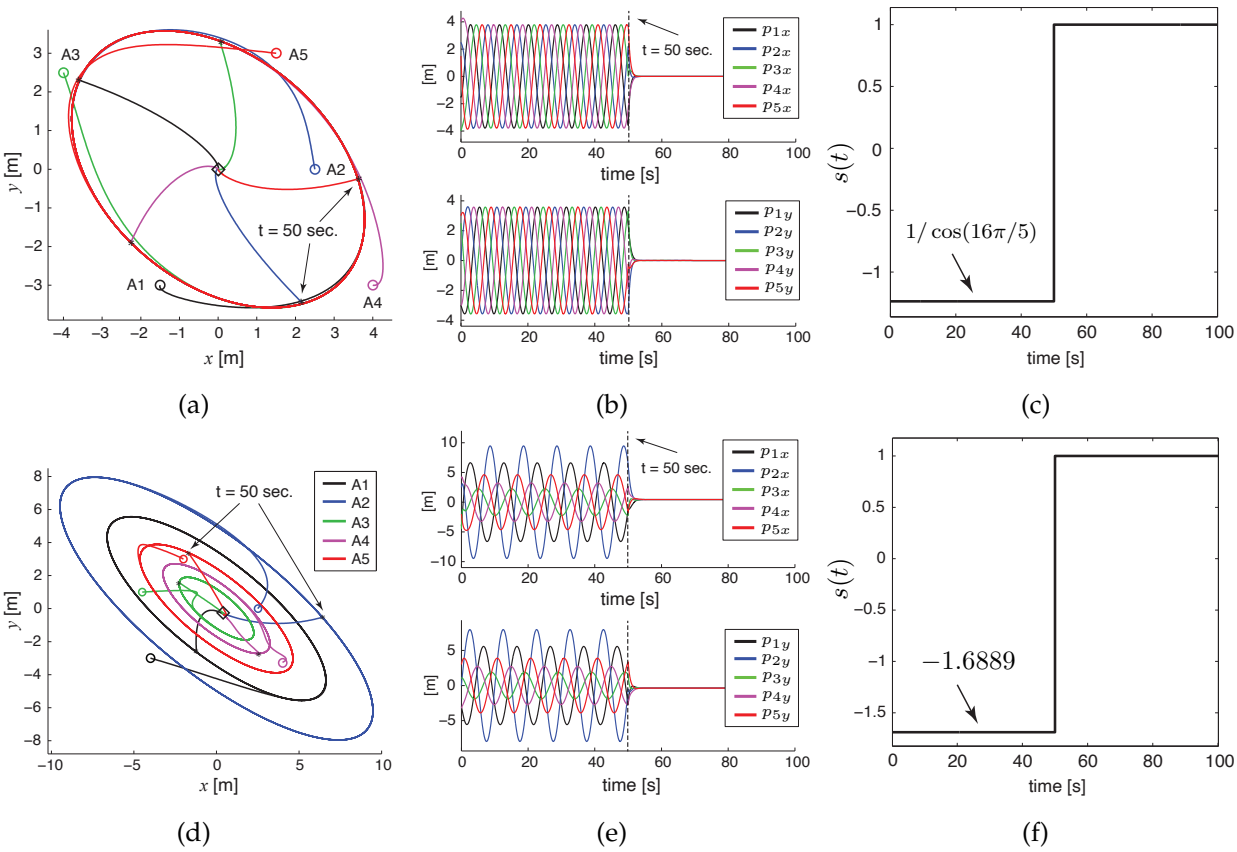


Figure 5.20: Simulation results – directed graph: *(a) Trajectory of the 5 vehicles: the communication digraph is the one in Fig. 5.18(a); (b) Time evolution of the x-, y-coordinates of the vehicles (top and bottom, respectively); (c) Time history of parameter s; (d) Trajectory of the 5 vehicles: the communication digraph is the one in Fig. 5.18(b); (e) Time evolution of the x-, y-coordinates of the vehicles; (f) Time history of parameter s.*

Directed communication graph

A scenario similar to that described in the previous subsection is considered in Fig. 5.23. In this case, 5 single-integrator agents implement the control law (5.40) and communicate using a directed graph. This leads to a closed-loop system of the form:

$$\dot{\mathbf{p}}(t) \;=\; (-\boldsymbol{\Delta}(\mathcal{D}(s)) \otimes \mathbf{I}_2)\, \mathbf{p}(t).$$

The results in Figs. 5.23(a)-(c) are relative to the digraph in Fig. 5.18(a), i.e., D_5. Fig. 5.23(a) shows the trajectory of the 5 vehicles. As shown in Fig. 5.23(c), $s(t) = 1/\cos(16\pi/5)$ for $t \in [0,\, 50)$ sec., and $s(t) = 1$ for $t \in [50,\, 100]$ sec. Fig. 5.23(b) displays the time evolution of the x-, y-coordinates of the agents. From Figs. 5.23(a) and 5.23(b), we can see that the vehicles first move counterclockwise along a common elliptical trajectory with

frequency $f(5) = \frac{1}{2\pi}\tan(16\pi/5)$ and phases $\phi_i(5) = \frac{2\pi(i-1)}{5\tan(16\pi/5)}$, $i \in \{1,\ldots,5\}$, and then rendezvous at the point $(\frac{1}{5}\sum_{i=1}^{5} p_{ix}(50),\ \frac{1}{5}\sum_{i=1}^{5} p_{iy}(50))$, i.e., they achieve average consensus (recall Proposition 5.12).

Finally, the results in Figs. 5.23(d)-(f), are relative to the digraph in Fig. 5.18(b). Fig. 5.23(d) shows the trajectory of the 5 agents. In this case (see Fig. 5.23(f)), $s(t) = -1.6889$ for $t \in [0, 50)$ sec., and $s(t) = 1$ for $t \in [50, 100]$ sec. From Figs. 5.23(d) and 5.23(e), we observe that the vehicles first move counterclockwise on separate closed orbits and then rendezvous at the same point.

5.4.4 Conclusions and future work

In this section, we have presented a generalization of the standard consensus protocol, called *deformed consensus protocol*, and we have analyzed its stability properties in terms of the real parameter s for some special families of undirected and directed graphs. Results for arbitrary graph topologies have been also provided: however, some work still needs to be done in order to precisely characterize, in graph-theoretical terms, the variegated behavior of the deformed consensus protocol. The proposed theory has been illustrated via extensive numerical simulations and examples.

In future works, we aim at studying the properties of the deformed consensus protocol when the (weighted) communication graph is not fixed but changes over time, at establishing a link with the existing cluster synchronization and group consensus literature (Xia and Cao 2011), and at investigating other "parametric" Laplacian-like matrices besides the deformed Laplacian (a first step in this direction, has been done in (Morbidi 2014)).

5.5 Subspace projectors for state-constrained consensus

In this final section, we study the state-constrained consensus problem and introduce a new family of distributed algorithms based on *subspace projection methods* which are simple to implement and which preserve, under some suitable conditions, the consensus value of the original discrete-time agreement protocol. The proposed theory is applied to the constrained rendezvous problem of single-integrator robots.

5.5.1 Introduction

As already discussed in Sect. 5.4.1, the consensus protocol has enjoyed growing success in robotics, and it has been widely employed as a basic building block to design distributed coordination strategies for multi-agent systems. However, in many practical situations, cooperative agents are *not* allowed to freely agree upon a common value, and their state variables are constrained within a given (convex) set. This happens, for example, when multiple mobile robots evolve in confined obstacle-ridden environments, when they are subject to motion constraints (e.g. their velocity is bounded), or they have a limited communication/sensing range. The resulting *constrained consensus problem* has been investigated from different angles in (Nedic et al. 2010, Lee and Mesbahi 2011, Wang and Zhou 2015, Lageman and Sun 2016, Qiu et al. 2016, Fontan et al. 2019). In fact, depending on the nature of constraints (linear vs. nonlinear, equality vs. inequality, including saturations), problems of different complexity arise. A recurring idea in (Nedic et al. 2010) (discrete-time formulation) and in (Wang and Zhou 2015, Lageman and Sun 2016) (continuous-time formulation), has been to project the state vector of each agent onto its constraint set. Similar projection techniques have also been used for distributed optimization (Shi et al. 2012, Qiu et al. 2016), and more recently, to study the synchronization of a network of Kuramoto oscillators (Jafarpour and Bullo 2019).

Inspired by the vast literature on Kalman filtering with state constraints (Andersson et al. 2019, Simon 2010), we will show that under suitable conditions, it is possible to constrain the joint state of a multi-agent system within a prescribed linear subspace, *without affecting* the consensus value of the corresponding unconstrained agreement protocol. In fact, differently from (Nedic et al. 2010, Wang and Zhou 2015), where the *local state* of each agent is kept inside a closed convex set known only to it, we adopt here an alternative approach based on subspace projectors which presents several attractive features: it is straightforward to implement and it leads to a new class of discrete-time constrained consensus protocols whose group decision properties can be easily characterized thanks to a well-established projection theory (Galántai 2004, Ben-Israel and Greville 2003). Our theoretical analysis is paired with numerical experiments for the constrained rendezvous problem in the 2D space.

The rest of this section is organized as follows. In Sect. 5.5.2, we recall some basic facts about orthogonal and oblique projections. Sect. 5.5.3 presents the main theoretical results. The theory is validated with the aid of numerical experiments in Sect. 5.5.4. Finally, in Sect. 5.5.5, we summarize the main contributions and outline some possible avenues for future research.

5.5.2 Preliminaries

In this section, we briefly review recall various definitions and results concerning orthogonal and oblique projections and reflections, that will support our subsequent developments: see (Galántai 2004, Ch. 1-2) and (Ben-Israel and Greville 2003).

5.8. DEFINITION (DEF. 2.1, (GALÁNTAI 2004)). *A matrix $\mathbf{\Pi} \in \mathbb{R}^{n\times n}$ is a projection (idempotent matrix), if and only if $\mathbf{\Pi}^2 = \mathbf{\Pi}$.* ◇

5.9. DEFINITION (DEF. 2.2, (GALÁNTAI 2004)). *Let $\mathcal{M}, \mathcal{N} \subset \mathbb{R}^n$ be complementary subspaces (i.e. $\mathcal{M} \oplus \mathcal{N} = \mathbb{R}^n$, where $\oplus$ denotes the direct sum). For each $\mathbf{z} \in \mathbb{R}^n$, there exists a unique decomposition of the form, $\mathbf{z} = \mathbf{x} + \mathbf{y}$ where $\mathbf{x} \in \mathcal{M}$, $\mathbf{y} \in \mathcal{N}$. The transformation $\mathbf{\Pi}_{\mathcal{M},\mathcal{N}}$ defined by $\mathbf{\Pi}_{\mathcal{M},\mathcal{N}}\,\mathbf{z} \triangleq \mathbf{x}$ is called projection onto $\mathcal{M}$ along $\mathcal{N}$. The transformation $\mathbf{\Pi}_{\mathcal{N},\mathcal{M}}$ defined by $\mathbf{\Pi}_{\mathcal{N},\mathcal{M}}\,\mathbf{z} \triangleq \mathbf{y}$ is called projection onto $\mathcal{N}$ along $\mathcal{M}$. The vector $\mathbf{x}$ is the projection of $\mathbf{z}$ onto $\mathcal{M}$ along $\mathcal{N}$, and $\mathbf{y}$ is the projection of $\mathbf{z}$ onto $\mathcal{N}$ along $\mathcal{M}$.* ◇

Let $\mathcal{M}^\perp = \{\mathbf{x} \in \mathbb{R}^n \,:\, \mathbf{x} \perp \mathbf{y} \text{ for all } \mathbf{y} \in \mathcal{M}\}$ be the orthogonal complement of $\mathcal{M} \subset \mathbb{R}^n$, a linear subspace.

5.10. DEFINITION (DEF. 2.3, (GALÁNTAI 2004)). *The projection $\mathbf{\Pi}$ onto $\mathcal{M}$ along $\mathcal{M}^\perp$ is called orthogonal.* ◇

5.11. DEFINITION (DEF. 2.4, (GALÁNTAI 2004)). *Let $\mathcal{M} \subset \mathbb{R}^n$ be a subspace. $\mathbf{\Pi} \in \mathbb{R}^{n\times n}$ is the orthogonal projection onto $\mathcal{M}$ if (a) $\mathrm{Im}(\mathbf{\Pi}) = \mathcal{M}$, (b) $\mathbf{\Pi}^2 = \mathbf{\Pi}$, (c) $\mathbf{\Pi}^T = \mathbf{\Pi}$.* ◇

Projections that are not orthogonal projections, are called *oblique projections*.

5.3. THEOREM (TH. 2.8, (GALÁNTAI 2004)). *$\mathbf{\Pi}$ is a projection if and only if $\mathbf{I}_n - \mathbf{\Pi}$ is also a projection. If $\mathbf{\Pi}$ is a projection onto $\mathcal{M}$ along $\mathcal{N}$, $\mathbf{I}_n - \mathbf{\Pi}$ is a projection onto $\mathcal{N}$ along $\mathcal{M}$.* ■

It is also straightforward to show that $\mathbf{\Pi}$ is a projection if and only if $\mathbf{\Pi}^T$ is also a projection.

5.4. THEOREM (TH. 2.10, (GALÁNTAI 2004)). *Any projection $\mathbf{\Pi} \in \mathbb{R}^{n\times n}$ may have only 0 and 1 eigenvalues. The multiplicity of the eigenvalue 1 is $\mathrm{rank}(\mathbf{\Pi}) = \mathrm{tr}(\mathbf{\Pi})$.* ■

5.5. THEOREM (TH. 2.16, (GALÁNTAI 2004)). *The linear transformation $\mathbf{\Pi}$ is an orthogonal projection if and only if $\mathbf{\Pi} = \mathbf{\Pi}^2 = \mathbf{\Pi}^T$. Moreover, the orthogonal projections are positive semidefinite and satisfy $\|\mathbf{\Pi}\,\mathbf{x}\|_2 \leq \|\mathbf{x}\|_2$ for all $\mathbf{x} \in \mathbb{R}^n$.* ■

5.12. DEFINITION (DEF. 3.1.1, (BERNSTEIN 2009)). *A matrix $\mathbf{\Psi} \in \mathbb{R}^{n\times n}$ is a reflection if $\mathbf{\Psi}$ is symmetric and orthogonal.* ◇

Any reflection $\mathbf{\Psi}$ is an involutory matrix, i.e. $\mathbf{\Psi}^2 = \mathbf{I}_n$, each of whose eigenvalues is 1 or -1. The following proposition provides an interesting link between projections and reflections.

5.13. PROPOSITION (FACT 3.14.1, (BERNSTEIN 2009)). *If $\mathbf{\Pi} \in \mathbb{R}^{n\times n}$ is an orthogonal projection, then $\mathbf{\Psi} \triangleq 2\mathbf{\Pi} - \mathbf{I}_n$ is a reflection. Conversely, if $\mathbf{\Psi} \in \mathbb{R}^{n\times n}$ is a reflection, then $\mathbf{\Pi} \triangleq \frac{1}{2}(\mathbf{\Psi} + \mathbf{I}_n)$ is an orthogonal projection.* ■

In fact, the affine mapping $\mathbf{f}(\mathbf{\Pi}) \triangleq 2\mathbf{\Pi} - \mathbf{I}_n$ from orthogonal projections to reflections, is bijective. We also observe that if $\mathbf{\Pi}_{\mathcal{M}}$ is the orthogonal projection onto $\mathcal{M}$, then $(2\mathbf{\Pi}_{\mathcal{M}} - \mathbf{I}_n)\mathbf{z}$ is the reflection of $\mathbf{z} \in \mathbb{R}^n$ across $\mathcal{M}^\perp$.

We report below some *representations* of projections that will be used in Sect. 5.5.3.

5.6. THEOREM (TH. 2.26, (GALÁNTAI 2004)). *Let $\mathcal{M}, \mathcal{N} \subset \mathbb{R}^n$ be complementary subspaces, and let the columns of $\mathbf{V} \in \mathbb{R}^{n\times r}$ and $\mathbf{W} \in \mathbb{R}^{n\times r}$ form a basis for $\mathcal{M}$ and $\mathcal{N}^\perp$, respectively. Then, $\mathbf{\Pi}_{\mathcal{M},\mathcal{N}} = \mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T$.* ■

Note that either $\mathbf{V}$ and $\mathbf{W}(\mathbf{W}^T\mathbf{V})^{-T}$ or $\mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1}$ and $\mathbf{W}$ are biorthogonal pairs[11], and that $(\mathbf{W}^T\mathbf{V})^{-1}$ can be interpreted as a "normalizing factor". In the 3D example of Fig. 5.21, we have:

$$\mathbf{V} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{W} = \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 0 & -2 \end{bmatrix},$$

being $\mathcal{N} = \text{span}\{[0,\, 1,\, 1]^T\}$. Note that $\mathbf{V}$ and $\mathbf{W}$ is not a biorthogonal pair, since $\mathbf{W}^T\mathbf{V} = \text{diag}(1,\, 2)$. In the more general case of an arbitrary positive definite weight matrix $\mathbf{\Omega} \in \mathbb{R}^{n\times n}$ defining an inner product $\langle \mathbf{x},\, \mathbf{y}\rangle_{\mathbf{\Omega}} = \mathbf{y}^T\mathbf{\Omega}\,\mathbf{x}$, the projection becomes $\mathbf{\Pi}^{\mathbf{\Omega}}_{\mathcal{M},\mathcal{N}} = \mathbf{V}(\mathbf{W}^T\mathbf{\Omega}\,\mathbf{V})^{-1}\mathbf{W}^T\mathbf{\Omega}$.

5.1. COROLLARY (COR. 2.27, (GALÁNTAI 2004)). *If the columns of $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r]$ form an orthonormal basis for a subspace $\mathcal{M} \subset \mathbb{R}^n$, then $\mathbf{\Pi}_{\mathcal{M}} = \mathbf{V}\mathbf{V}^T$ is the unique orthogonal projection onto $\mathcal{M}$.* ■

[11] $\mathbf{A} \in \mathbb{R}^{n\times m}$ and $\mathbf{B} \in \mathbb{R}^{n\times m}$ is a biorthogonal pair, if $\mathbf{B}^T\mathbf{A} = \mathbf{I}_m$.

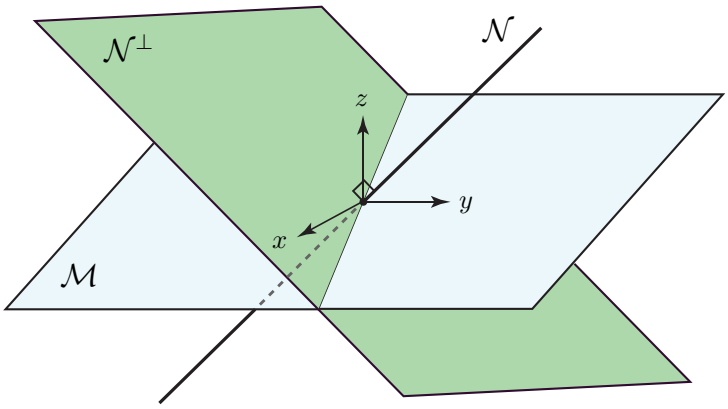


Figure 5.21: *Example of complementary subspaces $\mathcal{M}$ and $\mathcal{N}$ in $\mathbb{R}^3$.*

5.1. LEMMA (LEMMA 2.30, (GALÁNTAI 2004)). *Let $\mathcal{M}, \mathcal{N} \subset \mathbb{R}^n$ be complementary subspaces, and let $\mathbf{U} = [\mathbf{U}_1,\, \mathbf{U}_2]$ and $\mathbf{V} = [\mathbf{V}_1,\, \mathbf{V}_2]$ be orthogonal matrices such that $\mathbf{U}_1$ and $\mathbf{U}_2$ are bases for $\mathcal{M}$ and $\mathcal{M}^\perp$, respectively, and $\mathbf{V}_1$ and $\mathbf{V}_2$ are bases for $\mathcal{N}$ and $\mathcal{N}^\perp$, respectively. Then,*

$$\mathbf{\Pi}_{\mathcal{M},\mathcal{N}} = \mathbf{U} \begin{bmatrix} \mathbf{0} & (\mathbf{V}_2^T \mathbf{U}_1)^{-1} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{V}^T.$$

■

Matrices $\mathbf{U}$ and $\mathbf{V}$ can be chosen such that we obtain the Singular Value Decomposition (SVD) of $\mathbf{\Pi}_{\mathcal{M},\mathcal{N}}$.

Any projection is diagonalizable by a similarity transformation. In particular, the following result can be proved using Schur's theorem (Galántai 2004, Th. 1.28).

5.7. THEOREM (TH. 2.31, (GALÁNTAI 2004)). *If $\mathbf{\Pi} \in \mathbb{R}^{n\times n}$ is a projection of rank m, then there is an orthogonal matrix $\mathbf{V}$ such that,*

$$\mathbf{\Pi} = \mathbf{V} \begin{bmatrix} \mathbf{I}_m & \mathbf{B} \\ \mathbf{0} & \mathbf{0}_{n-m} \end{bmatrix} \mathbf{V}^T,$$

where matrix $\mathbf{B} \in \mathbb{R}^{m\times(n-m)}$ is arbitrary. ■

We finally recall here some connections between projections and generalized inverses (Ben-Israel and Greville 2003, Ch. 2). Moore's original definition of Moore-Penrose inverse is the following:

5.13. DEFINITION (DEF. 2.52, (GALÁNTAI 2004)). *If $\mathbf{M} \in \mathbb{R}^{m\times n}$, then the generalized inverse of $\mathbf{M}$ is defined to be the unique matrix $\mathbf{M}^+$ such that:*

a) $\mathbf{M}\mathbf{M}^{+} = \mathbf{\Pi}_{\mathrm{Im}(\mathbf{M})}$,

b) $\mathbf{M}^{+}\mathbf{M} = \mathbf{\Pi}_{\mathrm{Im}(\mathbf{M}^{+})} = \mathbf{\Pi}_{\mathrm{Im}(\mathbf{M}^{T})}$. ◇

To conclude, the next theorem provides useful properties of the Moore-Penrose inverse, cf. (Galántai 2004, Def. 1.54).

5.8. THEOREM (TH. 1.60, (GALÁNTAI 2004)). *Let* $\mathbf{M}^{+} \in \mathbb{R}^{n\times m}$ *be the Moore-Penrose inverse of* $\mathbf{M} \in \mathbb{R}^{m\times n}$. *Then,*

a) $\mathrm{Im}(\mathbf{M}^{+}) = \mathrm{Im}(\mathbf{M}^{T}) = \mathrm{Im}(\mathbf{M}^{+}\mathbf{M}) = \mathrm{Im}(\mathbf{M}^{T}\mathbf{M})$,

b) $\mathrm{Im}(\mathbf{I}_n - \mathbf{M}^{+}\mathbf{M}) = \ker(\mathbf{M}^{+}\mathbf{M}) = \ker(\mathbf{M}) = \mathrm{Im}(\mathbf{M}^{T})^{\perp}$. ■

5.5.3 State-constrained consensus protocols

This section presents our main theoretical results. In what follows, we make the standing assumption that the graph $\mathcal{G}$ is *connected*.

5.9. THEOREM (ORTHOGONAL PROJECTION). *Let us suppose that the following constraint is imposed on the state vector of consensus protocol* (3.6) *(cf. Sect. 3.2),*

$$\mathbf{x}(k+1) \in \mathcal{M} \subset \mathbb{R}^n, \quad k \in \{0, 1, 2, \ldots\}, \tag{5.41}$$

where the columns of $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r] \in \mathbb{R}^{n\times r}$ *form an orthonormal basis for the linear subspace* $\mathcal{M}$ *and* $\mathbb{1} \in \mathrm{span}\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r\}$. *Consider the* projected consensus protocol*,*

$$\mathbf{x}_{\mathrm{or}}(k+1) = \mathbf{\Pi}_{\mathcal{M}}\mathbf{P}\,\mathbf{x}_{\mathrm{or}}(k), \quad \mathbf{x}_{\mathrm{or}}(0) = \mathbf{x}_0, \tag{5.42}$$

where $\mathbf{\Pi}_{\mathcal{M}} = \mathbf{V}\mathbf{V}^{T} \in \mathbb{R}^{n\times n}$ *is the orthogonal projection onto* $\mathcal{M}$ *(cf. Corollary 5.1), and let us assume that*

$$\mathrm{rank}(\mathbf{I}_n - \mathbf{\Pi}_{\mathcal{M}}\mathbf{P}) = n - 1. \tag{5.43}$$

Then, the solution to (5.42) *satisfies,*

$$\lim_{k\to\infty} \mathbf{x}_{\mathrm{or}}(k) = \frac{1}{n}\mathbb{1}\mathbb{1}^{T}\mathbf{x}_0. \tag{5.44}$$

Proof : *See (Morbidi 2020).* ■

Note that (5.42) simply proceeds by projecting each step of consensus protocol (3.6) onto the constraint subspace $\mathcal{M}$, thus ensuring positive invariance (Blanchini and Miani 2015). The orthogonal projection $\mathbf{\Pi}_{\mathcal{M}}$ is instrumental in enforcing constraint (5.41), but it *does modify* the spectrum of $\mathbf{P}$ and affects the convergence rate to average consensus. Nevertheless, the sequences $\{\mathbf{x}_{\mathrm{or}}(k)\}$ and $\{\mathbf{x}(k)\}$ converge to the same consensus value.

5.7. REMARK (NONNEGATIVE PROJECTIONS). *If $\mathbf{\Pi}_{\mathcal{M}}$ is* nonnegative, *then $\mathbf{\Pi}_{\mathcal{M}}\mathbf{P}$ is nonnegative as well. Since the row-sums of $\mathbf{\Pi}_{\mathcal{M}}\mathbf{P}$ are all equal to 1, then $\mathbf{\Pi}_{\mathcal{M}}\mathbf{P}$ is row-stochastic and the statement of Theorem 5.9 becomes a direct consequence of(Bullo 2022, Th. 5.1). For further results on nonnegative projections, the reader is referred to (Galántai 2004, pp. 48-50).* ◇

The first of the next two corollaries states that Theorem 5.9 is still valid when the orthogonal projection $\mathbf{\Pi}_{\mathcal{M}}$ is replaced with the *reflection* $\mathbf{\Psi}_{\mathcal{M}} = 2\mathbf{\Pi}_{\mathcal{M}} - \mathbf{I}_n$ (recall Proposition 5.13), while the second one specializes it to *linear state equality constraints*.

5.2. COROLLARY (REFLECTION). *Let us suppose that the assumptions of Theorem 5.9 hold true. Then, provided that* $\operatorname{rank}(\mathbf{I}_n - \mathbf{\Psi}_{\mathcal{M}}\mathbf{P}) = n-1$, *the solution to the* reflected consensus protocol,

$$\mathbf{x}_{\text{re}}(k+1) = \mathbf{\Psi}_{\mathcal{M}}\mathbf{P}\,\mathbf{x}_{\text{re}}(k), \quad \mathbf{x}_{\text{re}}(0) = \mathbf{x}_0, \tag{5.45}$$

satisfies

$$\lim_{k\to\infty} \mathbf{x}_{\text{re}}(k) = \frac{1}{n}\mathbb{1}\mathbb{1}^T\mathbf{x}_0,$$

where $\mathbf{\Psi}_{\mathcal{M}} = 2\mathbf{\Pi}_{\mathcal{M}} - \mathbf{I}_n$ is the reflection across $\mathcal{M}^{\perp}$.

Proof : *See (Morbidi 2020).* ■

Note that unlike Theorem 5.9, in this case the state vector $\mathbf{x}_{\text{re}}$ of system (5.45) "bounces off" the subspace $\mathcal{M}^{\perp}$.

5.3. COROLLARY (ORTHOGONAL PROJECTION: LINEAR EQUALITY CONSTRAINTS). *Let us suppose that the following equality constraint is imposed on the state vector of protocol* (3.6),

$$\mathbf{M}\,\mathbf{x}(k+1) = \mathbf{0}, \quad k \in \{0,1,2,\ldots\}, \tag{5.46}$$

where matrix $\mathbf{M} \in \mathbb{R}^{q\times n}$ has row rank $m \leq q$ and satisfies $\mathbf{M}\,\mathbb{1} = \mathbf{0}$. Consider the projected consensus protocol,

$$\mathbf{x}_{\text{or}}(k+1) = \mathbf{\Pi}_{\ker(\mathbf{M})}\mathbf{P}\,\mathbf{x}_{\text{or}}(k), \quad \mathbf{x}_{\text{or}}(0) = \mathbf{x}_0, \tag{5.47}$$

where $\mathbf{\Pi}_{\ker(\mathbf{M})} = \mathbf{V}\mathbf{V}^T$ is the orthogonal projection onto the null space of $\mathbf{M}$, i.e. the columns of $\mathbf{V} \in \mathbb{R}^{n\times(n-m)}$ form a basis which satisfies $\mathbf{M}\mathbf{V} = \mathbf{0}$ and $\mathbf{V}^T\mathbf{V} = \mathbf{I}_{n-m}$. Then, under condition (5.43), *the state trajectories of protocol* (5.47) *satisfy* (5.44).

Proof : *Since $\mathbf{M}\,\mathbb{1} = \mathbf{0}$, then $\mathbb{1} \in \ker(\mathbf{M})$ and we can write the orthonormal basis of $\ker(\mathbf{M})$ as $\mathbf{V} = [\mathbb{1}/\sqrt{n},\ \overline{\mathbf{V}}]$ where $\overline{\mathbf{V}} \in \mathbb{R}^{n\times(n-m-1)}$. The rest of the proof is an immediate transcription of that of Theorem 5.9.* ■

As illustrated in Sect. 5.5.4, matrix $\mathbf{M}$ can be used to encode a variety of constraints frequently encountered in distributed robotics (e.g. collision avoidance or connectivity maintenance constraints). Note that $\mathbf{\Pi}_{\ker(\mathbf{M})}$ can be readily found by computing the

SVD of $\mathbf{M}$. In fact, if $\mathbf{M}$ is decomposed as $\mathbf{M} = [\mathbf{U}_m, \overline{\mathbf{U}}_m]\, \mathbf{\Sigma}\, [\mathbf{V}_m, \overline{\mathbf{V}}_m]^T$ where $[\mathbf{U}_m, \overline{\mathbf{U}}_m]$, $[\mathbf{V}_m, \overline{\mathbf{V}}_m]$ are orthogonal matrices with $\mathbf{U}_m \in \mathbb{R}^{q\times m}$, $\overline{\mathbf{U}}_m \in \mathbb{R}^{q\times(q-m)}$, $\mathbf{V}_m \in \mathbb{R}^{n\times m}$, $\overline{\mathbf{V}}_m \in \mathbb{R}^{n\times(n-m)}$, and $\mathbf{\Sigma} \in \mathbb{R}^{q\times n}$ is a rectangular diagonal matrix with nonnegative real numbers in decreasing order on the diagonal, then $\overline{\mathbf{V}}_m \overline{\mathbf{V}}_m^T$ is the orthogonal projection onto $\ker(\mathbf{M})$ (Galántai 2004, Th. 2.36). Moreover, $\mathbf{V}_m\mathbf{V}_m^T$, $\mathbf{U}_m\mathbf{U}_m^T$ and $\overline{\mathbf{U}}_m\overline{\mathbf{U}}_m^T$ are the orthogonal projections onto $\ker(\mathbf{M})^\perp$, $\mathrm{Im}(\mathbf{M})$ and $\mathrm{Im}(\mathbf{M})^\perp$, respectively.

The next theorem extends Theorem 5.9 to the more challenging case of *oblique projections* (recall Theorem 5.6).

5.10. THEOREM (OBLIQUE PROJECTION). *Let us suppose that constraint* (5.41) *is imposed on the state vector of protocol* (3.6). *Let $\mathcal{M}, \mathcal{N} \subset \mathbb{R}^n$ be complementary subspaces. Let the columns of $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r] \in \mathbb{R}^{n\times r}$ and $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_r] \in \mathbb{R}^{n\times r}$ form a basis for $\mathcal{M}$ and $\mathcal{N}^\perp$, respectively, and let $\mathbf{V}$ and $\mathbf{W}$ be a biorthogonal pair, i.e. $\mathbf{W}^T\mathbf{V} = \mathbf{I}_r$. Let $\mathbb{1} \in \mathrm{span}\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r\} \cap \mathrm{span}\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_r\}$. Consider the* projected consensus protocol*,*

$$\mathbf{x}_{\mathrm{ob}}(k+1) \;=\; \mathbf{\Pi}_{\mathcal{M},\mathcal{N}}\, \mathbf{P}\, \mathbf{x}_{\mathrm{ob}}(k), \quad \mathbf{x}_{\mathrm{ob}}(0) \;=\; \mathbf{x}_0, \tag{5.48}$$

where $\mathbf{\Pi}_{\mathcal{M},\mathcal{N}} = \mathbf{V}\mathbf{W}^T$ and assume that

$$\mathrm{rank}(\mathbf{I}_n - \mathbf{\Pi}_{\mathcal{M},\mathcal{N}}\mathbf{P}) \;=\; n-1. \tag{5.49}$$

Then, the solution to (5.48) *satisfies,*

$$\lim_{k\to\infty}\; \mathbf{x}_{\mathrm{ob}}(k) \;=\; \frac{1}{n}\, \mathbb{1}\mathbb{1}^T\mathbf{x}_0.$$

Proof : *See (Morbidi 2020).* ■

If $\mathbb{1} \notin \mathrm{span}\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_r\}$, system (5.48) may still achieve consensus asymptotically, but not average consensus.

5.8. REMARK (ON THE DISTRIBUTED NATURE OF PROJECTED CONSENSUS PROTOCOL). *Note that matrix $\mathbf{\Pi}_{\mathcal{M},\mathcal{N}}\,\mathbf{P}$ is* less *sparse than $\mathbf{P}$, in general, and it can be associated with a communication graph $\mathcal{G}' = (V, E')$ with $E \subseteq E'$, over which the dynamics of* (5.48) *evolve (cf. Fig. 5.24). This is not surprising, since to fulfill the state constraint* (5.41) *while still achieving average consensus, the communication between the robots must be more efficient. If the subspaces $\mathcal{M}$ and $\mathcal{N}$ are known a priori, protocol* (5.48) *(as well as protocol* (5.42)*) admits a distributed implementation over $\mathcal{G}'$.* ◇

In the next theorem, we leverage Theorems 5.3 and 5.8, and consider a special oblique projection for the case of *linear state equality constraints*. The "obliquity" of the projection hinges on the positive definite weight matrix $\mathbf{\Omega}$, which alters the group decision value of protocol (3.6).

5.11. THEOREM (WEIGHTED OBLIQUE PROJECTION: LINEAR EQUALITY CONSTRAINTS). *Let us suppose that constraint* (5.46) *is imposed on the state vector of protocol* (3.6)*, where* $\mathbf{M} \in \mathbb{R}^{q\times n}$ *has full row rank and it satisfies* $\mathbf{M}\mathbb{1} = \mathbf{0}$*. Consider the* projected consensus protocol,

$$\mathbf{x}_{\text{wob}}(k+1) = \mathbf{\Pi}^{\text{ob}}_{\ker(\mathbf{M})}\mathbf{P}\,\mathbf{x}_{\text{wob}}(k),\;\; \mathbf{x}_{\text{wob}}(0) = \mathbf{x}_0, \tag{5.50}$$

where $\mathbf{\Pi}^{\text{ob}}_{\ker(\mathbf{M})} = \mathbf{I}_n - \mathbf{M}^{\dagger}_R\mathbf{M}$ *is the oblique projection onto the null space of* $\mathbf{M}$ *and* $\mathbf{M}^{\dagger}_R = \mathbf{\Omega}^{-1}\mathbf{M}^T(\mathbf{M}\,\mathbf{\Omega}^{-1}\mathbf{M}^T)^{-1}$ *is the weighted right pseudo-inverse of* $\mathbf{M}$ *(Siciliano et al. 2009, Sect. A.7), where* $\mathbf{\Omega} \in \mathbb{R}^{n\times n}$ *is a positive definite matrix. Provided that* $\text{rank}(\mathbf{I}_n - \mathbf{\Pi}^{\text{ob}}_{\ker(\mathbf{M})}\mathbf{P}) = n-1$*, the state trajectories of* (5.50) *satisfy,*

$$\lim_{k\to\infty}\; \mathbf{x}_{\text{wob}}(k) = (\boldsymbol{\mu}^T\mathbf{x}_0)\mathbb{1}, \tag{5.51}$$

where $\boldsymbol{\mu}$ *is the left eigenvector of* $\mathbf{\Pi}^{\text{ob}}_{\ker(\mathbf{M})}\mathbf{P}$ *with eigenvalue 1, normalized to have unit sum, i.e.* $\mathbb{1}^T\boldsymbol{\mu} = 1$.

Proof : *See (Morbidi 2020).* ■

Theorem 5.11 can be generalized to arbitrary weight matrices $\mathbf{\Omega}_g \in \mathbb{R}^{n\times n}$ such that $\text{rank}(\mathbf{M}\mathbf{\Omega}_g\mathbf{M}^T) = \text{rank}(\mathbf{M})$. In this case, the weighted right pseudo-inverse of $\mathbf{M}$ becomes $\mathbf{M}^{\dagger}_R = \mathbf{\Omega}_g\mathbf{M}^T(\mathbf{M}\,\mathbf{\Omega}_g\mathbf{M}^T)^{-1}$ cf. (Galántai 2004, Th. 1.53(i)).

With reference to Theorem 5.9, in the following proposition we study the effect of a rotation matrix $\mathbf{R}$ on the basis vectors of the constraint subspace $\mathcal{M}$.

5.14. PROPOSITION. *Let the columns of* $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r] \in \mathbb{R}^{n\times r}$ *form an orthonormal basis for the subspace* $\mathcal{M}$*, and let* $\mathbf{\Pi}_{\mathcal{M}}$ *the orthogonal projection onto* $\mathcal{M}$*. Let* $\mathcal{M}^{\mathbf{R}}$ *be the subspace with orthonormal basis* $\mathbf{R}\mathbf{V}$*, where* $\mathbf{R} \in \text{SO}(n)$*. Then,* $\mathbf{\Pi}^{\mathbf{R}}_{\mathcal{M}} = \mathbf{R}\,\mathbf{\Pi}_{\mathcal{M}}\mathbf{R}^T$ *is the orthogonal projection onto* $\mathcal{M}^{\mathbf{R}}$*, and* $\mathbf{\Pi}^{\mathbf{R}}_{\mathcal{M}}$ *and* $\mathbf{\Pi}_{\mathcal{M}}$ *have the same spectrum.*

Proof : *We have that,*

$$(\mathbf{\Pi}^{\mathbf{R}}_{\mathcal{M}})^2 = (\mathbf{R}\,\mathbf{\Pi}_{\mathcal{M}}\mathbf{R}^T)(\mathbf{R}\,\mathbf{\Pi}_{\mathcal{M}}\mathbf{R}^T) = \mathbf{R}\,\mathbf{\Pi}^2_{\mathcal{M}}\mathbf{R}^T = \mathbf{\Pi}^{\mathbf{R}}_{\mathcal{M}},$$

and $(\mathbf{\Pi}^{\mathbf{R}}_{\mathcal{M}})^T = (\mathbf{R}\,\mathbf{\Pi}_{\mathcal{M}}\mathbf{R}^T)^T = \mathbf{R}\,\mathbf{\Pi}^T_{\mathcal{M}}\mathbf{R}^T = \mathbf{\Pi}^{\mathbf{R}}_{\mathcal{M}}$*, hence owing to Definition 5.11,* $\mathbf{\Pi}^{\mathbf{R}}_{\mathcal{M}}$ *is the orthogonal projection onto* $\mathcal{M}^{\mathbf{R}}$*. Finally, since* $\mathbf{R}^T = \mathbf{R}^{-1}$*, matrices* $\mathbf{\Pi}^{\mathbf{R}}_{\mathcal{M}}$ *and* $\mathbf{\Pi}_{\mathcal{M}}$ *are similar and they have the same spectrum.* ■

Note that the result of Proposition 5.14 can be extended to oblique projections, i.e. if we consider $\mathbf{\Pi}^{\mathbf{R}}_{\mathcal{M},\mathcal{N}} = \mathbf{R}\,\mathbf{\Pi}_{\mathcal{M},\mathcal{N}}\,\mathbf{R}^T$, cf. (Galántai 2004, Th. 2.35).

It is finally worth pointing out that in many practical applications, protocol (3.6) can be subject to *multiple* state constraints. For instance, one may wish to *simultaneously* account for the limited communication range of the robots and collision avoidance. The following theorem (Ben-Israel and Greville 2003, p. 183) elucidates the case of two subspace constraints.

5.12. THEOREM (ANDERSON & DUFFIN). *Let $\mathcal{M}_1,\ \mathcal{M}_2 \subset \mathbb{R}^n$ be two subspaces, with $\mathbf{\Pi}_{\mathcal{M}_1}$ and $\mathbf{\Pi}_{\mathcal{M}_2}$ the corresponding orthogonal projections. Then,*

$$\mathbf{\Pi}_{\mathcal{M}_1 \cap \mathcal{M}_2} \;=\; 2\,\mathbf{\Pi}_{\mathcal{M}_1}(\mathbf{\Pi}_{\mathcal{M}_1} + \mathbf{\Pi}_{\mathcal{M}_2})^{+}\,\mathbf{\Pi}_{\mathcal{M}_2} \;=\; 2\,\mathbf{\Pi}_{\mathcal{M}_2}(\mathbf{\Pi}_{\mathcal{M}_1} + \mathbf{\Pi}_{\mathcal{M}_2})^{+}\,\mathbf{\Pi}_{\mathcal{M}_1}.$$

■

In particular, if $\mathbf{\Pi}_{\mathcal{M}_1}$ and $\mathbf{\Pi}_{\mathcal{M}_2}$ commute, i.e. $\mathbf{\Pi}_{\mathcal{M}_1}\mathbf{\Pi}_{\mathcal{M}_2} = \mathbf{\Pi}_{\mathcal{M}_2}\mathbf{\Pi}_{\mathcal{M}_1}$, then $\mathbf{\Pi}_{\mathcal{M}_1 \cap \mathcal{M}_2} = \mathbf{\Pi}_{\mathcal{M}_1}\mathbf{\Pi}_{\mathcal{M}_2}$.

5.5.4 Numerical experiments

In this section, the theory presented in Sect. 5.5.3 is illustrated via numerical experiments. For its simplicity, the 2D rendezvous problem for single-integrator robots (Francis and Maggiore 2016), is chosen as a testing ground for the proposed projected consensus protocols. To improve readability, the discrete-time state trajectories will be displayed as continuous curves.

The $n = 10$ robots are initially arranged at equal intervals around a unit circle centered at point $(1,\,1)$ (dashed gray in Figs. 5.22(a)-(c) and 5.23(a)-(c)), i.e.

$$\mathbf{x}_0 \;=\; [\cos(0),\,\sin(0),\,\cos(\pi/5),\,\sin(\pi/5),\,\ldots,\cos(9\pi/5),\sin(9\pi/5)]^T + [1,\,1,\,\ldots,1]^T \in \mathbb{R}^{20},$$

and they have a prior knowledge of the surrounding environment. Fig. 5.22(a) shows the trajectory of the robots obtained with the *unconstrained* consensus protocol,

$$\mathbf{x}(k+1) \;=\; (\mathbf{P} \,\otimes\, \mathbf{I}_2)\,\mathbf{x}(k),\;\; \mathbf{x}(0) \;=\; \mathbf{x}_0, \tag{5.52}$$

with a cycle communication graph $\mathcal{G} = C_{10}$ and a step-size $\varepsilon = 0.3 < 1/d_{\max}(C_{10}) = 1/2$. The corresponding time-evolution of the x- and y-coordinates of the robots for the first half of the 100-iteration simulation, is reported in Fig. 5.22(d) (the same color convention as in Fig. 5.22(a) is adopted here). As expected, the robots meet at point $(1,\,1)$. For comparison, Fig. 5.22(b) shows the trajectory of the robots generated by the constrained consensus protocol with *orthogonal projection* (cf. Corollary 5.3),

$$\mathbf{x}_{\text{or}}(k+1) \;=\; (\mathbf{\Pi}_{\mathcal{M}}\mathbf{P} \,\otimes\, \mathbf{I}_2)\,\mathbf{x}_{\text{or}}(k),\;\; \mathbf{x}_{\text{or}}(0) \;=\; \mathbf{x}_0, \tag{5.53}$$

where $\mathcal{M} = \ker(\mathbf{M}_1) \subset \mathbb{R}^{10}$, with

$$\mathbf{M}_1 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -10 & 10 & \frac{1}{3} & -\frac{1}{3} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} \end{bmatrix}.$$

Matrix $\mathbf{M}_1$ allows here to encode "attraction forces" of various magnitude between pairs of robots. From Fig. 5.22(b) and Fig. 5.22(e), which reports the corresponding time-evolution of the x- and y-coordinates of the robots, we observe that the trajectories of

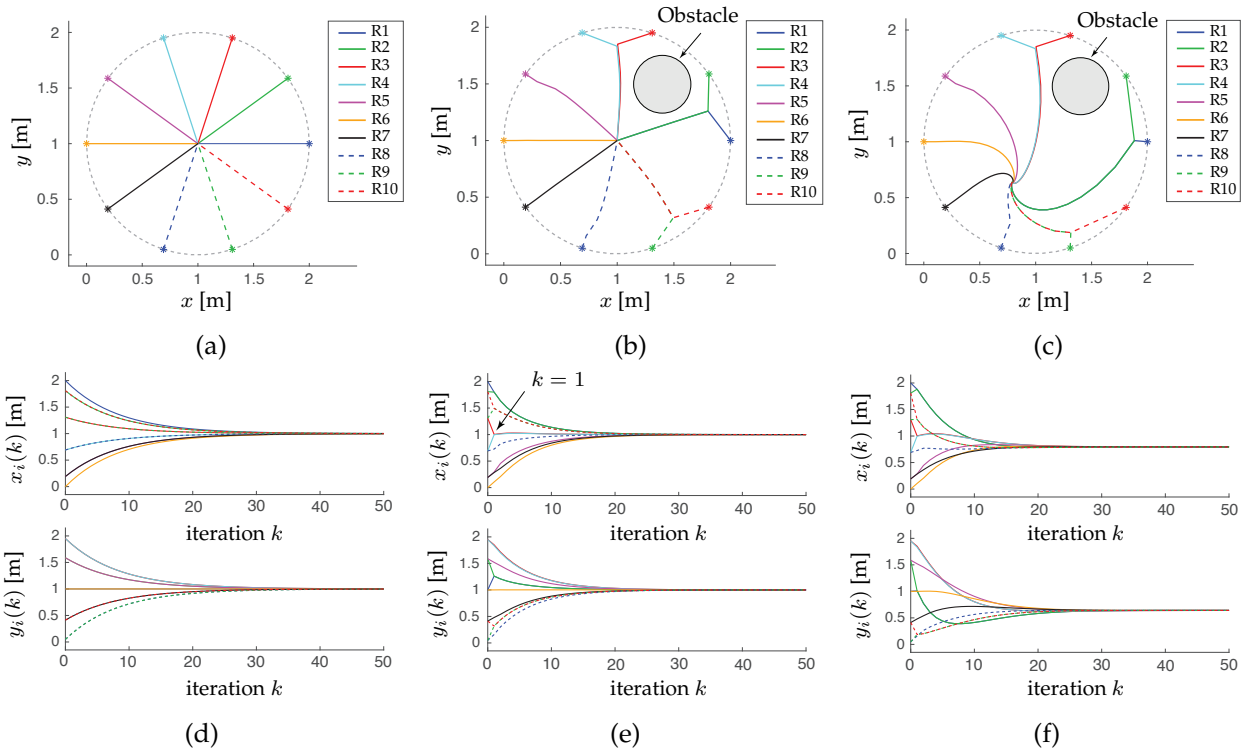


Figure 5.22: *Trajectory of the 10 robots (1st row), and corresponding time-evolution of their x- and y-coordinates (2nd row). In the two rows, the same color convention is used. (a),(d)* Unconstrained consensus protocol *with communication graph $\mathcal{G} = C_{10}$ and step-size $\varepsilon = 0.3$; (b),(e)* Constrained consensus protocol, orthogonal projection onto $\ker(\mathbf{M}_1)$*; (c),(f)* Constrained consensus protocol, weighted oblique projection onto $\ker(\mathbf{M}_1)$.

robots 1 and 2, 3 and 4, and 9 and 10, merge after one iteration (i.e. for $k = 1$), but that the rendezvous point $(1,\, 1)$ is unaffected. Robots 2 and 3 thus avoid the collision with the circular obstacle with radius $r = 0.25$ m, centered at $(1.45,\, 1.5)$, shown in Fig. 5.22(b) (shaded). Fig. 5.22(c) reports the trajectory of the robots generated by the constrained consensus protocol with *weighted oblique projection* (cf. Theorem 5.11),

$$\mathbf{x}_{\text{wob}}(k+1) \;=\; (\mathbf{\Pi}^{\text{ob}}_{\ker(\mathbf{M}_1)}\mathbf{P} \otimes \mathbf{I}_2)\,\mathbf{x}_{\text{wob}}(k), \quad \mathbf{x}_{\text{wob}}(0) \;=\; \mathbf{x}_0,$$

where the diagonal weight matrix $\mathbf{\Omega} \;=\; \text{diag}(50,\, 1,\, 1,\, 1,\, 50, 1,\, 1,\, 1,\, 10,\, 1)$, and Fig. 5.22(f) shows the time-evolution of the positions of the robots. In this case, the rendezvous point is $(0.79486,\, 0.64031)$, cf. equation (5.51). In Figs. 5.23(a),(d) the case of a *general oblique projection*,

$$\mathbf{x}_{\text{ob}}(k+1) \;=\; (\mathbf{\Pi}_{\mathcal{M},\mathcal{N}}\mathbf{P} \otimes \mathbf{I}_2)\,\mathbf{x}_{\text{ob}}(k), \quad \mathbf{x}_{\text{ob}}(0) \;=\; \mathbf{x}_0, \tag{5.54}$$

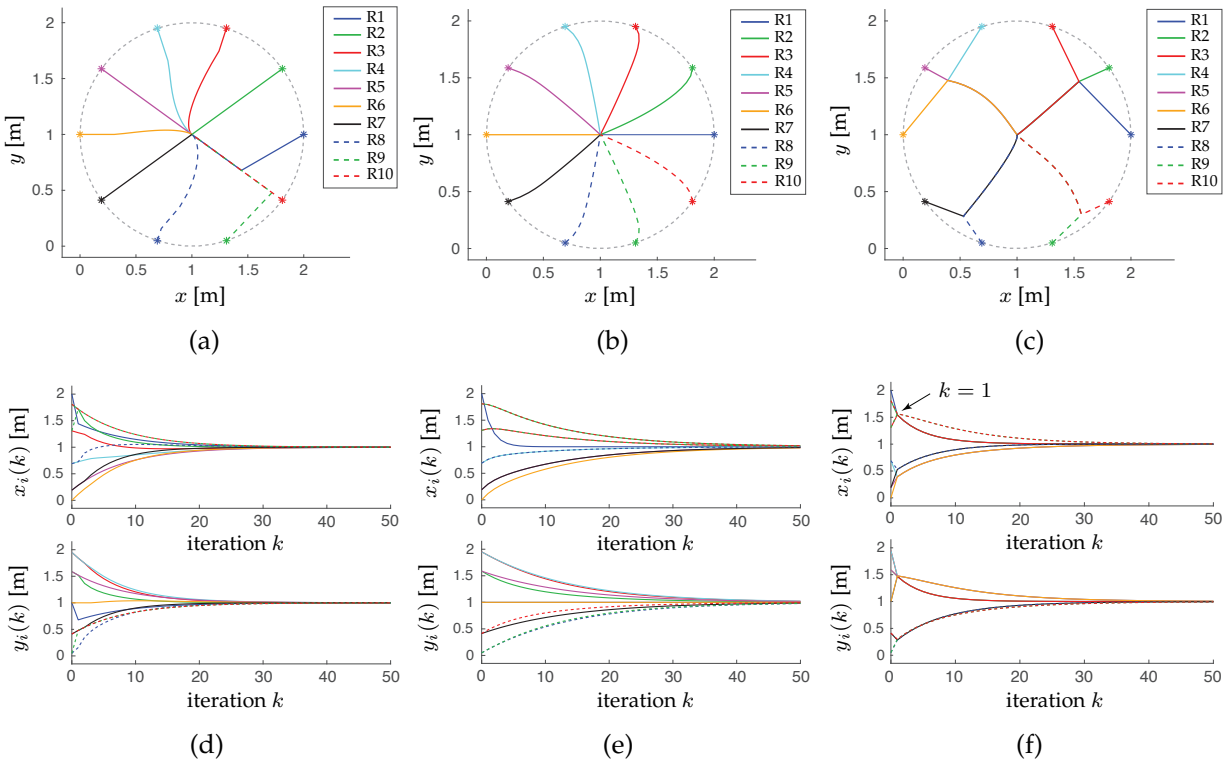


Figure 5.23: *Trajectory of the 10 robots (1st row), and corresponding time-evolution of their x- and y-coordinates (2nd row). In the two rows, the same color convention is used. (a),(d)* Constrained consensus protocol, general oblique projection $\mathbf{\Pi}_{\mathcal{M},\mathcal{N}}$; *(b),(e)* Unconstrained consensus protocol *with communication graph $\mathcal{G} = W_{10}$ and step-size $\varepsilon = 1/20$; (c),(f)* Constrained consensus protocol, orthogonal projection onto $\mathrm{Im}(\mathbf{M}_2^T)$.

is considered (cf. Theorem 5.10). Here, the nine columns of

$$\mathbf{V} = \begin{bmatrix} & & -1 \\ & \mathbf{I}_8 & 0 \\ & & \vdots \\ & & 0 \\ 0 & \cdots & 0 \quad 1 \\ 0 & \cdots & 0 \quad 1 \end{bmatrix}, \quad \mathbf{W} = \begin{bmatrix} & & 0 \\ & \mathbf{I}_8 & 0 \\ & & \vdots \\ & & 0 \\ 1 & \cdots & 0 \quad 0 \\ 0 & \cdots & 0 \quad 1 \end{bmatrix},$$

form a basis for the subspaces $\mathcal{M}, \mathcal{N}^\perp$, respectively, being $\mathcal{N} = \mathrm{span}([1, 0, \ldots, 0, -1, 0]^T)$. Matrices $\mathbf{V}$ and $\mathbf{W}$ satisfy the conditions of Theorem 5.10, and as shown in Fig. 5.23(a), the subspace constraint $\mathcal{M}$ primarily affects the behavior of robots 1 and 9. However, again, the rendezvous point is left unchanged, showing the convergence-preserving nature of protocol (5.54).

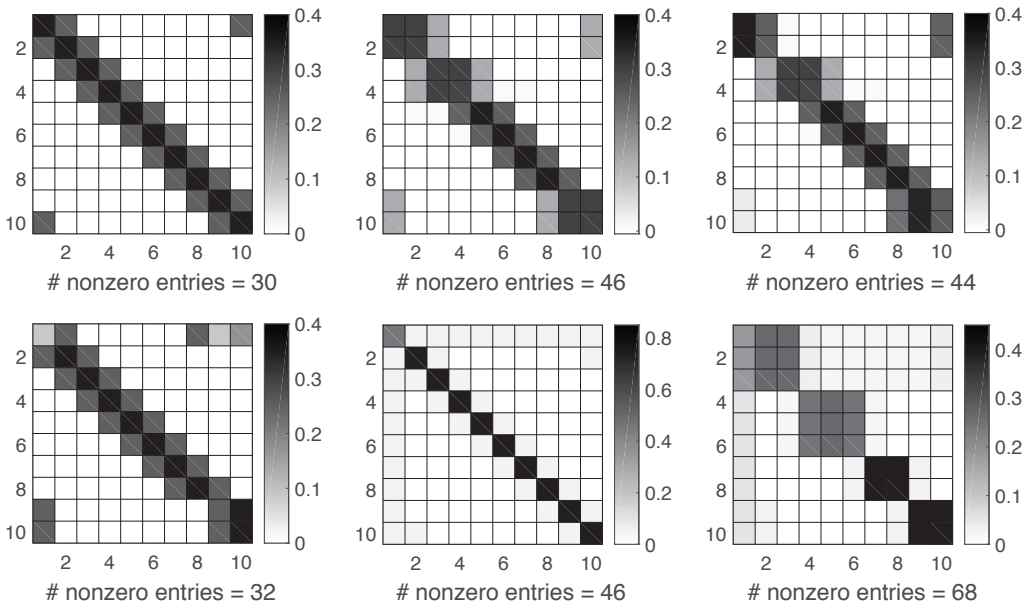


Figure 5.24: *(1st row) Sparsity pattern of matrices* $\mathbf{P}$, $\mathbf{\Pi}_{\mathcal{M}}\mathbf{P}$ *and* $\mathbf{\Pi}^{\text{ob}}_{\ker(\mathbf{M}_1)}$ *considered in Figs. 5.22(a)-(c); (2nd row) Sparsity pattern of matrices* $\mathbf{\Pi}_{\mathcal{M},\mathcal{N}}\mathbf{P}$, $\mathbf{P}$ *and* $\mathbf{\Pi}_{\mathcal{M}}\mathbf{P}$ *considered in Figs. 5.23(a)-(c).*

The trajectory of the robots obtained with protocol (5.52) using a wheel graph $\mathcal{G} = W_{10}$ and a step-size $\varepsilon = 1/20 < 1/d_{\max}(W_{10}) = 1/9$ is reported in Fig. 5.23(b). To draw a comparison, Fig. 5.23(c) shows the constrained trajectories of system (5.53) for $\mathcal{M} = \text{Im}(\mathbf{M}_2^T) \subset \mathbb{R}^{10}$, where

$$\mathbf{M}_2 = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}.$$

Similarly to $\mathbf{M}_1$, matrix $\mathbf{M}_2$ is used here for coupling the motion of two pairs and two triplets of robots. In Figs. 5.23(c),(f), we see that to satisfy the imposed subspace constraint, four robot clusters emerge after one iteration, but that the rendezvous point remains $(1, 1)$. Finally, to get a sense of the amount of communication between the robots required by the six protocols studied in this section, Fig. 5.24 reports the sparsity pattern of the corresponding dynamics matrices (grayscale map of matrix entries).

5.5.5 Conclusions and future work

In this final section of Chapter 5, we leveraged subspace projectors to design a new class of state-constrained consensus protocols. The group decision properties of several

variants of a prototypal algorithm have been studied, and numerical experiments for the constrained 2D rendezvous problem have validated the proposed theory.

In future works, we plan to extend our results to directed communication networks and characterize the convergence speed of new protocols. We also aim at generalizing the proposed theory to time-varying and nonlinear constraints (e.g. by locally projecting the joint state onto the tangent space of the constraint manifold (Lageman and Sun 2016)). Finally, for given a graph topology, we would like to check how restrictive is the rank condition in Theorem 5.9, for projection design.

5.6 Conclusion

In the first part of this chapter, we have studied three families of tasks (environmental monitoring, active target tracking, 3D reconstruction of unknown environments), which can be performed in a distributed fashion via the coordinated action of a group of aerial robots, interacting with the surrounding environment. In the second half of Chapter 5, we have considered the classical consensus protocol from a new angle and proposed two original extensions. We have introduced the so-called *deformed Laplacian* (a generalization of combinatorial Laplacian), and studied the stability properties of the corresponding deformed consensus protocol in terms of real parameter s, by establishing a connection with the well-known quadratic eigenvalue problem. Finally, we have presented a new geometric framework which allows to easily handle state constraints (e.g. arising from obstacles in the environment) in the discrete-time consensus problem, while preserving, under certain assumptions, the consensus value.

In the next chapter, the main contributions of the HDR thesis are briefly stated and possible avenues for future research are discussed.

Chapter 6

Conclusion and future outlook

New ideas pass through three periods: 1) It can't be done.
2) It probably can be done, but it's not worth doing.
3) I knew it was a good idea all along!

Arthur C. Clarke

Abstract

This closing chapter contains a summary of the main contributions of this HDR thesis, followed by a discussion about future research directions. This research activity will be carried out at the MIS laboratory, University of Picardie Jules Verne.

6.1 Summary of contributions

Since my doctoral studies, the focus of my research activity has been on the coordination of groups of autonomous vehicles, with special emphasis on the formation control problem, which I have investigated under different angles. This manuscript embodies the evolution of my research interests over the last decade, and presents the main results in a unified way. More specifically, the original contributions pertaining to the two central chapters of this HDR thesis, can be summarized as follows:

Chapter 4 ("*Formation control of mobile robots*") offers the following contributions:

- A new characterization of mobility for formations of unicycle robots specified by distance-bearing constraints, is presented. By introducing the notion of "macro-robot", the classification by *type* according to the degree of mobility and steerability proposed in (Campion et al. 1996), is extended to multi-robot systems.

- A new decentralized optimal strategy for cooperative adaptive cruise control with guaranteed string stability, is proposed and validated via numerical experiments.

- The general notion of Laplacian function, which traces back its origin to physics and network science (Michelitsch et al. 2019), is revisited in the light of multi-agent systems theory, and its application to shape-based formation control is explored for the first time.

Chapter 5 ("*Coordinated control of multiple robots*") offers the following contributions:

- Original solutions to three classical multi-agent problems, featuring multiple UAVs interacting with the surrounding environment, are presented (environmental monitoring, active target tracking and 3D reconstruction) and validated with numerical simulations and hardware experiments in order to introduce a sufficient degree of realism.

- New insights into the standard consensus protocol are provided. Notably, we have shown how to handle more general Graph Shift Operators (deformed Laplacian) and state constraints arising, for example, from the environment where the robots evolve. The classical 2D rendezvous problem served as a benchmark.

6.2 Future research directions

A cooperative task is a task performed by a team of n autonomous robots *in unison*. The robots should join forces to complete it, by relying on the on-board exteroceptive sensors to detect the surrounding environment and on the information shared with the teammates. As mentioned in Chapter 2, rendezvous, formation control, flocking, patrolling/surveillance, coverage, containment control (encirclement, entrapment and escorting missions alike), manipulation and transport, collaborative localization and SLAM, are typical examples of cooperative tasks for multi-robot systems.

While collision avoidance is an essential prerequisite, a cooperative task is typically subject to a number of additional physical, geometrical and logical constraints (see Fig. 6.1):

1. The robots have *limited sensing* and *communication capabilities*. This means that the knowledge of the world around them is local and incomplete: in fact, they can only perceive the environment in their immediate vicinity and send/receive messages to/from the nearest neighbors. Moreover, communication latency may cause the robots to be less responsive. Hence, there is the need for distributed estimation and control algorithms which do not rely on a single vulnerable coordination unit.

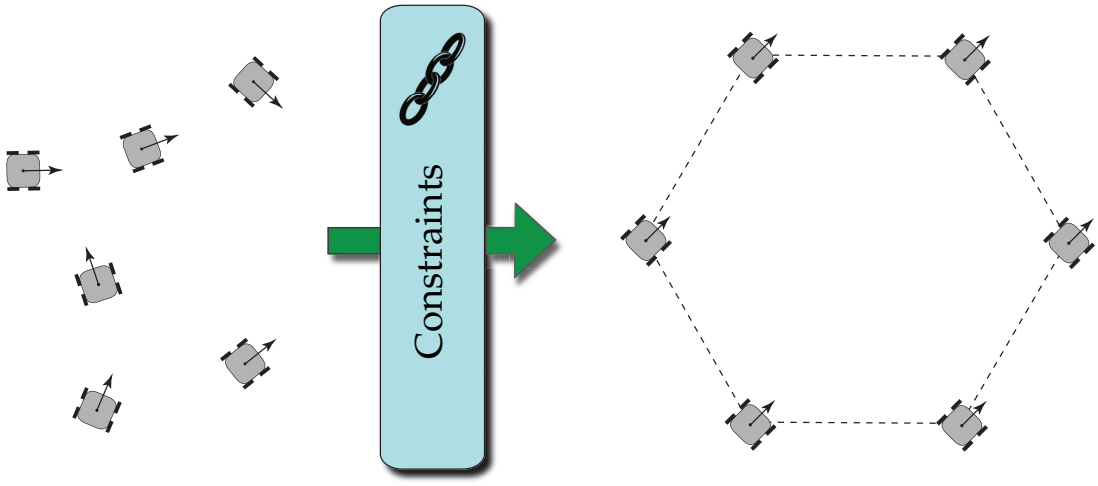


Figure 6.1: *In a standard 2D formation control problem, geometric constraints (relative distances and/or angles) are imposed on the robots. The robots may also have restricted mobility (e.g. unicycle or car-like robots) and limited communication/sensing capabilities.*

2. The robots may have *restricted mobility* (e.g. underactuation or nonholonomic constraints). As a consequence, certain directions of instantaneous motion are forbidden.

3. The *workload* should be equally balanced among the robots and a task assignment (or allocation) problem should be solved beforehand or in real time, during the mission. Collaborative behaviors should be promoted and the idle time of each robot should be minimized.

4. The coordination strategy should be *scalable*, i.e. insensitive to robots joining or leaving the team, *resilient*, i.e. able to quickly recover from unexpected external perturbations (e.g. a wind gust, a kick or other mishaps), and *robust* to disturbances (perception and actuation noise).

5. The multi-robot system should always be *safe* (or *dependable*), i.e. the associated state vector should always remain within a given set, where integrity is preserved. In other words, as long as the state vector lies in this set, we have the guarantee that no instability phenomena occur and that obstacle/inter-robot collisions are avoided.

6. The cooperative task may include *temporal/logic constraints* which specify the order of execution of a sequence of actions or events (e.g. action A should precede action B, and it should not take more than T seconds). Moreover, in certain applica-

tions, *asynchronous communication* between the robots (in contrast to conventional synchronous operation), is the only option.

7. The robots have stringent requirements in terms of *energy storage*. Battery-powered wheeled robots have an autonomy of a few hours, while the flight endurance of electric multi-rotor UAVs is typically up to 25 minutes (in general, in aerial robotics, researchers are faced with "Size Weight and Power" or SWaP constraints). This strongly restricts the class of missions that the robots can carry out.

8. Multi-robot systems may exhibit different sources of *heterogeneity*, which depend on different factors:

   - Type of locomotion: ground robots (wheeled, legged, tracked slip-skid), aerial robots (fixed-wing, rotary-wing, lighter-than-air), surface and underwater robots.
   - Role within the team (e.g. leader-follower, master-slave or shepherd-sheep models, hierarchical and hybrid structures).
   - Budget and available physical resources (e.g. computational power, sensing capabilities, battery capacity, communication range/bandwidth).

Unfortunately, the constraints mentioned above have been taken into account *separately* in the past, and only sparse results exist. For example, **Problem 1** has been addressed by using the well-known consensus protocol (Mesbahi and Egerstedt 2010, Olfati-Saber et al. 2007), and **Problem 2** has been widely studied in the robotics and control literature, especially within the formation-control framework (Egerstedt and Hu 2001, Tabuada et al. 2005, Consolini et al. 2008, Dong and Farrell 2008). For the task allocation problem (**Problem 3**), there is a rich body of work in the literature, see e.g. (Berman et al. 2009, Notomista et al. 2019), while network resilience, robustness and scalability (**Problem 4**) have only been the subject of recent research, see for example (LeBlanc et al. 2013, Guerrero-Bonilla et al. 2017) and (Franceschelli and Frasca 2020) for the so-called "open" multi-agent systems. Safety (**Problem 5**) has been taken into account via different approaches in the literature, the most popular being artificial potential fields, (reciprocal) velocity obstacles, constrained optimization (Alonso-Mora et al. 2019), and more recently control Lyapunov functions and control barrier functions (Wang et al. 2017, Ames et al. 2019). While the problem of asynchronous communication is well known, see (Cao et al. 2008) and (Boyd et al. 2006, Aysal et al. 2009) for gossip algorithms, there is scant research pertaining to temporal/logic constraints (**Problem 6**) in a multi-robot setting, see (Colledanchise and Ögren 2018, Lindemann et al. 2019) and the references therein. Likewise, as per **Problem 7**, only few results exist for a single car-like robot (Tokekar et al. 2014), and single (Morbidi et al. 2016, Zhang

et al. 2021) or multiple (Kemal Ure et al. 2014, Trotta et al. 2018) quadrotor UAVs. Finally, heterogeneous robot teams with agents belonging to different classes (**Problem 8**), have been recently considered in (Santos et al. 2018) for coverage control and in (Schillinger et al. 2018) for task allocation, but there are no results for *generic* cooperative multi-robot tasks.

From the previous discussion, we observe that:

- A *general formulation* encompassing the eight (conflicting) requirements mentioned above, is still missing. The nature and complexity of this formulation has yet to be established: for example, it is unclear how the mobility constraints conciliate and interrelate with the sensing, communication and safety constraints, and which constraint is more "critical" to task achievement. It would be interesting to cast a cooperative multi-robot task as a general nonlinear hierarchical optimal control problem, but it is not evident how to separate perception from action, and how to obtain a fully distributed solution using a principled approach, in general.
- **Problem 1** has been widely studied in the literature, but a fundamental question still remains unanswered. In fact, while distributed algorithms exist for computing generalized (or Hölder) means (Bauso et al. 2006, Cortés 2008) or solving linear equations (Wang et al. 2019), little is known about the *general family of tasks* (or more generally, mathematical operations), which lend themselves to a distributed implementation. We hope to break the decade-long drought of progress and shed new light on this family of tasks: this would provide crucial insights into distributed control and open new avenues for future research.

Other related problems which I am poised to investigate further in the near future, are the following:

1. The emerging field of **graph signal processing** (Sandryhaila and Moura 2014, Ortega et al. 2018, Ortega 2022), with its ramifications in image processing (Cheung et al. 2018, Cheung and Magli 2021) and machine learning (Graph Neural Networks, GNNs (Gama et al. 2020, Ruiz et al. 2020, Wu et al. 2021)), has strong connections with consensus theory (see Figs. 6.2(a),(b)). Several fundamental concepts (such as convolution, filtering or Fourier transform) have been generalized, and the mathematical tools developed in this field allow to process data defined over irregular domains (Shuman et al. 2016). However, their application to robotic networks is still in its infancy (Segarra, Marques and Ribeiro 2017), see also Sect. 4.3.3. GNNs excel in predicting and analyzing graphs, and they have met with more success. They have been recently applied to several classical problems in distributed control: path planning (Li et al. 2020, Li, Lin, Liu and Prorok 2021, Yu et al. 2023),

(a) (b) (c) (d) (e)

Figure 6.2: *Future research directions: (a), (b) Books on graph signal and graph spectral image processing; (c) Prophesee EVK4-HD event camera; (d) Ricoh Theta Z1 dual-fisheye camera; (e) Mixed human-robot team warehousing (image courtesy of* XPO Logistics*).*

target tracking (Zhou, Sharma, Li, Prorok, Ribeiro, Tokekar and Kumar 2022, Tzes et al. 2023), coverage control (Gosrich et al. 2022, Sharma et al. 2023, Agarwal et al. 2024, submitted), flocking (Tolstaya et al. 2020, Kortvelesy and Prorok 2021, Gama et al. 2022, Marino et al. 2023) and collaborative perception (Zhou, Xiao, Zhou and Loianno 2022). In most of these works, the authors trained a neural network with examples from a centralized controller in order to learn a distributed control law for the robots (which is much more difficult to devise, in general).

It is highly desirable to cross-fertilize ideas and build a bridge between robotics, control and signal theory, in order to design new local, asynchronous decision-making policies for the Perception-Action-Communication (PAC) loop (cf. Fig. 2.8).

2. **New sensing modalities** have recently emerged, opening up new opportunities for robotic research. For example, *event-based cameras* (Gallego et al. 2022) hold great promise for the coordination of agile multi-robot systems evolving in environments with challenging light conditions[1] (see Fig. 6.2(c)). In fact, these bio-inspired sensors capture visual information with low latency and minimal redundancy (they produce 10 to 1000 times less data), are less prone to motion blur than traditional cameras, and they correctly operate in high-dynamic-range scenes (Xie et al. 2017), which ushered in their widespread use today. For these reasons, the Robotic Perception group has submitted four research projects on the subject of (omnidirectional) event cameras in 2020-2023: the Evento project (funded by the Agence de l'Innovation de Défense and the UPJV), the ANR CERBERE project,

[1] `www.prophesee.ai/event-based-evaluation-kits`

the ANR DEVIN project and the ANR-FWF (France-Austria) EVELOC project (see Sect. 1.1.5 for more details). The first two projects are about halfway through, and the last two will start in January 2024. In particular, the goal of DEVIN project is to design new coordination algorithms for drone neutralization with a swarms of aerial robots equipped with event cameras, which is perfectly aligned with my personal research program.

The robotic community has also recently welcomed the advent of compact *twin-fisheye cameras* for spherical vision (Caron and Morbidi 2018, Benseddik et al. 2020, Delmas et al. 2023). Their small form factor (Ricoh Theta Z1 camera[2] only weighs 182 grams, see Fig. 6.2(d)) and low power consumption, make them very attractive for micro drones, and their miniaturization potential has yet to fully exploited, e.g. for use on-board insect-like robots (McGuire et al. 2019, Ndoye et al. 2023).

3. Human operators are increasingly confronted with teams of autonomous robots to perform a given task, especially in assembly lines, automated warehouses and industrial logistics (Sabattini et al. 2017, Ackerman 2022). While **mixed human-robot teams** are becoming increasingly popular, it is still poorly understood which is the most effective and intuitive means of communication between people and robots (see Fig. 6.2(e)). In particular, further research is needed to determine which sensory channel (visual, auditory (Che et al. 2020), kinaesthetic (Franchi et al. 2012, Scheggi et al. 2014), or a combination thereof) is the most appropriate, and which is the optimal stimulation frequency/intensity. HMIs (Human Machine Interfaces) tailored to meet the needs of multi-robot systems with a human-in-the-loop, still wait to be discovered.

To put the new knowledges into practice and validate the developed methods, I plan to adopt the same application scenario of the ScanBot project (Morbidi et al. 2018-2021), i.e. cooperative 3D reconstruction of large-scale environments with a heterogeneous team of ground and aerial robots. In fact, this problem is rich and general, it is in line with the scientific objectives of the Robotic Perception group of the MIS laboratory (cf. e-Cathédrale program (Mouaddib et al. 2019, Mouaddib 2015-2025) and EVELOC project). Moreover, it is well suited to highlight the challenges and opportunities of the proposed research program.

With the benefit of hindsight, a few words about **rigidity theory** (Asimow and Roth 1979, Tay and Whiteley 1985) applied to multi-robot formations, are finally in order here (see Fig. 6.3). In Chapter 5 of my Ph.D. thesis (Morbidi 2009), I pointed out that this theory held great promise in multi-agent systems, and it had not been applied to

[2] `https://ricohtheta.eu`

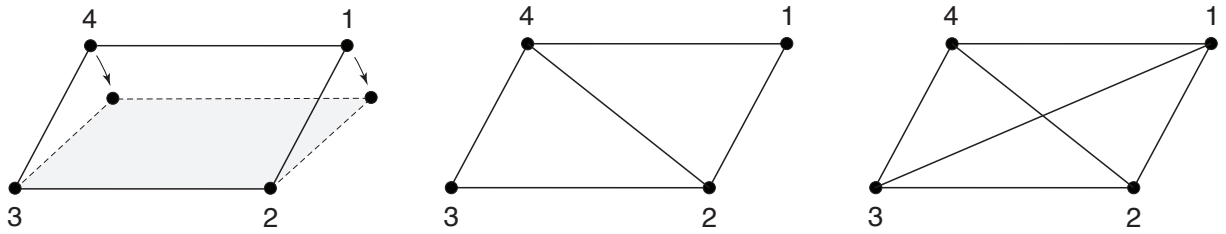


Figure 6.3: *Rigid and nonrigid formations: The formation shown in (a) is not rigid, since it can be deformed by a smooth motion without affecting the distance between the agents (black disks) connected by edges. The formations shown in (b) and (c) are rigid, since they cannot be deformed. In particular, the formation in (b) is* minimally rigid*, because the removal of any edge renders it nonrigid (Anderson et al. 2008).*

formations of nonholonomic robots yet. The foundational work of Eren, Olfati-Saber and coworkers (Eren et al. 2002, Olfati-Saber and Murray 2002), provided the basis for a systematic theory (Cao et al. 2006, Yu et al. 2007, Anderson et al. 2008, Krick et al. 2009), and a vast literature stemmed from it (Zelazo, Franchi, Bülthoff and Robuffo Giordano 2015, Stacey and Mahony 2018, Lin et al. 2022). Fifteen years later, this seminal work continues to inform and inspire, and the field is now mature (Queiroz et al. 2019, Ahn 2020). However, in spite of the resurgence of interest sparked by *bearing rigidity* theory (Zhao and Zelazo 2019, Arrigoni and Fusiello 2019, Michieletto et al. 2021) – in contrast to classical *distance rigidity* theory – the pace of publications has considerably slowed down in recent years. Nevertheless, the potential is far from being exhausted, and several promising research directions have yet to be investigated: for example, the 3D case, cf. (Schiano et al. 2016, Erskine et al. 2024), has not been as extensively explored as the 2D case, and robots with restricted mobility are seldom considered.

Bibliography

Ackerman, E.: 2022, Amazon Shows Off Impressive New Warehouse Robots: It's a lot of progress over just one year, *IEEE Spectrum* . [web]: `https://spectrum.ieee.org/amazon-warehouse-robots`.

Agarwal, S., Ribeiro, A. and Kumar, V.: 2024, submitted, Asynchronous Perception-Action-Communication with Graph Neural Networks, *IEEE Int. Conf. Robot. Automat.* arXiv:2309.10164.

Ahn, H.-S.: 2020, *Formation Control: Approaches for Distributed Agents*, Springer.

Alonso-Mora, J., Baker, S. and Rus, D.: 2017, Multi-robot formation control and object transport in dynamic environments via constrained optimization, *Int. J. Robot. Res.* **36**(9), 1000–1021.

Alonso-Mora, J., Montijano, E., Nägeli, T., Hilliges, O., Schwager, M. and Rus, D.: 2019, Distributed multi-robot formation control in dynamic environments, *Auton. Robot.* **43**, 1079–1100.

Ames, A., Coogan, S., Egerstedt, M., Notomista, G., Sreenath, K. and Tabuada, P.: 2019, Control Barrier Functions: Theory and Applications, *in Proc. European Contr. Conf.*, pp. 3420–3431.

Amigoni, F. and Gallo, A.: 2005, A Multi-Objective Exploration Strategy for Mobile Robots, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 3850–3855.

Anderson, B. and Moore, J.: 1989, *Optimal Control: Linear Quadratic Methods*, Prentice-Hall International.

Anderson, B., Yu, C., Fidan, B. and Hendrickx, J.: 2008, Rigid graph control architectures for autonomous formations, *IEEE Contr. Syst. Mag.* **28**(6), 48–63.

Andersson, L., Imsland, L., Brekke, E. and Scibilia, F.: 2019, On Kalman filtering with linear state equality constraints, *Automatica* **101**, 467–470.

Antonelli, G., Arrichiello, F., Caccavale, F. and Marino, A.: 2014, Decentralized time-varying formation control for multi-robot systems, *Int. J. Robot. Res.* **33**(7), 1029–1043.

Antonelli, G., Arrichiello, F. and Chiaverini, S.: 2008, The Entrapment/Escorting Mission, *IEEE Rob. Autom. Mag.* **15**(1), 22–29.

Aranda, M., Aragüés, R. and López-Nicolás, G.: 2023, Combined Leaderless Control of Translational, Shape-Preserving and Affine Multirobot Formations, *IEEE Robot. Autom. Lett.* **8**(11), 7567–7574.

Aranda, M., López-Nicolás, G. and Mezouar, Y.: 2021, Distributed Linear Control of Multirobot Formations Organized in Triads, *IEEE Robot. Autom. Lett.* **6**(4), 8498–8505.

Arrigo, F., Grindrod, P., Higham, D. and Noferini, V.: 2018, On the exponential generating function for non-backtracking walks, *Linear Algebra Appl.* **556**, 381–399.

Arrigo, F., Higham, D. and Noferini, V.: 2019, Non-Backtracking Alternating Walks, *SIAM J. Appl. Math.* **79**(3), 781–801.

Arrigo, F., Higham, D. and Noferini, V.: 2020, Beyond non-backtracking: non-cycling network centrality measures, *Proc. R. Soc. A* **476**, 20190653.

Arrigoni, F. and Fusiello, A.: 2019, Bearing-Based Network Localizability: A Unifying View, *IEEE Trans. Pattern Anal.* **41**(9), 2049–2069.

Asimow, L. and Roth, B.: 1979, The Rigidity of Graphs, II, *J. Math. Anal. Appl.* **68**(1), 171–190.

Atanasov, N., Le Ny, J., Daniilidis, K. and Pappas, G.: 2015, Decentralized active information acquisition: Theory and application to multi-robot SLAM, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 4775–4782.

Aysal, T., Yildiz, M., Sarwate, A. and Scaglione, A.: 2009, Broadcast gossip algorithms for consensus, *IEEE Trans. Signal Process.* **57**(7), 2748–2761.

Bai, H., Arcak, M. and Wen, J.: 2011, *Cooperative Control Design: A Systematic, Passivity-Based Approach*, Springer.

Balch, T. and Arkin, R.: 1998, Behavior-based Formation Control for Multirobot Teams, *IEEE Trans. Robot. Autom.* **14**(6), 926–939.

Bar-Shalom, Y., Li, X. and Kirubarajan, T.: 2001, *Estimation with Applications to Tracking and Navigation*, John Wiley & Sons.

Batinovic, A., Petrovic, T., Ivanovic, A., Petric, F. and Stjepan, B.: 2021, A Multi-Resolution Frontier-Based Planner for Autonomous 3D Exploration, *IEEE Robot. Autom. Lett.* **6**(3), 4528–4535.

Bauso, D., Giarré, L. and Pesenti, R.: 2006, Non-linear protocols for optimal distributed consensus in networks of dynamic agents, *Syst. Contr. Lett.* **55**(11), 918–928.

Bautista, E., Abry, P. and Gonçalves, P.: 2019, $L^{\gamma}-$PageRank for Semi-Supervised Learning, *Appl. Network Sci.* **4**(1), Article n. 57.

Belta, C. and Kumar, V.: 2004, Abstraction and Control for Groups of Robots, *IEEE Trans. Robot.* **20**(5), 865–875.

Ben-Israel, A. and Greville, T.: 2003, *Generalized Inverses: Theory and Applications*, 2nd edn, Springer.

Benseddik, H.-E., Morbidi, F. and Caron, G.: 2020, PanoraMIS: An ultra-wide field of view image dataset for vision-based robot-motion estimation, *Int. J. Robot. Res.* **39**(9), 1037–1051.

Benzi, M. and Golub, G.: 1999, Bounds for the Entries of Matrix Functions with Applications to Preconditioning, *BIT Numer. Math.* **39**(3), 417–438.

Benzi, M. and Razouk, N.: 2007, Decay Bounds and $O(n)$ Algorithms for Approximating Functions of Sparse Matrices, *Electron. Trans. Numer. Anal.* **28**, 16–39.

Berman, S., Halász, A., Hsieh, M. A. and Kumar, V.: 2009, Optimized Stochastic Policies for Task Allocation in Swarms of Robots, *IEEE Trans. Robot.* **25**(4), 927–937.

Bernstein, D.: 2009, *Matrix Mathematics: Theory, Facts, and Formulas*, Princeton University Press.

Best, G., Cliff, O., Patten, T., Mettu, R. and Fitch, R.: 2016, Decentralised Monte Carlo Tree Search for Active Perception, *in Proc. 12th Int. Workshop Algor. Found. Robot.* Article n. 50.

Bian, A., Buhmann, J., Krause, A. and Tschiatschek, S.: 2017, Guarantees for Greedy Maximization of Non-submodular Functions with Applications, *in Proc. 34th Int. Conf. Mach. Learn.*, pp. 498–507.

Bircher, A., Kamel, M., Alexis, K., Oleynikova, H. and Siegwart, R.: 2016, Receding Horizon "Next-Best-View" Planner for 3D Exploration, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 1462–1468.

Bircher, A., Kamel, M., Alexis, K., Oleynikova, H. and Siegwart, R.: 2018, Receding horizon path planning for 3D exploration and surface inspection, *Auton. Robot.* **42**(2), 291–306.

Blanchini, F., Colaneri, P. and Pellegrino, F.: 2008, Simultaneous performance achievement via compensator blending, *Automatica* **44**(1), 1–14.

Blanchini, F. and Miani, S.: 2015, *Set-Theoretic Methods in Control*, 2nd edn, Birkhäuser.

Border, R., Gammell, J. and Newman, P.: 2018, Surface Edge Explorer (SEE): Planning Next Best Views Directly from 3D Observations, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 6116–6123.

Boyd, S., Ghosh, A., Prabhakar, B. and Shah, D.: 2006, Randomized gossip algorithms, *IEEE Trans. Inf. Theory* **52**(6), 2508–2530.

Boyd, S. and Vandenberghe, L.: 2004, *Convex Optimization*, Cambridge University Press.

Bresenham, J.: 1965, Algorithm for computer control of a digital plotter, *IBM Syst. J.* **4**(1), 25–30.

Brouwer, A. and Haemers, W.: 2012, *Spectra of Graphs*, Universitext, Springer.

Bühler, T. and Hein, M.: 2009, Spectral clustering based on the graph p-Laplacian, *in Proc. 26th Annual Int. Conf. Machine Learning*, pp. 81–88.

Bullo, F.: 2022, *Lectures on Network Systems*, 1st edn, CreateSpace. With contributions by J. Cortes, F. Dörfler, and S. Martínez. [web]: `http://motion.me.ucsb.edu/book-lns`.

Bullo, F., Cortés, J. and Martínez, S.: 2009, *Distributed Control of Robotic Networks*, Applied Mathematics Series, Princeton.

Burgard, W., Moors, M., Stachniss, C. and Schneider, F.: 2005, Coordinated multi-robot exploration, *IEEE Trans. Robot.* **21**(3), 376–386.

Campion, G., Bastin, G. and D'Andréa-Novel, B.: 1996, Structural Properties and Classification of Kinematic and Dynamic Models of Wheeled Mobile Robots, *IEEE Trans. Robot. Autom.* **12**(1), 47–62.

Cao, M., Morse, A. and Anderson, B.: 2006, Sensor network localization with imprecise distances, *Syst. Contr. Lett.* **55**(11), 887–893.

Cao, M., Morse, A. and Anderson, B.: 2008, Agreeing Asynchronously, *IEEE Trans. Automat. Contr.* **53**(8), 1826–1838.

Cao, Y., Li, Y., Ren, W. and Chen, Y.: 2010, Distributed Coordination of Networked Fractional-Order Systems, *IEEE Trans. Systs. Man Cy. B* **40**(2), 362–370.

Caprari, G., Breitenmoser, A., Fischer, W., Hürzeler, C., Tâche, F., Siegwart, R., Nguyen, O., Moser, R., Schoeneich, P. and Mondada, F.: 2012, Highly Compact Robots for Inspection of Power Plants, *J. Field Robot.* **29**(1), 47–68.

CarMaker, IPG Automotive: 2023. [web]: `https://ipg-automotive.com/en/products-solutions/software/carmaker`.

Caron, G. and Morbidi, F.: 2018, Spherical Visual Gyroscope for Autonomous Robots Using the Mixture of Photometric Potentials, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 820–827.

Casbeer, D., Kingston, D., Beard, R., Mclain, T., Li, S. and Mehra, R.: 2006, Cooperative forest fire surveillance using a team of small unmanned air vehicles, *Int. J. Syst. Sci.* **37**(6), 351–360.

Charrow, B., Liu, S., Kumar, V. and Michael, N.: 2015, Information-theoretic mapping using Cauchy-Schwarz Quadratic Mutual Information, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 4791–4798.

Chaslot, G.-B.: 2010, *Monte-Carlo Tree Search*, PhD thesis, Maastricht University.

Che, Y., Okamura, A. and Sadigh, D.: 2020, Efficient and Trustworthy Social Navigation via Explicit and Implicit Robot-Human Communication, *IEEE Trans. Robot.* **36**(3), 692–707.

Chen, L., Mei, J., Li, C. and Ma, G.: 2020, Distributed Leader-Follower Affine Formation Maneuver Control for High-Order Multiagent Systems, *IEEE Trans. Automat. Contr.* **65**(11), 4941–4948.

Chen, X., Qi, H., Qi, L. and Teo, K.: 2004, Smooth convex approximation to the maximum eigenvalue function, *J. Global Optim.* **30**(2), 253–270.

Cheung, G. and Magli, E. (eds): 2021, *Graph Spectral Image Processing*, Wiley-ISTE.

Cheung, G., Magli, E., Tanaka, T. and Ng, M.: 2018, Graph Spectral Image Processing, *Proc. IEEE* **106**(5), 907–930.

Chipade, V. and Panagou, D.: 2023, Aerial Swarm Defense Using Interception and Herding Strategies, *IEEE Trans. Robot.* **39**(5), 3821–3837.

Choi, J., Oh, S. and Horowitz, R.: 2009, Distributed learning and cooperative control for multi-agent systems, *Automatica* **45**(12), 2802–2814.

Chung, T., Burdick, J. and Murray, R.: 2006, A decentralized motion coordination strategy for dynamic target tracking, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 2416–2422.

Chung, T., Gupta, V., Burdick, J. and Murray, R.: 2004, On a Decentralized Active Sensing Strategy using Mobile Sensor Platforms in a Network, *in Proc. 43rd IEEE Conf. Dec. Contr.*, pp. 1914–1919.

Chung, W. and Iagnemma, K.: 2016, Wheeled Robots, *in* B. Siciliano and O. Khatib (eds), *Handbook of Robotics*, 2nd edn, Springer, chapter 24, pp. 575–594.

Clark, J. and Fierro, R.: 2007, Mobile robotic sensors for perimeter detection and tracking, *ISA Trans.* **46**(1), 3–13.

Clavien, L., Lauria, M. and Michaud, F.: 2018, Estimation of the instantaneous centre of rotation with nonholonomic omnidirectional mobile robots, *Robot. Autonom. Syst.* **106**, 47–57.

Colaneri, P., Geromel, J. and Locatelli, A.: 1997, *Control Theory and Design: an RH_2 and RH_∞ Viewpoint*, Academic Press.

Colledanchise, M. and Ögren, P.: 2018, *Behavior Trees in Robotics and AI: An Introduction*, CRC Press.

Connolly, C.: 1985, The Determination of Next Best Views, *in Proc. IEEE Int. Conf. Robot. Automat.*, Vol. 2, pp. 432–435.

Consolini, L., Morbidi, F., Prattichizzo, D. and Tosques, M.: 2008, Leader-Follower Formation Control of Nonholonomic Mobile Robots with Input Constraints, *Automatica* **44**(5), 1343–1349.

Consolini, L., Morbidi, F., Prattichizzo, D. and Tosques, M.: 2009, Stabilization of a Hierarchical Formation of Unicycle Robots with Velocity and Curvature Constraints, *IEEE Trans. Robot.* **25**(5), 1176–1184.

Consolini, L., Morbidi, F., Prattichizzo, D. and Tosques, M.: 2012, On a Class of Hierarchical Formations of Unicycles and Their Internal Dynamics, *IEEE Trans. Automat. Contr.* **57**(4), 845–859.

Corah, M. and Michael, N.: 2019, Distributed matroid-constrained submodular maximization for multi-robot exploration: Theory and practice, *Auton. Robot.* **43**(2), 485–501.

Corah, M. and Michael, N.: 2021, Volumetric Objectives for Multi-Robot Exploration of Three-Dimensional Environments, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 9043–9050.

Corah, M., O'Meadhra, C., Goel, K. and Michael, N.: 2019, Communication-Efficient Planning and Mapping for Multi-Robot Exploration in Large Environments, *IEEE Robot. Autom. Lett.* **4**(2), 1715–1721.

Cortés, J.: 2008, Distributed algorithms for reaching consensus on general functions, *Automatica* **44**(3), 726–737.

Cortés, J. and Egerstedt, M.: 2017, Cooperative Heterogeneous Multi-Robot Systems: A Survey, *SICE J. Control Meas. Syst. Integr.* **10**(6), 495–503.

Cortés, J., Martínez, S., Karatas, T. and Bullo, F.: 2004, Coverage Control for Mobile Sensing Networks, *IEEE Trans. Robot. Autom.* **20**(2), 243–255.

Şucan, I., Moll, M. and Kavraki, L.: 2012, The Open Motion Planning Library, *IEEE Rob. Autom. Mag.* **19**(4), 72–82.

Curless, B. and Levoy, M.: 1996, A Volumetric Method for Building Complex Models from Range Images, *in Proc. 23rd Annual Conf. Comput. Graph. Interact. Tech.*, pp. 303–312.

Cvetković, D., Rowlinson, P. and Simić, S.: 2007, Signless Laplacians of finite graphs, *Linear Algebra Appl.* **423**, 155–171.

Dall'Amico, L., Couillet, R. and Tremblay, N.: 2019, Revisiting the Bethe-Hessian: Improved Community Detection in Sparse Heterogeneous Graphs, *in Proc. 33rd Conf. Neural Inf. Process. Syst.*, pp. 4039–4049.

Das, A., Fierro, R., Kumar, V., Ostrowsky, J., Spletzer, J. and Taylor, C.: 2002, A Vision-Based Formation Control Framework, *IEEE Trans. Robot. Autom.* **18**(5), 813–825.

Dasoulas, G., Lutzeyer, J. and Vazirgiannis, M.: 2021, Learning Parametrised Graph Shift Operators, *in Proc. 9th Int. Conf. Learn. Repres.*

Delmas, S., Morbidi, F., Caron, G., Babel, M. and Pasteau, F.: 2023, SpheriCol: A Driving Assistant for Power Wheelchairs Based on Spherical Vision, *IEEE Trans. Med. Rob. Bionics* **5**(2), 387–400.

Demmel, J.: 1997, *Applied Numerical Linear Algebra*, SIAM.

Dimarogonas, D. and Kyriakopoulos, K.: 2008, Connectedness Preserving Distributed Swarm Aggregation for Multiple Kinematic Robots, *IEEE Trans. Robot.* **24**(5), 1213–1223.

Dong, W. and Farrell, J.: 2008, Cooperative Control of Multiple Nonholonomic Mobile Agents, *IEEE Trans. Automat. Contr.* **53**(6), 1434–1448.

Dubins, L.: 1957, On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents, *American J. Math.* **79**(3), 497–516.

Duhautbout, T., Moras, J. and Marzat, J.: 2019, Distributed 3D TSDF Manifold Mapping for Multi-Robot Systems, *in Proc. 9th Europ. Conf. Mobile Robots*, pp. 1–8.

Edwards, D., Bean, T., Odell, D. and Anderson, M.: 2004, A Leader-Follower Algorithm for Multiple AUV Formations, *in Proc. IEEE/OES Autonomous Underwater Vehicles*, pp. 40–46.

Egerstedt, M. and Hu, X.: 2001, Formation constrained multi-agent control, *IEEE Trans. Robot. Autom.* **17**(6), 947–951.

Engel, J., Stückler, J. and Cremers, D.: 2015, Large-scale direct SLAM with stereo cameras, *in Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, pp. 1935–1942.

Eren, T., Belhumeur, P., Anderson, B. and Morse, A.: 2002, A framework for maintaining formations based on rigidity, *in Proc. 15th IFAC World Congress*, pp. 2752–2757.

Erskine, J., Briot, S., Fantoni, I. and Chriette, A.: 2024, Singularity Analysis of Rigid Directed Bearing Graphs for Quadrotor Formations, *IEEE Trans. Robot.* **40**, 139–157.

Fagnani, F. and Frasca, P.: 2018, *Introduction to Averaging Dynamics over Networks*, Springer.

Fallat, S. and Fan, Y.-Z.: 2012, Bipartiteness and the least eigenvalue of signless Laplacian of graphs, *Linear Algebra Appl.* **436**(9), 3254–3267.

Fanuel, M., Alaíz, C., Fernández, A. and Suykens, J.: 2018, Magnetic Eigenmaps for the visualization of directed networks, *Appl. Comput. Harmon. Anal.* **44**(1), 189–199.

Fanuel, M. and Suykens, J.: 2019, Deformed Laplacians and spectral ranking in directed networks, *Appl. Comput. Harmon. Anal.* **47**(2), 397–422.

Farinelli, A., Boscolo, N., Zanotto, E. and Pagello, E.: 2017, Advanced approaches for multi-robot coordination in logistic scenarios, *Robot. Autonom. Syst.* **90**, 34–44.

Fathian, K., Safaoui, S., Summers, T. and Gans, N.: 2021, Robust Distributed Planar Formation Control for Higher Order Holonomic and Nonholonomic Agents, *IEEE Trans. Robot.* **37**(1), 185–205.

Fisher, M., Nemhauser, G. and Wolsey, L.: 1978, An Analysis of Approximations for Maximizing Submodular Set Functions - II, *Math. Program. Stud.* **8**, 73–87.

Fontan, A., Shi, G., Hu, X. and Altafini, C.: 2019, Interval Consensus for Multiagent Networks, *IEEE Trans. Automat. Contr.* **65**(5), 1855–1869.

Fouss, F., Saerens, M. and Shimbo, M.: 2016, *Algorithms and Models for Network Data and Link Analysis*, Cambridge University Press.

Franceschelli, M. and Frasca, P.: 2020, Stability of Open Multiagent Systems and Applications to Dynamic Consensus, *IEEE Trans. Automat. Contr.* **66**(5), 2326–2331.

Franchi, A., Secchi, C., Son, H., Bulthoff, H. and Robuffo Giordano, P.: 2012, Bilateral Teleoperation of Groups of Mobile Robots With Time-Varying Topology, *IEEE Trans. Robot.* **28**(5), 1019–1033.

Franchi, A., Stegagno, P. and Oriolo, G.: 2016, Decentralized multi-robot encirclement of a 3D target with guaranteed collision avoidance, *Auton. Robot.* **40**(2), 245–265.

Francis, B. and Maggiore, M.: 2016, *Flocking and Rendezvous in Distributed Robotics*, Springer.

Frasca, P., Carli, R., Fagnani, F. and Zampieri, S.: 2009, Average consensus on networks with quantized communication, *Int. J. Robust Nonlin. Contr.* **19**(16), 1787–1816.

Fuhrmann, P. and Helmke, U.: 2015, *The Mathematics of Networks of Linear Systems*, Springer.

Furrer, F., Burri, M., Achtelik, M. and Siegwart, R.: 2016, RotorS – A Modular Gazebo MAV Simulator Framework, *in* A. Koubaa (ed.), *Robot Operating System (ROS)*, Vol. 625, Springer, pp. 595–625.

Galántai, A.: 2004, *Projectors and Projection Methods*, Vol. 6 of *Advances in Mathematics*, Springer.

Gallego, G., Delbrück, T., Orchard, G., Bartolozzi, C., Taba, B., Censi, A., Leutenegger, S., Davison, A., Conradt, J., Daniilidis, K. and Scaramuzza, D.: 2022, Event-Based Vision: A Survey, *IEEE Trans. Pattern Anal.* **44**(1), 154–180.

Gama, F., Isufi, E., Leus, G. and Ribeiro, A.: 2020, Graphs, Convolutions, and Neural Networks: From Graph Filters to Graph Neural Networks, *IEEE Signal Process. Mag.* **37**(6), 128–138.

Gama, F., Li, Q., Tolstaya, E., Prorok, A. and Ribeiro, A.: 2022, Synthesizing Decentralized Controllers With Graph Neural Networks and Imitation Learning, *IEEE Trans. Signal Process.* **70**, 1932–1946.

Gantmacher, F.: 2000, *The Theory of Matrices*, Vol. 2, AMS Chelsea Publishing.

García de Marina, H., Jayawardhana, B. and Cao, M.: 2016, Distributed rotational and translational maneuvering of rigid formations and their applications, *IEEE Trans. Robot.* **32**(3), 684–697.

Gasparri, A., Sabattini, L. and Ulivi, G.: 2017, Bounded Control Law for Global Connectivity Maintenance in Cooperative Multirobot Systems, *IEEE Trans. Robot.* **33**(3), 700–717.

Geiger, A., Roser, M. and Urtasun, R.: 2010, Efficient Large-Scale Stereo Matching, *in Proc. Asian Conf. Comp. Vis.*, pp. 25–38.

Gerndt, R., Seifert, D., Baltes, J., Sadeghnejad, S. and Behnke, S.: 2015, Humanoid robots in soccer: Robots versus humans in RoboCup 2050, *IEEE Rob. Autom. Mag.* **22**(3), 147–154.

Ghassemi, P. and Chowdhury, S.: 2022, Multi-robot task allocation in disaster response: Addressing dynamic tasks with deadlines and robots with range and payload constraints, *Robot. Autonom. Syst.* **147**, 103905.

Godsil, C. and Royle, G.: 2001, *Algebraic Graph Theory*, Springer.

Goldberg, F. and Kirkland, S.: 2014, On the sign patterns of the smallest signless Laplacian eigenvector, *Linear Algebra Appl.* **443**, 66–85.

Gong, C., Liu, T., Tao, D., Fu, K., Tu, E. and Yang, J.: 2015, Deformed Graph Laplacian for Semisupervised Learning, *IEEE Trans. Neur. Net. Lear.* **26**(10), 2261–2274.

González-Banos, H. and Latombe, J.-C.: 2002, Navigation Strategies for Exploring Indoor Environments, *Int. J. Robot. Res.* **21**(10-11), 829–848.

Gosrich, W., Mayya, S., Li, R., Paulos, J., Yim, M., Ribeiro, A. and Kumar, V.: 2022, Coverage Control in Multi-Robot Systems via Graph Neural Networks, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 8787–8793.

Gowal, S., Prorok, A. and Martinoli, A.: 2011, Two-Phase Online Calibration for Infrared-based Inter-Robot Positioning Modules, *in Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, pp. 3313–3319.

Grindrod, P., Higham, D. and Noferini, V.: 2018, The Deformed Graph Laplacian and Its Applications to Network Centrality Analysis, *SIAM J. Matrix Anal. A.* **39**(1), 310–341.

Grocholsky, B., Keller, J., Kumar, V. and Pappas, G.: 2006, Cooperative air and ground surveillance, *IEEE Rob. Autom. Mag.* **13**(3), 16–25.

Gu, K., Kharitonov, V. and Chen, J.: 2003, *Stability of Time-Delay Systems*, Birkhäuser.

Guerrero-Bonilla, L., Prorok, A. and Kumar, V.: 2017, Formations for Resilient Robot Teams, *IEEE Robot. Autom. Lett.* **2**(2), 841–848.

Guo, G. and Yue, W.: 2012, Autonomous Platoon Control Allowing Range-Limited Sensors, *IEEE Trans. Veh. Technol.* **61**(7), 2901–2912.

Gustavi, T. and Hu, X.: 2008, Observer-Based Leader-Following Formation Control using Onboard Sensor Information, *IEEE Trans. Robot.* **24**(6), 1457–1462.

Hadjicostis, C., Domínguez-García, A. and Charalambous, T.: 2018, Distributed Averaging and Balancing in Network Systems: with Applications to Coordination and Control, Vol. 5 of *Foundations and Trends® in Systems and Control*, Now Publishers, pp. 99–292.

Hardouin, G.: 2022, *A centralized and distributed multi-robot system for 3D surface reconstruction of unknown environments*, PhD thesis, Université de Picardie Jules Verne, Amiens, France. [web]: `www.theses.fr/s252746`.

Hardouin, G., Moras, J., Morbidi, F., Marzat, J. and Mouaddib, E.: 2020, Next-Best-View planning for surface reconstruction of large-scale 3D environments with multiple UAVs, *in Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, pp. 1567–1574.

Hardouin, G., Moras, J., Morbidi, F., Marzat, J. and Mouaddib, E.: 2023, A Multirobot System for 3-D Surface Reconstruction With Centralized and Distributed Architectures, *IEEE Trans. Robot.* **39**(4), 2623–2638.

Hardouin, G., Morbidi, F., Moras, J., Marzat, J. and Mouaddib, E.: 2020, Surface-driven Next-Best-View planning for exploration of large-scale 3D environments, *in Proc. IFAC World Cong.*, pp. 15501–15507.

Hatanaka, T., Chopra, N., Fujita, M. and Spong, M.: 2015, *Passivity-Based Control and Estimation in Networked Robotics*, Springer.

Hauser, K.: 2015, Lazy Collision Checking in Asymptotically-Optimal Motion Planning, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 2951–2957.

Helsgaun, K.: 2000, An effective implementation of the Lin–Kernighan traveling salesman heuristic, *Eur. J. Oper. Rer.* **126**(1), 106–130.

Henrot, A.: 2006, *Extremum Problems for Eigenvalues of Elliptic Operators*, Frontiers in Mathematics, Birkhäuser.

Higham, N.: 2008a, *Functions of Matrices: Theory and Computation*, SIAM.

Higham, N.: 2008b, The Matrix Function Toolbox. [web] `www.maths.manchester.ac.uk/~higham/mftoolbox`.

Hornung, A., Wurm, K., Bennewitz, M., Stachniss, C. and Burgard, W.: 2013, OctoMap: An efficient probabilistic 3D mapping framework based on octrees, *Auton. Robot.* **34**(3), 189–206.

Hou, Z. and Fantoni, I.: 2017, Interactive LeaderFollower Consensus of Multiple Quadrotors Based on Composite Nonlinear Feedback Control, *IEEE Trans. Contr. Syst. Tech.* **26**(5), 1732–1743.

Howard, A.: 2004, Multi-robot mapping using manifold representations, *in Proc. IEEE Int. Conf. Robot. Automat.*, Vol. 4, pp. 4198–4203.

Howard, A., Matarić, M. and Sukhatme, G.: 2002, An Incremental Self-Deployment Algorithm for Mobile Sensor Networks, *Auton. Robot.* **13**(2), 113–126.

Hu, J., Bhowmick, P. and Lanzon, A.: 2021, Group Coordinated Control of Networked Mobile Robots With Applications to Object Transportation, *IEEE Trans. Veh. Technol.* **70**(8), 8269–8274.

Hutin, N., Pégard, C. and Brassart, E.: 1998, A communication strategy for cooperative robots, *in Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, Vol. 1, pp. 114–119.

Jadbabaie, A., Lin, J. and Morse, A.: 2003, Coordination of groups of mobile autonomous agents using nearest neighbor rules, *IEEE Trans. Automat. Contr.* **48**(6), 988–1001.

Jafarpour, S. and Bullo, F.: 2019, Synchronization of Kuramoto Oscillators via Cutset Projections, *IEEE Trans. Automat. Contr.* **64**(7), 2830–2844.

Jawaid, S. and Smith, S.: 2015, Informative path planning as a maximum traveling salesman problem with submodular rewards, *Discrete Appl. Math.* **186**, 112–127.

Ji, M., Ferrari-Trecate, G., Egerstedt, M. and Buffa, A.: 2008, Containment Control in Mobile Networks, *IEEE Trans. Automat. Contr.* **53**(8), 1972–1975.

Jin, Z. and Murray, R.: 2006, Multi-Hop Relay Protocols for Fast Consensus Seeking, *in Proc. 45th IEEE Conf. Dec. Contr.*, pp. 1001–1006.

Juliá, M., Gil, A. and Reinoso, O.: 2012, A comparison of path planning strategies for autonomous exploration and mapping of unknown environments, *Auton. Robot.* **33**(4), 427–444.

Kamel, M., Stastny, T., Alexis, K. and Siegwart, R.: 2017, Model Predictive Control for Trajectory Tracking of Unmanned Aerial Vehicles Using Robot Operating System, *in* A. Koubaa (ed.), *Robot Operating System (ROS)*, Vol. 707, Springer, pp. 3–39.

Kar, S. and Moura, J.: 2009, Distributed Consensus Algorithms in Sensor Networks with Imperfect Communication: Link Failures and Channel Noise, *IEEE Trans. Signal Process.* **57**(1), 355–369.

Karaman, S. and Frazzoli, E.: 2011, Sampling-based algorithms for optimal motion planning, *Int. J. Robot. Res.* **30**(7), 846–894.

Kavraki, L., Svestka, P., Latombe, J.-C. and Overmars, M.: 1996, Probabilistic roadmaps for path planning in high-dimensional configuration spaces, *IEEE Trans. Robot. Autom.* **12**(4), 566–580.

Kemal Ure, N., Chowdhary, G., Toksoz, T., How, J., Vavrina, M. and Vian, J.: 2014, An Automated Battery Management System to Enable Persistent Missions With Multiple Aerial Vehicles, *IEEE/ASME Trans. Mechatron.* **20**(1), 275–286.

Keselman, L., Woodfill, J., Grunnet-Jepsen, A. and Bhowmik, A.: 2017, Intel RealSense Stereoscopic Depth Cameras, *Proc. IEEE Conf. Comp. Vis. Pattern Recogn. Workshops*, pp. 1–10.

Kia, S., Scoy, B. V., Cortés, J., Freeman, R., Lynch, K. and Martínez, S.: 2019, Tutorial on Dynamic Average Consensus: The Problem, its Applications, and the Algorithms, *IEEE Contr. Syst. Mag.* **39**(3), 40–72.

Klingensmith, M., Dryanovski, I., Srinivasa, S. and Xiao, J.: 2015, CHISEL: Real Time Large Scale 3D Reconstruction Onboard a Mobile Device using Spatially Hashed Signed Distance Fields, *in Proc. Robotics: Science and Systems XI*, Vol. 4. Article n. 40.

Kolmanovsky, I. and McClamroch, N.: 1995, Developments in nonholonomic control problems, *IEEE Contr. Syst. Mag.* **15**(6), 20–36.

Kompis, Y., Bartolomei, L., Mascaro, R., Teixeira, L. and Chli, M.: 2021, Informed Sampling Exploration Path Planner for 3D Reconstruction of Large Scenes, *IEEE Robot. Autom. Lett.* **6**(4), 7893–7900.

Kortvelesy, R. and Prorok, A.: 2021, ModGNN: Expert Policy Approximation in Multi-Agent Systems with a Modular Graph Neural Network Architecture, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 9161–9167.

Koung, D., Kermorgant, O., Fantoni, I. and Belouaer, L.: 2021, Cooperative Multi-Robot Object Transportation System Based on Hierarchical Quadratic Programming, *IEEE Robot. Autom. Lett.* **6**(4), 6466–6472.

Krick, L., Broucke, M. and Francis, B.: 2009, Stabilisation of infinitesimally rigid formations of multi-robot networks, *Int. J. Control* **82**(3), 423–439.

Kwok, A. and Martínez, S.: 2010, Unicycle Coverage Control via Hybrid Modeling, *IEEE Trans. Automat. Contr.* **55**(2), 528–532.

Lageman, C. and Sun, Z.: 2016, Consensus on spheres: convergence analysis and perturbation theory, *in Proc. 55th IEEE Conf. Dec. Contr.*, pp. 19–24.

Lague, D., Brodu, N. and Leroux, J.: 2013, Accurate 3D comparison of complex topography with terrestrial laser scanner: Application to the Rangitikei canyon (N-Z), *ISPRS J. photogramm.* **82**, 10–26.

Lalish, E., Morgansen, K. and Tsukamaki, T.: 2007, Oscillatory Control for Constant-Speed Unicycle-Type Vehicles, *in Proc. 46th IEEE Conf. Dec. Contr.*, pp. 5246–5251.

Lancaster, P. and Tismenetsky, M.: 1985, *The Theory of Matrices: with Applications*, 2nd edn, Academy Press.

Lauri, M., Pajarinen, J., Peters, J. and Frintrop, S.: 2020, Multi-Sensor Next-Best-View Planning as Matroid-Constrained Submodular Maximization, *IEEE Robot. Autom. Lett.* **5**(4), 5323–5330.

LaValle, S. and Kuffner Jr, J.: 2001, Randomized Kinodynamic Planning, *Int. J. Robot. Res.* **20**(5), 378–400.

Lawton, J., Beard, R. and Young, B.: 2003, A Decentralized Approach to Formation Maneuvers, *IEEE Trans. Robot. Autom.* **19**(6), 933–941.

LeBlanc, H., Zhang, H., Koutsoukos, X. and Sundaram, S.: 2013, Resilient Asymptotic Consensus in Robust Networks, *IEEE J. Sel. Areas Commun.* **31**(4), 766–781.

Léchevin, N., Rabbath, C. and Sicard, P.: 2006, Trajectory tracking of leader-follower formations characterized by constant line-of-sight angles, *Automatica* **42**(12), 2131–2141.

Lee, U. and Mesbahi, M.: 2011, Constrained Consensus via Logarithmic Barrier Functions, *in Proc. 50th IEEE Conf. Dec. Contr.*, pp. 3608–3613.

Leonard, N., Paley, D., Lekien, F., Sepulchre, R., Fratantoni, D. and Davis, R.: 2007, Collective Motion, Sensor Networks, and Ocean Sampling, *Proc. IEEE* **95**(1), 48–74.

Levie, R., Monti, F., Bresson, X. and Bronstein, M.: 2019, CayleyNets: Graph Convolutional Neural Networks With Complex Rational Spectral Filters, *IEEE Trans. Signal Process.* **67**(1), 97–109.

Lewis, F., Zhang, H., Hengster-Movric, K. and Das, A.: 2014, *Cooperative Control of Multi-Agent Systems: Optimal and Adaptive Design Approaches*, Springer.

Li, B. and Ma, H.: 2023, Double-Deck Multi-Agent Pickup and Delivery: Multi-Robot Rearrangement in Large-Scale Warehouses, *IEEE Robot. Autom. Lett.* **8**(6), 3701–3708.

Li, F. and Ng, M.: 2019, Image colorization by using graph bi-Laplacian, *Adv. Comput. Math.* **45**(3), 1521–1549.

Li, G., Ge, R. and Loianno, G.: 2021, Cooperative Transportation of Cable Suspended Payloads With MAVs Using Monocular Vision and Inertial Sensing, *IEEE Robot. Autom. Lett.* **6**(3), 5316–5323.

Li, Q., Gama, F., Ribeiro, A. and Prorok, A.: 2020, Graph Neural Networks for Decentralized Multi-Robot Path Planning, *in Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, pp. 11785–11792.

Li, Q., Lin, W., Liu, Z. and Prorok, A.: 2021, Message-Aware Graph Attention Networks for Large-Scale Multi-Robot Path Planning, *IEEE Robot. Autom. Lett.* **6**(3), 5533–5540.

Li, S., Li, K., Rajamani, R. and Wang, J.: 2011, Model Predictive Multi-Objective Vehicular Adaptive Cruise Control, *IEEE Trans. Contr. Syst. Tech.* **19**(3), 556–566.

Lin, Y., Lin, Z., Sun, Z. and Anderson, B.: 2022, A Unified Approach for Finite-Time Global Stabilization of Affine, Rigid, and Translational Formation, *IEEE Trans. Automat. Contr.* **67**(4), 1869–1881.

Lindemann, L., Nowak, J., Schönbächler, L., Guo, M., Tumova, J. and Dimarogonas, D.: 2019, Coupled Multi-Robot Systems Under Linear Temporal Logic and Signal Temporal Logic Tasks, *IEEE Trans. Contr. Syst. Tech.* **29**(2), 858–865.

Lischke, A., Pang, G., Gulian, M., Song, F., Glusa, C., Zheng, X., Mao, Z., Cai, W., Meerschaert, M., Ainsworth, M. and Karniadakis, G.: 2020, What is the fractional Laplacian? A comparative review with new results, *J. Comput. Phys.* **404**, 109009.

Liu, X., Goldsmith, A., Mahal, S. and Hedrick, J.: 2001, Effects of communication delay on string stability in vehicle platoons, *in Proc. IEEE Int. Conf. Intell. Transp. Syst.*, pp. 625–630.

López-Nicolás, G., Aranda, M. and Mezouar, Y.: 2020, Adaptive Multirobot Formation Planning to Enclose and Track a Target With Motion and Visibility Constraints, *IEEE Trans. Robot.* **36**(1), 142–156.

Lorensen, W. and Cline, H.: 1987, Marching cubes: A high resolution 3D surface construction algorithm, *in Proc. ACM SIGGRAPH Comp. Graph.*, Vol. 21, pp. 163–169.

Luo, D., Huang, H., Ding, C. and Nie, F.: 2010, On the eigenvectors of p-Laplacian, *Mach. Learn.* **81**(1), 37–51.

Lynch, K., Schwartz, I., Yang, P. and Freeman, R.: 2008, Decentralized Environmental Modeling by Mobile Sensor Networks, *IEEE Trans. Robot.* **24**(3), 710–724.

Manfredi, S.: 2013, Design of a multi-hop dynamic consensus algorithm over wireless sensor networks, *Control Eng. Pract.* **21**(4), 381–394.

Mannucci, A., Nardi, S. and Pallottino, L.: 2017, Autonomous 3D Exploration of Large Areas: A Cooperative Frontier-Based Approach, *in Proc. 4th Int. Conf. Model. Simul. Auton. Syst.*, pp. 18–39.

Mansouri, S., Kanellakis, C., Fresk, E., Kominiak, D. and Nikolakopoulos, G.: 2018, Cooperative coverage path planning for visual inspection, *Control Eng. Pract.* **74**, 118–131.

Marino, A., Pacchierotti, C. and Robuffo Giordano, P.: 2023, On the Stability of Gated Graph Neural Networks, *arXiv:2305.19235* .

Mariottini, G., Morbidi, F., Prattichizzo, D., Valk, N. V., Michael, N., Pappas, G. and Daniilidis, K.: 2009, Vision-Based Localization for Leader-Follower Formation Control, *IEEE Trans. Robot.* **25**(6), 1431–1438.

Marshall, J. and Broucke, M.: 2008, Symmetry Invariance of Multiagent Formations in Self-Pursuit, *IEEE Trans. Automat. Contr.* **53**(9), 2022–2032.

Marshall, J., Broucke, M. and Francis, B.: 2004, Formations of Vehicles in Cyclic Pursuit, *IEEE Trans. Automat. Contr.* **49**(11), 1963–1974.

Marthaler, D. and Bertozzi, A.: 2004, Tracking environmental level sets with autonomous vehicles, *in* S. Butenko, R. Murphey and P. Pardalos (eds), *Recent Developments in Cooperative Control and Optimization*, Kluwer, chapter 17, pp. 317–332.

Martínez, S.: 2010, Distributed interpolation schemes for field estimation by mobile sensor networks, *IEEE Trans. Contr. Syst. Tech.* **18**(2), 491–500.

Martínez, S. and Bullo, F.: 2006, Optimal sensor placement and motion coordination for target tracking, *Automatica* **42**(4), 661–668.

Marzbanrad, J. and Karimian, N.: 2011, Space control law design in adaptive cruise control vehicles using model predictive control, *Proc. I. Mech. Eng. D-J. Aut.* **225**, 870–884.

Maschuw, J., Diab, H., Abel, D. and Kowalewski, S.: 2011, Control Design for Generalized Platoon Problems, *Automatisierungstechnik* **59**(12), 721–729.

Massioni, P., Keviczky, T., Gill, E. and Verhaegen, M.: 2010, A Decomposition-Based Approach to Linear Time-Periodic Distributed Control of Satellite Formations, *IEEE Trans. Contr. Syst. Tech.* **19**(3), 481–492.

Matarić, M.: 1992, Behavior-based control: Main properties and implications, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 46–54.

McConnell, J., Huang, Y., Szenher, P., Collado-Gonzalez, I. and Englot, B.: 2022, DRACo-SLAM: Distributed Robust Acoustic Communication-efficient SLAM for Imaging Sonar Equipped Underwater Robot Teams, *in Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, pp. 8457–8464.

McGuire, K., Wagter, C. D., Tuyls, K., Kappen, H. and de Croon, G.: 2019, Minimal navigation solution for a swarm of tiny flying robots to explore an unknown environment, *Sci. Rob.* **4**(35), eaaw9710.

Meng, J. and Egerstedt, M.: 2007, Distributed Coordination Control of Multiagent Systems while Preserving Connectedness, *IEEE Trans. Robot.* **23**(4), 693–703.

Mesbahi, M. and Egerstedt, M.: 2010, *Graph Theoretic Methods in Multiagent Networks*, Applied Mathematics Series, Princeton University Press.

Michael, N. and Kumar, V.: 2009, Planning and Control of Ensembles of Robots with Nonholonomic Constraints, *Int. J. Robot. Res.* **28**(8), 962–975.

Michelitsch, T., Riascos, A. P., Collet, B., Nowakowski, A. and Nicolleau, F.: 2019, *Fractional Dynamics on Networks and Lattices*, ISTE-Wiley.

Michieletto, G., Cenedese, A. and Zelazo, D.: 2021, A Unified Dissertation on Bearing Rigidity Theory, *IEEE Trans. Control Network Syst.* **8**(4), 1624–1636.

Middleton, R. and Braslavsky, J.: 2010, String Instability in Classes of Linear Time Invariant Formation Control with Limited Communication Range, *IEEE Trans. Automat. Contr.* **55**(7), 1519–1530.

Millane, A., Taylor, Z., Oleynikova, H., Nieto, J., Siegwart, R. and Cadena, C.: 2018, C-blox: A Scalable and Consistent TSDF-based Dense Mapping Approach, *in Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, pp. 995–1002.

Miller, K. and Samko, S.: 2001, Completely monotonic functions, *Integral Transform Spec. Funct.* **12**(4), 389–402.

Monica, R. and Aleotti, J.: 2018, Contour-based next-best view planning from point cloud segmentation of unknown objects, *Auton. Robot.* **42**(2), 443–458.

Morbidi, F.: 2009, *Leader-Follower Formation Control and Visibility Maintenance of Nonholonomic Mobile Robots*, PhD thesis, Università di Siena, Dipartimento d'Ingegneria dell'Informazione, Siena, Italy.

Morbidi, F.: 2013a, The deformed consensus protocol, *Automatica* **49**(10), 3049–3055.

Morbidi, F.: 2013b, The Deformed Consensus Protocol: Extended Version, *arXiv:1304.2132* .

Morbidi, F.: 2014, The Second-order Parametric Consensus Protocol, *in Proc. European Contr. Conf.*, pp. 202–207.

Morbidi, F.: 2020, Subspace Projectors for State-Constrained Multi-Robot Consensus, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 7705–7711.

Morbidi, F.: 2022, Functions of the Laplacian Matrix With Application to Distributed Formation Control, *IEEE Trans. Control Network Syst.* **9**(3), 1459–1467.

Morbidi, F. and Bretagne, E.: 2018, A New Characterization of Mobility for Distance-Bearing Formations of Unicycle Robots, *in Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, pp. 4833–4839.

Morbidi, F., Cano, R. and Lara, D.: 2016, Minimum-energy path generation for a quadrotor UAV, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 1492–1498.

Morbidi, F., Colaneri, P. and Stanger, T.: 2013, Decentralized optimal control of a car platoon with guaranteed string stability, *in Proc. European Contr. Conf.*, pp. 3494–3499.

Morbidi, F., Consolini, L., Prattichizzo, D. and Tosques, M.: 2007, Leader-Follower Formation Control as a Disturbance Decoupling Problem, *in Proc. European Contr. Conf.*, pp. 1492–1497.

Morbidi, F., Freeman, R. and Lynch, K.: 2011, Estimation and Control of UAV Swarms for Distributed Monitoring Tasks, *in Proc. American Contr. Conf.*, pp. 1069–1075.

Morbidi, F. and Mariottini, G.: 2013, Active Target Tracking and Cooperative Localization for Teams of Aerial Vehicles, *IEEE Trans. Contr. Syst. Tech.* **21**(5), 1694–1707.

Morbidi, F., Mouaddib, E., Moras, J. and Marzat, J.: 2018-2021, ScanBot: Scanners RoBotisés pour la Numérisation Automatique du Patrimoine. Projet de recherche cofinancé par la région Hauts-de-France et l'ONERA DTIS.

Morbidi, F., Ray, C. and Mariottini, G.: 2011, Cooperative active target tracking for heterogeneous robots with application to gait monitoring, *in Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, pp. 3608–3613.

Morbidi, F., Ripaccioli, G. and Prattichizzo, D.: 2009, On Connectivity Maintenance in Linear Cyclic Pursuit, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 363–368.

Mouaddib, E.: 2015-2025, Programme E-Cathédrale. [web]: `https://mis.u-picardie.fr/e-cathedrale`.

Mouaddib, E., Caron, G., Groux-Leclet, D. and Morbidi, F.: 2019, Le patrimoine "in silico". Exemple de la cathédrale d'Amiens, *In Situ* **39**.

Mourikis, A. and Roumeliotis, S.: 2006, Performance Analysis of Multirobot Cooperative Localization, *IEEE Trans. Robot.* **22**(4), 666–681.

Mur-Artal, R. and Tardós, J.: 2017, ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras, *IEEE Trans. Robot.* **33**(5), 1255–1262.

Naus, G., Ploeg, J., de Molengraft, M. V., Heemels, W. and Steinbuch, M.: 2010, A Model Predictive Control Approach to Design a Parameterized Adaptive Cruise Control, *in* L. del Re, F. Allgöwer, L. Glielmo, C. Guardiola and I. Kolmanovsky (eds), *Automotive Model Predictive Control: Models, Methods and Applications*, Vol. 402 of *Lecture Notes in Control and Information Sciences*, Springer, chapter 17, pp. 273–283.

Naus, G., van den Bleek, R., Ploeg, J., Scheepers, B., van de Molengraft, R. and Steinbuch, M.: 2008, Explicit MPC design and performance evaluation of an ACC Stop-&-Go, *in Proc. American Contr. Conf.*, pp. 224–229.

Naus, G., Vugts, R., Ploeg, J., van de Molengraft, M. and Steinbuch, M.: 2010, String-Stable CACC Design and Experimental Validation: A Frequency-Domain Approach, *IEEE Trans. Veh. Technol.* **59**(9), 4268–4279.

Ndoye, A., Castillo-Zamora, J., Samorah-Laki, S., Miot, R., Ruymbeke, E. V. and Ruffier, F.: 2023, Vector Field Aided Trajectory Tracking by a 10-gram Flapping-Wing Micro Aerial Vehicle, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 5379–5385.

Nedic, A., Ozdaglar, A. and Parrilo, P.: 2010, Constrained Consensus and Optimization in Multi-Agent Networks, *IEEE Trans. Automat. Contr.* **55**(4), 922–938.

Nemhauser, G., Wolsey, L. and Fisher, M.: 1978, An Analysis of Approximations for Maximizing Submodular Set Functions - I, *Math. Program.* **14**, 265–294.

Newcombe, R., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A., Kohli, P., Shotton, J., Hodges, S. and Fitzgibbon, A.: 2011, KinectFusion: Real-time dense surface mapping and tracking, *in Proc. 10th IEEE Int. Symp. Mixed Augmen. Real.*, Vol. 11, pp. 127–136.

Newman, M.: 2018, *Networks*, 2nd edn, Oxford University Press.

Nguyen, C., Izadi, S. and Lovell, D.: 2012, Modeling Kinect Sensor Noise for Improved 3D Reconstruction and Tracking, *in Proc. 2nd IEEE Int. Conf. 3D Imag. Model. Proc. Visual. & Transm.*, pp. 524–530.

Nießner, M., Zollhöfer, M., Izadi, S. and Stamminger, M.: 2013, Real-time 3D reconstruction at scale using voxel hashing, *ACM Trans. Graphic* **32**(6), 1–11. Article no. 169.

Nikiforov, V.: 2017, Merging the A-and Q-spectral theories, *Appl. Anal. Discrete Math.* **11**(1), 81–107.

Noferini, V. and Quintana, M.: 2023, Generating functions of non-backtracking walks on weighted digraphs: radius of convergence and Ihara's theorem, *arXiv:2307.14200* .

Notomista, G., Mayya, S., Hutchinson, S. and Egerstedt, M.: 2019, An Optimal Task Allocation Strategy for Heterogeneous Multi-Robot Systems, *in Proc. European Contr. Conf.*, pp. 2071–2076.

Nuño, E., Loría, A., Paredes, A. and Hernández, T.: 2022, Consensus-based formation control of multiple nonholonomic vehicles under input constraints, *IEEE Control Syst. Lett.* **6**, 2767–2772.

Ögren, P., Fiorelli, E. and Leonard, N.: 2004, Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment, *IEEE Trans. Automat. Contr.* **49**(8), 1292–1302.

Oh, K.-K., Park, M.-C. and Ahn, H.-S.: 2015, A survey of multi-agent formation control, *Automatica* **53**, 424–440.

Okabe, A., Boots, B., Sugihara, K. and Chiu, S.: 2000, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, 2nd edn, John Wiley & Sons.

Oleynikova, H., Lanegger, C., Taylor, Z., Pantic, M., Millane, A., Siegwart, R. and Nieto, J.: 2020, An open-source system for vision-based micro-aerial vehicle mapping, planning, and flight in cluttered environments, *J. Field Robot.* **37**(4), 642–666.

Oleynikova, H., Taylor, Z., Fehr, M., Siegwart, R. and Nieto, J.: 2017, Voxblox: Incremental 3D Euclidean Signed Distance Fields for on-board MAV planning, *in Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, pp. 1366–1373.

Olfati-Saber, R.: 2005, Distributed Kalman filter with embedded consensus filters, *in Proc. 44th IEEE Conf. Dec. Contr.*, pp. 8179–8184.

Olfati-Saber, R.: 2007, Distributed tracking for mobile sensor networks with information-driven mobility, *in Proc. American Contr. Conf.*, pp. 4606–4612.

Olfati-Saber, R., Fax, J. and Murray, R.: 2007, Consensus and Cooperation in Networked Multi-Agent Systems, *Proc. IEEE* **95**(1), 215–233.

Olfati-Saber, R. and Jalalkamali, P.: 2012, Coupled Distributed Estimation and Control for Mobile Sensor Networks, *IEEE Trans. Automat. Contr.* **57**(10), 2609–2614.

Olfati-Saber, R. and Murray, R.: 2002, Graph rigidity and distributed formation stabilization of multi-vehicle systems, *in Proc. 41st IEEE Conf. Dec. Contr.*, Vol. 3, pp. 2965–2971.

Olfati-Saber, R. and Murray, R.: 2004, Consensus Problems in Networks of Agents with Switching Topology and Time-Delays, *IEEE Trans. Automat. Contr.* **49**(9), 1520–1533.

Öncü, S., van de Wouw, N. and Nijmeijer, H.: 2011, Cooperative Adaptive Cruise Control: Tradeoffs Between Control and Network Specifications, *in Proc. 14th IEEE Int. Conf. Intel. Trans. Syst.*, pp. 2051–2056.

Ongaro, F., Pane, S., Scheggi, S. and Misra, S.: 2019, Design of an Electromagnetic Setup for Independent Three-Dimensional Control of Pairs of Identical and Nonidentical Microrobots, *IEEE Trans. Robot.* **35**(1), 174–183.

Ortega, A.: 2022, *Introduction to Graph Signal Processing*, Cambridge University Press.

Ortega, A., Frossard, P., Kovačević, J., Moura, J. and Vandergheynst, P.: 2018, Graph Signal Processing: Overview, Challenges, and Applications, *Proc. IEEE* **106**(5), 808–828.

Papachristos, C., Kamel, M., Popović, M., Khattak, S., Bircher, A., Oleynikova, H., Dang, T., Mascarich, F., Alexis, K. and Siegwart, R.: 2019, Autonomous Exploration and Inspection Path Planning for Aerial Robots Using the Robot Operating System, *in* A. Koubaa (ed.), *Robot Operating System (ROS)*, Vol. 778, Springer, pp. 67–111.

Parker, L., Rus, D. and Sukhatme, G.: 2016, Multiple Mobile Robot Systems, *in* B. Siciliano and O. Khatib (eds), *Handbook of Robotics*, 2nd edn, Springer, chapter 53, pp. 1335–1384.

Passerieux, J. and Cappel, D. V.: 1998, Optimal observer maneuver for bearings-only tracking, *IEEE Trans. Aero. Elec. Sys.* **34**(3), 777–788.

Pavone, M. and Frazzoli, E.: 2007, Decentralized Policies for Geometric Pattern Formation and Path Coverage, *ASME J. Dyn. Syst. Meas. Contr.* **129**(5), 633–643.

Pégard, C., Arnould, J., Lebrun, A., Mouaddib, E. and Dolphin, B.: 1990, Hierarchy and communication for a team of mobile robots. Actions synchronization, *Revue d'automatique et de productique appliquées* **3**(2), 83–102.

Pereira, G., Campos, M. and Kumar, V.: 2004, Decentralized algorithms for multi-robot manipulation via caging, *Int. J. Robot. Res.* **23**(7–8), 783–795.

Pierson, A. and Schwager, M.: 2018, Controlling Noncooperative Herds with Robotic Herders, *IEEE Trans. Robot.* **34**(2), 517–525.

Polak, E.: 1997, *Optimization: Algorithms and Consistent Approximations*, Springer.

Qiu, Z., Liu, S. and Xie, L.: 2016, Distributed constrained optimal consensus of multi-agent systems, *Automatica* **68**, 209–215.

Queiroz, M. D., Cai, X. and Feemster, M.: 2019, *Formation Control of Multi-Agent Systems: A Graph Rigidity Approach*, John Wiley & Sons.

Quintero, S., Papi, F., Klein, D., Chisci, L. and Hespanha, J.: 2010, Optimal UAV Coordination for Target Tracking using Dynamic Programming, *in Proc. 49th IEEE Conf. Dec. Contr.*, pp. 4541–4546.

Ramachandra, K.: 2000, *Kalman Filtering Techniques for Radar Tracking*, CRC Press.

Reeds, J. and Shepp, L.: 1990, Optimal paths for a car that goes both forwards and backwards, *Pacific J. Math.* **145**(2), 367–393.

Ren, W. and Beard, R.: 2004, Decentralized Scheme for Spacecraft Formation Flying via the Virtual Structure Approach, *AIAA J. Guid. Contr. Dynam.* **27**(1), 73–82.

Ren, W. and Beard, R.: 2005, Consensus seeking in multiagent systems under dynamically changing interaction topologies, *IEEE Trans. Automat. Contr.* **50**(5), 655–661.

Respall, V. M., Devitt, D., Fedorenko, R. and Klimchik, A.: 2021, Fast Sampling-based Next-Best-View Exploration Algorithm for a MAV, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 89–95.

Reynolds, C.: 1987, Flocks, herds and schools: A distributed behavioral model, *in Proc. 14th Conf. Comp. Graph. Interact. Tech.*, pp. 25–34.

Robin, C. and Lacroix, S.: 2016, Multi-robot target detection and tracking: taxonomy and survey, *Auton. Robot.* **40**, 729–760.

Roza, A., Maggiore, M. and Scardovi, L.: 2022, *Distributed Coordination Theory for Robot Teams*, Vol. 490 of *Lecture Notes in Control and Information Sciences*, Springer.

Rubenstein, M., Cornejo, A. and Nagpal, R.: 2014, Programmable self-assembly in a thousand-robot swarm, *Science* **345**(6198), 795–799.

Ruiz, L., Gama, F. and Ribeiro, A.: 2020, Gated Graph Recurrent Neural Networks, *IEEE Trans. Signal Process.* **68**, 6303–6318.

Saade, A., Krzakala, F. and Zdeborová, L.: 2014, Spectral Clustering of Graphs with the Bethe Hessian, *in Proc. 28th Conf. Neural Inf. Process. Syst.*, pp. 406–414.

Sabattini, L., Aikio, M., Beinschob, P., Boehning, M., Cardarelli, E., Digani, V., Krengel, A., Magnani, M., Mandici, S., Oleari, F., Reinke, C., Ronzoni, D., Stimming, C., Varga, R., Vatavu, A., Lopez, S. C., Fantuzzi, C., Mäyrä, A., Nedevschi, S., Secchi, C. and Fuerstenber, K.: 2017, The PAN-Robots Project: Advanced Automated Guided Vehicle Systems for Industrial Logistics, *IEEE Rob. Autom. Mag.* **25**(1), 55–64.

Salam, T. and Hsieh, M.: 2023, Heterogeneous robot teams for modeling and prediction of multiscale environmental processes, *Auton. Robot.* **47**, 353–376.

Sandryhaila, A. and Moura, J.: 2014, Discrete Signal Processing on Graphs: Frequency Analysis, *IEEE Trans. Signal Process.* **62**(12), 3042–3054.

Sanfourche, M., Vittori, V. and Le Besnerais, G.: 2013, eVO: A realtime embedded stereo odometry for MAV applications, *in Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, pp. 2107–2114.

Santos, M., Diaz-Mercado, Y. and Egerstedt, M.: 2018, Coverage Control for Multirobot Teams With Heterogeneous Sensing Capabilities, *IEEE Robot. Autom. Lett.* **3**(2), 919–925.

Saravanos, A., Aoyama, Y., Zhu, H. and Theodorou, E.: 2023, Distributed Differential Dynamic Programming Architectures for Large-Scale Multiagent Control, *IEEE Trans. Robot.* **39**(6), 4387–4407.

Scheggi, S., Morbidi, F. and Prattichizzo, D.: 2014, Human-Robot Formation Control via Visual and Vibrotactile Haptic Feedback, *IEEE Trans. Haptics* **7**(4), 499–511.

Schiano, F., Franchi, A., Zelazo, D. and Rubuffo Giordano, P.: 2016, A rigidity-based decentralized bearing formation controller for groups of quadrotor UAVs, *in Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, pp. 5099–5106.

Schillinger, P., Bürger, M. and Dimarogonas, D.: 2018, Simultaneous task allocation and planning for temporal logic goals in heterogeneous multi-robot systems, *Int. J. Robot. Res.* **37**(7), 818–838.

Schmid, L., Pantic, M., Khanna, R., Ott, L., Siegwart, R. and Nieto, J.: 2020, An Efficient Sampling-based Method for Online Informative Path Planning in Unknown Environments, *IEEE Robot. Autom. Lett.* **5**(2), 1500–1507.

Schrank, D., Eisele, B. and Lomax, T.: 2019, Urban Mobility Report 2019, *Technical report*, Texas Transportation Institute, The Texas A&M University System. [web]: `https://rosap.ntl.bts.gov/view/dot/61408`.

Schrank, D., Lomax, T. and Eisele, B.: 2011, Urban Mobility Report 2011, *Technical report*, Texas Transportation Institute, The Texas A&M University System. [web]: `http://mobility.tamu.edu`.

Schwager, M., Rus, D. and Slotine, J.-J.: 2009, Decentralized, Adaptive Coverage Control for Networked Robots, *Int. J. Robot. Res.* **28**(3), 357–375.

Segarra, S., Marques, A., Mateos, G. and Ribeiro, A.: 2017, Network Topology Inference from Spectral Templates, *IEEE Trans. Signal Inf. Process. Networks* **3**(3), 467–483.

Segarra, S., Marques, A. and Ribeiro, A.: 2017, Optimal Graph-Filter Design and Applications to Distributed Linear Network Operators, *IEEE Trans. Signal Process.* **65**(15), 4117–4131.

Selin, M., Tiger, M., Duberg, D., Heintz, F. and Jensfelt, P.: 2019, Efficient Autonomous Exploration Planning of Large-Scale 3-D Environments, *IEEE Robot. Autom. Lett.* **4**(2), 1699–1706.

Seyfried, J., Szymanski, M., Bender, N., Estaña, R., Thiel, M. and Wörn, H.: 2005, The I-SWARM Project: Intelligent Small World Autonomous Robots for Micro-manipulation, *in* E. Şahin and W. Spears (eds), *Swarm Robotics*, Vol. 3342 of *Lecture Notes in Computer Science*, Springer, pp. 70–83.

Shamma, J. (ed.): 2008, *Cooperative Control of Distributed Multi-Agent Systems*, John Wiley & Sons.

Sharma, V., Zhou, L. and Tokekar, P.: 2023, D2CoPlan: A Differentiable Decentralized Planner for Multi-Robot Coverage, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 3425–3431.

Sheikholeslam, S. and Desoer, C.: 1993, Longitudinal Control of a Platoon of Vehicles with no Communication of Lead Vehicle Information: A System Level Study, *IEEE Trans. Contr. Syst. Tech.* **42**(4), 546–554.

Shi, G., Johansson, K. and Hong, Y.: 2012, Reaching an Optimal Consensus: Dynamical Systems That Compute Intersections of Convex Sets, *IEEE Trans. Automat. Contr.* **58**(3), 610–622.

Shuman, D., Ricaud, B. and Vandergheynst, P.: 2016, Vertex-frequency analysis on graphs, *Appl. Comput. Harmon. Anal.* **40**(2), 260–291.

Siciliano, B., Sciavicco, L., Villani, L. and Oriolo, G.: 2009, *Robotics: Modelling, Planning and Control*, Advanced Textbooks in Control and Signal Processing, Springer.

Siegwart, R., Nourbakhsh, I. and Scaramuzza, D.: 2011, *Introduction to Autonomous Mobile Robots*, MIT press.

Simon, D.: 2010, Kalman filtering with state constraints: a survey of linear and nonlinear algorithms, *IET Control Theory A.* **4**(8), 1303–1318.

Simonin, O.: 2023–2027, VORTEX: Vision-based Reconfigurable Drone Swarms for Fast Exploration. ANR project.

Singh, A., Krause, A., Guestrin, C. and Kaiser, W.: 2009, Efficient informative sensing using multiple robots, *J. Artif. Intell. Res.* **34**(1), 707–755.

Sinha, A. and Ghose, D.: 2006, Generalization of Linear Cyclic Pursuit with Application to Rendezvous of Multiple Autonomous Agents, *IEEE Trans. Automat. Contr.* **51**(11), 1819–1824.

Smith, R., Chao, Y., Li, P., Caron, D., Jones, B. and Sukhatme, G.: 2010, Planning and Implementing Trajectories for Autonomous Underwater Vehicles to Track Evolving Ocean Processes Based on Predictions from a Regional Ocean Model, *Int. J. Robot. Res.* **29**(12), 1475–1497.

Song, S. and Jo, S.: 2017, Online Inspection Path Planning for Autonomous 3D Modeling using a Micro-Aerial Vehicle, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 6217–6224.

Song, S. and Jo, S.: 2018, Surface-based Exploration for Autonomous 3D Modeling, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 4319–4326.

Song, S., Kim, D. and Choi, S.: 2022, View Path Planning via Online Multiview Stereo for 3-D Modeling of Large-Scale Structures, *IEEE Trans. Robot.* **38**(1), 372–390.

Stacey, G. and Mahony, R.: 2018, The Role of Symmetry in Rigidity Analysis: A Tool for Network Localisation and Formation Control, *IEEE Trans. Automat. Contr.* **63**(5), 1313–1328.

Stamouli, C., Bechlioulis, C. and Kyriakopoulos, K.: 2020, Multi-agent formation control based on distributed estimation with prescribed performance, *IEEE Robot. Autom. Lett.* **5**(2), 2929–2934.

Stanger, T. and del Re, L.: 2013, A model predictive Cooperative Adaptive Cruise Control approach, *in Proc. American Contr. Conf.*, pp. 1374–1379.

Stanković, S., Stanojević, M. and Šiljak, D.: 2000, Decentralized Overlapping Control of a Platoon of Vehicles, *IEEE Trans. Contr. Syst. Tech.* **8**(5), 816–832.

Stengel, R.: 1994, *Optimal Control and Estimation*, Dover Publications, Inc.

Stoll, M.: 2020, A Literature Survey of Matrix Methods for Data Science, *GAMM-Mitteilungen* **43**(3), e202000013.

Strogatz, S.: 2000, From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators, *Physica D* **143**(1–4), 1–20.

Strogatz, S.: 2012, *SYNC: How Order Emerges from Chaos In the Universe, Nature, and Daily Life*, Hachette, UK.

Stroupe, A. and Balch, T.: 2005, Value-based action selection for observation with robot teams using probabilistic techniques, *Robot. Autonom. Syst.* **50**(2–3), 85–97.

Stump, E., Kumar, V., Grocholsky, B. and Shiroma, P.: 2009, Control for Localization of Targets using Range-only Sensors, *Int. J. Robot. Res.* **28**(6), 743–757.

Sun, Z. and Anderson, B.: 2016, Formation feasibility on coordination control of networked heterogeneous systems with drift terms, *in Proc. 55th IEEE Conf. Dec. Contr.*, pp. 3462–3467.

Sun, Z., Mou, S., Anderson, B. and Yu, C.: 2018, Conservation and decay laws in distributed coordination control systems, *Automatica* **87**, 1–7.

Susca, S., Bullo, F. and Martinez, S.: 2008, Monitoring Environmental Boundaries with a Robotic Sensor Network, *IEEE Trans. Contr. Syst. Tech.* **16**(2), 288–296.

Swaroop, D. and Rajagopal, K.: 2001, A review of Constant Time Headway Policy for Automatic Vehicle Following, *in Proc. IEEE Int. Conf. Intel. Trans. Syst.*, pp. 65–69.

Tabuada, P., Pappas, G. and Lima, P.: 2005, Motion Feasibility of Multi-Agent Formations, *IEEE Trans. Robot.* **21**(3), 387–391.

Tahbaz-Salehi, A. and Jadbabaie, A.: 2008, A Necessary and Sufficient Condition for Consensus Over Random Networks, *IEEE Trans. Automat. Contr.* **53**(3), 791–795.

Tan, K. and Lewis, M.: 1997, High Precision Formation Control of Mobile Robots Using Virtual Structures, *Auton. Robot.* **4**(4), 387–403.

Tanner, H., Pappas, G. and Kumar, V.: 2004, Leader-to-Formation Stability, *IEEE Trans. Robot. Autom.* **20**(3), 443–455.

Tay, T. and Whiteley, W.: 1985, Generating Isostatic Frameworks, *Structural Topology* **11**, 21–69.

Taylor, C., Rahimi, A., Bachrach, J., Shrobe, H. and Grue, A.: 2006, Simultaneous Localization, Calibration, and Tracking in an ad Hoc Sensor Network, *in Proc. 5th ACM Int. Conf. Inf. Proc. Sens. Net.*, pp. 27–33.

Thuilot, B., d'Andréa-Novel, B. and Micaelli, A.: 1996, Modeling and Feedback Control of Mobile Robots Equipped with Several Steering Wheels, *IEEE Trans. Robot. Autom.* **12**(3), 375–390.

Tisseur, F. and Meerbergen, K.: 2001, The Quadratic Eigenvalue Problem, *SIAM Rev.* **43**(1), 235–286.

Tokekar, P., Karnad, N. and Isler, V.: 2014, Energy-optimal trajectory planning for car-like robots, *Auton. Robot.* **37**, 279–300.

Tolstaya, E., Gama, F., Paulos, J., Pappas, G., Kumar, V. and Ribeiro, A.: 2020, Learning Decentralized Controllers for Robot Swarms with Graph Neural Networks, *in Proc. 3rd Conf. Robot Learn.*, pp. 671–682.

Triplett, B., Klein, D. and Morgansen, K.: 2009, Cooperative Estimation for Coordinated Target Tracking in a Cluttered Environment, *Mobile Netw. Appl.* **14**(3), 336–349.

Trotta, A., Di Felice, M., Montori, F., Chowdhury, K. and Bononi, L.: 2018, Joint Coverage, Connectivity, and Charging Strategies for Distributed UAV Networks, *IEEE Trans. Robot.* **34**(4), 883–900.

Tsitsiklis, J.: 1984, *Problems in decentralized decision making and computation*, PhD thesis, EECS Department, Massachusetts Institute of Technology.

Tzes, M., Bousias, N., Chatzipantazis, E. and Pappas, G.: 2023, Graph Neural Networks for Multi-Robot Active Information Acquisition, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 3497–3503.

Uciński, D.: 2005, *Optimal Measurement Methods for Distributed Parameter System Identification*, CRC Press.

Vasquez-Gomez, J., Sucar, L., Murrieta-Cid, R. and Lopez-Damian, E.: 2014, Volumetric Next-best-view Planning for 3D Object Reconstruction with Positioning Error, *Int. J. Adv. Robot. Syst.* **11**(10).

Vicsek, T. and Zafeiris, A.: 2012, Collective motion, *Phys. Rep.* **517**(3-4), 71–140.

Walter, E. and Pronzato, L.: 1994, *Identification of Parametric Models: from Experimental Data*, Springer.

Wang, L., Ames, A. and Egerstedt, M.: 2017, Safety Barrier Certificates for Collisions-Free Multirobot Systems, *IEEE Trans. Robot.* **33**(3), 661–674.

Wang, X., Mou, S. and Anderson, B.: 2019, Scalable, distributed algorithms for solving linear equations via double-layered networks, *IEEE Trans. Automat. Contr.* **65**(3), 1132–1143.

Wang, X. and Zhou, Z.: 2015, Projection-Based Consensus for Continuous-Time Multi-Agent Systems with State Constraints, *in Proc. 54th IEEE Conf. Dec. Contr.*, pp. 1060–1065.

Wilson, R.: 1973, An Introduction to Matroid Theory, *Am. Math. Mon.* **80**(5), 500–525.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C. and Philip, S.: 2021, A Comprehensive Survey on Graph Neural Networks, *IEEE Trans. Neural Networks Learn. Syst.* **32**(1), 4–24.

Wurman, P., D'Andrea, R. and Mountz, M.: 2008, Coordinating Hundreds of Cooperative, Autonomous Vehicles in Warehouses, *AI Magazine* **29**(1), 9.

Xia, W. and Cao, M.: 2011, Clustering in diffusively coupled networks, *Automatica* **47**(11), 2395–2405.

Xie, Z., Chen, S. and Orchard, G.: 2017, Event-Based Stereo Depth Estimation Using Belief Propagation, *Front. Neurosci.* **11**, 535.

Xu, Z., Deng, D. and Shimada, K.: 2021, Autonomous UAV Exploration of Dynamic Environments Via Incremental Sampling and Probabilistic Roadmap, *IEEE Robot. Autom. Lett.* **6**(2), 2729–2736.

Yang, P., Freeman, R. and Lynch, K.: 2007, Distributed cooperative active sensing using consensus filters, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 405–410.

Yang, P., Freeman, R. and Lynch, K.: 2008, Multi-Agent Coordination by Decentralized Estimation and Control, *IEEE Trans. Automat. Contr.* **53**(11), 2480–2496.

Yao, W., García de Marina, H., Sun, Z. and Cao, M.: 2023, Guiding Vector Fields for the Distributed Motion Coordination of Mobile Robots, *IEEE Trans. Robot.* **39**(2), 1119–1135.

Yoder, L. and Scherer, S.: 2016, Autonomous Exploration for Infrastructure Modeling with a Micro Aerial Vehicle, *in* D. Wettergreen and T. Barfoot (eds), *Field and Service Robotics: Results of the 10th Int. Conf.*, Springer, pp. 427–440.

Yu, C., Hendrickx, J., Fidan, B., Anderson, B. and Blondel, V.: 2007, Three and higher dimensional autonomous formations: Rigidity, persistence and structural persistence, *Automatica* **43**(3), 387–402.

Yu, C., Li, Q., Gao, S. and Prorok, A.: 2023, Accelerating Multi-Agent Planning Using Graph Transformers with Bounded Suboptimality, *in Proc. IEEE Int. Conf. Robot. Automat.*, pp. 3432–3439.

Zavlanos, M., Egerstedt, M. and Pappas, G.: 2011, Graph-Theoretic Connectivity Control of Mobile Robot Networks, *Proc. IEEE* **99**(9), 1525–1540.

Zelazo, D., Franchi, A., Bülthoff, H. and Robuffo Giordano, P.: 2015, Decentralized rigidity maintenance control with range measurements for multi-robot systems, *Int. J. Robot. Res.* **34**(1), 105–128.

Zelazo, D., Robuffo Giordano, P. and Franchi, A.: 2015, Bearing-Only Formation Control Using an SE(2) Rigidity Theory, *in Proc. 54th IEEE Conf. Dec. Contr.*, pp. 6121–6126.

Zeng, A., Song, S., Nießner, M., Fisher, M., Xiao, J. and Funkhouser, T.: 2017, 3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions, *in Proc. IEEE Conf. Comp. Vis. Pattern Recogn.*, pp. 1802–1811.

Zhang, F. and Leonard, N.: 2010, Cooperative Filters and Control for Cooperative Exploration, *IEEE Trans. Automat. Contr.* **55**(3), 650–663.

Zhang, H. and Gurfil, P.: 2016, Distributed Control for Satellite Cluster Flight Under Different Communication Topologies, *J. Guid. Control Dyn.* **39**(3), 617–627.

Zhang, J., Campbell, J., Sweeney II, D. and Hupman, A.: 2021, Energy consumption models for delivery drones: A comparison and assessment, *Transp. Res. Part D Transp. Environ.* **90**, 102668.

Zhao, F., Shin, J. and Reich, J.: 2002, Information-driven dynamic sensor collaboration for tracking applications, *IEEE Signal Proc. Mag.* **19**(2), 61–72.

Zhao, S.: 2018, Affine Formation Maneuver Control of Multiagent Systems, *IEEE Trans. Automat. Contr.* **63**(12), 4140–4155.

Zhao, S., Dimarogonas, D., Sun, Z. and Bauso, D.: 2018, A General Approach to Coordination Control of Mobile Agents with Motion Constraints, *IEEE Trans. Automat. Contr.* **63**(5), 1509–1516.

Zhao, S. and Zelazo, D.: 2016, Bearing Rigidity and Almost Global Bearing-Only Formation Stabilization, *IEEE Trans. Automat. Contr.* **61**(5), 1255–1268.

Zhao, S. and Zelazo, D.: 2019, Bearing Rigidity Theory and Its Applications for Control and Estimation of Network Systems: Life Beyond Distance Rigidity, *IEEE Contr. Syst. Mag.* **39**(2), 66–83.

Zhou, K. and Roumeliotis, S.: 2008, Optimal motion strategies for range-only constrained multisensor target tracking, *IEEE Trans. Robot.* **24**(5), 1168–1185.

Zhou, K. and Roumeliotis, S.: 2011, Multirobot Active Target Tracking with Combinations of Relative Observations, *IEEE Trans. Robot.* **27**(4), 678–695.

Zhou, L., Sharma, V., Li, Q., Prorok, A., Ribeiro, A., Tokekar, P. and Kumar, V.: 2022, Graph Neural Networks for Decentralized Multi-Robot Target Tracking, *in Proc. IEEE Int. Symp. Safety, Security, and Rescue Robotics*, pp. 195–202.

Zhou, Y., Xiao, J., Zhou, Y. and Loianno, G.: 2022, Multi-Robot Collaborative Perception With Graph Neural Networks, *IEEE Robot. Autom. Lett.* **7**(2), 2289–2296.

Zhu, M. and Martínez, S.: 2015, *Distributed Optimization-Based Control of Multi-Agent Networks in Complex Environments*, Springer.