Decentralized control of multi-agent systems : a hybrid formalism

Tommaso Borzone

École doctorale IAEM Lorraine
DFD Automatique et Production Automatisée
Université de Lorraine

Decentralized control of multi-agent systems: a hybrid formalism

THÈSE

présentée et soutenue publiquement le 9 septembre 2019

pour l'obtention du

Doctorat de l'Université de Lorraine

Spécialité Automatique, Traitement du Signal et des Images, Génie Informatique

par

Tommaso BORZONE

Composition du jury

Président :	W. PERRUQUETTI	Professeur des Universités, École Centrale de Lille, CRIStAL-CNRS, Lille
Rapporteurs :	E. PANTELEY	Directrice de recherche CNRS, L2S-CentraleSupelec, Gif-sur-Yvette
	A. SEURET	Directeur de recherche CNRS, LAAS-CNRS, Toulouse
Examinateurs :	F. GARIN	Chargée de recherche, INRIA Rhône-Alpes, Grenoble
	I.-C. MORĂRESCU	Professeur des Universités, Université de Lorraine, CRAN, Vandœuvre-lès-Nancy (Directeur de thèse)
	M. JUNGERS	Directeur de recherche CNRS, CRAN, Vandœuvre-lès-Nancy (Co-directeur de thèse)
	M. BOC	Chef de projets, CEA LIST Nano-INNOV, Palaiseau
Invités :	C. JANNETEAU	Chef de laboratoire LSC, CEA LIST Nano-INNOV, Palaiseau
	B. MARCHAND	Responsable Partenariats Industriels et Plateforme Usine du Future FFLOR, CEA Tech, Metz

Centre de Recherche en Automatique de Nancy
UMR 7039 CNRS – Université de Lorraine
2, avenue de la forêt de Haye 54516 Vandœuvre-lès-Nancy
Tél.+33 (0)3 83 59 59 59 Fax +33 (0)3 83 59 56 44

Acknowledgments

I would like to express my deepest gratitude to my supervisors Constantin et Marc for their trust at the beginning of this journey and for their precious advice, help and motivation throughout the entire route. At the beginning of my PhD, from an academic research perspective, I felt like I was raw material. However, their guidance allowed me to shape myself and to acquire the essential skills to work in scientific research and to present my results in the most effective way. It was a wonderful experience from a professional and human perspective.

I would like to thank the CEA tech which made this project possible and especially Mathias and Christophe for their trust and support for the implementation aspects of this work. Many thanks to Benoit, Manuel, Romain, Alice and all the staff and colleagues at the Plate-forme régionale de transfert technologique (PRTT) CEA Tech in Metz and at the Fucture of Factory Lorraine facility in Trémery.

I would like to express my gratitude to the members of my thesis commitee for agreeing to evaluate my thesis work and for attending the presentation. Special thanks for their work and their valuable remarks.

My deepest thanks go to my family, my parents Rita and Andrea and my sister Matilde for their constant love, support and for always believing in me. Many thanks also to Francesca and Patrick, my second family.

I would also like to thank the Centre de Recherche en Automatique de Nancy (CRAN) starting with Christine and all the permanent staff and ending with my PhD colleagues, especially Balou, Florian, Blinco, Hervé, Mathieu and JP. Special thanks for Prisca and Daniele who, beside science and politics talks, were the first that made me feel at home. Then my greatest thanks to the GREEN colleagues Davide, Bomber, Max, Ivano, Dom. I wish to all of them the best for their careers and for their PhD. Daniele and Davide, after these three years I wondered how I survived the previous 27 years of my life without having known them. Simply thanks.

Four years ago I met a PhD engineer who had travelled 4000 km from the University of Pisa to move with his beautiful family to a small village in Norway. His passion and care, his ability to analyze and study inspired me and pushed me to make important life choices, including to start a PhD. That is why I would like to thank Dr. Vincenzo Calabrò.

I would like to thank my friend and colleague Marco for his advise and the wonderful experience that we made together. And of course many thanks to all the Italian friends Gianluca, Andrea, Valentina, Filippo, Sara for always being there in time of joke and in time of need.

Finally, the most sincere and heartfelt gratitude goes to Laura. She supported me in my choices. She turned her life upside down so that I could realize my dream. These words are maybe not enough but I say thank you Principessa.

To my family and to my dear Laura

Contents

Notation

$\mathbb{N}$	-	*set of non-negative integers.*
$\mathbb{R}$	-	*set of real numbers.*
$\mathbb{R}_+$	-	*set of non-negative real numbers.*
$\mathbb{R}^{n\times m}$	-	*set of $n \times m$ real matrices.*
$\mathbb{R}^n$	-	*set of n dimension real vectors.*
I_n	-	*$n \times n$ identity matrix*
$x_{(h)}$ $(x_{i,(h)})$	-	*h-th component of vector x (x_i respectively).*
$x - y > 0$	-	*component wise inequality* $: x_{(h)} - y_{(h)} > 0, \forall\, h = 1, \ldots, n.$
$P \succ 0$	-	*positive definite matrix P.*
$P > 0$	-	*positive matrix P, all entries are strictly positive.*
$P^\top$	-	*transpose of matrix P.*
$x^\top$	-	*transpose of vector x.*
$\|x\|$	-	*Euclidean norm of vector $x \in \mathbb{R}^n$, see Appendix A.*
$\|x\|$	-	*infinity norm of vector $x \in \mathbb{R}^n$, see Appendix A.*
$\|x\|_1$	-	*1-norm of vector $x \in \mathbb{R}^n$, see Appendix A.*
$\mathbb{1}_n$	-	*column vector in $\mathbb{R}^n$ whose elements are all equal to one.*
$\mathbb{1}_{n\times m}$	-	*$n \times m$ matrix whose elements are all equal to one* $: (\mathbb{1}_n^\top, \ldots, \mathbb{1}_n^\top) \in \mathbb{R}^{n\times m}$.

Commande décentralisée de systèmes multi agents: un formalisme hybride

Dans l'analyse des systèmes multi agents, nous pouvons essentiellement nous concentrer sur deux caractéristiques principales : la dynamique de l'agent et l'interaction par laquelle les agents sont interconnectés. La première composante est profondément liée au cadre de l'analyse. En effet, la dynamique des sous-systèmes peut varier énormément si l'on traite, par exemple, un système économique ou un réseau social. D'autre part, les interactions ou l'échange d'informations entre les agents impliquent, quel que soit le cadre d'analyse, la théorie des graphes. Il est donc possible d'envisager les problèmes des systèmes multi agents sous deux points de vue, en se concentrant soit sur la dynamique des agents (linéaire ou non linéaire, homogène ou hétérogène, etc.), soit sur la nature et la topologie du réseau (statique ou switching, ouvert à l'admission de nouveaux agents, etc.).

L'étude des systèmes multi agents a également attiré l'attention de la communauté de contrôle, pour s'attaquer à des problèmes très complexes et très difficiles à résoudre de manière centralisée. Le problème centralisé peut ainsi être réduit à un ensemble de sous-problèmes locaux : les agents interagissent entre eux et appliquent un algorithme de contrôle décentralisé afin d'obtenir un comportement global émergent, par exemple pour parvenir à un accord sur des valeurs spécifiques ou pour synchroniser différents agents. A cet égard, le problème du consensus, analysé dans cet ouvrage, est une représentation mathématique typique d'un tel scénario, dans lequel les agents interagissent pour trouver une conformité dans leur comportement. Les essaims robotiques, les réseaux de capteurs et les réseaux d'alimentation sont quelques exemples où l'accord peut jouer un rôle crucial dans la réalisation d'un comportement souhaité.

Souvent, la dynamique des exemples cités précédemment présente une nature mixte, c'est-à-dire qu'elle nécessite explicitement des processus continus pour interagir avec (ou être supervisés par) des processus discrets (par exemple un réseau de convertisseurs de commutation, ou un essaim de robots qui, en se déplaçant, sentent sporadiquement l'environnement autour de ceux-ci). Ces systèmes sont appelés systèmes dynamiques hybrides. Les systèmes impulsifs représentent une classe particulière de systèmes hybrides, caractérisée par la combinaison d'un processus continu décrit par des équations différentielles et des sauts sur l'état. Ces événements discrets peuvent être associés à des conditions temporelles périodiques ou à des événements en relation avec l'évolution de leur propre dynamique.

L'objectif du présent travail est d'analyser le processus du consensus d'une classe particulière de systèmes multi agents en tenant compte des phénomènes hybrides qui

apparaissent en considérant les interactions sporadiques entre les agents du réseau. L'analyse est réalisée avec une attention particulière sur la topologie du réseau, avec un accent particulier sur les applications aux flottes de robots. Cette recherche s'inscrit d'ailleurs dans le cadre du projet [CEA tech, 2019], développé par le département de recherche technologique du *Commissariat à l'énergie atomique et aux énergies alternatives*. (CEA tech). La plate-forme Future of Factory Lorraine (FFLOR) développe et teste en conditions réelles les technologies numériques pour l'usine du futur, impliquant différents aspects de l'automatisation industrielle moderne, y compris le contrôle des systèmes robotisés multi agents.

Future of Factory Lorraine (FFLOR)

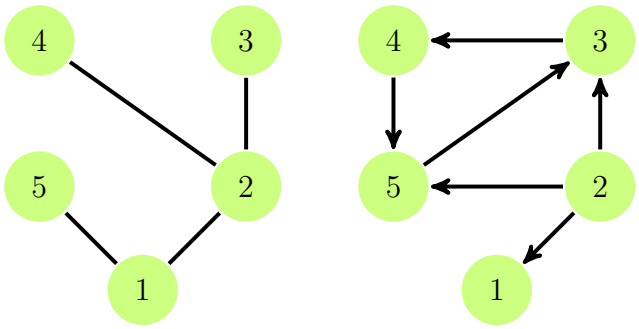
La plateforme est organisée autour de quatre lignes directrices de R&D portant sur la réalité virtuelle, les robots collaboratifs de manutention de charges, la coopération homme-robot et la logistique et les assemblages intelligents. Les solutions développées dans ce contexte visent à accroître la flexibilité, la productivité et la sécurité des usines de fabrication tout en facilitant la connectivité et l'interopérabilité entre les systèmes de production et d'information. La plate-forme est située dans une usine PSA Peugeot Citroën de 3400 salariés à Trémery, Moselle, France. La plate-forme s'appuie sur des technologies disponibles sur le marché ou développées par CEA Tech Institute List dans ses solutions. Toutes les solutions sont testées avec les opérateurs de fabrication, améliorées et intégrées dans les systèmes pour faciliter un transfert rapide à l'industrie. A l'issue des travaux de thèse, un système de gestion de flotte pour le suivi et la collecte des données a été conçu, développé et installé sur le site FFLOR. Le système de gestion de flotte contrôle un réseau de robots mobiles de petite taille capables de s'assembler en formation et de naviguer dans un environnement fermé.

Préliminaires

Dans le Chapitre 1 dédié aux notions préliminaires, nous passons en revue quelques concepts de base et outils théoriques qui ont été utilisés dans ce travail. Nous commençons par une brève introduction aux méthodes de la théorie des graphes qui sont à la base de l'analyse des interactions dans un système multi agents. Plus spécifiquement, nous définissons les éléments théoriques qui caractérisent deux topologies d'interaction souvent présentées dans la littérature : dirigées et non dirigées (voir Figure 1). De plus, nous introduisons le problème du consensus et nous analysons les résultats existants dans la littérature concernant sa convergence pour les topologies décrites par les graphes non dirigés et dirigés.

Une introduction théorique similaire est ensuite donnée pour les systèmes dynamiques hybrides, où nous rappelons le concept de domaine temporel hybride, de solution de système hybride et de trajectoire [Goebel et al., 2012] et où nous introduisons une formulation particulière de la condition de stabilité asymptotique du système hybride.

La dernière section est consacrée à l'introduction de la dynamique d'un type particulier de véhicules robotiques appartenant à la classe plus large des systèmes dynamiques non-holonomiques. En particulier, nous concentrons notre attention sur le



(a) Topologie décrite par un graphe non dirigé.

(b) Topologie décrite par un graphe dirigé

Figure 1: Exemple d'un graphe non dirigé Figure 1(a) et dirigé Figure 1(b).

modèle cinématique unicycle et décrivons brièvement les difficultés qui se posent dans l'étude de la stabilisation de ce type de systèmes, principalement liées aux singularités et au fait que les conditions suffisantes du théorème de Brocket [Brockett, 1983] ne sont pas vérifiées.

Cadre hybride pour le consensus des systèmes multi agents

Dans cette partie correspondant au Chapitre 2, nous présentons les principaux résultats théoriques de nos travaux basés sur les nouveaux résultats rapportés dans [Borzone et al., 2018b, Borzone et al., 2018a, Borzone et al., 2019]. Nous concentrons notre attention sur le problème de consensus d'une classe particulière de dynamique des agents, à savoir les systèmes non linéaires de suivi de référence constant par morceau. Nous proposons une approche en deux étapes basée sur la séparation entre la tâche décentralisé de consensus et la tâche de suivi des références locales (voir Figure 2). Les agents mettent à jour leurs références grâce à des mesures relatives sporadiques de l'état du voisinage. La nature hybride de cette configuration est prise en compte dans l'analyse de stabilité qui suit.

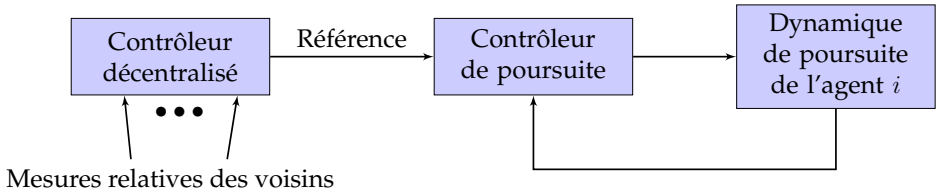


Figure 2: Schéma fonctionnel montrant la stratégie de contrôle de l'agent i. Le suivi est mis en œuvre comme une tâche distincte et indépendante du niveau de planification décentralisé qui utilise les informations de détection sur le voisinage de l'agent i.

Tout d'abord, nous étudions le cas d'une topologie décrite par un graphe non dirigé, variant dans le temps et connecté. En particulier, nous nous concentrons sur la séquence d'échantillonnage pendant laquelle l'algorithme décentralisé est mis à jour, séquence

qui n'est pas connue à priori. L'idée principale est de formuler une condition suffisante pour la séquence de mise à jour qui assure la convergence de l'algorithme de consensus. Cette condition suffisante garantit qu'un laps de temps suffisamment long s'est écoulé entre une mise à jour et la suivante afin de permettre la convergence de la boucle interne de suivi. Ceci se traduit par une condition de temps de séjour sur la séquence de mise à jour du réseau.

Par la suite, nous concentrons notre attention sur les graphes dirigés et variable dans le temps, mais nous relâchons l'hypothèse de connectivité. Nous introduisons le concept de connectivité commune d'une ensemble de graphes et, en utilisant la même idée que celle décrite précédemment, nous formulons une nouvelle condition suffisante pour la stabilité concernant la séquence de mise à jour de chaque agent séparément.

Enfin, certains résultats portant sur la robustesse de la solution par rapport aux mesures relatives perturbées sont fournis dans la dernière section.

Consensus dans les réseaux de robots non-holonomes

Nous appliquons dans le Chapitre 3 le cadre d'analyse développé précédemment à un réseau de véhicules non holonomes. L'application s'inspire du projet de flotte multi robots de la plateforme FFLOR du CEA qui faisait partie de ces travaux. Le même cas d'étude a été utilisé dans [Borzone et al., 2018a, Borzone et al., 2019, Borzone et al., 2018b] pour valider les résultats théoriques.

L'approche en deux étapes est appliquée à un système multi agents de véhicules non-holonomes. Une fois qu'il est démontré que la dynamique locale de suivi de référence constant par morceau satisfait à l'ensemble des hypothèses formulées dans le chapitre précédent, les résultats de stabilité sont appliqués au problème de rendez-vous et au problème de réalisation de la formation pour un ensemble de robots mobiles à roues différentielles.

Dans la dernière section, nous validons et testons la solution proposée dans un environnement de simulation Matlab, en nous référant aux trois cas d'étude décrits dans l'analyse théorique, à savoir le cas de graphe non dirigé et variant dans le temps, le cas de graphe dirigé et variant dans le temps et le cas contenant du bruit et des biais de mesure sur les capteur.

Gestion et surveillance d'une flotte de robots mobiles autonomes (AMR)

Ce dernier partie correspond au Chapitre 4 et décrit le système de gestion de flotte développé au cours des travaux de thèse. Les systèmes de gestion de flotte peuvent être considérés comme un ensemble de logiciels et de matériels informatiques qui interagissent pour faciliter la réalisation de tâches et d'objectifs communs à une flotte de robots. Prenons par exemple les systèmes de gestion de flotte des robots mobiles Kiva [Wurman et al., 2008] utilisés pour le transport des marchandises dans les entrepôts d'Amazon. Autre exemple : les véhicules terrestres autonomes (AGV) Asti

Easybot déjà utilisés sur le site de FFLOR dans le cadre des lignes directrices du CEA pour la recherche en logistique intelligente. En général, ces robots sont programmés pour se déplacer de manière autonome le long de pistes prédéfinies transportant des marchandises. Cependant, les itinéraires et les horaires sont fournis par un planificateur centralisé qui effectue également l'allocation des ressources et gère l'attribution des tâches à chaque robot.

Le système de gestion de flotte étudié dans ce travail traite d'un type plus flexible de robots mobiles autonomes, conçus pour des tâches moins répétitives telles que la surveillance et la collecte de données. Ces applications ont fait l'objet d'études et de développements ces dernières années pour l'environnement public intérieur et les serres automatisées [Roldán et al., 2015]. L'application de cette technologie à un environnement industriel est motivée soit par un simple objectif de production et de contrôle des performances, soit par des missions de sauvetage à haut risque dans un environnement dangereux provoqué par des accidents ou des catastrophes.

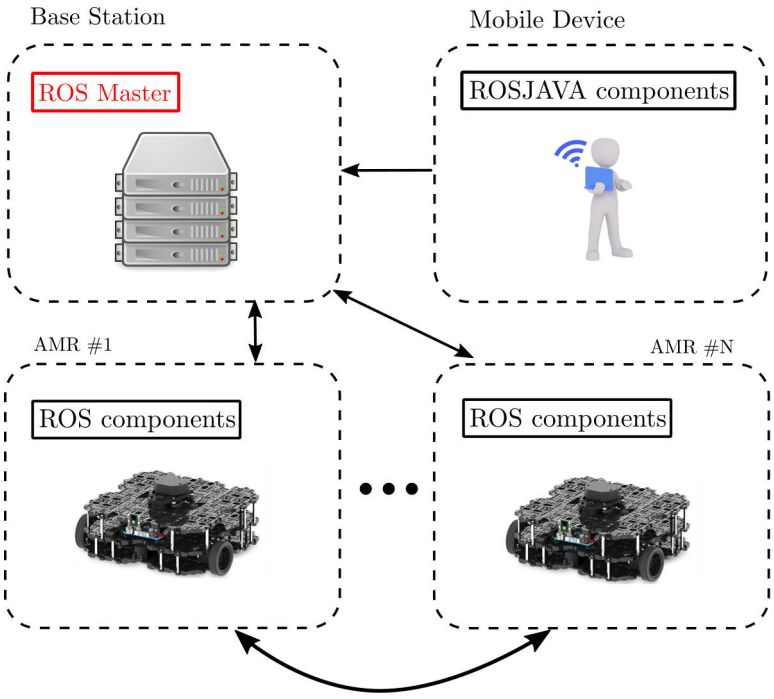


Figure 3: Le logiciel de gestion de flotte FFLOR.

L'ensemble de la solution de gestion de flotte (FMS) a été développé en utilisant les bibliothèques clients ROS. Le FMS exige que les robots soient capables d'estimer leur propre position et de détecter les obstacles mobiles ou fixes. Ces composants sont utilisés pour mettre à jour la carte de l'environnement qui, à son tour, est utilisée par le planificateur pour créer de nouveaux chemins pour les robots. L'opérateur fournit les objectifs pour la flotte dans une formation prédéfinie. L'autonomie de chaque robot est assurée par le module de navigation qui implémente SLAM (Localisation et cartographie simultanées).

Comme représenté dans Figure 3, le FMS conçu se compose principalement de trois entités physiques : un opérateur qui fournit une forme souhaitée et qui commande les véhicules à distance ; un poste central de contrôle pour la surveillance, la collecte et

l'analyse des données et le réseau de robots mobiles autonomes (AMR) déployés dans un environnement non structuré.

Conclusions

Le but de ce travail est d'analyser le consensus d'une classe de systèmes multi agents en tenant compte des phénomènes hybrides qui se produisent lorsque l'on considère les interactions sporadiques entre les agents du réseau. L'analyse est réalisée avec une attention particulière sur la topologie du réseau, avec un accent particulier sur les applications aux flottes de robots. Ce choix d'application est motivé par le fait que ces travaux s'inscrivent dans le cadre du projet technologique CEA Future of Factory Lorraine (FFLOR), où des technologies numériques innovantes pour l'automatisation industrielle sont développées et testées.

Les résultats des travaux développés dans cette thèse conduisent à diverses extensions possibles et à de nouvelles recherches. Par exemple, il pourrait être intéressant d'essayer d'assouplir l'hypothèse de convergence exponentielle et d'étudier les conditions suffisantes dans le cas d'une propriété de convergence moins restrictive de la dynamique de suivi local.

En outre, la deuxième étape de la stratégie en deux étapes et l'analyse connexe dont il est question au chapitre 2 sont fondées sur un suivi de référence par élément d'état. Inspirés par les travaux de [De Campos et al., 2012] sur l'accord des systèmes linéaires hétérogènes et les travaux de synchronisation des systèmes hétérogènes non linéaires par [Isidori et al., 2014] nous pourrions penser à étendre notre analyse au cas du suivi de référence en sortie.

Un autre aspect important qui est négligé dans l'analyse théorique est l'impact qu'une politique d'évitement des obstacles et des collisions auraient sur la stabilité (voir exemple [Wang et al., 2017]). Ces deux algorithmes peuvent affecter l'hypothèse de convergence exponentielle, ce qui est crucial dans notre cadre théorique. Une solution possible est de considérer la correction additive introduite par l'algorithme présenté dans 4.2.2 comme une nouvelle perturbation et donc d'exploiter les propriétés ISS de la dynamique globale.

Introduction

> "We are caught in an inescapable network of mutuality, tied in a single garment of destiny."
>
> *Martin Luther King Jr.*

The concept of complexity of large-scale natural and artificial physical phenomena and social phenomena is often difficult to assimilate outside the scientific community. And yet complexity pervades every sphere of knowledge and human society, and it is, for the most part, due to interconnections of individual processes and systems which evolves in the world where we live in. Moreover, the degree of interdependence is so high that simple solutions such as isolation and denial, if ever they worked in the past, have no reason to exist anymore. The opening quote suggests us how this characterizes our society, but the same applies, in a more theoretic and scientific way, to most natural and artificial physical phenomena which began to rise the interest of the scientific community in the last decades. Multi-agent systems and networked systems turned out to be the most effective way to model complex systems dynamics, such as animal collective behavior (birds flocking), biology (the spread of a disease in an organism or a population), social networks (echo-room creation and fake news spreading), opinion dynamics, economic modeling and many others.

In the analysis of multi-agent system we can essentially focus on two main features: the dynamics of the single agent and the interaction through which the agents are interconnected. The first component is deeply related to the framework of the analysis. Indeed the sub-systems dynamics can vary enormously if we treat, for instance, an economic system or a social network. On the other hand the interactions or the exchange of information between the agents involve, regardless the analysis framework, the graph theory and its fundamental tools. It is therefore possible to consider multi-agent systems problems from two main perspectives, focusing whether on the dynamics of the agents (linear or non-linear, homogeneous or heterogeneous and so on) or on the nature and topology of the network (static or switching, open or close to the admission of new agents, etc.).

The study of multi-agent systems has also attracted the attention of the control community, to tackle some highly complex problems very difficult to solve in a centralized manner. The centralized problem can thus be reduced to a set of local sub-problems: the agents interact each other and apply a decentralized control algorithm in order to achieve an emergent global behaviour, for instance to reach some sort of agreement on specific values or to synchronize together different agents. With this respect, the consensus problem, analysed in this work, is a typical mathematical representation

of such a scenario, in which agents interact to find an accordance in their behavior. Robotic swarms, sensors networks, and power networks are some examples where the agreement can play a crucial role in achieving a desired behaviour.

Often, the dynamics of the previously cited examples present a mixed nature, that is they explicitly requires continuous processes to interact with (or to be supervised by) discrete processes (for instance a network of switching converter, or a swarm of robot which, while moving, sporadically sense the environment around them). This class of systems is referred as hybrid dynamic systems. The impulsive systems represent a particular class of hybrid systems, characterized by the combination of a continuous process described by differential equations and jumps on the state. These discrete events can be associated with periodic temporal conditions or events in relation to the evolution of their own dynamics.

The purpose of the present work is to analyse the agreement behaviour of a particular class of multi-agent systems taking into account hybrid phenomena that appear considering sporadic interactions between the agents of the network. The analysis is carried out with particular attention on the topology of the network, with special focus on applications to fleets of robots. This research is in fact part of the Future of Factory Lorraine (FFLOR) project [CEA tech, 2019], developed by the technological research department of the *Commissariat à l'énergie atomique et aux énergies alternatives* (CEA tech). The FFLOR platform develops and tests in actual operating conditions digital technologies for the factory of the future, involving different aspects of modern industrial automation including the control of multi-agent robotic systems.

Future of Factory Lorraine (FFLOR)

The platform is organized around four R&D guidelines addressing virtual reality, load-handling collaborative robots, human-robot cooperation, and smart logistics and assemblies. The solutions developed in this context aim to increase the manufacturing plants flexibility, productivity, and safety while facilitating connectivity and interoperability between production and information systems. The platform is located in a PSA Peugeot Citroën factory with 3,400 employees in Trémery, Moselle, France. The platform leverages commercially-available technologies or technologies developed by CEA Tech institute List in its solutions. All solutions are tested with manufacturing operators, improved and integrated into systems to facilitate a fast transfer to industry. As completion of the thesis work, a fleet management system for monitoring and data collection purpose has been designed, developed and installed at the FFLOR site. The fleet management system controls a network of small size mobile robots capable to assemble in formation and navigate in indoor environment.

Structure of the thesis

The following summary briefly presents the contents of the thesis and its main contributions.

Chapter 1: Preliminaries

In this first chapter we review some basic concepts and theoretical tools that have been used in this work. We start with a brief introduction to the graph theoretic methods that are the basis of the analysis of the interactions in a multi-agent systems. After that, we introduce the agreement problem and we analyse the existent results concerning its convergence. A similar theoretic introduction is then given for the hybrid dynamic systems, where we recall the concept of hybrid time domain, hybrid system solution and trajectory and where we introduce a particular formulation of the asymptotic stability condition for hybrid system. The last section is dedicated to introduce the dynamics of a particular type of robotic vehicles belonging to the wider class of the non-holonomic dynamic systems.

Chapter 2: Hybrid framework for agreement in multi-agent systems

In this chapter we present the main theoretic result of our work based on the novel results reported in [Borzone et al., 2018b, Borzone et al., 2018a, Borzone et al., 2019]. We focus our attention to the agreement problem of a particular class of agent dynamics, that is the piece-wise reference tracking non-linear systems. A two stage approach based on the separation between the decentralised reference agreement task and the local reference tracking task is proposed. The agents update their references thanks to sporadic relative measurements of the state of the neighborhood. The hybrid nature of this set-up is considered in the stability analysis that follows. A sufficient stability condition involving the time between one update and the following one is proposed in the case of connected undirected interaction topology and jointly connected directed topology. Finally, some results dealing with the robustness of the solution with respect to perturbed relative measurements are provided in the last section.

Chapter 3: Consensus in networks of non-holonomics robots

Here we apply the framework of analysis developed in the previous chapter to a network of non-holonomic vehicles. The application is inspired by the multi-robot fleet project at the CEA FFLOR platform which was part of this work. The same case of study has been used in [Borzone et al., 2018a, Borzone et al., 2019, Borzone et al., 2018b] to validate the theoretic results. The two stage approach is applied to a multi-agent system of non-holonomic vehicles. Once the local piece-wise reference tracking dynamic is shown to fulfil the set of assumptions formulated in the previous chapter, the stability results are applied for the rendez-vous problem and the formation realization problem for a set of differential wheels mobile robots. In the last section, we validate and test the proposed solution in a Matlab simulation environment.

Chapter 4: Management and monitoring of a fleet of Autonomous Mobile Robots (AMR)

In this chapter we introduce the fleet management system developed in the framework of the CEA tech FFLOR platform in Trémery. The system aims at remotely controlling a fleet of robotic vehicles with relative sensing capabilities, for monitoring and data

collection purpose. The system is composed by a remote operator, a base station and a group of Autonomous Mobile Robots to deployed in an industrial hazardous environment. After an introduction about the Robotic Operating System (ROS) libraries used as development framework for the software, we give an insight of the implementation of the algorithm proposed in the previous chapters. The implementation is carried out both on simulated and real mobile robotic devices. Then we describe the architecture of the fleet management system and we finally test our set up via a comparison of some chosen performance index extracted from the real platform and the simulated one.

Publications

The research carried out during this PhD has inspired and given rise to the following publications

International journal

- T. Borzone, I.-C. Morărescu, M. Jungers, M. Boc, C. Janneteau, "Hybrid framework for consensus in directed and asynchronous network of non-holonomic agents", *IEEE Control Systems Letters, 2018*, Vol. 2(4), pages 707-712.

International conferences

- T. Borzone, I.-C. Morărescu, M. Jungers, M. Boc, C. Janneteau, "Hybrid framework for consensus in fleets of non-holonomic robots", *American Control Conference (ACC), June 2018, Milwaukee, USA*, pages: 4299-4304.
- T. Borzone, I.-C. Morărescu, M. Jungers, M. Boc, C. Janneteau, "Hybrid formalism for consensus of a general class of multi-agent systems with biased measurements", *European Control Conference (ECC), June 2019, Naples, Italy*.

French national conferences

- T. Borzone, I.-C. Morărescu, M. Jungers, M. Boc, C. Janneteau, "Approche hybride pour le consensus pour les réseaux dirigés et asynchrones d'agents non-holonomes", *Journées Doctorales - Journées Nationales MACS, JD-JN-MACS 2019, June 2019, Bordeaux, France*.

Chapter 1

Preliminaries

This chapter provides mathematical and theoretical tools and concepts useful for the further developments presented in this manuscript. First we provide few definitions on graph theory, then the notion of undirected and directed graphs and networks are introduced. We will then introduce the agreement problem for networks of multi-agent systems in discrete time, providing some instrumental results from seminal works in the literature. In the third section, a brief introduction to hybrid systems theory is provided, together with some standard definitions concerning the stability of such systems.

1.1 Graph Theory

We call agent a dynamical system which interacts with others forming a network, i.e. a finite set of agents, each one with his dynamics, possibly different one from another. The agents in the network are interconnected according to some specific interaction topology, which describes the flow of information among the agents in the network and consequently it determines the set of neighbors of each agent. The decentralized control techniques aim at finding suitable control laws that, using only local interactions from the neighborhood of one agent, achieve some emergent behaviours for the whole network. Interactions can be interpreted in several ways: from a more high level perspective the interactions can take place with the communication of information through some sort of physical infrastructure, however it is always possible to see interactions as the exchange of relative sensing. Whatever the nature of the interaction, the study of graph theory is a fundamental component for the analysis of multi-agent systems.

To this purpose will make extensively use of the definitions given in [Godsil and Royle, 2013] and [Mesbahi and Egerstedt, 2010] about graphs.

Definition 1.1. *The time varying weighted graph $\mathcal{G}(t)$ is the triplet $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t), P(t))$, where $\mathcal{V}$ is a set of N vertices or nodes $\mathcal{V} = \{1, \cdots, N\}$, $\mathcal{E}(t) \subset \mathcal{V}^2$ is a set of edges or pairs of distinct vertices $(j, i), i \neq j$ which represents the flow of information from j to i and finally $P(t) \in \mathbb{R}^{N \times N}$ is a matrix whose elements $P_{ij}(t) \geq 0$ weight the flow of information from node j to node i.*

Remark 1.1. *The case of un-weighted graphs can be defined as a special case, by imposing that* $P_{ij}(t) \in \{0, 1\}$*, meaning that* $P_{ij}(t) = 1$ *when* $(j, i) \in \mathcal{E}(t)$ *and* $P_{ij}(t) = 0$ *if there is no connection from the two nodes* $(j, i) \notin \mathcal{E}(t)$*.*

The notion of directed or undirected graphs come naturally as follows:

Definition 1.2. *A time varying weighted graph is* **undirected** *if the edges are bidirectional, i.e. for each* $j, i \in \mathcal{V}$*,* $(j, i) \in \mathcal{E}(t) \Leftrightarrow (i, j) \in \mathcal{E}(t)$ *and* $P_{ij}(t) = P_{ji}(t)$ *that is the weights matrix is symmetric* $P(t) = P(t)^\top$*; it is called* **directed** *otherwise (we will then call it* **digraph***).*

Even if this mathematical object is inherently set theoretic it is also suitable to be represented through graphical schemes where the elements of $\mathcal{V}$ correspond to vertices. In the case of undirected graph the elements of $\mathcal{E}(t)$ are lines which link the vertices one another, whereas in the digraph case, the edges are represented as arrows. In the graphical representation of a digraph, for each $(j, i) \in \mathcal{E}(t)$ we draw an arrow whose tail and head are respectively on the vertices j and i. Hence, we say that vertex j is **adjacent** to vertex i and of course, in the undirected case, adjacency is a reflexive property of vertices. Thanks to the graphical representation the next definitions come more naturally

Definition 1.3. *A* **path** *of length* l *in a graph* $\mathcal{G}(t)$ *is a union of directed edges* $\bigcup_{k=1}^{l}(i_k, j_k)$ *such that* $i_{k+1} = j_k$*,* $\forall k \in \{1, \dots, l-1\}$*. Two nodes* i *and* j *are* **connected nodes** *of graph* $\mathcal{G}(t)$ *if there exists at least a path in* $\mathcal{G}(t)$ *from* i *to* j *(i.e.* $i_1 = i$ *and* $j_l = j$*).*

For what concerns undirected graphs, a path is graphically identified as a sequence of lines which links two nodes, whereas for digraph the path is simply a sequence of oriented arrows (see Figure 1.1). When $i_1 = j_l$, that is the starting and ending vertices of a path coincide, the path is called a **cycle**. We define **in-degree** of a node $i \in \mathcal{V}$ as the number of adjacent nodes to i.

Definition 1.4. *A* **connected** *graph (digraph)* $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t), P(t))$ *at time* $t \in \mathbb{R}$ *is such that there exists a node* $i \in \mathcal{V}$ *such that for any other node* $j \in \mathcal{V} \setminus \{i\}$ *there is a path from* i *to* j*. Then* i *is called* **root** *of the graph.*

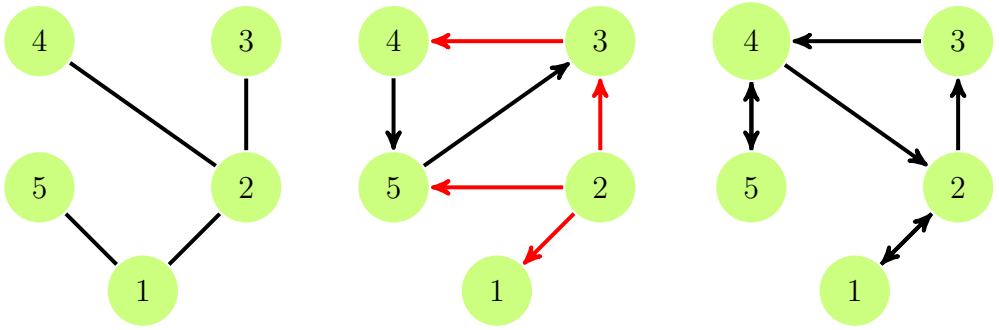
For a given connected digraph $\mathcal{G}(t)$ a **spanning tree** is the sub-graph $\mathcal{G}^*(t) = (\mathcal{V}^*, \mathcal{E}^*(t), P^*(t))$ obtained with the same set of vertices $\mathcal{V}^* = \mathcal{V}$, without cycles, such that all vertices have in-degree 1 except the root node (in-degree 0), and $P^*(t)$ obtained as in Definition 1.1 considering the newly created edge set $\mathcal{E}^*(t)$. We can therefore say that a graph is connected at time t if and only if there exists a spanning tree.

A stronger notion of connectivity is given in the following definition.

Definition 1.5. *A* **strongly connected** *graph (digraph)* $\mathcal{G}(t)$ *is such that any* $i \in \mathcal{V}$ *is root of the graph. In other words for any couple* (i, j) *there exists a path between these two nodes.*

Remark 1.2. *For undirected graphs the connectivity and the strong connectivity are equivalent, because the flow of information is directionless. In Figure 1.1 these notions are made more clear.*

Beside the instantaneous connectivity, when one deals with time-varying graph topology, it is also interestning to consider the connection properties over time. In order to do this we should define the union graph

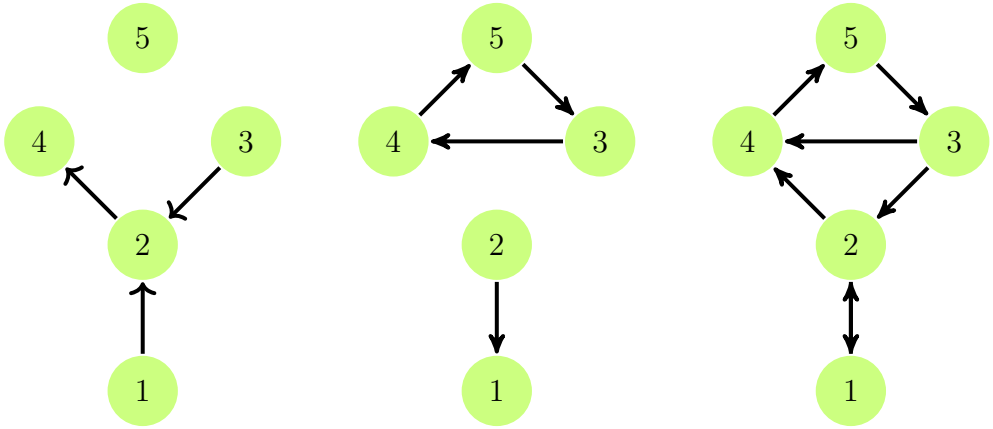


(a) Connected graph. (b) Connected digraph. (c) Strongly connected di- graph.

Figure 1.1: Graphical example of a connected graph in Figure 1.1(a) and digraph in Figure 1.1(b) where node 2 corresponds to the root of the graph. Indeed from the root node any other node can be reached an there exists a spanning tree (represented in red), but there is no strong connectivity because there is no way to reach 2 from other nodes. The stronlgy connectivity is represented in Figure 1.1(c).

Definition 1.6. *Given two graphs* $\mathcal{G}(t) = \{\mathcal{V}, \mathcal{E}(t), P(t)\}$ *and* $\mathcal{G}(s) = \{\mathcal{V}, \mathcal{E}(s), P(s)\}$ *associated with the time instant* t *and* s *we define the* **union graph** *as* $\bar{\mathcal{G}} = \mathcal{G}(t) \cup \mathcal{G}(s) = \{\mathcal{V}, \mathcal{E}(t) \cup \mathcal{E}(s), P(t) + P(s)\}$.

According to this Definition we will say that two graphs are **jointly connected** if their union is connected and **jointly strongly connected** if their union is strongly connected (see for instance Figure 1.2).



(a) $\mathcal{G}(t)$. (b) $\mathcal{G}(s)$. (c) $\bar{\mathcal{G}} = \mathcal{G}(t) \cup \mathcal{G}(s)$.

Figure 1.2: Graphical example of two non connected graphs $\mathcal{G}(t)$ and $\mathcal{G}(s)$ whose union $\bar{\mathcal{G}}$ is indeed jointly strongly connected.

1.2 Consensus in multi-agent systems

One of the most interesting and also most extensively studied decentralized control task consists in reaching an agreement between the agents over the value of some specific quantities of interest. The agreement can be seen as the final stage of a dynamic process that we can identify with the consensus during which the agents in the network exchange information to accomplish their task. The network can be considered having static or time varying topology. This work will focus on the latter case. Considerations about parallel and distributed computation in network of processors motivated in the early 1980s the studies of [Tsitsiklis, 1984, Bertsekas and Tsitsiklis, 1989] and at in the same years the implications in the domain of the chaotic iterations inspired the works of [Bru et al., 1988]. Later the heading synchronization and the flocking problem introduced by the Vicsek model [Vicsek et al., 1995] inspired the seminal work of [Jadbabaie et al., 2003, Moreau, 2005, Ren and Beard, 2005]. These first works focused on the consensus for networks of agents with single integrator dynamics. The same problem has been later analyzed and studied for several different applications and in several different frameworks. For example: in [Olfati-Saber and Murray, 2004] the continuous time version of the consensus is formulated for a switching topology with the aid of a time varying Laplacian matrix associated with the graph. The work in [Seuret et al., 2008] goes further in details of what happens in presence of communication delays between the nodes.

Our interest here is to introduce some fundamental results on the discrete time version of the consensus algorithm. We take a time varying network of N agents described by the graph $\mathcal{G}(k) = (\mathcal{V}, \mathcal{E}(k), P(k))$ and we associate with each agent a state value $x_i \in \mathbb{R}^{n_i}$. In the following, we will assume that the agents have to reach an agreement over the value of their state x_i, assumed to have the same dimension independently of the agent, namely $x_i \in \mathbb{R}^n$, $\forall i \in \mathcal{V}$. The evolution of the state follows the discrete time dynamics

$$x_i(k+1) = u_i(k), \text{ for } k \in \mathbb{N}, \tag{1.1}$$

where $u_i \in \mathbb{R}^n$ is the input signal. The consensus algorithm represents a particular choice of u_i of the form

$$x_i(k+1) = \sum_{j=1}^{N} P_{ij}(k) x_j(k), \text{ for } k \in \mathbb{N}. \tag{1.2}$$

We define $\mathcal{N}_i(k)$ the set of neighbors of agent i at time k, $\mathcal{N}_i(k) = \{j \in \mathcal{V} : (j,i) \in \mathcal{E}(k)\}$. From Definition 1.1, it follows that $P_{ij}(k) = 0 \Leftrightarrow j \notin \mathcal{N}_i(k)$.

Collecting the state of all agents in a single vector $x = (x_1^\top, \ldots, x_N^\top)^\top \in \mathbb{R}^{n \cdot N}$, it is possible to write down the dynamics in matrix form as

$$x(k+1) = P(k) \otimes \mathbf{I}_n x(k), \text{ for } k \in \mathbb{N}, \tag{1.3}$$

where we used the Kroneker product to represent that the consensus algorithm expands to all the component of x_i. The agreement condition can be now formally defined from a set theoretic point of view when the set

$$\mathscr{A} = \left\{ x \in \mathbb{R}^{n \cdot N} : x_i = x_j, \forall (i,j) \in \mathcal{V}^2 \right\}, \tag{1.4}$$

is globally attractive and positive invariant with reference to dynamics (1.2).

A fundamental assumption that several convergence results have in common concerns the weight matrix $P(k)$.

Assumption 1.1. *Given a time varying graph $\mathcal{G}(k) = (\mathcal{V}, \mathcal{E}(k), P(k))$ the weight matrix $P(k)$ is such that:*

(I) $P_{ii}(k) \geq b$.

(II) $P_{ij}(k) = 0$ *if* $(j,i) \notin \mathcal{E}(k)$ *and* $P_{ij}(k) \geq a$ *if* $(j,i) \in \mathcal{E}(k)$.

(III) $\sum_{j=1}^{N} P_{i,j}(k) = 1$, *for all* $i \in \mathcal{V}$ *and for all* $k \in \mathbb{N}$.

for some $a, b > 0$.

Remark 1.3. *According to Condition I and II in Assumption 1.1, $P(k)$ has always positive elements on the diagonal and his entries are zero only if $(j,i) \notin \mathcal{E}(k)$. Assumption III is called row stochasticity and in the undirected graph case this property implies the column stochasticity as a direct consequence of the symmetric structure of $P(k)$. The fact of using two different bounds a, b for the elements of $P(k)$ might seem useless (it would be enough to choose the larger of the two), nevertheless it will be more clear in Chapter 2 how this additional degree of freedom in the design impacts on the result of this work. In the following, if not specified, we will consider a = b, i.e. the same lower bound over the influence strength for the connected agents and the i^{th} agent itself.*

Thanks to Assumption 1.1, $P_{ii}(k) = 1 - \sum_{\substack{j=1 \\ j \neq i}}^{N} P_{ij}(k)$ for all $k \in \mathbb{N}$, hence we can easily verify that the decentralized discrete consensus algorithm (1.2) is equivalent to the following

$$x_i(k+1) = x_i(k) + \sum_{\substack{j=1 \\ j \neq i}}^{N} P_{ij}(k)(x_j(k) - x_i(k)), \text{ for } k \in \mathbb{N}. \tag{1.5}$$

1.2.1 Time varying directed networks

As already discussed, some works on this particular case exist in literature since the early 80s as the one of [Bertsekas and Tsitsiklis, 1989] with a convergence result, later exploited also from [Blondel et al., 2005], which hold on the following additional assumptions

Assumption 1.2 (Joint connectivity)**.** *The union graph $\bar{\mathcal{G}}(k_0) = \cup_{k \geq k_0} \mathcal{G}(k)$ is jointly connected for all $k_0 \in \mathbb{N}$.*

Assumption 1.3 (Bounded Intercommunication Interval)**.** *If j interacts with i an infinite number of times (that is, if $(j,i) \in \mathcal{E}(k)$ infinitely often), then there is some $L \in \mathbb{N}$ such that, for all $k \in \mathbb{N}$, $(j,i) \in \mathcal{E}(k) \cup \mathcal{E}(k+1) \cup \cdots \cup \mathcal{E}(k+L-1)$.*

The first assumption ensures that any agent has a certain influence on any other agent over time. Assumtion 1.3 is a persistance of excitation condition which states that interactions occur sufficiently often in time.

The convergence is guaranteed thanks to the following theorem.

Theorem 1.1. *Given a time varying network of N agents with state x_i, described by the directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}(k), \mathcal{P}(k))$, let Assumption 1.1, 1.2 and 1.3 hold, then the attractor $\mathscr{A}$ is globally attractive and forward invariant with reference to dynamics* (1.2).

The proof proposed by [Bertsekas and Tsitsiklis, 1989] goes beyond the purpose of this work but the main idea, which is often used in results of this type (see for istance [Borzone et al., 2018a] or [Morărescu et al., 2015]), is as follows. We consider the simplified case where $x_i \in \mathbb{R}$ and we define $m(k) = \min_{i \in \mathcal{V}} x_i$ and $M(k) = \max_{i \in \mathcal{V}} x_i$. We can easily show that $M(k)$ and $m(k)$ are respectively non increasing and non decreasing thanks to row stochasticity of $P(k)$. Eventually we need to verify that the difference $\Delta(k) = M(k) - m(k)$ contracts by a constant factor after a sufficiently large amount of time; the value of this time interval is chosen so that each agent in the network interacts with every other agent (directly or indirectly); the result follows from the fact that each interaction has a non-zero strength defined by $P_{ij}(k) \geq a$ and the chain of this minimal interactions characterizes the contraction of $\Delta(k)$.

If the bounded intercommunication interval assumption is not fulfilled, the consensus algorithm does not guarantee the attractivity of $\mathscr{A}$ as shown for instance in Exercise 3.1, p.517 of [Bertsekas and Tsitsiklis, 1989].

Some years later, a new formulation of Theorem 1.1 was presented by [Ren and Beard, 2005], where the assumption on the bounded intercommunication interval and on the joint strong connectivity were condensed in an assumption over the existence of a spanning tree frequently enough for the topology of the network.

1.2.2 Time varying undirected networks

The networks whose topology is described by an undirected graph were not historically the first to be analyzed. Works on this subject usually exploit spectral and algebraic properties of the time varying weight matrix $P(k)$ that in the case of undirected graph is symmetric $P(k) = P^\top(k)$.

One of the most known convergence result was proposed in [Jadbabaie et al., 2003] and takes advantage of Assumption 1.2 to formulate the following theorem.

Theorem 1.2. *Given a time varying network of N agents with state x_i, described by the undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}(k), \mathcal{P}(k))$, let Assumption 1.1 and 1.2 hold, then the attractor $\mathscr{A}$ is globally attractive and forward invariant with respect to dynamics* (1.2).

1.2.3 Convergence analysis

Theorems 1.1 and 1.2 can be reformulated as results on the convergence of products of stochastic matrices.

Corollary 1.1 ([Blondel et al., 2005]). *Consider an infinite sequence of stochastic matrices $\{P(k)\}_{k\in\mathbb{N}}$ that satisfies Assumptions 1.1 and 1.2. If either Assumption 1.3 (bounded intercommunication interval) is satisfied, or if the graph is undirected, then there exists a non-negative vector $d \in \mathbb{R}^N$ such that*

$$\lim_{k\to\infty} P(k)P(k-1)\cdots P(1)P(0) = \mathbb{1}_N d^\top.$$

The proof of this goes beyond the purpose of our work (see [Jadbabaie et al., 2003] or [Blondel et al., 2005] for further details), but what is interesting is the idea at the basic tools and ideas. We consider again a set of N agents with scalar state $x_i \in \mathbb{R}$; a useful property of row stochastic matrices is that $P(k)\mathbb{1}_N = \mathbb{1}_N$ which is a direct consequence of the fact that the sum along the row is unitary. This also implies that $\text{span}\{\mathbb{1}_N\}$ is a $P(k)$-invariant subspace. Consider now a matrix $U \in \mathbb{R}^{(N-1)\times N}$, defining an orthogonal projection on the space orthogonal to $\text{span}\{\mathbb{1}_n\}$, implying $\ker(U) = \text{span}\{\mathbb{1}_N\}$ or equivalently $U\mathbb{1}_n = 0$. Given such a matrix U the equations

$$UP(k) = Q(k)U, \text{ for } k \in \mathbb{N} \tag{1.6}$$

have a unique solution $Q(k) \in \mathbb{R}^{(N-1)\times(N-1)}$; moreover the spectrum of $Q(k)$ is exactly the spectrum of $P(k)$ except the unitary eigenvalue, i.e. $\sigma(P(k)) = 1 \cup \sigma(Q(k))$ where σ denotes the spectrum of a matrix.

Using these considerations, the attractiveness of set $\mathscr{A}$ for system (1.3) is proven showing the attractiveness of the origin for the new dynamics concerning the projected state:

$$\tilde{x}(k+1) = Q(k)\tilde{x}(k), \text{ for } k \in \mathbb{N}, \tag{1.7}$$

with $\tilde{x} \in \mathbb{R}^{N-1}$, $\tilde{x} = Ux$.

1.3 Hybrid dynamical systems

To easily follow the developments proposed in this manuscript we also need some basic results from hybrid systems theory. According to the definition in [Goebel et al., 2012] a hybrid dynamical system, or just a hybrid system, is a dynamical system that exhibits characteristics of both continuous-time and discrete-time dynamical systems or a dynamical system that is modeled with a combination of common modeling tools for continuous time and discrete-time dynamical systems. One of the hybrid systems model commonly used in literature can be represented in the form

$$\begin{cases} \dot{q} = F(q,d) & (q,d) \in C \\ q^+ = G(q,d) & (q,d) \in D, \end{cases} \tag{1.8}$$

where $q \in \mathbb{R}^{n_q}$, $d \in \mathbb{R}^{n_d}$ and $F : C \to \mathbb{R}^{n_q}$ and $G : D \to \mathbb{R}^{n_q}$ are continuous. The state of the hybrid system, represented by q, can change according to a differential equation $\dot{q} = F(q,d)$, while in the **flow set** C, and it can change according to a difference equation $q^+ = G(q,d)$ while in the **jump set** D. The q^+ represents the shortening for the right limit of $q(t)$, namely $q^+ = q(t^+) = \lim_{\tau\searrow t} q(\tau)$. Vector d denotes the input.

In order to define a notion of solution for system (1.8) it is important to define a concept of time parametrization. Hence, the first step is to introduce the concept of hybrid time domains.

Definition 1.7 ([Goebel et al., 2012]). *A subset $E \subset \mathbb{R}_+ \times \mathbb{N}$ is a* **compact hybrid time domain** *if $E = \cup_{k=0}^{K}([t_k, t_{k+1}], k)$ for some finite sequence of times $0 = t_0 \leq t_1 \leq \cdots \leq t_{K+1}$. E is a* **hybrid time domain** *if for all $(T, K) \in E$, the set $E \cap ([0, T] \times \{0, 1, \ldots, K\})$ is a compact hybrid time domain. On a hybrid time domain there is a natural ordering of points, that is $(t, j) \leq (s, k)$ if $t + j \leq s + k$, and $(t, j) < (s, k)$ if $t + j < s + k$.*

Equivalently, E is a compact hybrid time domain if E is a union of a finite sequence of intervals $[t_k, t_{k+1}] \times \{k\}$, while E is a hybrid time domain if it is a union of a finite or infinite sequence of intervals $[t_k, t_{k+1}] \times \{k\}$, with the last interval (if existent) possibly of the form $[t_k, T)$ with T finite or $T = \infty$.

We call **hybrid signal** a function which is defined over a hybrid time domain. For instance the **hybrid input** is a hybrid signal $d : \text{dom } d \to \mathbb{R}^{n_d}$ such that $d(\cdot, k)$ is Lebesgue measurable and locally essentially bounded for every k, where we denoted by $\text{dom } d$ the hybrid time domain of the hybrid signal d.

An other important hybrid signal is the hybrid arc.

Definition 1.8 ([Cai and Teel, 2009]). *An hybrid signal $q : \text{dom } q \to \mathbb{R}^{n_q}$ is called a* **hybrid arc** *if $q(\cdot, k)$ is locally absolutely continuous for every k.*

The pair hybrid arc and hybrid input are the fundamental ingredients to define a solution to a hybrid system.

Definition 1.9 ([Cai and Teel, 2009]). *A hybrid arc $q : \text{dom } q \to \mathbb{R}^{n_q}$ and a hybrid input $d : \text{dom } d \to \mathbb{R}^{n_d}$ are a solution pair to the hybrid system (1.8) if*

(I) $\text{dom } q = \text{dom } d$ *and* $(q(0, 0), d(0, 0)) \in C$;

(II) *for all $k \in \mathbb{N}$ and almost all $t \in \mathbb{R}_+$ such that $(t, k) \in \text{dom } q$ we have $(q(t, k), d(t, k)) \in C$ $\frac{\mathrm{d}}{\mathrm{d}t} q(t, k) = F(q(t, k), d(t, k))$;*

(III) *for every $(t, k) \in \text{dom } q$ such that $(t, k + 1) \in \text{dom } q$, $(q(t, k), d(t, k)) \in D$ and $q(t, k + 1) \in G(q(t, k), d(t, k))$.*

We are interested in the asymptotic properties of the hybrid system (1.8) with respect to the origin $q = 0$. In order to analyze it we should define special comparison functions, known as class $\mathcal{K}$ and class $\mathcal{KL}$ functions.

Definition 1.10 ([Khalil, 2001]). *A continuous function $\gamma : [0, \alpha) \to [0, \infty)$ is said to belong to class $\mathcal{K}$ if it is strictly increasing and $\gamma(0) = 0$. It is said to belong to class $\mathcal{K}_\infty$ if it belongs to class $\mathcal{K}$ and in addition $\alpha = \infty$ and $\gamma(r) \longrightarrow \infty$ as $r \to \infty$.*

Definition 1.11 ([Khalil, 2001]). *A continuous function $\beta : [0, \alpha) \times [0, \infty) \to [0, \infty)$ is said to belong to class $\mathcal{KL}$ if for each s, $\beta(r, s) \in \mathcal{K}$ with respect to r and, for every fixed r, $\beta(r, s)$ is decreasing with respect to s and $\beta(r, s) \to 0$ as $s \to \infty$.*

The concept of comparison function allows us to introduce a standard definition of asymptotic stability for the hybrid system when the input $d \equiv 0$.

Definition 1.12 ([Liberzon et al., 2014]). *The origin $q = 0$ is* **globally pre-asymptotically stable** *for system (1.8) if there exists a function β of class $\mathcal{KL}$ such that*

$$|q(t, k)| \leq \beta(|q(0, 0)|, t + k), \quad \textit{for all } (t, k) \in \text{dom } q. \tag{1.9}$$

The asymptotic properties can be redefined in presence of input d thanks to the following Definition.

Definition 1.13. *The hybrid system (1.8) is* **pre-input-to-state stable** *or pre-ISS with respect to the input d and the origin $q = 0$ if there exist a function β of class $\mathcal{KL}$ and a function γ of class $\mathcal{K}_\infty$ such that*

$$|q(t,k)| \leq \beta(|q(0,0)|, t+k) + \gamma(|d|_{(t,k)}), \quad \text{for all } (t,k) \in \operatorname{dom} q, \tag{1.10}$$

where $|d|_{(t,k)}$ stands for the supremum of d up to the hybrid time (t,k).

Remark 1.4. *Since $\max\{x,y\} \leq x+y$ for positive x and y, the pre-ISS condition is often stated in a different form (see for example [Cai and Teel, 2009]), namely, asking for the existence of some β and γ functions of class $\mathcal{KL}$ and $\mathcal{K}_\infty$ respectively (in general different from the ones in the pre-ISS definition) such that*

$$|q(t,k)| \leq \max\{\beta(|q(0,0)|, t+k), \gamma(|d|_{(t,k)})\}, \quad \text{for all } (t,k) \in \operatorname{dom} q.$$

1.4 Alternative representation

The hybrid system structure which is considered in this work presents discrete time events that are regulated by some external element like a timer. Moreover in our case the input d appears only as a disturbance in the jump dynamics. Our approach is to redefine such a simplified case considering a sequence of time events $\mathcal{T} = \{t_k\}_{k\in\mathbb{N}}$, $t_0 = 0$, then to represent the model in a form which does not depend on the definition of C and D.

$$\dot{q}(t) = \widetilde{F}(q(t)) \qquad t \in \mathbb{R}_+ \setminus \mathcal{T} \tag{1.11}$$

$$q(t_k) = \widetilde{G}(q(t_k^-), d(t_k^-)) \qquad t_k \in \mathcal{T}, \tag{1.12}$$

with $q(0^-) = q(0)$, $d(0^-) = d(0)$. The flow and jump maps now correspond to $\widetilde{F} : \mathbb{R}^{n_q} \to \mathbb{R}^{n_q}$ and $\widetilde{G} : \mathbb{R}^{n_q} \times \mathbb{R}^{n_d} \to \mathbb{R}^{n_q}$. Note that in (1.12), differently from (1.8), we expressed the updated value of q as a function of the left limit $q(t)$, $q(t^-) = \lim_{\tau \nearrow t} q(\tau)$.

Remark 1.5. *The notation proposed in (1.11)-(1.12) is inspired by the literature on impulsive systems (see for instance [Hetel et al., 2013, Morărescu et al., 2015]) where the usage of jump set, flow set and hybrid time parametrization described in the previous section is replaced by the usage of a discrete time events sequence $\mathcal{T}$, and a standard time parametrization. The advantage of such notation is that it allows to analyze the stability of the global hybrid system by first studying the stability of the discrete subsystem.*

A practical feature often used to characterize a discrete time events sequence is the dwell time.

Definition 1.14. *The real positive number $\underline{\tau} > 0$ is called* **dwell time** *of a sequence of discrete time events $\mathcal{T}$ if $t_{k+1} - t_k > \underline{\tau}$, for all $k \in \mathbb{N}$.*

With this new formulation, we can define a solution parametrized in time t, $q : \mathbb{R} \to \mathbb{R}^{n_q}$, possibly discontinuous, and we can separately analyze the asymptotic properties of this trajectory for all $t_k \in \mathcal{T}$ or between one jump and the other for all $t \in [t_k, t_{k+1})$.

For instance if we decide to evaluate $q : \mathbb{R} \to \mathbb{R}^{n_q}$ for all $t_k \in \mathcal{T}$ we obtain a discrete time evolution and we can define the following asymptotic property respectively in absence or in presence of the discrete input d.

Definition 1.15 ([Jiang and Wang, 2001]). *The origin $q = 0$ is* **globally asymptotically stable**, *with respect to the zero input discrete dynamics (1.12) if there exists a function $\tilde{\beta}$ of class $\mathcal{KL}$ such that*

$$|q(t_k)| \leq \tilde{\beta}(|q(0)|, k), \quad \forall t_k \in \mathcal{T}. \tag{1.13}$$

Definition 1.16 ([Jiang and Wang, 2001]). *The discrete dynamics (1.12) is* **input-to-state stable**, *ISS with respect to the input d and the origin $q = 0$ if there exists a function $\tilde{\beta}$ of class $\mathcal{KL}$ and a function $\tilde{\gamma}$ of class $\mathcal{K}_\infty$ such that*

$$|q(t_k)| \leq \tilde{\beta}(|q(0)|, k) + \tilde{\gamma}(|d|_{\{0,\dots,t_k\}}), \quad \forall t_k \in \mathcal{T}, \tag{1.14}$$

where $|d|_{\{0,\dots,t_k\}}$ stands for the supremum of d up to the discrete event at time t_k.

The previous inequality corresponds to an ISS condition for discrete systems and as it has already been showed for the pre-ISS condition, it can be restated in terms of the $\max\{\cdot,\cdot\}$ operator.

1.5 Non-holonomic mobile vehicles

As already introduced, our research is focused on the application of decentralized control techniques for fleets of robotic vehicles. This kind of mobile devices are becoming increasingly important in view of their potential for autonomous intervention. The autonomous control of one particular category of mobile robots has attracted a lot of attention since the early 90s, namely the wheeled mobile robots. This kind of systems are all characterized by the presence of a kinematic constraint which motivates their classification as non-holonomic systems. The literature on such systems is very rich starting from the mechanical classification, modelling and analysis addressed for instance in [Campion et al., 1993] to finely detailed surveys and books on the control techniques for such systems (see [Siegwart et al., 2011], for one of the most recent). As for this introductory section the concepts presented here are mainly inspired by [Siciliano et al., 2010].

Consider a mechanical system whose configuration is described by a vector of generalized coordinates $\xi \in \mathbb{R}^{n_\xi}$. The motion of the system which is represented by the evolution of ξ over time may be free to move everywhere or it may be subject to constraints that can be differently classified (described with equalities or inequalities, time varying or time invariant etc). A constraint represented by the set of n_C equalities

$$C_i(\xi, t) = 0, \qquad i = 1, \dots, n_C < n_\xi, \tag{1.15}$$

where $C_i : \mathbb{R}^{n_\xi} \times \mathbb{R}_+ \to \mathbb{R}$ are of class C^∞ and independent, is called an **holonomic rheonomic constraint** (or just **holonomic constraint** if it does not depend on time t,

$C_i(\xi) = 0$). Constraints which involve the generalized velocities of the mechanical systems $\dot{\xi} \in \mathbb{R}^{n_\xi}$ are called **kinematic constraint**:

$$A_i(\xi, \dot{\xi}) = 0, \qquad i = 1, \dots, n_C < n_\xi, \tag{1.16}$$

with $A_i : \mathbb{R}^{n_\xi} \times \mathbb{R}^{n_\xi} \to \mathbb{R}$ of class C^∞ and independent. It is often possible to express these constraints in matrix form linearly dependent in the generalized velocities $\dot{\xi}$, using the so called Pfaffian form:

$$A(\xi)^\top \dot{\xi} = 0.$$

They constraint the instantaneous admissible motion of the mechanical system reducing the set of generalized velocity directions that can be employed at each configuration.
The existence of n_C holonomic constraints implies the existence of an equal number of kinematic constraints which we can obtain by differentiating:

$$\frac{\mathrm{d}C_i(\xi)}{\mathrm{d}t} = \frac{\partial C_i(\xi)}{\partial \xi}\dot{\xi} = 0, \qquad i = 1, \dots, n_C < n_\xi.$$

However, the converse is not true in general, that is a system of kinematic constraints as the one in (1.16) may or may not be integrable to obtain (1.15). If it is non integrable the kinematic constraints are said to be **non-holonomic**. A mechanical system which is subject to at list one such constraint is called a **non-holonomic system**.

Example 1.1. *Most of the wheeled vehicles are non-holonomic mechanical systems. In particular, this work focuses on the unicycle, whose kinematic is usually described using the generalized coordinates $\xi = (x, y, \theta)$, that is the Cartesian 2D position (x, y) and the orientation θ represented as the angle between the front axis of the vehicle and the x axis of the reference frame (see Figure 1.3). The kinematic constraint expresses the fact that the projection of generalized translation velocity of the vehicle (the vectors $(\dot{x}, 0)$ and $(0, \dot{y})$) is zero along the direction orthogonal to the sagittal plane (the vehicle does not translate along the wheels rotation axis), namely*

$$\dot{x}\sin\theta - \dot{y}\cos\theta = \begin{bmatrix}\sin\theta & -\cos\theta & 0\end{bmatrix}\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix} = A(\xi)^\top\dot{\xi} = 0. \tag{1.17}$$

where the constraint is formulated using the vector field $A : \mathbb{R}^3 \to \mathbb{R}^3$, as a linear combination of the generalized velocities.

The system of n_C constraints implies that the admissible generalized velocities at each configuration ξ belong to a reduced dimension $(n_\xi - n_C)$ subspace which corresponds to the null space of the matrix $A^\top(\xi)$. In fact, if we denote by $\{b_1(\xi), b_2(\xi), \dots, b_{n_\xi - n_C}(\xi)\}$ a basis of $\ker(A^\top(\xi))$, the trajectories of the generalized coordinates $\xi(t)$ can be characterized as solutions of the nonlinear dynamic system

$$\dot{\xi}(t) = \sum_{j=1}^{n_\xi - n_C} b_j(\xi(t))u_j(t), \quad \text{for } t \in \mathbb{R}_+, \tag{1.18}$$

where u_j, $j = \{1, \dots, n_\xi - n_C\}$ are the control inputs.

Example 1.2. *Considering again the unicycle, we can verify that a possible and convenient choice for $b_1(\xi)$ and $b_2(\xi)$ is*

$$\operatorname{span}\{b_1(\xi), b_2(\xi)\} = \operatorname{span}\left\{\begin{bmatrix}\cos\theta\\ \sin\theta\\ 0\end{bmatrix}, \begin{bmatrix}0\\0\\1\end{bmatrix}\right\} = \ker(A^\top(\xi)) = \ker\left(\begin{bmatrix}\sin\theta & -\cos\theta & 0\end{bmatrix}\right).$$

Following the formulation in (1.18) the dynamics of the unicycle can be characterized as the solutions of

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}\cos\theta\\ \sin\theta\\ 0\end{bmatrix} v + \begin{bmatrix}0\\0\\1\end{bmatrix} w, \tag{1.19}$$

where the forward velocity v and the angular velocity w have been chosen as control input, namely $u_1 = v$, $u_2 = w$. A graphical representation of the state and inputs variable are provided in Figure 1.3.

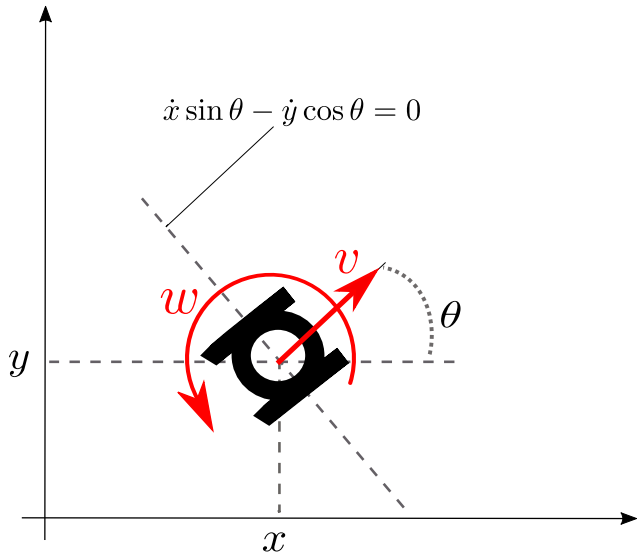


Figure 1.3: A representation of the unicycle non-holonomic system.

1.5.1 Non-holonomic vehicle control

As already anticipated, the problem of controlling such wheeled systems have been deeply studied in the last decades and they essentially divide in two separate groups. In one case the control is designed for a trajectory tracking problem where a desired time varying trajectory is created out of the loop, solving the dynamics

$$\begin{bmatrix}\dot{r}_x\\ \dot{r}_y\\ \dot{r}_\theta\end{bmatrix} = \begin{bmatrix}\cos r_\theta\\ \sin r_\theta\\ 0\end{bmatrix} v_r + \begin{bmatrix}0\\0\\1\end{bmatrix} w_r, \tag{1.20}$$

for some given reference inputs v_r, w_r. Afterward a nonlinear controller is designed to track this trajectory, imposing some constraints on the reference inputs v_r and w_r. Different approaches for this problem have been shown in [Kanayama et al., 1990], [Jiang and Nijmeijier, 1997], [Panteley et al., 1998] and several others.

The second big category of controller regroups the strategies which aims at stabilizing the vehicle in a particular pose. Such control problems are usually referred in literature as point-stabilization problems (see for instance [Canudas de Wit et al., 1993] for an overview on different point stabilization techniques).

There exist in literature some successful attempts to unify these two control problems, considering the point stabilization as a particular case of reference tracking where $v_r, w_r \to 0$ for $t \to \infty$; for instance the works in [Wang et al., 2015], [Morin and Samson, 2009] address this problem. Nevertheless, because of the so-called Brockett necessary conditions [Brockett, 1983], both the unifying and the specific pose stabilization approach to the problem of finding a continuously differentiable and time invariant control law to stabilize the position of the systems turn out to be an impossible task.

In order to formalize the Brockett necessary condition, we start by considering a nonlinear system of the form

$$\dot{\xi}(t) = f\big(\xi(t), u(t)\big), \text{ for } t \in \mathbb{R}_+, \tag{1.21}$$

where $u \in \mathbb{R}^{n_u}$ and $f : \mathbb{R}^{n_\xi} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_\xi}$ is locally Lipschitz.

Remark 1.6. *Note that the unicycle dynamics itself* (1.19) *can be expressed in terms of a map* $f : \mathbb{R}^3 \times \mathbb{R}^2 \to \mathbb{R}^3$ *as*

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v\cos\theta \\ v\sin\theta \\ w \end{bmatrix}. \tag{1.22}$$

We say that a locally Lipschitz feedback law $u = \kappa(\xi)$, $\kappa : \mathbb{R}^{n_\xi} \to \mathbb{R}^{n_u}$ stabilizes the origin $\xi = 0$ if the origin is proven to be Lyapunov stable and an attractor for the closed loop dynamics

$$\dot{\xi}(t) = f\big(\xi(t), \kappa(\xi)\big), \text{ for } t \in \mathbb{R}_+. \tag{1.23}$$

We introduce the Brockett Theorem as the following results:

Theorem 1.3. *Let* $f(\cdot,\cdot)$ *be continuously differentiable* $f \in C^1$ *and assume to have found a continuously differentiable stabilizing control* $\kappa(\cdot) \in C^1$ *for* (1.21)*. Then the image of the mapping* $f : \mathbb{R}^{n_\xi} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_\xi}$ *defined in* (1.21) *contains some neighborhood of the origin* $\xi = 0$*.*

Example 1.3. *In the unicycle case, all points of the form* $(0, a, b)$ *with* $a \neq 0$ *and* $|b| < \pi$ *cannot belong to the image of* f*, since* $v\sin\theta = a \neq 0$ *and* $w = b \neq 0$ *imply* $v\cos\theta \neq 0$ *too. It follows that, in the unicycle case, the image of* f *as defined in* (1.22) *cannot contain any ball centered in the origin, thus it does not exist a stabilizing* $\kappa(\cdot) \in C^1$ *for dynamics* (1.21)*.*

It is possible to prove that for systems in the general non-holonomic form (1.18) the necessary condition is never met, thus in order to stabilize a system like the unicycle in a given position it is necessary to use a piece-wise continuous or time varying control law as proposed in [Canudas de Wit and Sordalen, 1992, Samson and Ait-Abderrahim,

1991] or to switch to a new polar coordinate system where the Brockett necessary condition is met as done in [Astolfi, 1999, Aicardi et al., 1995]. The authors in [Zhong Ping, 2000, Floquet et al., 2003] proposed some time varying solutions in order to overcome the robustness issues that affect the unicycles control.

Chapter 2

Hybrid framework for agreement in multi-agent systems

This chapter presents our main results on the decentralized control design for multi-agent systems with heterogeneous nonlinear dynamics. The study is motivated by the decentralized control of fleets of mobile ground robots with non-holonomic dynamics [Canudas de Wit et al., 1993, Samson and Ait-Abderrahim, 1991], other classes of aerial [Mahony et al., 2012], submarine vehicles [Fossen, 1994], Lagrangian systems and nonlinear systems in general. In such cases the techniques described in Section 1.2 are less effective because of the growing complexity of the agents dynamics. Consequently, consensus algorithms which address the problem for different classes of nonlinear systems received an increasing attention, see for example the review [Cao et al., 2013] and the references therein.

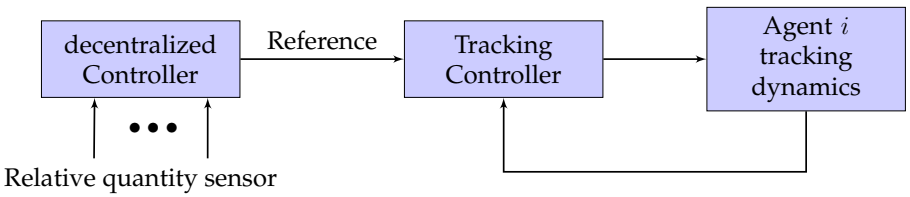


Figure 2.1: Block diagram showing the control strategy for agent i. The tracking is implemented as a separate task and independent from the planning decentralized level which uses the sensing information about the neighborhood of agent i.

The strategy proposed in this chapter consists of a two stage approach, one concerning the decentralized control which generates piece-wise constant reference signals and the other concerning the local convergence of the reference tracking error as represented in Figure 2.1. This results in a hybrid closed-loop dynamics due to the jumping (non-smooth) references that agents have to track. The proposed algorithm follows the idea in [Buşoniu and Morărescu, 2014] in which agents with discrete-time nonlinear dynamics sense sporadically other agents in their neighborhood and, based on their relative measurements, they update their references. A similar two stage approach has been used for the output consensus of linear heterogeneous systems in [De Campos et al., 2012] and [Wieland et al., 2011], as well as in the nonlinear system

synchronization context by [Isidori et al., 2014] and [Casadei and Astolfi, 2018], where the agents interact via a continuous coupling of their references. In [Zhu et al., 2016] the authors present an approach considering that the agents exchange relative output measurements. The same output coupling case is addressed in the analysis framework for synchronization of heterogeneous nonlinear systems proposed by [Panteley and Loría, 2017], where, the coupling term directly intervenes in the agent nonlinear dynamics control.

The previous results apply to very general classes of nonlinear systems and to different kind of interactions, which are modeled with continuous time coupling. In this work, we investigate further the outcomes of a switching network topology and we try to grasp the hybrid characteristics that the approach outlined in the Figure 2.2 brings with it. In fact, such approach couples together a discrete decentralized dynamics with the continuous reference tracking one.

Moreover, in the last section we introduce in the hybrid framework the concept of noisy relative measurements which is reflected in a perturbed coupling between the agents of the network. The perturbed coupling has been extensively studied in literature in particular for what concerns its implication on the consensus process. For instance, noisy coupling has been considered when evaluating mean square based performance of consensus algorithm by [Xiao et al., 2007, Garin and Schenato, 2011]. The work by [Huang and Manton, 2009] considers consensus algorithm with additive measurement noise, whereas in [Nedic et al., 2009] the authors consider the effects of quantization and the work by [Carli et al., 2009] takes into account digital noisy coupling channels.

In the first part, we develop the basic concept and assumptions of the proposed strategy. Then we analyze the asymptotic properties of this approach, deriving some condition on the value of the dwell-time that characterizes the discrete time events sequence of interactions. The latter condition will be developed in sections 2.3 and 2.4 for undirected and directed graph topology respectively. In section 2.5 the robustness with respect to additive sensor noise is analyzed for directed graph topology case.

2.1 Agents heterogeneous error dynamics

We define a set $\mathcal{V}$ of N agents and we associate with each one a state $x_i \in \mathbb{R}^n$, the same dimension for every agent. We assume that the dynamics of the i^{th} system in the network is described by the following differential equations

$$\dot{x}_i(t) = \varphi_i(x_i(t), u_i(t)), \tag{2.1}$$

where $u_i \in \mathbb{R}^s$ is the control action and $\varphi_i : \mathbb{R}^n \times \mathbb{R}^s \to \mathbb{R}^n$ is assumed to be locally Lipschitz. To simplify the presentation, we remove the argument t when it is not explicitly needed. As already anticipated we underline that the considered network is heterogeneous, meaning that each agent can possibly own a different dynamic.

Throughout the work, we denote by $r_i \in \mathbb{R}^n$ the reference that agent i has to track and by $e_i \in \mathbb{R}^n$ the associated error coordinates

$$e_i = x_i - r_i. \tag{2.2}$$

Furthermore, considering a constant reference $\dot{r}_i = 0$, it is possible to model the tracking error dynamics as

$$\dot{e}_i = \dot{x}_i - \dot{r}_i = \varphi_i(x_i, u_i). \tag{2.3}$$

We impose the following assumption on the dynamics of each agent.

Assumption 2.1 (Exponential Convergence of e_i)**.** *For all $i \in \mathcal{V}$, there exists $u_i = \kappa(x_i, e_i)$ such that, the tracking error closed loop dynamics is autonomous*

$$\dot{e}_i = f_i(e_i), \tag{2.4}$$

for some $f_i : \mathbb{R}^n \times \mathbb{R}^s \to \mathbb{R}^m$. Moreover, $u_i = \kappa(x_i, e_i)$ is such that

$$|e_i(t)| \leq c_i \, |e_i(t_0)| \, \mathrm{e}^{-\lambda_i (t - t_0)}, \quad \text{for } t \geq t_0. \tag{2.5}$$

Remark 2.1. *The previous assumption states that as far as the reference is constant there must exist a control feedback for which the tracking error closed loop dynamics is an autonomous system described by the map f_i. In addition, we impose such a controller to make the tracking error exponentially converge to $e_i = 0$.*

Remark 2.2. *Note that Assumption 2.1 inherently implies that every agent i has access to its state x_i, for all t. If this is not the case, we assume the existence of a nonlinear observer (one for each agent i) capable to reconstruct x_i starting from a set of measured output, with an arbitrarily fast convergence time (see for instance the solutions proposed in [Reif et al., 1998]).*

2.2 Agreement problem formulation via hybrid formalism

The agents interact with each other through a decentralized control policy with sporadic state updates, we introduce the sequence of update instants as

$$\mathcal{T} = \left\{ t_k : t_k \in \mathbb{R}_+, t_k < t_{k+1}, \forall k \in \mathbb{N}, \lim_{k \to \infty} t_k = \infty, t_0 = 0 \right\}. \tag{2.6}$$

We define $\mathcal{T}_i$ as the infinite countable subset of $\mathcal{T}$ collecting the update instants only for the agent i. Formally

$$\mathcal{T}_i = \{ t_k \in \mathcal{T} | \exists j \in \mathcal{V}, (i, j) \in \mathcal{E}(t_k) \}. \tag{2.7}$$

Using the definition of $\mathcal{T}_i$ we introduce the last update which affected the agent i as $\underline{t}_{k,i} = \max\{ t_h \in \mathcal{T}_i | h < k \}$.

To represent the decentralized algorithm, we introduce the collective vectors of relative quantities $\bar{r}_i = \left((r_i - r_1)^\top, \cdots, (r_i - r_N)^\top \right)^\top$, $\bar{e}_i = \left((e_i - e_1)^\top, \cdots, (e_i - e_N)^\top \right)^\top$ and $\bar{x}_i = \left((x_i - x_1)^\top, \cdots, (x_i - x_N)^\top \right)^\top$ belonging to $\mathbb{R}^{n \cdot N}$ and related by (2.2).

When an update occurs we should be tempted to apply a map $g_i : \mathbb{R}^{n \cdot (N+1)} \to \mathbb{R}^n$ which describes a general decentralized algorithm directly on the references of other agents in the network to which we have access, that is

$$r_i(t_k) = g_i\big(r_i(t_k^-), \bar{r}_i(t_k^-) \big), \quad \text{for } t_k \in \mathcal{T}_i. \tag{2.8}$$

The usage of $\bar{r}_i$ requires to know the references from other agents, thus it implies a communication between the players in order to pass the information over the network which in some cases can be costly and not reliable. Typical examples are the submarine environment where communications via acoustic modem are difficult (see for example the works of [Ghabcheloo et al., 2009, Soares et al., 2012]) or in multi UGV (Unmanned Ground Vehicle) scenarios where the exchange of information can be subject to severe interferences depending on the environment. On the contrary, equipping the subsystems with relative sensors (e.g relative positioning range and bearing sensors or sonar and Doppler sensors), would lead to the replacement of r_i and $\bar{r}_i$ in (2.8) with quantities related to the real state of the agents

$$r_i(t_k) = g_i\big(x_i(t_k^-), \bar{x}_i(t_k^-)\big), \text{ for } t_k \in \mathcal{T}_i. \tag{2.9}$$

Recalling the definition of neighborhood, in order to enforce algorithm (2.9), we assume the accessibility $\forall t_k \in \mathcal{T}_i$ of the relative state of $\bar{x}_{i,(j)},\ \forall j \in \mathcal{N}_i(t_k)$.

It is worth noting that the error e_i is also affected by the jumps. Indeed, following (2.2), the jumps propagate to the errors since the latter are obtained from the real state, $x_i(t_k) = x_i(t_k^-)$ and the reference $r_i(t_k) \neq r_i(t_k^-)$ through

$$e_i(t_k) = x_i(t_k) - r_i(t_k) = x_i(t_k^-) - g_i\big(x_i(t_k^-), \bar{x}_i(t_k^-)\big), \text{ for } t_k \in \mathcal{T}_i. \tag{2.10}$$

Collecting together the dynamics of the systems during the continuous time and during the reference update instants leads to the following hybrid formalization for all $i \in \mathcal{V}$

$$\begin{cases} \dot{r}_i(t) = 0 \\ \dot{e}_i(t) = f_i\big(e_i(t)\big) \end{cases} \quad \text{for } t \in \mathbb{R}_+ \setminus \mathcal{T}_i, \tag{2.11}$$

$$\begin{cases} r_i(t_k) = g_i\big(r_i(t_k^-) + e_i(t_k^-), \bar{r}_i(t_k^-) + \bar{e}_i(t_k^-)\big) \\ e_i(t_k) = e_i(t_k^-) + r_i(t_k^-) - g_i\big(r_i(t_k^-) + e_i(t_k^-), \bar{r}_i(t_k^-) + \bar{e}_i(t_k^-)\big) \end{cases} \quad \text{for } t_k \in \mathcal{T}_i, \tag{2.12}$$

where for consistency we impose $e(0^-) = 0,\ r(0^-) = x(0)$.

Remark 2.3. *Note that the flow dynamics (2.11) is completely decentralized meaning that each agent tracks its reference and no interaction with other agents is required in continuous time as it is schematized by the block diagram in Figure 2.2.*

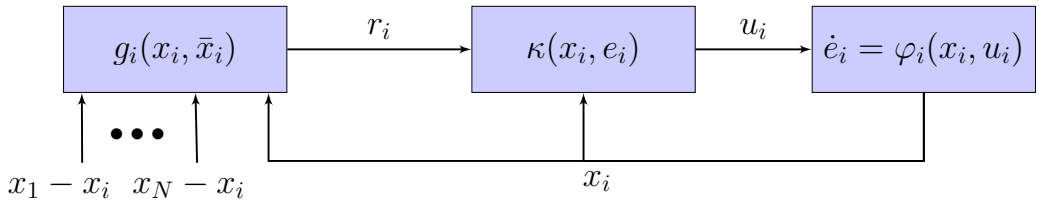


Figure 2.2: Block diagram showing the implementation of the control strategy for the i^{th} agent.

Modifying the algorithm in (2.8) with (2.9), we introduced a coupling between the discrete evolution of the errors and the references inducing a new convergence and robustness analysis of the overall system.

In the following sections, we will make extensive use of the following vectors which collect the references, the errors and the real states for all the agents: $r = (r_1^\top, \cdots, r_N^\top)^\top$, $e = (e_1^\top, \cdots, e_N^\top)^\top$ and $x = (x_1^\top, \cdots, x_N^\top)^\top$ belonging $\mathbb{R}^{n \cdot N}$ and related by (2.2).

2.2.1 Agreement problem

As already anticipated, one of our aims in this work is to design and analyze a decentralized algorithm allowing the agents to reach an agreement on the value of their state. Since from Assumption 2.1 we consider that the agents are able to track exponentially fast a reference we will focus on the design of a jump map g_i as defined in (2.8) which can achieve this emergent behaviour. This can be done in a decentralized way applying a discrete-time linear consensus algorithm (1.5) to the references r_i, $\forall t_k \in \mathcal{T}_i$

$$r_i(t_k) = r_i(t_k^-) + \sum_{\substack{j=1 \\ j \neq i}}^{N} P_{i,j}(t_k)(r_j(t_k^-) - r_i(t_k^-)), \tag{2.13}$$

where $P_{i,j}(t_k)$ are the elements of the row stochastic matrix $P(t_k) \in \mathbb{R}^{N \times N}$ chosen according to Assumption 1.1.

Remark 2.4. *Note that the jump map* (2.13) *for one agent requires only information from the neighbors (in the interconnection graph). Once again, following conditions of Assumption 1.1, if $(i,j) \notin \mathcal{E}(t_k)$, that is if no information is available from agent j at time t_k, the corresponding element $P_{i,j}(t_k) = 0$.*

Applying either the results in Theorem 1.2 or in Theorem 1.1 for respectively undirected or directed time varying network topology, the discrete-time updating rule (2.13) ensures asymptotic agreement (*i.e.* $\lim_{k \to \infty} |r_i(t_k) - r_j(t_k)| = 0$, for all $(i,j) \in \mathcal{V}^2$). Recalling Section 1.2, namely we obtain that the set

$$\mathscr{X} = \left\{ r \in \mathbb{R}^{n \cdot N} : r_i = r_j, \forall (i,j) \in \mathcal{V}^2 \right\},$$

is global asymptotically attractive with reference to dynamics (2.13). As for the errors e_i their convergence is ensured thanks to Assumption 2.1.

Nevertheless, as already discussed in Section 2.2, to avoid communication between the agents we replace the reference related quantities with state related quantities obtaining $\forall t_k \in \mathcal{T}_i$

$$r_i(t_k) = x_i(t_k^-) + \sum_{\substack{j=1 \\ j \neq i}}^{N} P_{i,j}(t_k)(x_j(t_k^-) - x_i(t_k^-)). \tag{2.14}$$

Now, thanks to the error coordinate transformation 2.2, we can write $\forall t_k \in \mathcal{T}$

$$\begin{aligned} r_i(t_k) &= e_i(t_k^-) + r_i(t_k^-) + \sum_{\substack{j=1 \\ j \neq i}}^{N} P_{i,j}(t_k)(e_j(t_k^-) + r_j(t_k^-) - e_i(t_k^-) - r_i(t_k^-)) \\ &= \sum_{j=1}^{N} P_{i,j}(t_k) e_j(t_k^-) + \sum_{j=1}^{N} P_{i,j}(t_k) r_j(t_k^-). \end{aligned} \tag{2.15}$$

Using once again the error coordinate transformation we proceed in the same way from (2.10) to

$$e_i(t_k) = x_i(t_k^-) - r_i(t_k) = e_i(t_k^-) + r_i(t_k^-) - \sum_{j=1}^{N} P_{i,j}(t_k) e_j(t_k^-) - \sum_{j=1}^{N} P_{i,j}(t_k) r_j(t_k^-) \quad (2.16)$$

Finally, putting together (2.15)-(2.16) we can reformulate the dynamics of each agent as a hybrid system where the consensus algorithm appears in place of the map g_i, namely $\forall i \in \mathcal{V}$

$$\begin{cases} \dot{r}_i(t) = 0 \\ \dot{e}_i(t) = f_i\big(e_i(t)\big) \end{cases} \quad \text{for } t \in \mathbb{R}_+ \setminus \mathcal{T}_i, \quad (2.17)$$

$$\begin{cases} r_i(t_k) = \sum_{j=1}^{N} P_{i,j}(t_k) e_j(t_k^-) + \sum_{j=1}^{N} P_{i,j}(t_k) r_j(t_k^-) \\ e_i(t_k) = e_i(t_k^-) + r_i(t_k^-) - \sum_{j=1}^{N} P_{i,j}(t_k) e_j(t_k^-) - \sum_{j=1}^{N} P_{i,j}(t_k) r_j(t_k^-) \end{cases} \quad \text{for } t_k \in \mathcal{T}_i, \quad (2.18)$$

Coupling the jump dynamics of r_i and e_i it will be no longer possible to apply neither Theorem 1.2 for undirected time varying networks topology nor Theorem 1.1 for directed time varying networks topology. For that reason, a new convergence analysis will be addressed in Section 2.3 and 2.4 respectively for undirected and directed networks topology. In Section 2.5 a general class of disturbance signal will be introduced to study the robustness of the algorithm.

2.2.2 Heuristic considerations on stability

Before to formulate a new convergence and robustness analysis it is worth to spend more words on the behaviour of the studied systems. It is our concern to develop few heuristic and practical considerations which help to understand of the following sections.

In order to do so, we start by considering the simplified case where $r_i,\ e_i,\ x_i$ are scalar. Inspecting the hybrid dynamics of agent i in (2.17)-(2.18) we recall that during the flow time the reference r_i is constant and the closed loop error dynamics is such that the norm $|e_i|$ decreases exponentially thanks to Assumption 2.1. A possible example of reference and the error norm trajectories is represented in Figure 2.3.

Looking at the figure we can see how the discrete time events at time $\underline{t}_{1,i}$ and $\underline{t}_{5,i}$ are executed when the error norm is almost zero, $|e_i| \approx 0$, whereas at time $\underline{t}_{2,i}$, $\underline{t}_{3,i}$ and $\underline{t}_{4,i}$ the updates of the reference and consequently the jumps on the errors are executed before $|e_i|$ had converged.

Intuitively if we let converge the closed loop dynamics of each tracking subsystem for a sufficiently large amount of time, this will cause $|e_i| \to 0$ and consequently $x_i \to r_i$. The update law (2.14) which uses the actual state of the agent x_i and the relative state with respect to the neighbors $\bar{x}_i$ will thus tends to the formulation with the references r_i and $\bar{r}_i$, presented in (2.13), which we know to be stable and robust from the literature results presented in Section 1.2.3.

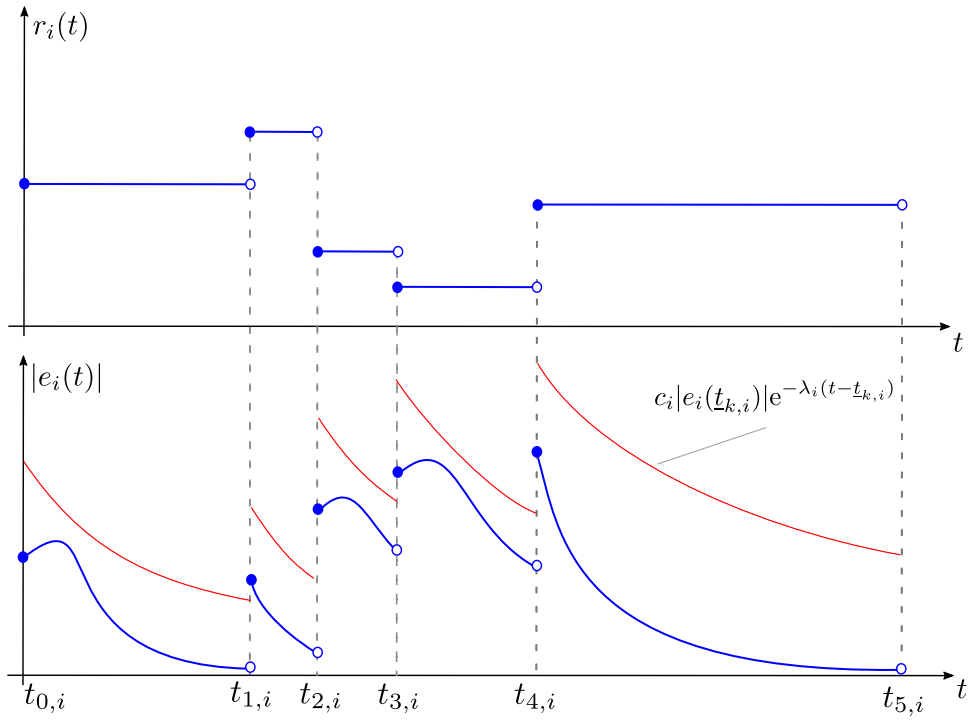


Figure 2.3: Example of trajectories in blue for the reference r_i and the error norm $|e_i|$. In red the bound due to the exponential convergence of the error norm during the flow is represented.

These considerations lead to the question of whether it is possible to find a minimum time between two consecutive discrete time updates, that can ensure the convergence of the overall interconnection. In the next sections, we will try to give an answer to this question formulating two dwell-time conditions for time varying networks with undirected and directed graph topology. In the undirected graph case, the condition will be given over the discrete time events of the overall network (the elements of sequence $\mathcal{T}$), whereas for the directed case, a condition over the time events of each agent i will be formulated (the elements of $\mathcal{T}_i$).

2.3 Undirected time varying topology

The results illustrated in this section have been presented in [Borzone et al., 2018b] which focused on networks described by undirected graph topology. We start extending and reformulating Assumption 1.1 in terms of sequence $\mathcal{T}$.

Assumption 2.2. *Given a time varying graph $\mathcal{G}(t_k) = (\mathcal{V}, \mathcal{E}(t_k), P(t_k))$ the weight matrix $P(t_k)$ is such that*

(I) Assumption 1.1 holds for some $a \in (0,1)$ and $b \in (\underline{b}, 1]$ and $P_{i,j}(t_k)$ is diagonally dominant $\forall t_k \in \mathcal{T}$.

(II) for every $t_k, t_h \in \mathcal{T}$, $t_k \neq t_h$ if $\mathcal{E}(t_k) = \mathcal{E}(t_h)$ then $P(t_k) = P(t_h)$.

Remark 2.5. *The second point in Assumption 2.2 represents a one-to-one correspondence between the topology of the graph and the choice of weight matrix. By mean of this formulation, the possible choices of weight matrices are finite.*

Furthermore, for the rest of the section, we assume the connectivity of $\mathcal{G}$ with the following:

Assumption 2.3 (Connectivity)**.** *The graph topology is such that $\mathcal{G}(t_k)$ is connected for all $t_k \in \mathcal{T}$.*

Taking advantage of the collective vectors $r, e, x \in \mathbb{R}^{n \cdot N}$ we define the new set whose asymptotic properties are of interest to us.

$$\mathscr{X}' = \left\{ e, r \in \mathbb{R}^{n \cdot N} : e = 0,\ r_i = r_j, \forall (i,j) \in \mathcal{V}^2 \right\}. \tag{2.19}$$

The first requirement in (2.19) is equivalent to demand that each robot reaches its own reference and the second requirement means that all the references achieve consensus.

We start from the analysis of the flow subsystem (2.17) stating the following result.

Lemma 2.1. *There exist positive constants c and λ such that for all $t \in [t_k, t_{k+1})$ and $k \in \mathbb{N}$*

$$\begin{aligned} \|r(t)\| &= \|r(t_k)\| \\ \|e(t)\| &\leq c \, \|e(t_k)\| \, \mathrm{e}^{-\lambda (t - t_k)}. \end{aligned} \tag{2.20}$$

Proof. The first inequality comes directly form the fact that according to (2.17) the references r_i does not change between two consecutive jumps. As for the second one, thanks to Assumption 2.1 we know that, for agent i, $|e_i(t)| \leq c_i \, |e_i(t_k)| \, \mathrm{e}^{-\lambda_i (t - t_k)}$ is true for all $t \in [t_k, t_{k+1})$ and $k \in \mathbb{N}$. The latter implies $\|e_i(t)\| \leq \sqrt{n} c_i \, \|e_i(t_k)\| \, \mathrm{e}^{-\lambda_i (t - t_k)}$ using the 2-norm. The decreasing of e_i for all $i \in \mathcal{V}$, yields to the decreasing of e for $c = \sqrt{n} \max_{i \in \mathcal{V}} \{c_i\}$ and $\lambda = \min_{i \in \mathcal{V}} \{\lambda_i\}$. ■

Lemma 2.1 basically states that, as far as the reference is fixed, the tracking dynamics exponentially converges toward $e = 0$, but nothing can be said about the reference r which is kept constant during the flow.

In order to study the asymptotic properties of $\mathscr{X}'$, let us investigate the behavior of the discrete time subsystem (2.18) by rewriting the dynamics in a more compact way using the matrix form (1.3):

$$\begin{aligned} r(t_k) &= \mathcal{P}(t_k) r(t_k^-) + \mathcal{P}(t_k) e(t_k^-) \\ e(t_k) &= (\mathbf{I}_{n \cdot N} - \mathcal{P}(t_k)) e(t_k^-) + (\mathbf{I}_{n \cdot N} - \mathcal{P}(t_k)) r(t_k^-), \end{aligned} \tag{2.21}$$

for $t_k \in \mathcal{T}$ and where $\mathcal{P}(t_k) = P(t_k) \otimes \mathbf{I}_n$.

For the sake of clarity, let us consider a simplified case for $n = 1$ agents with scalar state, reference and error. Taking inspiration from the convergence results illustrated in Corollary 1.1 we know that for any orthogonal projection matrix $U \in \mathbb{R}^{(N-1) \times N}$ the equations

$$U P(t_k) = Q(t_k) U, \text{ for all } t_k \in \mathcal{T} \tag{2.22}$$

have a unique solution $Q(t_k)$. Moreover, the couple $(Q(t_k), P(t_k))$ has the same spectrum except for the eigenvalue 1. Without loss of generality we can choose U such that

$UU^\top = \mathbf{I}_{N-1}$. With such a choice, $\|U^\top\| = 1$ and the proof follows directly from the definition of matrix norm

$$\|U^\top\| = \sup_{\|x\|=1} \frac{\|U^\top x\|}{\|x\|} = \sup_{x^T x=1} \frac{x^\top U U^\top x}{x^T x} = 1, \quad \text{for } x \in \mathbb{R}^{N-1}.$$

Thanks to the properties of the projection matrices we know that

$$\mathbb{R}^N = \mathrm{im}(U) \oplus \ker(U).$$

We denote by $\tilde{r} \in \mathbb{R}^{N-1}$, $\tilde{r} = Ur$ the projection of the reference collective vector; the main consequences of the choice of U and the previous equality is that, for a given $r(t_k)$, there always exists a scalar σ such that

$$r(t_k) = U^\top \tilde{r}(t_k) + \sigma \mathbb{1}_N. \tag{2.23}$$

Now we come back to the original problem, considering once again agents with $n > 1$. Multiplying the first equation in (2.21) to the left by $\mathcal{U} = U \otimes \mathbf{I}_n$ and taking into account (2.22), one obtains for $t_k \in \mathcal{T}$

$$\begin{cases} \mathcal{U} r(t_k) = \mathcal{Q}(t_k)\mathcal{U} r(t_k^-) + \mathcal{Q}(t_k)\mathcal{U} e(t_k^-) \\ e(t_k) = (\mathbf{I}_{n\cdot N} - \mathcal{P}(t_k)) r(t_k^-) + (\mathbf{I}_{n\cdot N} - \mathcal{P}(t_k)) e(t_k^-), \end{cases} \tag{2.24}$$

where $\mathcal{Q}(t_k) = Q(t_k) \otimes \mathbf{I}_n$.
Now taking the second equation in (2.21) and using (2.23) one has

$$\begin{aligned} (\mathbf{I}_{n\cdot N} - \mathcal{P}(t_k)) r(t_k^-) &= (\mathbf{I}_{n\cdot N} - \mathcal{P}(t_k))(\mathcal{U}^\top \tilde{r}(t_k^-) + \sigma 1_{n\cdot N}) \\ &= (\mathbf{I}_{n\cdot N} - \mathcal{P}(t_k))\mathcal{U}^\top \tilde{r}(t_k^-) + \sigma 1_{n\cdot N} - \mathcal{P}(t_k)\sigma 1_{n\cdot N} \\ &= (\mathbf{I}_{n\cdot N} - \mathcal{P}(t_k))\mathcal{U}^\top \tilde{r}(t_k^-), \end{aligned}$$

Consequently, (2.24) rewrites for $t_k \in \mathcal{T}$ as

$$\begin{cases} \tilde{r}(t_k) = \mathcal{Q}(t_k)\tilde{r}(t_k^-) + \mathcal{Q}(t_k)\mathcal{U} e(t_k^-) \\ e(t_k) = (\mathbf{I}_{n\cdot N} - \mathcal{P}(t_k))\mathcal{U}^\top \tilde{r}(t_k^-) + (\mathbf{I}_{n\cdot N} - \mathcal{P}(t_k)) e(t_k^-). \end{cases} \tag{2.25}$$

Under Assumptions 2.2 and 2.3, the following quantities are well defined (*i.e.* the maximum exists):

$$\begin{aligned} \gamma_{11} &= \max_{t_k \in \mathcal{T}} \|\mathcal{Q}(t_k)\|, \\ \gamma_{12} &= \max_{t_k \in \mathcal{T}} \|\mathcal{Q}(t_k)\mathcal{U}\|, \\ \gamma_{21} &= \max_{t_k \in \mathcal{T}} \left\|(\mathbf{I}_{n\cdot N} - \mathcal{P}(t_k))\mathcal{U}^\top\right\|, \\ \gamma_{22} &= \max_{t_k \in \mathcal{T}} \|(\mathbf{I}_{n\cdot N} - \mathcal{P}(t_k))\|. \end{aligned} \tag{2.26}$$

Lemma 2.2. *Under Assumption 2.2, 2.3, $\forall\, t_k \in \mathcal{T}$, one has that:*

$$\begin{aligned} \|\tilde{r}(t_k)\| &\leq \gamma_{11}\|\tilde{r}(t_k^-)\| + \gamma_{12}\|e(t_k^-)\|, \\ \|e(t_k)\| &\leq \gamma_{21}\|\tilde{r}(t_k^-)\| + \gamma_{22}\|e(t_k^-)\|. \end{aligned} \tag{2.27}$$

Proof. The proof is straightforward applying the 2-norm to (2.25) and from the definition of $\gamma_{11}, \gamma_{12}, \gamma_{21}$ and γ_{22} in (2.26). ■

2.3.1 Stability analysis

In the sequel, we consider the matrices $\Gamma \in \mathbb{R}^{2\times 2}$ and $M : \mathbb{R}_+ \to \mathbb{R}^{2\times 2}$:

$$\Gamma = \begin{bmatrix} \gamma_{11} & \gamma_{12} \\ \gamma_{21} & \gamma_{22} \end{bmatrix}, \quad M(\tau) = \begin{bmatrix} 1 & 0 \\ 0 & c\, \mathrm{e}^{-\lambda\tau} \end{bmatrix}.$$

We emphasize that the behavior of $\|e(t)\|$ and $\|r(t)\|$ (thus $\|\tilde{r}(t)\|$) is characterized between two consecutive discrete time jumps $t_k,\ t_{k+1}$ by Corollary (2.1) in term of the matrix $M(t_{k+1} - t_k)$, namely

$$\begin{bmatrix} \|\tilde{r}(t_{k+1})\| \\ \|e(t_{k+1})\| \end{bmatrix} \leq M(t_{k+1} - t_k) \begin{bmatrix} \|\tilde{r}(t_k)\| \\ \|e(t_k)\| \end{bmatrix} = \begin{bmatrix} \|\tilde{r}(t_k)\| \\ c\,\|e(t_k)\|\, \mathrm{e}^{-\lambda(t_{k+1}-t_k)} \end{bmatrix}.$$

On the other hand, Lemma 2.2 gives an upper-bound, in term of Γ, on the jumps of $\|e(t)\|$ and $\|\tilde{r}(t)\|$ at time $t_k \in \mathcal{T}$.

Lemma 2.3. *Under Assumptions 2.2 and 2.3, let $\underline{\tau} \geq 0$ be such that the positive matrix $\Gamma M(\underline{\tau})$ is Schur. Then, for all sequences $\mathcal{T} = \{t_k\}_{k\geq 0}$ of jump times satisfying the dwell-time property $t_{k+1} - t_k \geq \underline{\tau}$, for all $k \in \mathbb{N}$, $\mathscr{X}'$ is globally attractive with respect to dynamics (2.18).*

Proof. From Corollary 2.1 and Lemma 2.2, it follows that for all $k \in \mathbb{N}$,

$$\begin{bmatrix} \|\tilde{r}(t_k)\| \\ \|e(t_k)\| \end{bmatrix} \leq \Gamma M(\tau_{k-1}) \ldots \Gamma M(\tau_0) \begin{bmatrix} \|\tilde{r}(t_0)\| \\ \|e(t_0)\| \end{bmatrix},$$

where $\tau_k = t_{k+1} - t_k,\ \forall k \in \mathbb{N}$. Remarking that the coefficients of the positive matrix $M(\tau)$ are non-increasing with respect to τ, it follows that

$$\begin{bmatrix} \|\tilde{r}(t_k)\| \\ \|e(t_k)\| \end{bmatrix} \leq \left(\Gamma M(\underline{\tau})\right)^k \begin{bmatrix} \|\tilde{r}(t_0)\| \\ \|e(t_0)\| \end{bmatrix}.$$

Hence, if the positive matrix $\Gamma M(\underline{\tau})$ is Schur, then both sequences $\{\|\tilde{r}(t_k)\|\}_{k\geq 0}$ and $\{\|e(t_k)\|\}_{k\geq 0}$ go to 0, and the system (2.18) converges to some point in $\mathscr{X}'$. ■

Given the previous result, the stability of $\mathscr{X}'$ with respect to the overall hybrid dynamics can be investigated by studying the spectral properties of the positive matrix $\Gamma M(\underline{\tau})$. Let us remark that values $\underline{\tau}$ such that $\Gamma M(\underline{\tau})$ is Schur provide lower bounds on the minimal dwell-time between two events that ensures the set $\mathscr{X}'$ to be globally attractive. In the following, we establish sufficient conditions for deriving such values $\underline{\tau}$.

At this point, it is interesting to emphasize that we have transformed the problem of stability analysis of the overall hybrid system in a problem of stabilization of a positive discrete-time system.

Theorem 2.1. *Considering Assumptions 2.2 and 2.3, a sufficient condition that guarantees the set $\mathscr{X}'$ to be a globally attractive set for the dynamics (2.17)-(2.18) $\forall i \in \mathcal{V}$ is that $\underline{\tau} > \underline{\tau}^*$ with*

$$\underline{\tau}^* = \frac{1}{\lambda} \max\left\{ \ln \frac{\gamma_{12} c}{1 - \gamma_{11}},\ \ln \frac{\gamma_{22} c}{1 - \gamma_{21}} \right\} > 0. \tag{2.28}$$

Proof. Let us remark that

$$\Gamma M(\underline{\tau}^*) = \begin{bmatrix} \gamma_{11} & \gamma_{12} c \mathrm{e}^{-\lambda \underline{\tau}} \\ \gamma_{21} & \gamma_{22} c \mathrm{e}^{-\lambda \underline{\tau}} \end{bmatrix}.$$

Moreover, the positive matrix $\Gamma M(\underline{\tau}^*)$ is Schur if and only if there exists $z \in \mathbb{R}^2_+$, such that $\Gamma M(\underline{\tau}^*)z < z$ (see e.g. [Rantzer, 2011]). Choosing $z = \mathbb{1}_2$, one obtains that $\Gamma M(\underline{\tau}^*)z < z$ is equivalent to

$$\begin{cases} \gamma_{11} + \gamma_{12} c \mathrm{e}^{-\lambda \underline{\tau}^*} < 1, \\ \gamma_{21} + \gamma_{22} c \mathrm{e}^{-\lambda \underline{\tau}^*} < 1. \end{cases} \tag{2.29}$$

The first inequality in (2.29) has the solution

$$\underline{\tau}^* > \frac{1}{\lambda} \ln \frac{\gamma_{12} c}{1 - \gamma_{11}},$$

while the second one is solved by

$$\underline{\tau}^* > \frac{1}{\lambda} \ln \frac{\gamma_{22} c}{1 - \gamma_{21}}.$$

Combining the two conditions above, one obtains the result in (2.28). For consistency of the previous result, we have to prove that either $\gamma_{11} < 1$ or $\gamma_{21} < 1$. Let us note that $Q(t_k) = UP(t_k)U^\top$ is a symmetric matrix since $P(t_k)$ is symmetric for undirected graphs. Therefore $\mathcal{Q}(t_k)$ is also symmetric with real eigenvalues. Furthermore, thanks to Assumption 2.3, the spectral radius $\rho(\mathcal{Q}(t_k)) < 1$, which means that $\|\mathcal{Q}(t_k)\| \leq \rho(\mathcal{Q}(t_k)) < 1$. This simply ensures that $\gamma_{11} < 1$. Although not needed, we can also prove that $\gamma_{21} < 1$. Recalling the choice of U, we can use the Kronecker product property introduced in Proposition A.1 and write $\|\mathcal{U}^\top\| = \|U^\top \otimes \mathbf{I}_n\| = \|U^\top\|\|\mathbf{I}_n\| = 1$.

Next we show that $\|\mathbf{I}_{n \cdot N} - \mathcal{P}(t_k)\| \leq 1$. First of all, thanks to Assumption 2.2 and to the stochasticity of $P(t_k)$, we know that $|\mathbf{I}_N - P(t_k)|_1 < 1$ and $|\mathbf{I}_N - P(t_k)| < 1$ (it will suffice to apply the definition of the 1-norm and ∞-norm as the maximum of respectively columns and rows sums). Then using the following matrix norm property (see Proposition A.3)

$$\|\mathbf{I}_N - P(t_k)\| \leq \sqrt{|\mathbf{I}_N - P(t_k)|_1 |\mathbf{I}_N - P(t_k)|} < 1.$$

The condition $\|\mathbf{I}_{n \cdot N} - \mathcal{P}(t_k)\| \leq 1$ follows once again from applying the Kronecker product property in Proposition A.1. ■

Remark 2.6. *The combination of Lemma 2.3 and Theorem 2.1 do not define the discrete time sequence $\mathcal{T}$ for the updates of the network but they define a centralized sufficient condition for $\mathcal{T}$, in terms of a dwell-time lower bound, that ensures the asymptotic stability of the interconnection.*

2.4 Directed time varying topology

This section collects some results inspired by the works in [Borzone et al., 2018a] and [Borzone et al., 2019]. The directed topology translates in the loss of symmetry of matrix $P(t_k)$ and consequently in the loss of some spectral properties that had been exploited in section 2.3. Moreover, the symmetry of $P(t_k)$ implies the synchronization of the interactions between pair of nodes (i.e. if j interacts with i, then, at the same time, i interacts with j), whereas in case of directional interactions the pair of nodes are more likely asynchronous. Finally, we should note that the dwell-time condition given in section 2.3 is referred to a sequence $\mathcal{T}$ which, from definition, collects the discrete time events of all the network and from this point of view, it corresponds to a degree of residual centralization of the control design (all the agents should agree over the last discrete time events occurred in the network). Our aim is rather to reformulate the dwell-time condition referred to sequence $\mathcal{T}_i$ which leads to the plain decentralization of the control policy (each agent decides autonomously when to execute an update of their own reference). This motivated the authors to use an approach mostly inspired by the analysis in [Bertsekas and Tsitsiklis, 1989] and afterwards used by several other authors (see for instance [Blondel et al., 2005] [Morărescu et al., 2015]).

First of all, Assumption 1.1 is reformulated to comply with our case of study.

Assumption 2.4. *Given a time varying graph $\mathcal{G}(t_k) = (\mathcal{V}, \mathcal{E}(t_k), P(t_k))$ there exist positive constants $a \in (0,1)$ and $b \in (\underline{b}, 1]$ such that Assumption 1.1 holds for $\underline{b} = \frac{nL-1}{nL}$, where L has been defined in Assumption 1.3.*

Moreover, we consider that Assumptions 1.2 and 1.3 hold for $t_k \in \mathcal{T}$.

Differently from what has been done in section 2.3, here we assume the existence of a lower and upper bounded dwell-time.

Assumption 2.5. *There exist positive constants $\underline{\tau}, \bar{\tau} > 0$ such that, for all $t_k \in \mathcal{T}_i$*

$$\underline{\tau} \leq t_k - \underline{t}_{k,i} \leq \bar{\tau}, \quad \forall i \in \mathcal{V}.$$

In other words, we assume the existence of a lower and an upper bound for the time elapsed between two consecutive updates of agent i.

We now introduce some quantities useful for the following analysis.

Definition 2.1. *We define the vectors which collect minimum and the maximum among the same components of vectors r_i as $m, M \in \mathbb{R}^n$*

$$m = (\min_{i\in\mathcal{V}} r_{i,(1)}, \cdots, \min_{i\in\mathcal{V}} r_{i,(n)})^\top, \quad M = (\max_{i\in\mathcal{V}} r_{i,(1)}, \cdots, \max_{i\in\mathcal{V}} r_{i,(n)})^\top.$$

The difference between these maximum and minimum vectors will be called **diameter** *and denoted by $\Delta = M - m$. Finally with $m_{(h)}$, $M_{(h)}$, and $\Delta_{(h)}$ we will denote the h^{th} component of the corresponding vector.*

The aim of the section is to analyze the asymptotic properties of the set

$$\mathscr{X}'' = \left\{ e, r \in \mathbb{R}^{n\cdot N} : e = 0,\ \Delta = 0 \right\}, \tag{2.30}$$

for the N interconnected hybrid systems (2.17)-(2.18). The set in (2.30) requires that each agent reaches its own reference ($e = 0$) and that all the references achieve consensus over all the n components ($\Delta = 0$).

In the following, in order to prove the convergence of the proposed strategy and to obtain a more compact formulation, we reduce the analysis of the interconnected hybrid dynamics in (2.17)-(2.18) to the analysis of a hybrid positive system with the following dynamics:

$$\dot{q}(t) = F(q(t)) \qquad t \in \mathbb{R}_+ \setminus \mathcal{T} \tag{2.31}$$

$$q(t_k) = G(q(t_k^-)) \qquad t_k \in \mathcal{T}, \tag{2.32}$$

where $q = (\Delta^\top, |e_1|, \dots, |e_N|)^\top \in \mathbb{R}^p_+$, $q(0^-) = q(0)$, $F : \mathbb{R}^p_+ \to \mathbb{R}^p_+$ and $G : \mathbb{R}^p_+ \to \mathbb{R}^p_+$ are continuous with $p = n + N$.

With this choice of state variable, the asymptotic properties of $\mathscr{X}''$ for dynamics (2.17)-(2.18) are equivalent to the ones of the origin $q = 0$ with respect to dynamics (2.31)-(2.32). In the following section, we will start studying the convergence of the discrete evolution of q. Precisely, we show that $q = 0$ is globally asymptotically stable for the discrete time dynamics (2.32) according to Definition 1.15. Then, we extend the result to the whole hybrid system (2.31)-(2.32).

2.4.1 Stability analysis of the jump dynamics

Thanks to Assumption 2.1, the agents converge toward their reference during the flow (i.e. $e_i \to 0$ for $t \in [t_k, t_{k+1})$ but the reference r_i remains constant. Hence we are interested in investigating the behavior of the discrete part of the hybrid interconnection (2.17)-(2.18), i.e.

$$r_i(t_k) = \sum_{j=1}^{N} P_{i,j}(t_k)\big(e_j(t_k^-) + r_j(t_k^-)\big), \tag{2.33}$$

$$e_i(t_k) = e_i(t_k^-) + r_i(t_k^-) - \sum_{j=1}^{N} P_{i,j}(t_k)\big(e_j(t_k^-) + r_j(t_k^-)\big), \tag{2.34}$$

for $t_k \in \mathcal{T}_i$. The aim of this section is to prove that after at most L updates of the values of the references diameters Δ and the errors norms $|e_i|$ contract.

We start with two instrumental results concerning the update of the reference (2.33). Recalling Definition 2.1, the result of the following Lemma shows that the update of the minimum m and of the maximum M of the references are respectively lower and upper bounded.

Lemma 2.4. *Under Assumptions 2.4, 1.2 and 1.3, for all $t_k \in \mathcal{T}$, the following holds*

$$m(t_k) \geq m(t_{k-1}) - \sum_{j=1}^{N} \left|e_j(t_k^-)\right| \mathbb{1}_n \tag{2.35}$$

and

$$M(t_k) \leq M(t_{k-1}) + \sum_{j=1}^{N} \left|e_j(t_k^-)\right| \mathbb{1}_n. \tag{2.36}$$

Proof. We start from (2.33)

$$\begin{aligned} r_i(t_k) &= \sum_{j=1}^{N} P_{i,j}(t_k)\big(r_j(t_k^-) + e_j(t_k^-)\big) \\ &\geq \sum_{j=1}^{N} P_{i,j}(t_k)\big(m(t_k^-) - \left|e_j(t_k^-)\right| \mathbb{1}_n\big) \\ &\geq m(t_k^-) - \sum_{j=1}^{N} \left|e_j(t_k^-)\right| \mathbb{1}_n, \end{aligned}$$

where, we first extracted the constant term $m(t_k^-)$ from the sum and then we exploited the stochasticity of the matrix $\sum_{j=1}^{N} P_{i,j}(t_k) = 1$. The references are constant between t_k and t_{k+1} and so is the minimum vector $m(t_k^-) = m(t_{k-1})$. We obtain for every agent i and every $t_k \in \mathcal{T}$

$$r_i(t_k) \geq m(t_{k-1}) - \sum_{j=1}^{N} \left|e_j(t_k^-)\right| \mathbb{1}_n.$$

Since the previous expression is true for all $r_i(t_k)$ component-wise it is true also for $m(t_k)$ which brings us to (2.35). Inequality (2.36) is proven by a similar argument involving the maximum M. ■

The next Lemma is essentially a generalization of the previous one, which concerns the bound over the reference jump of any agent i in the network.

Lemma 2.5. *Let Assumptions 2.4, 1.2 and 1.3 hold. Then for all $(\bar{\jmath}, \bar{\imath}) \in \mathcal{E}(t_k)$, $t_k \in \mathcal{T}_i$, the following is also true*

$$r_{\bar{\imath}}(t_k) \geq m(t_k^-) + a\left(r_{\bar{\jmath}}(t_k^-) - m(t_k^-)\right) - \sum_{j=1}^{N} \left|e_j(t_k^-)\right| \mathbb{1}_n \tag{2.37}$$

and

$$r_{\bar{\imath}}(t_k) \leq M(t_k^-) - a\left(M(t_k^-) - r_{\bar{\jmath}}(t_k^-)\right) + \sum_{j=1}^{N} \left|e_j(t_k^-)\right| \mathbb{1}_n. \tag{2.38}$$

Proof. Since $P(t_k)$ is stochastic, one has $\sum_{j=1, j\neq\bar{\jmath}}^{N} P_{\bar{\imath},j}(t_k) + P_{\bar{\imath},\bar{\jmath}}(t_k) = 1$ for all t_k, thus from (2.33) one obtains

$$r_{\bar{\imath}}(t_k) - m(t_k^-) \geq \sum_{j=1, j\neq\bar{\jmath}}^{N} P_{\bar{\imath},j}(t_k)\left(r_j(t_k^-) - m(t_k^-)\right) + P_{\bar{\imath},\bar{\jmath}}(t_k)\left(r_{\bar{\jmath}}(t_k^-) - m(t_k^-)\right) + \sum_{j=1}^{N} P_{\bar{\imath},j}(t_k) e_j(t_k^-)$$

From Assumption 2.4, $\sum_{j=1, j\neq\bar{\jmath}}^{N} P_{\bar{\imath},j}(t_k)\left(r_j(t_k^-) - m(t_k^-)\right)$ can be neglected because it is always a positive quantity and $P_{\bar{\imath},j}(t_k) > a$. Moreover, we note that thanks to the norm properties and to the stochasticity of $P(t_k)$, $P_{\bar{\imath},j}(t_k) e_j(t_k^-) \geq -P_{\bar{\imath},j}(t_k)\left|e_j(t_k^-)\right| \mathbb{1}_n \geq -\left|e_j(t_k^-)\right| \mathbb{1}_n$. Replacing these inequalities we obtain (2.37). Inequality (2.38) is proven by a similar argument involving the maximum M. ■

Now we recursively extend the result in Lemma 2.5 to the last L discrete time events of $\mathcal{T}$ (recalling that L represents the maximum number of updates that occur in the network between two activations of any $(j,i) \in \mathcal{E}$, see Assumption 1.3). The following result is inspired by Lemma 3.2 in [Bertsekas and Tsitsiklis, 1989].

Lemma 2.6. *Let Assumptions 2.4, 1.2 and 1.3 hold. Then, given $\bar{\imath} \in \mathcal{V}$, for all $i \in \mathcal{V}$, and for all $k > L, k \in \mathbb{N}$ we have*

$$r_i(t_k) \geq m(t_{k-L}) + a^L\big(r_{\bar{\imath}}(t_{k-L}) - m(t_{k-L})\big) - \sum_{j=1}^{N}\sum_{\ell=0}^{L-1}(2-a)a^\ell \left|e_j(t_{k-\ell}^-)\right| \mathbb{1}_n \tag{2.39}$$

and

$$r_i(t_k) \leq M(t_{k-L}) - a^L\big(M(t_{k-L}) - r_{\bar{\imath}}(t_{k-L})\big) + \sum_{j=1}^{N}\sum_{\ell=0}^{L-1}(2-a)a^\ell \left|e_j(t_{k-\ell}^-)\right| \mathbb{1}_n. \tag{2.40}$$

Proof. We focus on inequality (2.39). Thanks to Assumption 1.2 starting from $\bar{\imath}$ there exists a forward path to i, with at most L elements $\{(\bar{\imath}, i_{k-L+1}), \dots, (i_{k-1}, i)\}$. Using expression (2.37) in Lemma 2.5 for the current iteration in $t_k \in \mathcal{T}_i$ leads to

$$r_i(t_k) \geq m(t_{k-1}) + a\big(r_{i_{k-1}}(t_{k-1}) - m(t_{k-1})\big) - \sum_{j=1}^{N}\left|e_j(t_k^-)\right| \mathbb{1}_n,$$

where we used the fact that the reference is constant during the flow $r_{i_{k-1}}(t_k^-) = r_{i_{k-1}}(t_{k-1})$. Then we proceed backward recursively for $t_{k-1} \in \mathcal{T}_{i_{k-1}}$ and so on till $\bar{\imath}$ obtaining

$$r_i(t_k) \geq a^L r_{\bar{\imath}}(t_{k-L}) + \sum_{\ell=0}^{L-1}(1-a)a^\ell m(t_{k-\ell-1}) - \sum_{j=1}^{N}\sum_{\ell=0}^{L-1} a^\ell \left|e_j(t_{k-\ell}^-)\right| \mathbb{1}_n.$$

Thanks to Lemma 2.4 we can replace

$$m(t_{k-\ell-1}) \geq m(t_{k-L}) - \sum_{\ell=0}^{L-1}\sum_{j=1}^{N}\left|e_j(t_{k-\ell}^-)\right| \mathbb{1}_n,$$

since $\ell = \{0, \dots, L-1\}$. Finally exploiting the convergence of the sum $\sum_{\ell=0}^{L-1}(1-a)a^\ell = (1-a^L)$ we find inequality (2.39). As for inequality (2.40), it can be proven by a similar argument involving the maximum M. ■

Exploiting the exponential decreasing of the error in continuous time according to Assumption 2.1 and defining $c = \max_{i\in\mathcal{V}} c_i$ and $\lambda = \min_{i\in\mathcal{V}} \lambda_i$ we obtain the following result.

Proposition 2.1. *Under the additional Assumption 2.5, there exist $\eta_1, \eta_2 \in \mathbb{R}_+$ such that for all $k > 2L, k \in \mathbb{N}$ one has*

$$\Delta(t_k) \leq \eta_1 \Delta(t_{k-L}) + \sum_{j=1}^{N}\sum_{\ell=0}^{2L-1} \eta_2 c \mathrm{e}^{-\lambda\tau} \left|e_j(t_{k-\ell-1})\right| \mathbb{1}_n. \tag{2.41}$$

Moreover, possible values for η_1, η_2 are $\eta_1 = (1 - a^L)$, $\eta_2 = 2(2 - a)$.

Proof. To prove this result, we should reason component-wise for all $h = \{1, \dots, n\}$. Lemma 2.6 has been proven for all $i \in \mathcal{V}$, so it will hold also for $\mathbf{i} = \operatorname{argmin}_{i \in \mathcal{V}} r_{i,(h)}(t_k)$ and $\mathbf{j} = \operatorname{argmax}_{j \in \mathcal{V}} r_{j,(h)}(t_k)$, or equivalently for $m_{(h)}(t_k)$ and $M_{(h)}(t_k)$. Thus we can write

$$m_{(h)}(t_k) \geq m_{(h)}(t_{k-L}) + a^L \left(r_{\bar{\imath}}(t_{k-L}) - m_{(h)}(h)(t_{k-L}) \right) - \sum_{j=1}^{N} \sum_{\ell=0}^{L-1} (2-a) a^\ell \left| e_j(t^-_{k-\ell}) \right|$$

$$M_{(h)}(t_k) \leq M_{(h)}(t_{k-L}) - a^L \left(M_{(h)}(t_{k-L} - r_{\bar{\imath}}(t_{k-L})) \right) + \sum_{j=1}^{N} \sum_{\ell=0}^{L-1} (2-a) a^\ell \left| e_j(t^-_{k-\ell}) \right|.$$

We subtract the first inequality from the second one obtaining

$$\Delta_{(h)}(t_k) \leq (1 - a^L) \Delta_{(h)}(t_{k-L}) + \sum_{j=1}^{N} \sum_{\ell=0}^{L-1} 2(2-a) a^\ell \left| e_j(t^-_{k-\ell}) \right|. \tag{2.42}$$

Thanks to Assumptions 2.1 and 1.3 and 2.5 and considering c and λ which take care of heterogeneity we can write

$$a^\ell \left| e_j(t^-_{k-\ell}) \right| \leq c_j \left| e_j(\underline{t}_{k-\ell,j}) \right| \mathrm{e}^{-\lambda_j (t_{k-\ell} - \underline{t}_{k-\ell,j})} \leq \sum_{\bar{l}=0}^{L-1} c \left| e_j(t_{k-\ell-\bar{l}-1}) \right| \mathrm{e}^{-\lambda \underline{\tau}}. \tag{2.43}$$

Indeed we know that the previous update instant of j^{th} reference satisfies $\underline{t}_{k-\ell,j} \in \{t_{k-\ell-L}, \dots, t_{k-\ell-1}\}$. Hence, the value of the error is bounded by the sum of the errors at these instants $|e_j(\underline{t}_{k-\ell,j})| \leq \sum_{\bar{l}=0}^{L} |e_j(t_{k-\ell-\bar{l}-1})|$ (note that we also replace a^ℓ because $0 < a < 1$). We used Assumption 2.5 to bound the time elapsed between two consecutive updates of the agent j, that is $\underline{\tau} \leq t_{k-\ell} - \underline{t}_{k-\ell,j}$. Replacing inequality (2.43) in (2.42) and stacking the inequalities for all $h = \{1, \dots, n\}$ leads to (2.41) with $\eta_1 = (1 - a^L)$ and $\eta_2 = 2(2 - a)$. ■

As it has been done for the reference diameters we now analyze the update law of the error (2.34). The following Proposition characterizes the error norm evolution.

Proposition 2.2. *Let the additional Assumption 2.5 holds. Then there exists $\eta_3 \in \mathbb{R}_+$, such that for all $k > L, k \in \mathbb{N}$ one has*

$$|e_i(t_k)| \leq \sum_{\ell=0}^{L-1} c \mathrm{e}^{-\lambda \underline{\tau}} |e_i(t_{k-\ell-1})| + \sum_{j=1, j \neq i}^{N} \sum_{\ell=0}^{L-1} \eta_3 c \mathrm{e}^{-\lambda \underline{\tau}} |e_j(t_{k-\ell-1})| + \sum_{\ell=0}^{L-1} \eta_3 \mathbb{1}_n^\top \Delta(t_{k-\ell-1}) \tag{2.44}$$

for all $i \in \mathcal{V}$. Moreover a possible value is $\eta_3 = (1 - b)$.

Proof. We start taking $t_k \in \mathcal{T}_i$ and we decompose the update law (2.34) in

$$e_i(t_k) = (1 - P_{i,i}(t_k))\, r_i(t_k^-) - \sum_{j=1, j \neq i}^{N} P_{i,j}(t_k) r_j(t_k^-) + (1 - P_{i,i}(t_k))\, e_i(t_k^-) - \sum_{j=1, j \neq i}^{N} P_{i,j}(t_k) e_j(t_k^-).$$

We use the fact that thanks to Assumption 2.4 the quantity $\sum_{j=1,j\neq i}^{n} P_{i,j}(t_k)$ is positive and $P_{i,i}(t_k) \geq b$ together with the stochasticity of the matrix $P(t_k)$ to write

$$\begin{aligned} e_i(t_k) \leq & (1 - P_{i,i}(t_k))\, M(t_k^-) - (1 - P_{i,i}(t_k))\, m(t_k^-) + \\ & (1 - P_{i,i}(t_k))\, e_i(t_k^-) - \sum_{j=1,j\neq i}^{N} P_{i,j}(t_k) e_j(t_k^-) \\ \leq & (1-b)\Delta(t_k^-) + (1-b) e_i(t_k^-) - \sum_{j=1,j\neq i}^{N} P_{i,j}(t_k) e_j(t_k^-). \end{aligned}$$

Using the norm on the left and on the right and noticing that $|\Delta| \leq \sum_{h=1}^{n} \Delta_{(h)} \leq \mathbb{1}_n^\top \Delta$ ($\Delta_{(h)} \geq 0$ from definition), we obtain

$$|e_i(t_k)| \leq \; (1-b)\left|e_i(t_k^-)\right| + \sum_{j=1,j\neq i}^{N} (1-b)\left|e_j(t_k^-)\right| + (1-b)\mathbb{1}_n^\top \Delta(t_k^-).$$

Considering $t_k \notin \mathcal{T}_i$ instead leads to the trivial $|e_i(t_k)| = \left|e_i(t_k^-)\right|$, hence putting together the two conditions we can formalize a new inequality for all $t_k \in \mathcal{T}$ as

$$|e_i(t_k)| \leq \; \left|e_i(t_k^-)\right| + \sum_{j=1,j\neq i}^{N} (1-b)\left|e_j(t_k^-)\right| + (1-b)\mathbb{1}_n^\top \Delta(t_k^-). \tag{2.45}$$

Thanks to Assumption 2.1, 1.3 and 2.5 we can write

$$\left|e_j(t_k^-)\right| \leq c_j \left|e_j(\underline{t}_{k,j})\right| \mathrm{e}^{-\lambda_j (t_k - \underline{t}_{k,j})} \leq \sum_{\ell=0}^{L-1} c \left|e_j(t_{k-\ell-1})\right| \mathrm{e}^{-\lambda \tau}. \tag{2.46}$$

Indeed we know that $\underline{t}_{k,j} \in \{t_{k-L}, \dots, t_{k-1}\}$, hence the value of the error is bounded by the sum the errors at those instants $|e_j(\underline{t}_{k,j})| \leq \sum_{\ell=0}^{L-1} |e_j(t_{k-\ell-1})|$. In the same way we can bound $\Delta(t_k^-) \leq \Delta(\underline{t}_{k,j}) \leq \sum_{\ell=0}^{L-1} \Delta(t_{k-\ell-1})$ and replace the latter expression and (2.46) in inequality (2.45). We obtain condition (2.44) for $\eta_3 = (1-b)$. ■

Let us remark that until now, thanks to Proposition 2.1-2.2, we obtained n inequalities for the reference diameters and N inequalities for the error norms. To characterize the stability properties of the jumps dynamics we rewrite inequalities (2.41)-(2.44) in a more compact form.

In the sequel we will make use of the following matrices, whose values depend on c and η_1, η_2, η_3 introduced in Propositions 2.1-2.2. In particular we define $B_1, B_2, B_3, \Gamma_1, \Gamma_2 \in \mathbb{R}_+^{(n+N)\times(n+N)}$

$$\begin{aligned} B_1 = \begin{bmatrix} \eta_1 \mathbf{I}_n & 0 \\ 0 & 0 \end{bmatrix}, B_2 = \begin{bmatrix} 0 & 0 \\ 0 & c\mathbf{I}_N \end{bmatrix} B_3 = \begin{bmatrix} 0 & 0 \\ \eta_3 \mathbb{1}_{N\times n} & 0 \end{bmatrix}, \\ \Gamma_1 = \begin{bmatrix} 0 & \eta_2 c \mathbb{1}_{n\times N} \\ 0 & \eta_3 c \mathbf{J}_N \end{bmatrix}, \Gamma_2 = \begin{bmatrix} 0 & \eta_2 c \mathbb{1}_{n\times N} \\ 0 & 0 \end{bmatrix} \end{aligned} \tag{2.47}$$

where the matrix $\mathbf{J}_N = (\mathbb{1}_{N\times N} - \mathbf{I}_N)$. We use these blocks to condense the results expressed in Propositions 2.1-2.2 in the following lemma.

Lemma 2.7. *Under Assumptions 2.4, 1.2, 1.3 and 2.5, $\forall\, t_k \in \mathcal{T}, k > 2L$, one has that:*

$$\bar{q}(t_k) \leq (B + \Gamma \mathrm{e}^{-\lambda \underline{\tau}})\bar{q}(t_{k-1}), \tag{2.48}$$

where $B, \Gamma \in \mathbb{R}_+^{p\times p}$ and $\bar{q} \in \mathbb{R}_+^p$, $p = (N+n)\cdot 2L$, with structures

$$\bar{q}(t_k) = \begin{bmatrix} q(t_k) \\ \vdots \\ q(t_{k-L+1}) \\ \vdots \\ q(t_{k-2L+1}) \end{bmatrix}, \bar{q}(t_{k-1}) = \begin{bmatrix} q(t_{k-1}) \\ \vdots \\ q(t_{k-L}) \\ \vdots \\ q(t_{k-2L}) \end{bmatrix}, \Gamma = \begin{bmatrix} \overbrace{\Gamma_1 + B_2 & \dots & \Gamma_1 + B_2}^{L\cdot(n+N)} & \overbrace{\Gamma_2 & \dots & \Gamma_2}^{L\cdot(n+N)} \\ 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & \dots & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} \overbrace{B_3 & \dots & B_3 & B_3 + B_1}^{L\cdot(n+N)} & \overbrace{0 & 0}^{L\cdot(n+N)} \\ \mathbf{I}_{n+m} & \dots & \dots & 0 & 0 & 0 \\ \vdots & & & \vdots & \vdots & \vdots \\ 0 & \dots & \dots & \mathbf{I}_{n+m} & 0 & 0 \\ 0 & \dots & \dots & 0 & \mathbf{I}_{(L-1)(n+m)} & 0 \end{bmatrix}. \tag{2.49}$$

Proof. This Lemma reformulates in a more compact form the results in Proposition 2.1-2.2. The proof follows from the action of stacking inequality (2.41) and inequalities (2.44) $\forall i \in \mathcal{V}$. Collecting all the $|e_i(t_{k-\ell})|$ and $\Delta(t_{k-\ell})$ for all $\ell \in \{0, \dots, L-1\}$ in the vector q we obtain the upper part of the block matrices in (2.49). All the terms weighted by $\mathrm{e}^{-\lambda \underline{\tau}}$ are gathered in matrix Γ, whereas the others in B. For what concerns the lower part of the matrices in (2.49) the identities in matrix B represent the tautology for which each delayed value $q(t_{k-\ell})$ is equal to itself. ■

The following result formalizes the implication of the closed loops discrete dynamics (2.48) on the stability of the origin $q = 0$.

Theorem 2.2. *Under Assumptions 2.4, 1.2, 1.3 and 2.5, let $\underline{\tau}^* > 0$ such that the positive matrix $B + \Gamma \mathrm{e}^{-\lambda \underline{\tau}^*}$ is Schur. Then, for all sequences $\mathcal{T} = \{t_k\}_{k\geq 0}$ of discrete time events satisfying the condition $\underline{\tau} \geq \underline{\tau}^*$, $q = 0$ is globally exponentially stable with respect to the jump dynamics (2.32), according to Definition 1.15. Furthermore, there exists $\beta_d, K_d > 0$ such that*

$$|q(t_k)| \leq K_d \mathrm{e}^{-\beta_d k} |q(0)|, \quad \forall t_k \in \mathcal{T}. \tag{2.50}$$

Proof. Since $\underline{\tau} \geq \underline{\tau}^*$, if such a $\underline{\tau}^*$ exists and the matrix $B + \Gamma \mathrm{e}^{-\lambda \underline{\tau}^*}$ is Schur, then $\bar{q} = 0$ is globally asymptotically stable for the dynamics (2.48) and so it is $q = 0$. This implies the existence of class $\mathcal{KL}$ function $\tilde{\beta}$ such that the condition 1.13 is verified. In particular we could find $K_d,\ \beta_d > 0$ such that condition (2.50) is true (take for example $\mathrm{e}^{-\beta_d} = \rho(B + \Gamma \mathrm{e}^{-\lambda \underline{\tau}^*})$ where $\rho(A)$ is the spectral radius of matrix A). ■

Now it is our concern to discuss the existence and possibly provide a value for $\underline{\tau}^*$ which makes the matrix $B + \Gamma \mathrm{e}^{-\lambda \underline{\tau}^*}$ Schur as in the statement of Theorem 2.2.

Theorem 2.3. *Let Assumptions 2.4, 1.2, 1.3 and 2.5 hold. A sufficient condition that ensures the origin $q = 0$ is globally asymptotically stable for the jump dynamics (2.32) is that $\underline{\tau} > \underline{\tau}^*$ with*

$$\underline{\tau}^* = \frac{1}{\lambda} \ln \max \left\{ \frac{\eta_2 cN}{1 - \eta_1 \alpha^{-(L-1)}} \sum_{\ell=0}^{2L-1} \alpha^{-\ell}, \frac{c(\eta_3(N-1)+1)}{1 - \eta_3 nL\alpha^{-(L-1)}} \sum_{\ell=0}^{L-1} \alpha^{-\ell} \right\}, \tag{2.51}$$

and $\forall i \in \mathcal{V}$, with α such that $\max\left\{\eta_1^{\frac{1}{L-1}}, \eta_3 nL^{\frac{1}{L-1}}\right\} < \alpha < 1$ and $0 < \alpha < 1$ for $L = 1$.

Proof. We know from the real positive system theory (see e.g. [Rantzer, 2011]) that the positive matrix $B + \Gamma e^{-\lambda \underline{\tau}^*}$ is Schur if and only if there exists $z \in \mathbb{R}_+^p$, such that

$$Bz + \Gamma e^{-\lambda \underline{\tau}^*} z < z. \tag{2.52}$$

We start from the partial result of finding a $z = (z_0 \mathbb{1}_{n+N}^\top, z_1 \mathbb{1}_{n+N}^\top, \dots, z_{2L-1} \mathbb{1}_{n+N}^\top)^\top$ such that $Bz < z$. We choose the monotone increasing sequence $z_\ell = \alpha^{-\ell}, \ \ell = \{0, \dots, 2L-1\}$ with $0 < \alpha < 1$ (with this choice $z_0 = 1/\alpha^0 = 1$). Exploiting the block structure of matrix B in (2.49) we obtain the element-wise inequality

$$Bz = \begin{bmatrix} \eta_1 z_{L-1} \mathbb{1}_n \\ \eta_3 n \sum_{\ell=0}^{L-1} z_\ell \mathbb{1}_N \\ z_0 \mathbb{1}_{n+N} \\ \vdots \\ z_{2L-2} \mathbb{1}_{n+N} \end{bmatrix} < \begin{bmatrix} z_0 \mathbb{1}_n \\ z_0 \mathbb{1}_N \\ z_1 \mathbb{1}_{n+N} \\ \vdots \\ z_{2L-1} \mathbb{1}_{n+N} \end{bmatrix}.$$

The previous set of inequalities, is characterized by some relevant conditions that are repeated n, N or $n+N$ times. We therefore extract those relevant inequalities in

$$\begin{cases} \eta_1 z_{L-1} < z_0 \\ \eta_3 n \sum_{\ell=0}^{L-1} z_\ell < z_0 \\ z_0 < z_1 \\ \vdots \\ z_{2L-2} < z_{2L-1} \end{cases}.$$

Now it is possible to exploit the choice of the vector z previously described, replacing the values of $z_\ell, \forall \ell = \{0, \dots, 2L-1\}$ obtaining

$$\begin{cases} \eta_1 \alpha^{-(L-1)} < 1 \\ \eta_3 n \sum_{\ell=0}^{L-1} \alpha^{-\ell} < 1 \\ z_0 < z_1 \\ \vdots \\ z_{2L-2} < z_{2L-1} \end{cases} \Rightarrow \begin{cases} \eta_1 \alpha^{-(L-1)} < 1 \\ \eta_3 nL\alpha^{-(L-1)} < 1 \\ z_0 < z_1 \\ \vdots \\ z_{2L-2} < z_{2L-1} \end{cases},$$

where we bound the second inequality summing L times the maximum term of the sequence $\alpha^{-(L-1)}$.
While the rest of the inequalities are ensured by the fact that z_ℓ is monotone increasing, the first two inequalities are non trivial. From the first condition we obtain $\eta_1^{\frac{1}{L-1}} < \alpha < 1$, and from the second one we obtain $\eta_3 nL^{\frac{1}{L-1}} < \alpha < 1$. We can conclude that thanks

to Assumption 2.4 there always exists α such that $\max\{\eta_1^{\frac{1}{L-1}}, \eta_3 n L^{\frac{1}{L-1}}\} < \alpha < 1$. In the limit case for $L = 1$ the condition will be $0 < \alpha < 1$.
Then we define the vector $\varepsilon = z - Bz > 0$ and we rewrite inequality (2.52) as $\Gamma z e^{-\lambda \underline{\tau}^*} < \varepsilon$. It remains to solve this inequality component wise for all $(\Gamma z)_{(i)}$, in symbols

$$\underline{\tau}^* > \frac{1}{\lambda} \ln \max_i \frac{(\Gamma z)_{(i)}}{\varepsilon_{(i)}}. \tag{2.53}$$

In order to find the explicit value of $\underline{\tau}^*$ we look at the block structures in (2.47)-(2.49) and we compute Γz explicitly as

$$\Gamma z = \left[\eta_2 c N \sum_{\ell=0}^{2L-1} z_\ell \mathbb{1}_n^\top,\ c(\eta_3(N-1)+1) \sum_{\ell=0}^{L-1} z_\ell \mathbb{1}_N^\top,\ 0, \dots,\ 0 \right]^\top.$$

Since the last $(n+N)\times(2L-1)$ components of Γz are zeros, in order to apply inequality (2.53) we are just interested in the first $(n+N)$ components of ε (we denote the rest with the symbol$*$)

$$\varepsilon = z - Bz > \left[(1 - \eta_1 \alpha^{-(L-1)}) \mathbb{1}_n^\top, (1 - \eta_3 n L \alpha^{-(L-1)}) \mathbb{1}_N^\top, *, \dots, * \right]^\top.$$

Finally inequality (2.51) is obtained replacing in (2.53) the values of $\varepsilon_{(i)}$ and $(\Gamma z)_{(i)}$. ■

Remark 2.7. *In the directed case, the combination of Theorem 2.2 and Theorem 2.3 does not define the update sequences $\mathcal{T}_i$ for the agents. It rather defines a sufficient condition in terms of a dwell-time lower bound for the distributed choices of $\mathcal{T}_i$. This condition ensures the asymptotic stability of the interconnection. Note also that the existence of α is granted by the value of $\underline{b}$ in Assumption 2.4.*

2.4.2 Stability analysis of the overall hybrid dynamics

In this section, we complete the analysis of the asymptotic properties with a second main result which concerns the stability of the overall hybrid system (2.31)-(2.32). To do so, we first introduce the following Lemma that proves the exponential decreasing of the trajectories of q during the flow.

Lemma 2.8. *Let Assumptions 2.4, 1.2, 1.3 and 2.5 hold. There exist a $K_c > 0$ and an arbitrarily small $\beta_c > 0$, with $\beta_c \leq \lambda$, such that for all $t_k \in \mathcal{T}$*

$$|q(t)| \leq K_c e^{-\beta_c (t - t_k)} |q(t_k)|, \quad \forall t \in [t_k, t_{k+1}). \tag{2.54}$$

Moreover, inequality (2.54) holds for

$$K_c \geq e^{\beta_c \bar{\tau}} + c. \tag{2.55}$$

Proof. Recall that from its definition, $\Delta \geq 0$ always. Thus $|\Delta| \leq |\Delta|_1 = \sum_{h=1}^{n} \Delta_h = \mathbb{1}_n^\top \Delta$. Expanding inequality (2.54) on the right side we end up with

$$\begin{aligned} |q(t)| \leq & \mathbb{1}_n^\top \Delta(t) + \sum_{j=1}^{N} |e_j(t)| \\ \leq & \mathbb{1}_n^\top \Delta(t_k) + c\mathrm{e}^{-\lambda(t-t_k)} \sum_{j=1}^{N} |e_j(t_k)| \leq K_c \mathrm{e}^{-\beta_c(t-t_k)} \big(\mathbb{1}_n^\top \Delta(t_k) + \sum_{j=1}^{N} |e_j(t_k)|\big) \\ \leq & K_c \mathrm{e}^{-\beta_c(t-t_k)} |q(t_k)|, \end{aligned} \tag{2.56}$$

for some sufficiently large $K_c > 0$ and a sufficiently small $\beta_c > 0$. Note that, in (2.56) we also used the exponential convergence of the error (see Assumption 2.1). In order to obtain an expression for K_c, β_c we take the inner part of (2.56) and we divide by $\mathbb{1}_n^\top \Delta(t_k) + \sum_{j=1}^{N} |e_j(t_k)|$ obtaining inequality

$$\frac{\mathbb{1}_n^\top \Delta(t_k)}{\mathbb{1}_n^\top \Delta(t_k) + \sum_{j=1}^{N} |e_j(t_k)|} + c\mathrm{e}^{-\lambda(t-t_k)} \frac{\sum_{j=1}^{N} |e_j(t_k)|}{\mathbb{1}_n^\top \Delta(t_k) + \sum_{j=1}^{N} |e_j(t_k)|} \leq K_c \mathrm{e}^{-\beta_c(t-t_k)}. \tag{2.57}$$

Since the ratios are less then one we can infer that if $1 + c\mathrm{e}^{-\lambda(t-t_k)} \leq K_c \mathrm{e}^{-\beta_c(t-t_k)}$ it will comply with inequality (2.57) too. It follows the condition on K_c

$$K_c \geq \mathrm{e}^{\beta_c(t-t_k)} + c\mathrm{e}^{(\beta_c-\lambda)(t-t_k)}. \tag{2.58}$$

First, we impose $\beta_c \leq \lambda$ to let the second exponential be decreasing. Since we are dealing with the updates of the whole network, we cannot use the lower bound $\underline{\tau}$ in Assumption 2.5 to bound the value of the second exponential term in (2.58) (indeed the dwell-time concerns two consecutive updates of just one agent whereas, from a network point of view, two discrete time events may even happen one immediately after the other). On the other hand, the upper bound $\bar{\tau}$ for one agent is a good upper limit for the dwell-time of the network and consequently, to bound the second exponential term in (2.58). Hence we obtain expression in (2.55) as a possible value for K_c just imposing $0 < t - t_k \leq \bar{\tau}$. ■

Finally it is possible to complete the stability analysis for the hybrid system (2.31)-(2.32) using Definition 1.12. In order to do so it is necessary to treat the hybrid solutions of the systems defined over an hybrid time domain as in Definition 1.7 and 1.9. We will then use the map $q : \mathrm{dom}\ q \to \mathbb{R}^{n+N}$ to formulate the following result.

Theorem 2.4. *If the sufficient condition in Theorem 2.3 is met and for all sequences $\mathcal{T} = \{t_k\}_{k\geq 0}$ of jump times $\bar{\tau} \leq \bar{\tau}^* < \frac{\beta_d}{\beta_c}$, then the origin $q = 0$ is globally pre-asymptotically stable for system (2.31)-(2.32) according to Definition 1.12. Furthermore, there exist $\bar{K}, \bar{\beta} > 0$ such that for all $k \in \mathbb{N}$ and $t \in [t_k, t_{k+1})$*

$$|q(t,k)| \leq \bar{K} \mathrm{e}^{-\bar{\beta}(t+k)} |q(0,0)|. \tag{2.59}$$

Proof. We expand the previous results which were expressed in terms of $q : \mathbb{R}_+ \to \mathbb{R}^{n+N}$ using the hybrid solutions over the hybrid time domain $q : \mathrm{dom}\ q \to \mathbb{R}^{n+N}$ obtaining from Theorem 2.2

$$|q(t_k,k)| \leq K_d \mathrm{e}^{-\beta_d k} |q(0,0)|, \quad \forall k \in \mathbb{N}.$$

and from Lemma 2.8

$$|q(t,k)| \leq K_c \mathrm{e}^{-\beta_c(t-t_k)} |q(t_k,k)|, \quad \forall t \in [t_k, t_{k+1}), \forall k \in \mathbb{N}.$$

Then we compose the two results yielding, for all $k \in \mathbb{N}$ and $t \in [t_k, t_{k+1})$, to

$$|q(t,k)| \leq K_c \mathrm{e}^{-\beta_c(t-t_k)} K_d \mathrm{e}^{-\beta_d k} |q(0,0)| \leq K_c K_d \mathrm{e}^{-\beta_c t} \mathrm{e}^{\beta_c t_k - \beta_d k} |q(0,0)|.$$

The upper bound $\bar{\tau}$ can be used as an upper bound for the time elapsed between two consecutive discrete events of the whole network leading to the intuitive condition $t_k \leq k\bar{\tau}$. Using this inequality, it results

$$|q(t,k)| \leq K_c K_d \mathrm{e}^{-\beta_c t - (\beta_d - \beta_c \bar{\tau})k},$$

for all $k \in \mathbb{N}$, $t \in [t_k, t_{k+1})$. The previous expression corresponds to (2.59) for $\bar{\beta} = \min\{\beta_c, \beta_d - \beta_c\bar{\tau}\}$ and $\bar{K} = K_c K_d$. The sufficient condition over the maximum elapsed time comes directly from the fact that $\beta_d - \beta_c\bar{\tau} > 0$ to ensure the convergence. ■

2.5 Directed time varying topology with disturbance

The addition of sensor noise as exogenous inputs to the discrete dynamics of the interconnected systems (2.17)-(2.18) represents the natural extension of the analysis seen so far. In fact, if we consider that both the reconstruction of the state x_i and the relative state vector $\tilde{x}_i$ are affected by some kind of exogenous disturbance (which can be identified with measurement noise), the reference update dynamics (2.14) changes its form as it follows

$$r_i(t_k) = x_i(t_k^-) + \omega_i(t_k^-) + \sum_{\substack{j=1 \\ j\neq i}}^{N} P_{i,j}(t_k)(x_j(t_k^-) - x_i(t_k^-) + \omega_{ij}(t_k^-)), \tag{2.60}$$

where we used the bounded exogenous inputs $\omega_i,\ \omega_{ij} \in \mathbb{R}^n$, with $\left|\omega_i(t_k^-)\right| \leq \bar{\omega}$ and $\left|\omega_{ij}(t_k^-)\right| \leq \bar{\omega}$ for all $t_k \in \mathcal{T}_i$ and $i \in \mathcal{V}$. Since the inputs are additive one can rewrite the previous expression as

$$\begin{aligned} r_i(t_k) =& x_i(t_k^-) + \sum_{\substack{j=1 \\ j\neq i}}^{N} P_{i,j}(t_k)(x_j(t_k^-) - x_i(t_k^-)) + \omega_i(t_k^-) + \sum_{\substack{j=1 \\ j\neq i}}^{N} P_{i,j}(t_k)\omega_{ij}(t_k^-) \\ =& x_i(t_k^-) + \sum_{\substack{j=1 \\ j\neq i}}^{N} P_{i,j}(t_k)(x_j(t_k^-) - x_i(t_k^-)) + \delta_i(t_k^-), \end{aligned}$$

where the new exogenous input $\delta_i(t) = \omega_i(t) + \sum_{\substack{j=1 \\ j\neq i}}^{N} |P_{i,j}(t)|\, \omega_{ij}(t)$ has been considered.

Thanks to the stochasticity of $P(t_k)$ in Assumption 2.4, the new exogenous input δ_i is

such that

$$\left|\delta_i(t_k^-)\right| \leq \bar{\omega} + \sum_{\substack{j=1 \\ j\neq i}}^{N} |P_{i,j}(t_k)|\,\bar{\omega} \leq \bar{\omega} + (1-b)\bar{\omega} = (2-b)\bar{\omega} = \bar{\delta}, \quad \forall t_k \in \mathcal{T}_i. \tag{2.61}$$

Considering the new discrete input the hybrid interconnected systems (2.17)-(2.18) should change assuming the following form for all $i \in \mathcal{V}$.

$$\begin{cases} \dot{r}_i(t) = 0 \\ \dot{e}_i(t) = f_i\big(e_i(t)\big) \end{cases} \qquad \text{for } t \in \mathbb{R}_+ \setminus \mathcal{T}_i, \tag{2.62}$$

$$\begin{cases} r_i(t_k) = \displaystyle\sum_{j=1}^{N} P_{i,j}(t_k) e_j(t_k^-) + \sum_{j=1}^{N} P_{i,j}(t_k) r_j(t_k^-) + \delta_i(t_k^-) \\ e_i(t_k) = e_i(t_k^-) + r_i(t_k^-) - \displaystyle\sum_{j=1}^{N} P_{i,j}(t_k) e_j(t_k^-) - \sum_{j=1}^{N} P_{i,j}(t_k) r_j(t_k^-) - \delta_i(t_k^-) \end{cases} \qquad \text{for } t_k \in \mathcal{T}_i. \tag{2.63}$$

Our aim in this section is to analyze how the asymptotic properties of the closed loop interconnected systems (2.62)-(2.63) behave in the presence of the exogenous inputs δ_i. In order to do so, we will keep the same framework used for the directed network, therefore we will investigate the stability of the set $\mathscr{X}''$ in (2.30), for the N interconnected hybrid systems in presence of inputs δ_i.

Once again we reduce the analysis of the interconnected hybrid dynamics in (2.62)-(2.63) to a new hybrid positive system where this time in presence of the constant input $d(t_k^-) = \bar{\delta}$.

$$\dot{q}(t) = F(q(t)) \qquad t \in \mathbb{R}_+ \setminus \mathcal{T} \tag{2.64}$$

$$q(t_k) = \widetilde{G}(q(t_k^-), d(t_k^-)) \qquad t_k \in \mathcal{T}, \tag{2.65}$$

where $q = (\Delta^\top, |e_1|, \ldots, |e_N|)^\top \in \mathbb{R}^p_+$, $q(0^-) = q(0), d(0^-) = d(0)$, $F : \mathbb{R}^p_+ \to \mathbb{R}^p_+$ and $\tilde{G} : \mathbb{R}^p_+ \times \mathbb{R}_+ \to \mathbb{R}^p_+$ are continuous with $p = n + N$.

The asymptotic properties of $\mathscr{X}''$ for dynamics (2.62)-(2.63) are equivalent to the ones of the origin $q = 0$ with respect to dynamics (2.64)-(2.65). In the following section we show that the discrete time dynamics (2.65) is input-to-state stable ISS with reference to the input d, according to Definition 1.16. Then, similarly to the case without disturbance, we extend the result to the whole hybrid system.

2.5.1 Input-to-state stability of the jump dynamics

In this section, we focus our attention on the behavior of the discrete part of the hybrid system (2.62)- (2.63), i.e.

$$r_i(t_k) = \sum_{j=1}^{N} P_{i,j}(t_k)\big(e_j(t_k^-) + r_j(t_k^-)\big) + \delta_i(t_k^-) \tag{2.66}$$

$$e_i(t_k) = e_i(t_k^-) + r_i(t_k^-) - \sum_{j=1}^{N} P_{i,j}(t_k)\big(e_j(t_k^-) + r_j(t_k^-)\big) - \delta_i(t_k^-), \tag{2.67}$$

for $t_k \in \mathcal{T}_i$.

We start developing once again the same results obtained in Section 2.4 for the reference updates in directed networks, this time adapted to the case with exogenous input δ_i.

Lemma 2.9. *Under Assumptions 2.4, 1.2 and 1.3, for all $t_k \in \mathcal{T}$, the following holds*

$$m(t_k) \geq m(t_{k-1}) - \sum_{j=1}^{N} \left|e_j(t_k^-)\right| \mathbb{1}_n - \bar{\delta}\mathbb{1}_n \tag{2.68}$$

and

$$M(t_k) \leq M(t_{k-1}) + \sum_{j=1}^{N} \left|e_j(t_k^-)\right| \mathbb{1}_n + \bar{\delta}\mathbb{1}_n. \tag{2.69}$$

Proof. The proof is essentially equivalent to the Proof of Lemma 2.4 but this time we start from (2.66)

$$\begin{aligned} r_i(t_k) =& \sum_{j=1}^{N} P_{i,j}(t_k)\big(r_j(t_k^-) + e_j(t_k^-)\big) + \delta_i(t_k^-) \\ \geq& \sum_{j=1}^{N} P_{i,j}(t_k)\big(m(t_k^-) - \left|e_j(t_k^-)\right| \mathbb{1}_n\big) - \left|\delta_i(t_k^-)\right| \mathbb{1}_n \\ \geq& m(t_k^-) - \sum_{j=1}^{N} \left|e_j(t_k^-)\right| \mathbb{1}_n - \bar{\delta}\mathbb{1}_n. \end{aligned}$$

The references are constant between t_k and t_{k+1} and so is the minimum vector $m(t_k^-) = m(t_{k-1})$. We obtain for every agent i and every $t_k \in \mathcal{T}$

$$r_i(t_k) \geq m(t_{k-1}) - \sum_{j=1}^{N} \left|e_j(t_k^-)\right| \mathbb{1}_n - \bar{\delta}\mathbb{1}_n.$$

Since the previous expression is true for all $r_i(t_k)$ component-wise it is true also for $m(t_k)$ which brings us to (2.68). Inequality (2.69) is proven by a similar argument involving the maximum M. ■

Lemma 2.10. *Let Assumptions 2.4, 1.2 and 1.3 hold. Then for all $(\bar{\jmath}, \bar{\imath}) \in \mathcal{E}(t_k)$, $t_k \in \mathcal{T}_i$, the following is also true*

$$r_{\bar{\imath}}(t_k) \geq m(t_k^-) + a\left(r_{\bar{\jmath}}(t_k^-) - m(t_k^-)\right) - \sum_{j=1}^{N} \left|e_j(t_k^-)\right| \mathbb{1}_n - \bar{\delta}\mathbb{1}_n \tag{2.70}$$

and

$$r_{\bar{\imath}}(t_k) \leq M(t_k^-) - a\left(M(t_k^-) - r_{\bar{\jmath}}(t_k^-)\right) + \sum_{j=1}^{N} \left|e_j(t_k^-)\right| \mathbb{1}_n + \bar{\delta}\mathbb{1}_n. \tag{2.71}$$

Proof. As in the proof of Lemma 2.5, from the stochasticity of $P(t_k)$ we find

$$\begin{aligned} r_{\bar{\imath}}(t_k) - m(t_k^-) \geq & \sum_{j=1, j\neq\bar{\jmath}}^{N} P_{\bar{\imath},j}(t_k)\left(r_j(t_k^-) - m(t_k^-)\right) + P_{\bar{\imath},\bar{\jmath}}(t_k)\left(r_{\bar{\jmath}}(t_k^-) - m(t_k^-)\right) \\ & + \sum_{j=1}^{N} P_{\bar{\imath},j}(t_k) e_j(t_k^-) + \delta_{\bar{\imath}}(t_k^-) \\ \geq & a\left(r_{\bar{\jmath}}(t_k^-) - m(t_k^-)\right) - \sum_{j=1}^{N} \left|e_j(t_k^-)\right| \mathbb{1}_n + \delta_{\bar{\imath}}(t_k^-). \end{aligned}$$

As for $\delta_{\bar{\imath}}$ we know $\delta_{\bar{\imath}}(t_k^-) \geq -\left|\delta_{\bar{\imath}}(t_k^-)\right| \mathbb{1}_n \geq -\bar{\delta}\mathbb{1}_n$. Replacing the last inequality we obtain (2.70). Inequality (2.71) is is proven by a similar argument involving the maximum M. ■

Once again we recursively extend the result in Lemma 2.10 to the last L discrete time events of $\mathcal{T}$.

Lemma 2.11. *Let Assumptions 2.4, 1.2 and 1.3 hold. Then, given $\bar{\imath} \in \mathcal{V}$, for all $i \in \mathcal{V}$, and for all $k > L, k \in \mathbb{N}$ we have*

$$r_i(t_k) \geq m(t_{k-L}) + a^L\left(r_{\bar{\imath}}(t_{k-L}) - m(t_{k-L})\right) - \sum_{j=1}^{N}\sum_{\ell=0}^{L-1}(2-a)a^\ell \left|e_j(t_{k-\ell}^-)\right| \mathbb{1}_n - \sum_{\ell=0}^{L-1}(2-a)a^\ell\bar{\delta}\mathbb{1}_n \tag{2.72}$$

and

$$r_i(t_k) \leq M(t_{k-L}) - a^L\left(M(t_{k-L}) - r_{\bar{\imath}}(t_{k-L})\right) + \sum_{j=1}^{N}\sum_{\ell=0}^{L-1}(2-a)a^\ell \left|e_j(t_{k-\ell}^-)\right| \mathbb{1}_n + \sum_{\ell=0}^{L-1}(2-a)a^\ell\bar{\delta}\mathbb{1}_n. \tag{2.73}$$

Proof. We will consider the path $\{(\bar{\imath}, i_{k-L+1}), \ldots, (i_{k-1}, i)\}$ like it has been done in the proof of Lemma 2.6. Using inequality (2.70) for one iteration in Lemma 2.10 leads to

$$r_i(t_k) \geq m(t_{k-1}) + a\left(r_{i_{k-1}}(t_{k-1}) - m(t_{k-1})\right) - \sum_{j=1}^{N}\left|e_j(t_k^-)\right| \mathbb{1}_n - \left|\delta_i(t_k^-)\right| \mathbb{1}_n.$$

We proceed backward recursively for $t_{k-1} \in \mathcal{T}_{i_{k-1}}$ and so on till $\bar{\imath}$ obtaining

$$r_i(t_k) \geq a^L r_{\bar{\imath}}(t_{k-L}) + \sum_{\ell=0}^{L-1}(1-a)a^\ell m(t_{k-\ell-1}) - \sum_{j=1}^{N}\sum_{\ell=0}^{L-1} a^\ell \left|e_j(t_{k-\ell}^-)\right| \mathbb{1}_n - \sum_{\ell=0}^{L-1} a^\ell\bar{\delta}\mathbb{1}_n.$$

where we bounded the input thanks to the fact $\left|\delta_i(t_k^-)\right| \leq \bar{\delta}, \forall i \in \mathcal{V}$. Thanks to Lemma 2.9 we can replace

$$m(t_{k-\ell-1}) \geq m(t_{k-L}) - \sum_{\ell=0}^{L-1}\sum_{j=1}^{N}\left|e_j(t_{k-\ell}^-)\right| \mathbb{1}_n - \sum_{\ell=0}^{L-1}\bar{\delta}\mathbb{1}_n,$$

since $\ell = \{0, \ldots, L-1\}$. Finally exploiting the convergence of the sum $\sum_{\ell=0}^{L-1}(1-a)a^\ell = (1-a^L)$ we find inequality (2.72). As for inequality (2.73), it can be proven by a similar argument involving the maximum M. ■

Proposition 2.3. *Under the additional Assumption 2.5, there exist $\eta_1, \eta_2 \in \mathbb{R}_+$ such that for all $k > 2L, k \in \mathbb{N}$ one has*

$$\Delta(t_k) \leq \eta_1 \Delta(t_{k-L}) + \sum_{j=1}^{N} \sum_{\ell=0}^{2L-1} \eta_2 c \mathrm{e}^{-\lambda_\tau} \left| e_j(t_{k-\ell-1}) \right| \mathbb{1}_n + \sum_{\ell=0}^{L-1} \eta_2 a^\ell \bar{\delta} \mathbb{1}_n. \tag{2.74}$$

Moreover, possible values of η_1, η_2 are $\eta_1 = (1 - a^L)$, $\eta_2 = 2(2 - a)$.

Proof. The proof of this Proposition is equivalent to the one of Proposition 2.1, except for the presence of the term $\sum_{\ell=0}^{L-1}(2-a)a^\ell \bar{\delta} \mathbb{1}_n$ in expression of the lower and upper bound (2.72) and (2.73). This two terms sum up in the component-wise diameter computation obtaining:

$$\Delta_{(h)}(t_k) \leq (1 - a^L) \Delta_{(h)}(t_{k-L}) + \sum_{j=1}^{N} \sum_{\ell=0}^{L-1} 2(2-a) a^\ell \left| e_j(t_{k-\ell}^-) \right| \sum_{\ell=0}^{L-1} 2(2-a) a^\ell \bar{\delta}. \tag{2.75}$$

Inequality (2.74) follows from the same arguments. ■

We now take care of the new update law of the error (2.67) in presence of input δ_i. The following Proposition characterizes the error norm discrete time evolution.

Proposition 2.4. *Let the additional Assumption 2.5 holds. Then there exists $\eta_3 \in \mathbb{R}_+$, such that for all $k > L, k \in \mathbb{N}$ one has*

$$|e_i(t_k)| \leq \sum_{\ell=0}^{L-1} c \mathrm{e}^{-\lambda_\tau} |e_i(t_{k-\ell-1})| + \sum_{j=1, j\neq i}^{N} \sum_{\ell=0}^{L-1} \eta_3 c \mathrm{e}^{-\lambda_\tau} |e_j(t_{k-\ell-1})| + \sum_{\ell=0}^{L-1} \eta_3 \mathbb{1}_n^\top \Delta(t_{k-\ell-1}) + \bar{\delta} \tag{2.76}$$

for all $i \in \mathcal{V}$. Moreover a possible value of η_3 is $\eta_3 = (1 - b)$.

Proof. We decompose the error discrete update law (2.67) in

$$\begin{aligned}
e_i(t_k) = &(1 - P_{i,i}(t_k))\, r_i(t_k^-) - \sum_{j=1, j\neq i}^{N} P_{i,j}(t_k) r_j(t_k^-) + \\
&(1 - P_{i,i}(t_k))\, e_i(t_k^-) - \sum_{j=1, j\neq i}^{N} P_{i,j}(t_k) e_j(t_k^-) + \delta_i(t_k^-) \\
\leq &(1 - P_{i,i}(t_k))\, M(t_k^-) - (1 - P_{i,i}(t_k))\, m(t_k^-) + \\
&(1 - P_{i,i}(t_k))\, e_i(t_k^-) - \sum_{j=1, j\neq i}^{N} P_{i,j}(t_k) e_j(t_k^-) + \delta_i(t_k^-) \\
\leq &(1-b)\Delta(t_k^-) + (1-b) e_i(t_k^-) - \sum_{j=1, j\neq i}^{N} P_{i,j}(t_k) e_j(t_k^-) + \delta_i(t_k^-) \\
\leq &(1-b) \left| e_i(t_k^-) \right| + \sum_{j=1, j\neq i}^{N} (1-b) \left| e_j(t_k^-) \right| + (1-b) \mathbb{1}_n^\top \Delta(t_k^-) + \bar{\delta}.
\end{aligned}$$

The proof can be concluded by the same argument used in proof of Proposition 2.2, with the difference that now the constant input $\bar{\delta}$ appears in the inequality. ■

Thanks to Proposition 2.3-2.4, we obtained n inequalities for the reference diameters and N inequalities for the error norms, where the bound on the exogenous input $\bar{\delta}$ appears as constant term in all the inequalities. To characterize the stability properties of the jumps dynamics we rewrite inequalities (2.74)-(2.76) in a more compact form.

Lemma 2.12. *Under Assumptions 2.4, 1.2, 1.3 and 2.5, $\forall\, t_k \in \mathcal{T}, k > 2L$, one has that:*

$$\bar{q}(t_k) \leq (B + \Gamma \mathrm{e}^{-\lambda_{\underline{\tau}}})\bar{q}(t_{k-1}) + Dd(t_{k-1}), \tag{2.77}$$

where $B, \Gamma \in \mathbb{R}_+^{p\times p}$ and $\bar{q} \in \mathbb{R}_+^p$, $p = (N+n)\cdot 2L$ have been defined in Lemma 2.7. Furthermore the constant input $d(t_k) = \bar{\delta},\ \forall t_k \in \mathcal{T}$ id weighted by the input matrix $D \in \mathbb{R}_+^p$

$$D = \left[\eta_2 \sum_{\ell=0}^{L-1} a^\ell \mathbb{1}_n^\top, \mathbb{1}_N^\top, 0, \ldots, 0\right]^\top. \tag{2.78}$$

Proof. The Lemma reformulates in a more compact form the results in Proposition 2.3-2.4. The proof follows from the action of stacking inequality (2.74) and inequalities (2.76) $\forall i \in \mathcal{V}$ as it has been done in Lemma 2.7. ■

In order to conclude about the input-to-state stability of the jump dynamics (2.65) we recall that the result of Theorem 2.3 is still valid in the current framework. In other words if we are capable of finding a valid $\underline{\tau}^*$ such that the zero input jump dynamics is globally asymptotically stable as proven in Theorem 2.2, then, thanks to the boundedness of both the input and the norm of the D, we can conclude about the input-to-state stability. We formalize these considerations in the following Theorem.

Theorem 2.5. *Suppose to have chosen a value of $\underline{\tau}^* > 0$ which satisfies Theorem 2.3, namely a $\underline{\tau}^*$ such that the positive matrix $B + \Gamma \mathrm{e}^{-\lambda_{\underline{\tau}^*}}$ is Schur. Then, the jump dynamics (2.32), is input-to-state stable with reference to the origin $q = 0$ and input d according to Definition 1.16. Furthermore, there exist $K_d, \beta_d, \gamma > 0$ such that*

$$|q(t_k)| \leq K_d \mathrm{e}^{-\beta_d k} |q(0)| + \gamma |d|_{\{0,\ldots,t_k\}}, \quad \forall t_k \in \mathcal{T}. \tag{2.79}$$

Proof. If the matrix $B+\Gamma \mathrm{e}^{-\lambda_{\underline{\tau}^*}}$ is Schur, then Theorem 2.2 is valid and the zero input dynamics (2.65) is globally asymptotically stable. This implies the existence of K_d and β_d such that $|q(t_k)| \leq K_d \mathrm{e}^{-\beta_d k} |q(0)|, \quad \forall t_k \in \mathcal{T}$. Since the input is bounded $\forall t_k$, $|d|_{\{0,\ldots,t_k\}} \leq \bar{\delta}$. Furthermore the infinite norm of matrix is always finite $|D| = \eta_2 \sum_{\ell=0}^{L-1} a^\ell = \eta_2 \frac{1-a^L}{1-a}$, so we can conclude about the existence of a finite real $\gamma > 0$ such that $|D| \leq \gamma$. The previous elements allow us to conclude about the input-to-state stability of the jump dynamics. ■

2.5.2 Input-to-state stability of the overall hybrid dynamics

Now it is our concern to complete the analysis of the asymptotic properties of the overall hybrid system (2.31)-(2.32) in presence of exogenous input. First of all it is important to note that, since the input affects only the jump dynamics, the behaviour of the system during the flow continues to be perfectly compliant to the result of Lemma

2.8. Hence, the robustness analysis for the hybrid system (2.64)-(2.65) can be completed using Definition 1.13. In order to do so it is necessary to treat the hybrid solutions of the systems defined over an hybrid time domain as in Definitions 1.7 and 1.9. We will then use once again the map $q : \text{dom } q \to \mathbb{R}^{n+N}$ to formulate the following result.

Theorem 2.6. *If the sufficient condition in Theorem 2.3 is met and for all sequences $\mathcal{T} = \{t_k\}_{k\geq 0}$ of jump times $\bar{\tau} \leq \bar{\tau}^* < \frac{\beta_d}{\beta_c}$, then the the hybrid system (2.31)-(2.32) is pre-input-to-state stable with respect to the origin $q = 0$ and the exogenous input d according to Definition 1.12. Furthermore, there exist $\bar{K},\ \bar{\beta},\ \bar{\gamma}$ such that for all $k \in \mathbb{N}$ and $t \in [t_k, t_{k+1})$*

$$|q(t,k)| \leq \bar{K}\mathrm{e}^{-\bar{\beta}(t+k)} |q(0,0)| + \bar{\gamma} |d|_{\{0,\dots,t_k\}} . \tag{2.80}$$

Proof. We expand inequalities (2.79), (2.54) which were expressed in terms of $q : \mathbb{R}_+ \to \mathbb{R}^{n+N}$ using the hybrid solutions over the hybrid time domain $q : \text{dom } q \to \mathbb{R}^{n+N}$ obtaining for Theorem 2.5

$$|q(t_k,k)| \leq K_d\mathrm{e}^{-\beta_d k} |q(0,0)| + \gamma |d|_{\{0,\dots,t_k\}} , \quad \forall k \in \mathbb{N}.$$

and for Lemma 2.8

$$|q(t,k)| \leq K_c\mathrm{e}^{-\beta_c(t-t_k)} |q(t_k,k)| , \quad \forall t \in [t_k, t_{k+1}), \forall k \in \mathbb{N}.$$

Then we compose the two results obtaining for all $k \in \mathbb{N}$ and $t \in [t_k, t_{k+1})$

$$\begin{aligned} |q(t,k)| \leq & K_cK_d\mathrm{e}^{-\beta_c(t-t_k)}\mathrm{e}^{-\beta_d k} |q(0,0)| + K_c\gamma |d|_{\{0,\dots,t_k\}} \\ \leq & K_cK_d\mathrm{e}^{-\beta_c t}\mathrm{e}^{\beta_c t_k - \beta_d k} |q(0,0)| + K_c\gamma |d|_{\{0,\dots,t_k\}} . \end{aligned}$$

The upper bound $\bar{\tau}$ can be used as an upper bound for the time elapsed between two consecutive discrete events of the whole network leading to the intuitive condition $t_k \leq k\bar{\tau}$. Using this inequality we obtain

$$|q(t,k)| \leq K_cK_d\mathrm{e}^{-\beta_c t - (\beta_d - \beta_c\bar{\tau})k} + K_c\gamma |d|_{\{0,\dots,t_k\}} ,$$

for all $k \in \mathbb{N},\ t \in [t_k, t_{k+1})$. The previous expression corresponds to (2.80) for $\bar{\beta} = \min\{\beta_c, \beta_d - \beta_c\bar{\tau}\}$, $\bar{K} = K_cK_d$ and $\bar{\gamma} = K_c\gamma$. The sufficient condition over the maximum elapsed time comes directly from the fact that, in order to ensure the convergence, the inequality $\beta_d - \beta_c\bar{\tau} > 0$ holds. ■

2.6 Application example

In order to justify the results discussed so far, in this section we propose a simple example which represents a useful framework of application for the stability sufficient condition proposed in the chapter. The scenario is composed by two agents with linear dynamics which are required to track their own reference. The references are chosen using a consensus discrete algorithm as the one described in Section 2.2.1.

The linear dynamics of each agent is described by the following system

$$\dot{x}_i(t) = A_i x_i(t) + B_i u_i(t), \tag{2.81}$$

with $x_i \in \mathbb{R}^n$, $u_i \in \mathbb{R}^s$, $A_i \in \mathbb{R}^{n\times n}$, and $B_i = \mathbf{I}_n$.

The objective is to make the state x_i track the reference $r_i \in \mathbb{R}^n$. Using the tracking error signal $e_i = x_i - r_i$ we can think to apply, for instance, the following feedback

$$u_i(t) = K_{1,i}x_i(t) + K_{2,i}e_i(t), \tag{2.82}$$

with an appropriate choice of matrices $K_{1,i},\ K_{2,i} \in \mathbb{R}^{n\times n}$.

Choosing $K_{1,i} = -A_i$, the closed loop dynamics of the tracking error becomes

$$\dot{e}_i(t) = A_i x_i(t) + \mathbf{I}_n\big(-A_i x_i(t) + K_{2,i}e_i(t)\big) = K_{2,i}e_i(t), \tag{2.83}$$

which complies with Assumption 2.1, if the matrix $K_{2,i}$ is Hurwitz. In such case the exponential convergence of the tracking error is also granted.

As already introduced, we consider a network of two agents, with state $x_1, x_2 \in \mathbb{R}^3$ and dynamics

$$\dot{x}_1 = A_1x_1 + B_1u_1, \qquad \dot{x}_2 = A_2x_2 + B_2u_2.$$

with $A_1,\ A_2 \in \mathbb{R}^{3\times 3}$, inputs $u_1,\ u_2 \in \mathbb{R}^3$ and $B_1 = B_2 = \mathbf{I}_3$.

The agents interact with a discrete time consensus algorithm in order to reach an agreement on the value of the first reference component $r_{i,(1)}$ as described in 2.2.1. We consider that the other reference components are kept constant and equal to zero $r_{i,(2)} \equiv 0$ and $r_{i,(3)} \equiv 0$. The interaction is considered undirected.

Suppose that, after applying the steps in (2.82) and (2.83), we take $K_{2,1}, K_{2,2} \in \mathbb{R}^{3\times 3}$ such that the tracking error closed loop dynamics becomes

$$\dot{e}_1 = \begin{bmatrix} 0 & 1.02 & -1.02 \\ 1 & -3.07 & -3.055 \\ 0 & 2 & 0 \end{bmatrix} e_1, \qquad \dot{e}_2 = \begin{bmatrix} 0 & 2.51 & -1.255 \\ 1 & -1.57 & -3.34 \\ 0 & 2 & 0 \end{bmatrix} e_2.$$

for $t \in \mathbb{R}_+ \setminus \mathcal{T}_1$ and $t \in \mathbb{R}_+ \setminus \mathcal{T}_2$ respectively.

The resulting hybrid dynamics which characterizes agent 1 is represented by the following set of equations

$$\begin{cases} \dot{r}_{1,(1)}(t) &= 0 \\ \dot{r}_{1,(2)}(t) &= 0 \\ \dot{r}_{1,(3)}(t) &= 0 \\ \dot{e}_1(t) &= B_1K_{1,1}e_1(t) \end{cases} \quad \text{for } t \in \mathbb{R}_+ \setminus \mathcal{T}_1, \tag{2.84}$$

$$\begin{cases} r_{1,(1)}(t_k) &= P_{1,2}(t_k)e_{2,(1)}(t_k^-) + P_{1,2}(t_k)r_{2,(1)}(t_k^-) \\ r_{1,(2)}(t_k) &= r_{1,(2)}(t_k^-) \\ r_{1,(3)}(t_k) &= r_{1,(3)}(t_k^-) \\ e_{1,(1)}(t_k) &= (1 - P_{1,2}(t_k))e_{2,(1)}(t_k^-) - (1 - P_{1,2}(t_k))r_{2,(1)}(t_k^-) \\ e_{1,(2)}(t_k) &= e_{1,(2)}(t_k^-) \\ e_{1,(3)}(t_k) &= e_{1,(3)}(t_k^-) \end{cases} \quad \text{for } t_k \in \mathcal{T}_1, \tag{2.85}$$

with initial conditions $r_{i,(1)}(0) = r_0$, $r_{i,(2)}(0) = r_{i,(3)}(0) = 0$ and $e_i(0) = 0$.

The interaction between the couple of agents is described by the symmetric stochastic weight matrix

$$P(t_k) = \begin{bmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{bmatrix}.$$

Note that the choice of the previous weight matrix implies $\mathcal{T}_1 = \mathcal{T}_2 = \mathcal{T}$.

The exponential convergence constants are evaluated for the closed loop dynamics of the two agents obtaining $c_1 = 5.3$, $\lambda_1 = 0.5$ and $c_2 = 8.3$, $\lambda_2 = 0.4$. From the convergence of the tracking errors e_1 and e_2 we can infer the convergence of the error first components $e_{1,(1)}$ and $e_{2,(1)}$:

$$|e_i(t)| \leq c_i \, |e_i(\underline{t}_{k,i})| \, \mathrm{e}^{-\lambda_i (t - \underline{t}_{k,i})} \implies \left|e_{i,(1)}(t)\right| \leq c_i \left|e_{i,(1)}(\underline{t}_{k,i})\right| \mathrm{e}^{-\lambda_i (t - \underline{t}_{k,i})}, \text{for } t \geq \underline{t}_{k,i}.$$

Since the network is described by an undirected graph it is possible to apply Theorem 2.1 in order to find a suitable lower bound on the dwell time between two consecutive reset, that ensures the stability of the two interconnected system (2.84)-(2.85). From the formula in (2.28) we obtain $\underline{\tau}^* = 6.4$ s. In Figure 2.4 the convergence of the overall dynamics is shown. The discrete time events sequence $\mathcal{T}$ has been chosen such that $\forall t_k \in \mathcal{T}$, $\underline{\tau}^* < t_{k+1} - t_k \leq \bar{\tau}$ where $\bar{\tau}$ has been arbitrarily taken as 8.0 s.

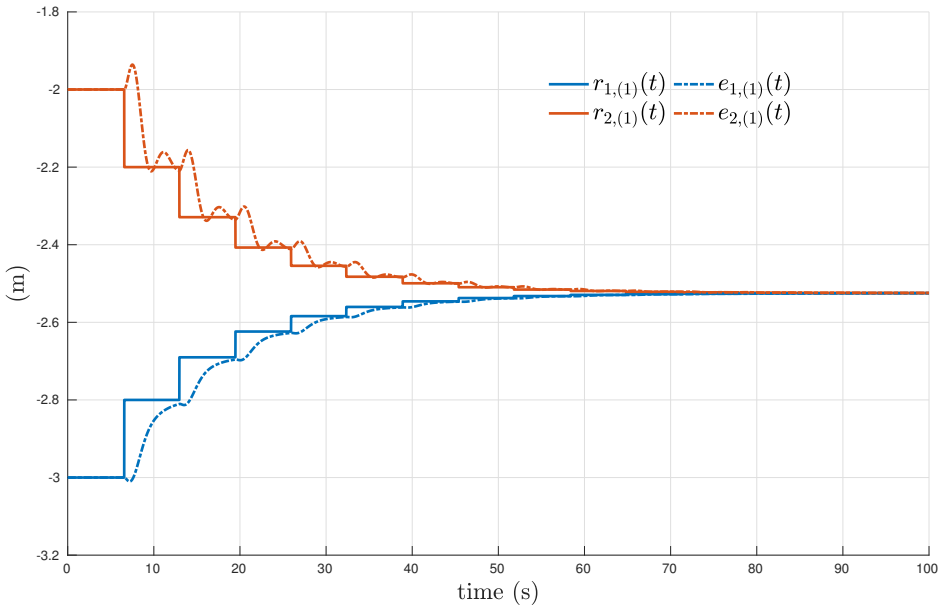


Figure 2.4: The trajectories of the reference and the tracking error for the two agents network with $\underline{\tau}^* = 6.4$ s $< t_{k+1} - t_k \leq 8.0$ s.

Note that this choice of dwell-time corresponds to a sufficient condition for the convergence. In other words, for a sequence $\mathcal{T}'$ which does not comply with Theorem 2.1 it may happens that the overall dynamics diverge. This is the case, for instance, in the scenario shown in Figure 2.5, where the dwell-time lower bound was set to $\underline{\tau}^* = 1.4$ s.

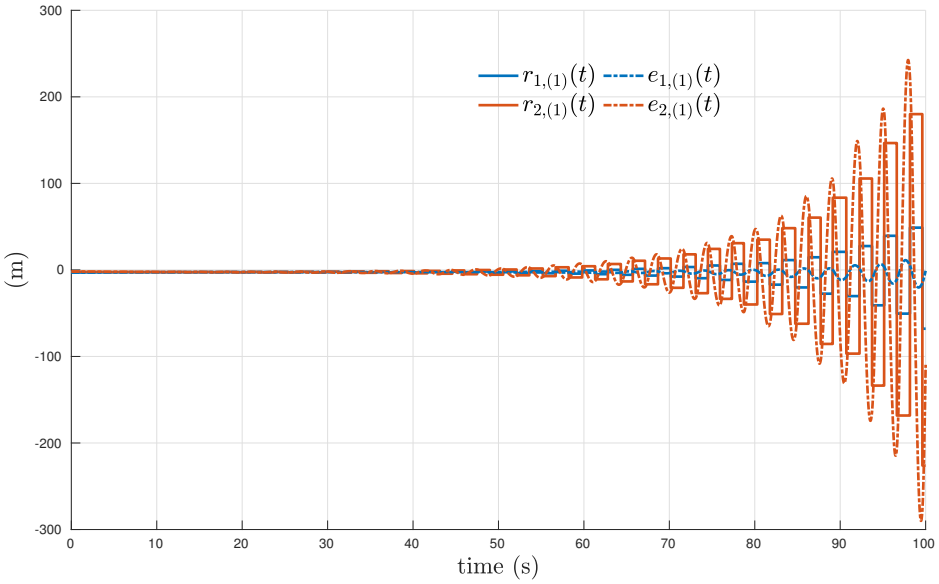


Figure 2.5: The trajectories of the reference and the tracking error for the two agents network with $\underline{\tau}^* = 1.4$ s $< t_{k+1} - t_k \leq 3.0$ s.

2.7 Conclusions

This Chapter presented an hybrid framework for the synthesis and the stability and robustness analysis of a two stage agreement solution for multi-agent systems. The proposed decentralized control strategy is applied to a network of nonlinear, heterogeneous and exponentially convergent tracking subsystems. The interactions between agents are modeled with specific features: first of all they are sporadic, hence the updates happen following a sequence of discrete time events. Secondly they are sensor driven interactions, avoiding to consider package loss or disturbances of standard communication infrastructure. Indeed, when an interaction occurs, an agent updates the reference to track according to its neighbors output, based on some relative inter-distance measurements. The resulting closed-loop dynamics is represented as an interconnection of hybrid subsystems.

Two different time varying topology have been considered. In Section 2.4 a sufficient condition for the convergence of the algorithm is formulated in term of a minimum dwell-time condition for network described by undirected graphs. Later in section 2.4 an other sufficient condition is presented in term of a minimum and maximum dwell-time condition, for network described by directed graphs.

Once the convergence has been showed in both cases, in Section 2.5 we analyzed the robustness of the proposed strategy for directed graph network, with respect to the measurement noise that affects both their own output and the relative output sensors.

Chapter 3

Consensus in networks of non-holonomics robots

Decentralized control of multi-robot systems has attracted a lot of interest in recent years since it is typically a scalable and versatile strategy that applies to a large number of applications such as search and rescue missions, ground coverage, object transport among others.

In this field a consensus process is usually meant to reach an agreement on some set of spacial coordinates in order to achieve a Rendez-Vous in some points, to maintain the connectivity of the network, to realize a formation among other tasks ([Cortés, 2006, Bullo et al., 2009, Beard and Stepanyan, 2003]). However many results in literature cover the simplified single or double integrator vehicle dynamics, hence they cannot be rigorously applied to multi-agent systems with non-holonomics constraints.

In Section 1.5.1, we introduced the Brockett necessary condition which represents the main issue in dealing with this kind of systems. To address this problem in the consensus-like control objectives, several different approaches have been proposed so far; in [Dimarogonas and Kyriakopoulos, 2007] a discontinuous and time invariant control is proposed to achieve the consensus and solve the rendez-vous problem for an undirected graph network of multiple unicycle robots. The same context has been analyzed in [Dong and Farrell, 2008] for network links characterized by bounded delay. The problem of the leader-less formation realization for networks of non-holonomic robots has been addressed in [Peng et al., 2015], [Maghenem et al., 2019]. In [Ajorlou et al., 2015] the case of input bounded disturbances is taken into account and in [Defoort et al., 2008] a sliding mode control solution is proposed to address the robustness issues with respect to unicycle modelling errors. The previous works on formation realization have in common to consider a continuous time coupling between the agents in the networks and to consider statics undirected or directed graph interactions. The time varying network topology case has been taken into account for the rendez-vous and the formation realization problem by the works in [Liu et al., 2013] and [Léchevin et al., 2006] respectively. Nevertheless, even in those cases the authors consider a continuous time interactions between the agents.

In this Chapter we aim to present the rendez-vous and the formation realization problem for fleets of non-holonomic vehicles as a framework of application of the theoretic results described in Chapter 2. The stabilization controller used to drive the

vehicles with unicycle dynamics to their reference is described in Section 3.1. Afterward we take care of verifying that the overall system obtained using the two stage approach complies with the the exponential convergence Assumption 2.1 and in Section 3.2 the hybrid dynamics which characterizes the agents is introduced. Finally in Section 3.3 we give some numerical examples applying the stability results to the formation realization problem of a fleet of unicycle robot, whose interactions are modeled by undirected switching and directed switching topology.

3.1 Point stabilization for unicycle dynamics

We consider now a network of N unicycles and we assume that a 2D reference position, denoted by $r_i = (r_{i,x}, r_{i,y})^\top$, is provided to each vehicle. Since the problem under study does not require to obtain a specific final orientation of the robots we also assume that the heading reference, denoted by $r_{i,\theta}$ is set to 0 for all robots. The generalized coordinates of the center of mass of each vehicle with respect to the fixed frame are denoted using the same representation as the unicycle Example 1.1, $(X_i, \theta_i)^\top \in \mathbb{R}^3$ with $X_i = (x_i, y_i)^\top$.

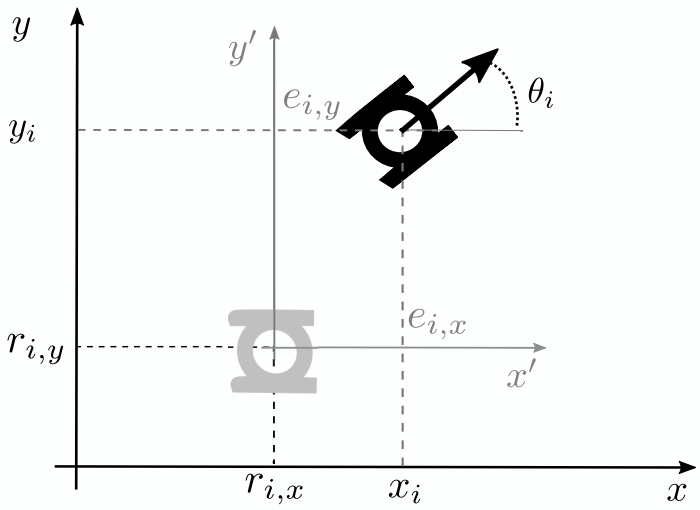


Figure 3.1: The error coordinate representation. The fixed $x - y$ reference frame is shown in black, whereas the reference frame $x' - y'$ is represented in gray.

To address our stabilization problem we choose a set of error coordinates $\xi_i = (e_i, e_{i,\theta})^\top \in \mathbb{R}^3$ where $e_i = (e_{i,x}, e_{i,y})^\top \in \mathbb{R}^2$ is the Cartesian 2D positioning error. As it is clear from the representation in Figure 3.1 the coordinates transformation (2.2) is still valid, namely

$$\begin{aligned} e_i &= X_i - r_i \\ e_{i,\theta} &= \theta_i - r_{i,\theta} = \theta_i. \end{aligned} \tag{3.1}$$

The non-holonomic dynamics of the unicycle in (1.19) can be rewritten in the more

compact form

$$\dot{\xi}_i = b(\xi_i)u_i, \quad b(\xi_i) = \begin{bmatrix} \cos e_{i,\theta} & 0 \\ \sin e_{i,\theta} & 0 \\ 0 & 1 \end{bmatrix}, \quad u_i = \begin{bmatrix} v_i \\ w_i \end{bmatrix}. \tag{3.2}$$

Thanks to this choice of ξ_i the stabilization problem to any reference position r_i can be reduced to the problem of the stabilization to the set $\mathscr{O} = \{\xi_i : \xi_i = (0, 0, 2\pi n)^\top\}$ for $n \in \mathbb{Z}$.

In this work we used a piecewise smooth control law, first proposed by [Canudas de Wit and Sordalen, 1992], which is based on a state coordinate transformation. Consider the following circle family $\mathscr{P}$ (see Figure 3.2)

$$\mathscr{P} = \left\{e_i : e_{i,x}^2 + (e_{i,y} - R_i)^2 = R_i^2\right\} \tag{3.3}$$

as the set of circles with radius R_i. They pass through r_i and e_i and are centred on the y'-axis of the new reference frame. Let $\theta_{i,d}$ be the angle of the tangent of $\mathscr{P}$ at e_i defined as

$$\theta_{i,d} = \begin{cases} 2\arctan\dfrac{e_{i,y}}{e_{i,x}} & e_i \neq 0 \\ 0 & e_i = 0, \end{cases} \tag{3.4}$$

where $\theta_{i,d}$, by convention, belongs to $(-\pi, \pi]$. Hence, $\theta_{i,d}$ has discontinuities on the y'-axis with respect to $e_{i,x}$. The discontinuity surface is defined as

$$\mathscr{D} = \{\xi_i : e_{i,x} = 0, e_{i,y} \neq 0\}. \tag{3.5}$$

In view of these s, the authors introduce the following change of coordinates

$$\begin{aligned} a_i(\xi_i) &= R_i\theta_{i,d} = \frac{\|e_i\|}{2e_{i,y}}\theta_{i,d} \\ \alpha_i(\xi_i) &= e_{i,\theta} - 2\pi n(e_{i,\theta}) - \theta_{i,d}. \end{aligned} \tag{3.6}$$

In Figure 3.2 a schematic representation of the state transformation is shown. The variable a_i defines the arc length from the r_i to e_i along the circle centred on the y'-axis that passes through these two points. a_i may be positive or negative. In the degenerate case, when $e_{i,y} = 0$, we define $a_i = 0$, which makes a_i continuous with respect to $e_{i,y}$. On the other hand the orientation error $\alpha_i \in (-\pi, pi]$ is a periodic and piecewise continuous function. The function $n : \mathbb{R} \to \mathbb{Z}$ is introduced to let $\alpha_i \in [-\pi, \pi)$. We define as $\mathscr{E}$ the set where α_i is discontinuous i.e.

$$\mathscr{E} = \{\xi_i : \alpha(\xi_i) = \pi\}. \tag{3.7}$$

The coordinates transformation in (3.6) leads to the map $\Phi : \mathbb{R}^3 \longmapsto \mathbb{R} \times (-\pi, \pi]$ from the state space $\xi_i \in \mathbb{R}^3$ to the two-dimensional state space $z_i \in \mathbb{R} \times (-\pi, \pi]$, namely $z_i = \Phi(\xi_i)$ where $\Phi(\xi_i) = \big(a_i(\xi_i), \alpha_i(\xi_i)\big)^\top$.

It is important to notice that, as shown in [Canudas de Wit and Sordalen, 1992, Lemma 1], the chosen coordinates transformation Φ maps the elements of attractor $\mathscr{O}$ in the origin of the new coordinate system, namely $\Phi(\xi_i) = 0$, for $\xi_i \in \mathscr{O}$. The

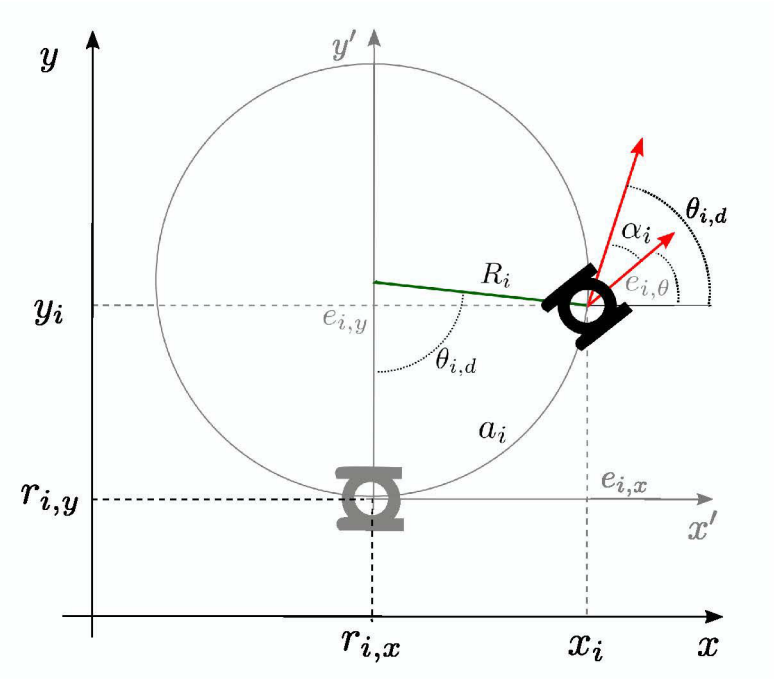


Figure 3.2: Scheme showing the state transformation used for the point stabilization.

stabilization of the set $\mathscr{O}$ for dynamics (3.2) can be thus reduced to the stabilization of the origin $z_i = 0$ for the following dynamics

$$\dot{z}_i = \frac{\partial \Phi(\xi_i)}{\partial \xi_i}\dot{\xi}_i = \frac{\partial \Phi(\xi_i)}{\partial \xi_i} b(\xi_i)u_i = \tilde{b}(\xi_i)u_i, \quad \tilde{b}(\xi_i) = \begin{bmatrix} \tilde{b}_1(\xi_i) & 0 \\ \tilde{b}_2(\xi_i) & 1 \end{bmatrix}. \tag{3.8}$$

The authors propose the feedback control law $u_i = \kappa(\xi_i)$ with positive gains $k, \gamma > 0$

$$\begin{aligned} v_i &= -\gamma \tilde{b}_1(\xi_i) a_i, \\ w_i &= -\tilde{b}_2(\xi_i) v_i - k\alpha_i. \end{aligned} \tag{3.9}$$

Replacing the feedback law in (3.8) leads to the completely decoupled closed loop dynamics

$$\begin{aligned} \dot{a}_i &= -\gamma \tilde{b}_1(\xi_i)^2 a_i, \\ \dot{\alpha}_i &= -k\alpha_i. \end{aligned} \tag{3.10}$$

It is now possible to state the following convergence result.

Theorem 3.1 (Theorem 1 in [Canudas de Wit and Sordalen, 1992]). *The closed loop dynamics (3.10) converges exponentially towards the origin $z_i = 0$, i.e. there exists some positive constants c_z and λ_z such that*

$$\|z_i(t)\| \leq c_z \|z_i(t_0)\| \, \mathrm{e}^{-\lambda_z (t-t_0)}, \quad \textit{for } t \geq t_0. \tag{3.11}$$

Moreover there exist $c_a, \lambda_a > 0$ such that, for $t \geq t_0$

$$|\alpha_i(t)| \leq |\alpha_i(t_0)| \, \mathrm{e}^{-k(t-t_0)} \tag{3.12}$$

$$|a_i(t)| \leq c_a \, |a_i(t_0)| \, \mathrm{e}^{-\lambda_a (t-t_0)}. \tag{3.13}$$

In the proof proposed by the authors it is shown how the discontinuity surfaces $\mathscr{D}$ and $\mathscr{E}$ are repulsive with respect to the trajectories of the system $\xi_i(t)$ and how the norm of $z_i(t)$ remains constant when $\xi_i(t)$ crosses those surfaces. Integrating the closed loop dynamics (3.10) the result in inequality (3.12) is trivial, whereas the convergence of the arc trajectory $a_i(t) = a_i(t_0)\mathrm{e}^{-\gamma\rho(t)}$ with $\rho(t) = \int_{t_0}^{t} \tilde{b}_1^2\big(\xi_i(\tau)\big)\mathrm{d}\tau$, involves some considerations over the boundedness of $\tilde{b_1}(\xi_i)$. In particular the authors show that for a given $T > t_0$ there exists a big enough $\varepsilon(T) > 0$ such that

$$\left|\tilde{b}_1(\xi_i(t))^2 - 1\right| \leq \varepsilon(T), \quad \text{for } t \geq T. \tag{3.14}$$

This bound is used to prove inequality (3.13) with

$$\begin{aligned} c_a =& \mathrm{e}^{\gamma(1-\varepsilon(T))T} \\ \lambda_a =& \gamma(1-\varepsilon(T)). \end{aligned} \tag{3.15}$$

By replacing the feedback law $\kappa(\xi_i)$ (3.9) in (3.2) is possible to obtain the closed loop dynamics of the error coordinates ξ_i as

$$\dot{\xi}_i = b(\xi_i)\kappa(\xi_i). \tag{3.16}$$

Moreover, the results expressed in Theorem 3.1 lead to the following Lemma about the exponential convergence of the 2D positioning error for each agent.

Lemma 3.1. *There exist positive constants c and λ such that*

$$\|e_i(t)\| \leq c\,\|e_i(t_0)\|\,\mathrm{e}^{-\lambda(t-t_0)}, \quad \textit{for } t \geq t_0. \tag{3.17}$$

Furthermore, the latter is true for $c = \frac{\pi}{2}c_a$ and $\lambda = \lambda_a$.

Proof. From geometric consideration $\|e_i\| = 2R_i\left|\sin\frac{\theta_{i,d}}{2}\right|$ and $|a_i| = R_i\,|\theta_{i,d}|$. These two expressions can be combined, and considering $\left|\frac{\frac{\theta_{i,d}}{2}}{\sin\frac{\theta_{i,d}}{2}}\right| \leq \frac{\pi}{2}$, for all $\theta_{i,d} \in (-\pi, \pi]$ one obtains

$$|a_i(t)| = \|e_i(t)\|\left|\frac{\frac{\theta_{i,d}(t)}{2}}{\sin\frac{\theta_{i,d}(t)}{2}}\right| \leq \frac{\pi}{2}\,\|e_i(t)\|, \qquad \text{for } t \geq t_0.$$

On the other side given a circle, an arc is always greater or equal to the subtended chord, thus for $t \geq t_0$

$$\begin{aligned} \|e_i(t)\| \leq& |a_i(t)| \leq c_a\,|a_i(t_0)|\,\mathrm{e}^{-\lambda_a(t-t_0)} \\ \leq& \frac{\pi}{2}c_a\,\|e_i(t_0)\|\,\mathrm{e}^{-\lambda_a(t-t_0)}. \end{aligned}$$

■

3.2 Consensus for multi-agent vehicle systems with relative sensing

We assume to have sensors on board of each vehicle which can provide at time $t_k \in \mathcal{T}_i$ the absolute 2D position of the i^{th} robot X_i and the relative positions of the neighbors, namely $X_j - X_i, \forall j \in \mathcal{N}_i(t_k)$.

We will refer to the **Rendez-Vous** problem when the robots are required to meet in some point in the Cartesian 2D space. The robots should own the same reference position in order to reach such point. As discussed in section 2.2, this agreement can be obtained using sensing information rather then communication, applying the decentralized consensus algorithm (2.14) and obtaining

$$r_i(t_k) = X_i(t_k^-) + \sum_{\substack{j=1 \\ j\neq i}}^{N} P_{i,j}(t_k)(X_j(t_k^-) - X_i(t_k^-)), \qquad \text{for } t_k \in \mathcal{T}_i. \tag{3.18}$$

Once a new reference r_i is obtained we use expression (3.1) to obtain the error coordinates ξ_i. Then the feedback law $u_i = \kappa(\xi_i)$ in (3.9) takes care of driving the i^{th} vehicle to his own updated reference r_i. This sequence is schematized by the block diagram in Figure 3.3 and the effects of the jump at time $t_k \in \mathcal{T}_i$ on both the references and the errors are geometrically described in Figure 3.4.

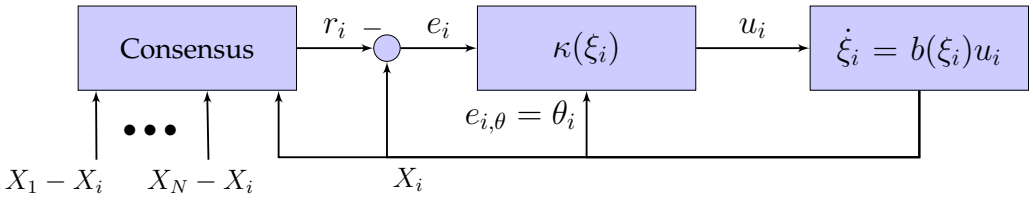


Figure 3.3: Block diagram showing how the control startegy has been implemented for robot i.

The trajectories of $\xi_i(t)$ for $t \in [t_k, t_{k+1})$ are characterized by the closed loop dynamics (3.16). Moreover, thanks to the result in Lemma 3.1, the trajectories of the positioning error e_i comply with the exponential convergence Assumption 2.1 for $t \in [t_k, t_{k+1})$.

We obtain the following closed loop hybrid dynamics for all $i \in \mathcal{V}$:

$$\begin{cases} \dot{r}_i(t) = 0 \\ \dot{r}_{i,\theta}(t) = 0 \\ \dot{\xi}_i(t) = b\big(\xi_i(t)\big)\kappa\big(\xi_i(t)\big) \end{cases} \qquad \text{for } t \in \mathbb{R}_+ \setminus \mathcal{T}_i, \tag{3.19}$$

$$\begin{cases} r_i(t_k) = \displaystyle\sum_{j=1}^{N} P_{i,j}(t_k) e_j(t_k^-) + \sum_{j=1}^{N} P_{i,j}(t_k) r_j(t_k^-) \\ r_{i,\theta}(t_k) = r_{i,\theta}(t_k^-) \\ e_i(t_k) = e_i(t_k^-) + r_i(t_k^-) - \displaystyle\sum_{j=1}^{N} P_{i,j}(t_k) e_j(t_k^-) - \sum_{j=1}^{N} P_{i,j}(t_k) r_j(t_k^-) \\ e_{i,\theta}(t_k) = e_{i,\theta}(t_k^-) \end{cases} \qquad \text{for } t_k \in \mathcal{T}_i, \tag{3.20}$$

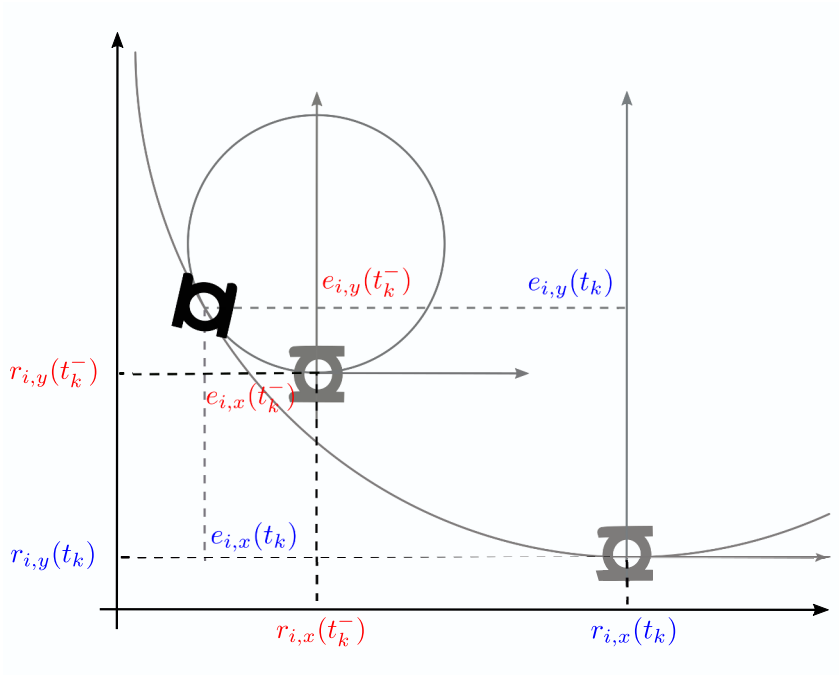


Figure 3.4: Scheme showing the effects of the discrete time event occurred at time t_k on the variable r_i and e_i. The value of the variables before and after the jump are denoted in red and blue respectively.

for $r_i(0) = X(0)$, $r_{i,\theta}(0) = 0$ and $\xi_i(0) = 0$. The equations $e_{i,\theta}(t_k) = e_{i,\theta}(t_k^-)$, $r_{i,\theta}(t_k) = r_{i,\theta}(t_k^-)$ come from the fact that in our case, only a 2D reference position r_i is provided to each vehicle, whereas $r_{i,\theta} \equiv 0$.

3.2.1 Formation realization

We focus our attention on the formation realization problem. We would like that the N non-holonomic mobile robots realize a formation somewhere in the Cartesian plane (a so called translation invariant formation). Following the idea by [Mesbahi and Egerstedt, 2010] the latter problem is fundamentally identical to the consensus problem. Indeed, let Π be the formation that robots have to achieve, i.e. $\Pi = \{(p_1^\top, \dots, p_N^\top)^\top \mid p_i \in \mathbb{R}^2, \ i = 1, \cdots, N\} \in \mathbb{R}^{2 \cdot N}$. Any collection of agents reference positions $r_1, \cdots, r_N \in \mathbb{R}^2$ is then considered to satisfy the formation requirements if

$$r_i = p_i + \underline{r}^*, \qquad \forall i = 1, \cdots, N,$$

for some arbitrary translation $\underline{r}^* \in \mathbb{R}^2$, the same for all the robots. It is possible to define $\underline{r}_i$ as the displacement of r_i from the target location p_i, i.e. $\underline{r}_i = r_i - p_i$. Then, algorithm (2.13) applied to the displacements $\underline{r}_i$ leads to

$$\underline{r}_i(t_k) = \underline{r}_i(t_k^-) + \sum_{\substack{j=1 \\ j \neq i}}^{N} P_{i,j}(t_k)(\underline{r}_i(t_k^-) - \underline{r}_j(t_k^-)), \qquad \text{for } t_k \in \mathcal{T}_i, \tag{3.21}$$

which ensures the convergence to the agreement value denoted by $\underline{r}^*$. If all the N robots agree over the value of $\underline{r}^*$, it means that they have reached the formation. The previous idea is well represented in Figure 3.5.

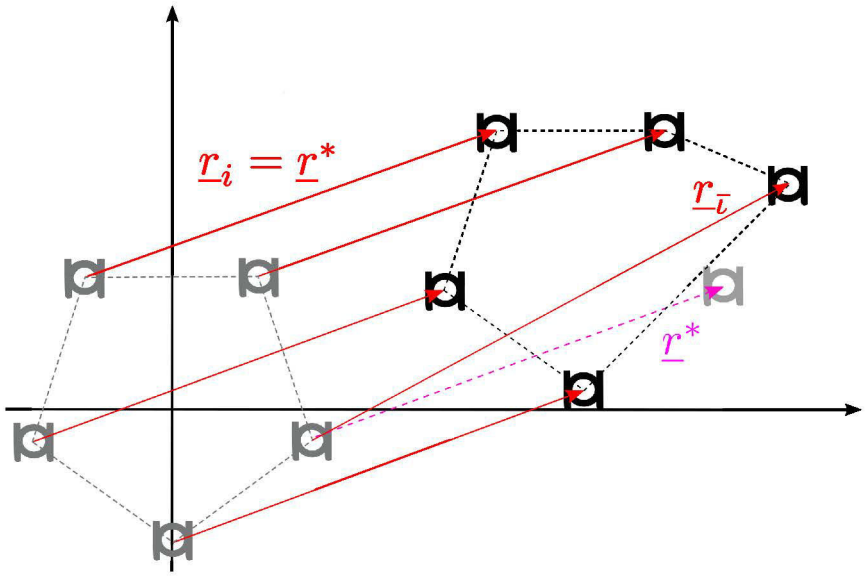


Figure 3.5: The result of the consensus on $\underline{r}_i$ for a network of 5 agents in pentagon formation. In this case agent $\bar{\imath}$ has not yet converged to the agreement value $\underline{r}^*$.

Once again, our aim is to implement a decentralized algorithm based on sensing information, hence we apply the modification proposed in (2.14) obtaining a new update law

$$\underline{r}_i(t_k) = X_i(t_k^-) - p_i + \sum_{\substack{j=1 \\ j\neq i}}^{N} P_{i,j}(t_k)(X_j(t_k^-) - X_i(t_k^-) + p_{ij}), \qquad \text{for } t_k \in \mathcal{T}_i, \tag{3.22}$$

where $p_{ij} = p_j - p_i$ is the distance that the neighbors should have in the desired formation. From an actualized value of $\underline{r}_i$ we obtain a new reference using $r_i = \underline{r}_i + p_i$ and the new value of the error coordinates ξ_i from (3.1). This value is then used to feed the control law $u_i = \kappa(\xi_i)$ in (3.9).

The closed loop hybrid dynamics that is obtained using (3.22) can be easily reconducted to the hybrid dynamics of the Rendez-Vous problem (3.19)-(3.20) considering the special case for which $p_i = 0,\ \forall i \in \mathcal{V}$. For this reason, in the following, the stability results developed in Chapter 2 are applied to the Rendez-Vous hybrid multi-agent dynamics.

3.3 Numerical examples

In this section we provide some numerical example as validation of the theoretic results obtained in Chapter 2, applying the stability analysis to a formation realization

scenario for a network of 7 robots. The validation has been performed with a simulator developed in Matlab, with an object oriented architecture. The fundamental results concerning the dwell-time have been applied to networks of robots whose interactions are described by undirected and directed graph. Eventually, we evaluated the convergence in case of measurements noise.

The values of gains γ and k for the feedback law (3.9) are the same for all the vehicles in the network and they have been chosen using the approach described in [Canudas de Wit and Sordalen, 1992, Section D]. We use as design specification the times T_a and T_α as the amount of times needed to decrease respectively $a_i(t)$ and $\alpha_i(t)$, from their initial values to some prescribed values $a_i(T_a)$ and $\alpha_i(T_\alpha)$, i.e. for $t_0 = 0$

$$\left|\frac{\alpha_i(T_\alpha)}{\alpha_i(0)}\right| \leq n_\alpha, \ \forall t \geq T_\alpha, \text{ and } \left|\frac{a_i(T_a)}{a_i(0)}\right| \leq n_a, \ \forall t \geq T_a > T_\alpha,$$

where n_α and n_a are the specified rate of convergence and where we imposed α_i to converge more rapidly than a_i. From the exponential convergence of α_i in (3.12) it is straightforward to obtain k as

$$k = \frac{1}{T_\alpha} \ln \frac{1}{n_\alpha}.$$

The authors show that from T_α on (i.e. when α_i is small) the bound $\epsilon(T_\alpha)$ introduced in (3.14) can be approximated by

$$\epsilon(T_\alpha) \leq \varepsilon_0 = n_\alpha^2 \pi^2 \left(\frac{\pi^2}{4} - 1\right) + \pi^2 n_\alpha, \quad \text{for } t \geq T_\alpha.$$

Using the values of c_a and λ_a in (3.15) the exponential convergence condition of a_i in (3.13) becomes

$$\begin{aligned} |a_i(T_a)| &\leq |a_i(0)| \, \mathrm{e}^{\gamma(1-\varepsilon(T_\alpha))T_\alpha} \mathrm{e}^{-\gamma(1-\varepsilon(T_\alpha))T_a} \\ &\leq |a_i(0)| \, \mathrm{e}^{\gamma(1-\varepsilon_0)T_\alpha} \mathrm{e}^{-\gamma(1-\varepsilon_0)T_a} \\ &\leq |a_i(0)| \, \mathrm{e}^{-\gamma(1-\varepsilon_0)(T_a - T_\alpha)}, \end{aligned}$$

which leads to

$$\gamma = \frac{1}{(1-\varepsilon_0)(T_a - T_\alpha)} \ln \frac{1}{n_a}.$$

Moreover the exponential convergence constants of the error e_i in (3.17) can be expressed according to (3.15) as

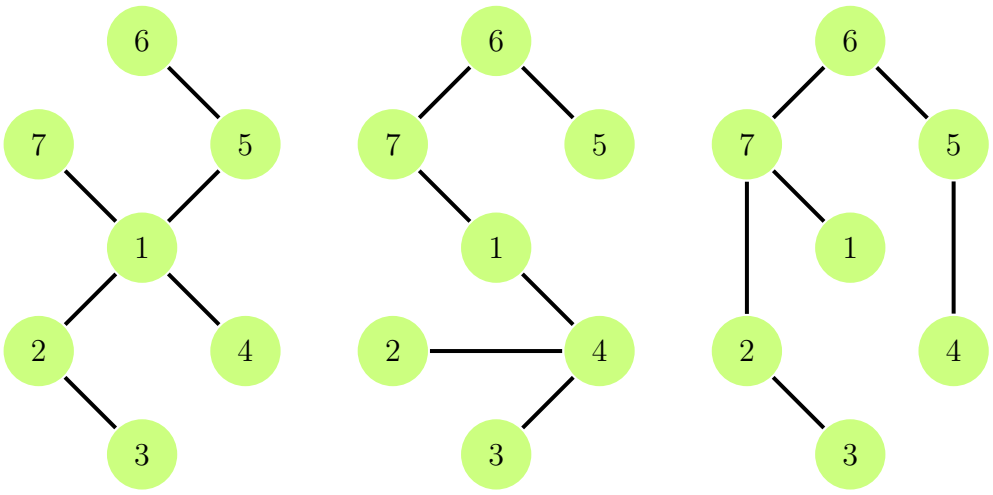
$$c_i = \frac{\pi}{2} \mathrm{e}^{\gamma(1-\varepsilon_0)T_\alpha} \text{ and } \lambda_i = \gamma(1-\varepsilon_0).$$

Throughout the rest of the section, as design specification we use $T_a = 5T_\alpha = 5$ sec, $n_a = n_\alpha = 0.01$ obtaining

$$k = 4.61, \quad \gamma = 1.29, \quad c_i = 4.97, \quad \lambda_i = 1.15, \quad \forall i \in \mathcal{V}.$$

3.3.1 Undirected network graph case

The fleet of robots is required to realize a translation invariant linear formation specified by the set of positions $\Pi = \{(-3,-3),(-2,-2),\ (-1,-1),(0,0),\ (1,1)(2,2),(3,3)\}$ in meters. The interaction between the agents switches randomly between the ones described by the three graphs represented in Figure 3.6 following a switch function $\sigma : \mathcal{T} \to \{1,2,3\}$. The three graphs topologies have been chosen such that the graphs $\mathcal{G}_1, \mathcal{G}_2$ and $\mathcal{G}_3$ are connected, in order to comply with Assumption 2.3.



(a) $\sigma(t_k) = 1$. (b) $\sigma(t_k) = 2$. (c) $\sigma(t_k) = 3$.

Figure 3.6: Undirected graphs used to implement the sensing interaction between the vehicles in the network.

A different weight stochastic matrix $P_{\sigma(t_k)} = P(t_k)$ identifies each graph, where the elements $P_{ij}(t_k)$ are chosen locally in order to fulfil the requirements of Assumption 2.2 taking for instance $a \in \left(0, \frac{1}{2(N-1)}\right)$ and $b = 1 - n_i a,\ b \in \left(\frac{1}{2}, 1\right]$ where n_i is the number of neighbors of i at time t_k. For $a = \frac{2}{25}$ we obtain

$$P_1 = \frac{1}{25}\begin{bmatrix} 17 & 2 & 0 & 2 & 2 & 0 & 2 \\ 2 & 21 & 2 & 0 & 0 & 0 & 0 \\ 0 & 2 & 23 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 23 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 21 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 & 23 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 & 23 \end{bmatrix}, \quad P_2 = \frac{1}{25}\begin{bmatrix} 21 & 0 & 0 & 2 & 0 & 0 & 2 \\ 0 & 23 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 23 & 2 & 0 & 0 & 0 \\ 2 & 2 & 2 & 19 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 23 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 & 21 & 2 \\ 2 & 0 & 0 & 0 & 0 & 2 & 23 \end{bmatrix},$$

$$P_3 = \frac{1}{25}\begin{bmatrix} 23 & 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 21 & 2 & 0 & 0 & 0 & 2 \\ 0 & 2 & 23 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 23 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 21 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 & 21 & 2 \\ 2 & 2 & 0 & 0 & 0 & 2 & 19 \end{bmatrix}. \tag{3.23}$$

Since the network is described by an undirected graph, it is possible to apply Theorem 2.1 to find a suitable lower bound on the dwell time between two consecutive resets, that guarantees the stability of the hybrid interconnection. From the formula in (2.28) we obtain $\underline{\tau}^* = 4.9$ s. Thus, the discrete time events sequence $\mathcal{T}$ of the simulation has been chosen such $\forall t_k \in \mathcal{T}$, $t_{k+1} - t_k > 4.9$ s.

The initial Cartesian position of the agents have been chosen as $X_1(0) = (-6, -1)$, $X_2(0) = (-9, -4)$, $X_3(0) = (-6, 3.5)$, $X_4(0) = (1, -2)$, $X_5(0) = (-11, 6)$, $X_6(0) = (2, 5)$ and $X_7(0) = (-2, 5)$ where the coordinates are expressed in meters; the initial heading angles are $\theta_1(0) = -1.5$, $\theta_2(0) = -3$, $\theta_3(0) = -0.9$, $\theta_4(0) = -0.3$, $\theta_5(0) = 1.5$, $\theta_6(0) = 1.5$ and $\theta_7(0) = -4$ all expressed in radians. The initial reference vectors $r_i(0)$ have been initialized to $X_i(0)$.

The trajectories of the robots are shown in Figure 3.7 with the sequence of reference positions r_i updated through the consensus, emphasizing the final positions of the agents that realize the formation defined by Π but in a different place of the 2D plane (i.e. what is called a translation invariant formation realization).

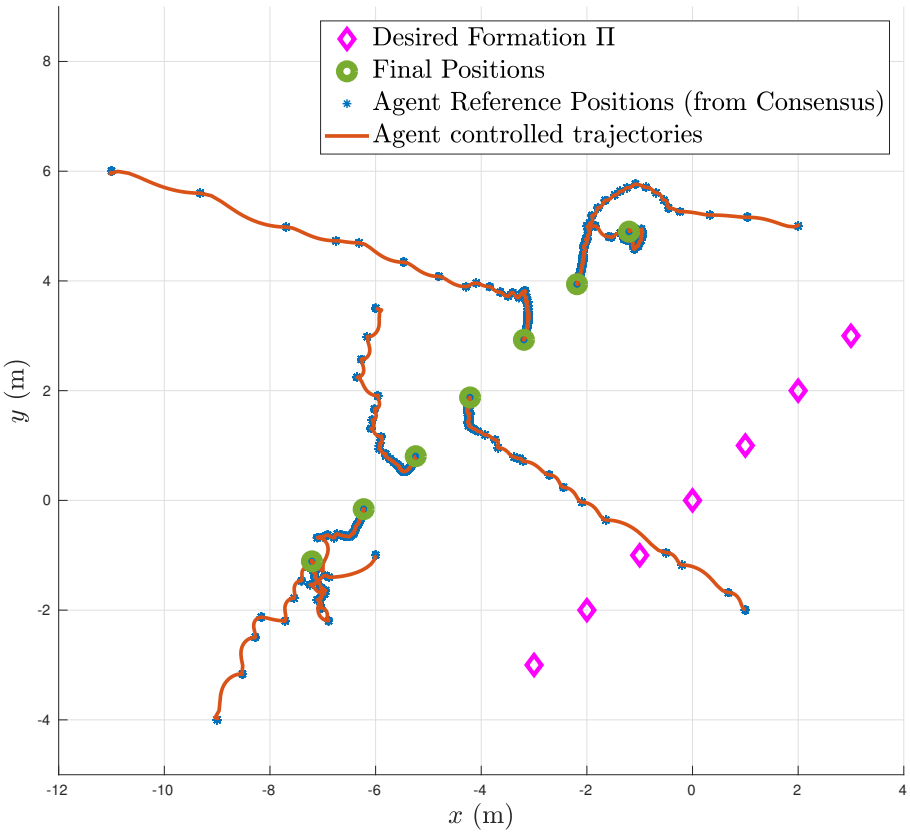


Figure 3.7: A picture showing the robots reaching the line formation in the plane.

The convergence of the robots toward an agreement over the value of $r_i(t)$, is represented in Figure 3.8 and Figure 3.9, respectively for the x and the y component. In the same figure we represented two detailed view of the same trajectories where it is

possible to appreciate as the dwell time condition allows discrete time events only once the error e_i has already converged to the reference r_i.

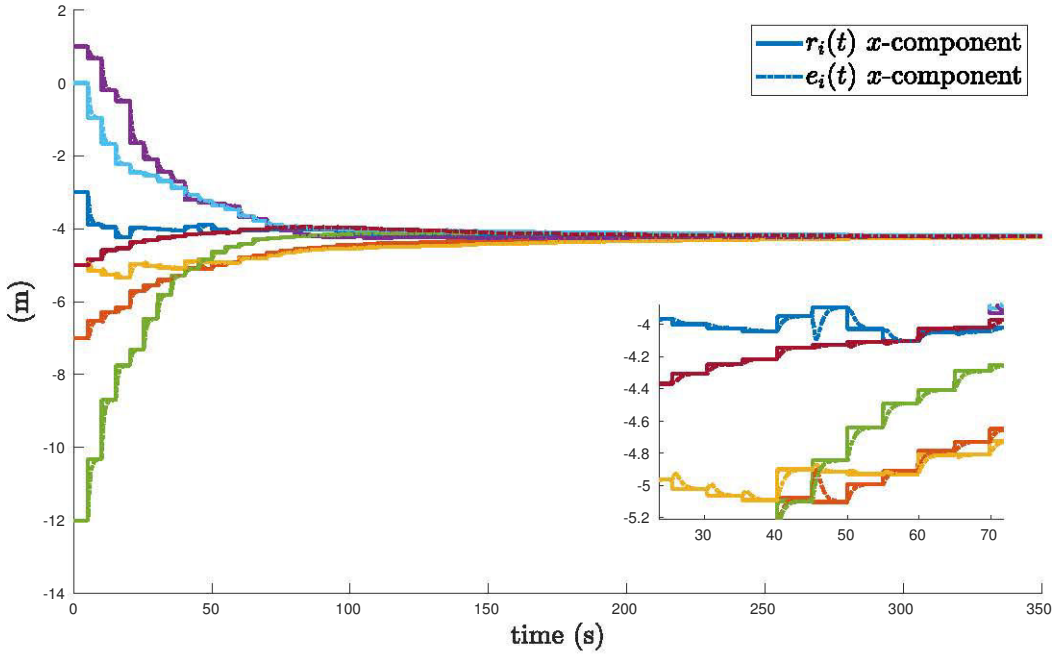


Figure 3.8: Agreement on the x component for the undirected graph case.

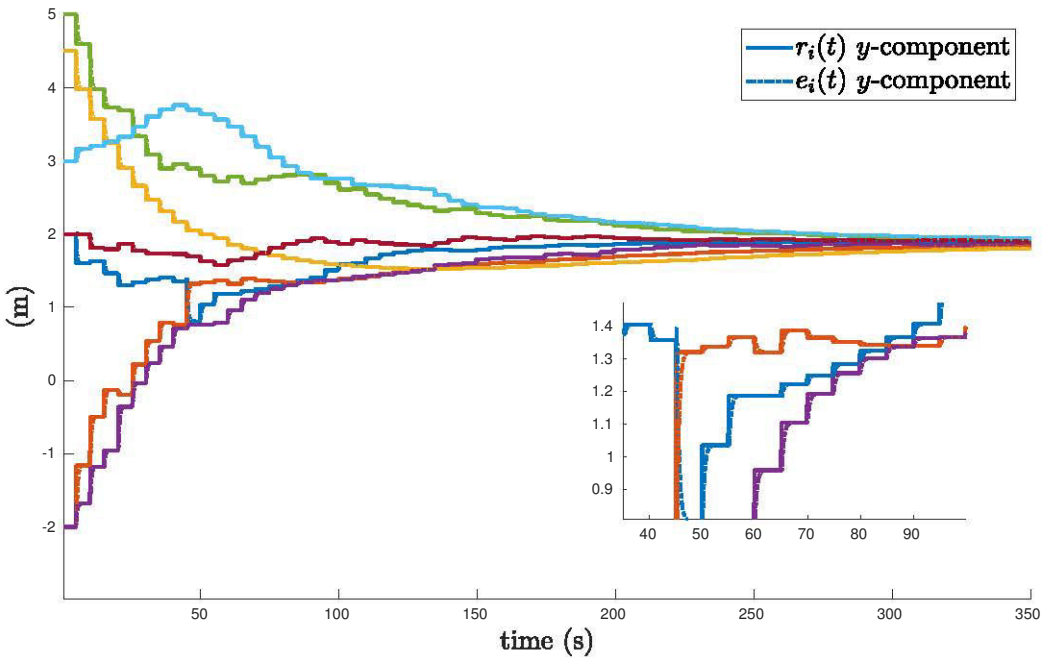


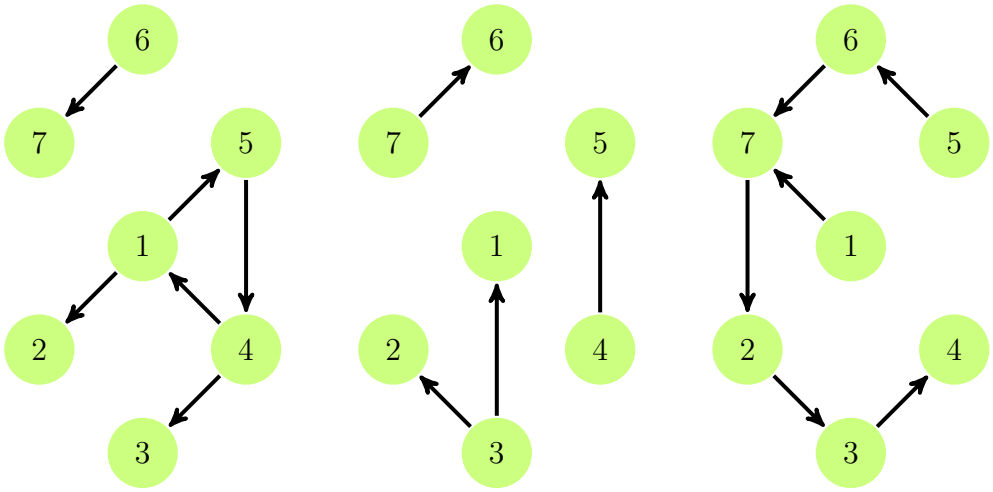
Figure 3.9: Agreement on the y component for the undirected graph case.

3.3.2 Directed network graph case

For this scenario, the fleet of robots is required to realize a circle formation specified by the set of positions $\Pi = \{(-3,-3),(-2,-2),\ (-1,-1),(0,0),\ (1,1)(2,2),(3,3)\}$ in meters. In order to fulfil Assumption 1.3 in this case the interaction between the agents switches quasi-randomly (we enforce the intercommunication interval to $L = 4$) between the three directed topologies represented in Figure 3.10 following once again a switch function $\sigma : \mathcal{T} \to \{1,2,3\}$. The three graphs have been chosen such that the union graph $\mathcal{G} = \mathcal{G}_1 \bigcup \mathcal{G}_2 \bigcup \mathcal{G}_3$ is connected, in order to comply with Assumption 1.2.

In this example, the network is described by a directed graph, hence it is possible to apply Theorem 2.3 to find a new lower bound on the dwell time between two consecutive resets. From the formula in (2.51) we obtain $\underline{\tau}^* = 19.1$ s. The discrete time events sequence $\mathcal{T}$ of the simulation has been chosen such $\forall t_k \in \mathcal{T}$, $t_{k+1} - t_k > 19.1$ s.

The initial Cartesian position of the agents and the initial heading angles have been kept the same as in the previous example. The initial reference vectors $r_i(0)$ have been initialized to $X_i(0)$.



(a) $\sigma(t_k) = 1$. (b) $\sigma(t_k) = 2$. (c) $\sigma(t_k) = 3$.

Figure 3.10: Digraphs used to implement the sensing interaction between the vehicles in the network.

Each weight stochastic matrix $P_{\sigma(t_k)} = P(t_k)$ identifies one of the graph and the elements of $P_{ij}(t_k)$ are chosen locally according to Assumption 2.4 taking for instance $b \in \left(\frac{Ln-1}{Ln}, 1\right) = \left(\frac{7}{8}, 1\right)$ and $a = \frac{1-b}{n_i}$ where n_i is the number of agent linked to i at time t_k. For $b = \frac{11}{12}$ we obtain

$$P_1 = \frac{1}{24}\begin{bmatrix} 22 & 0 & 0 & 2 & 0 & 0 & 0 \\ 2 & 22 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 22 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 22 & 2 & 0 & 0 \\ 2 & 0 & 0 & 0 & 22 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 24 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 22 \end{bmatrix}, \quad P_2 = \frac{1}{24}\begin{bmatrix} 22 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 22 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 24 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 24 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 22 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 22 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 24 \end{bmatrix},$$

$$P_3 = \frac{1}{24}\begin{bmatrix} 24 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 22 & 0 & 0 & 0 & 0 & 2 \\ 0 & 2 & 22 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 22 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 24 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 22 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 22 \end{bmatrix}. \tag{3.24}$$

The robots follow the trajectories represented in Figure 3.11, where, once again, we emphasize the final positions of the agents that realize the formation defined by Π. The convergence of the robots toward an agreement over the value of $r_i(t)$, is represented in Figure 3.12 and Figure 3.13, respectively for the x and the y component. In the same figure, we represent a detailed view of the trajectories. In this case, we can see how the new dwell-time condition results in a more conservative approach, compared for instance to Figure 3.9 where the discrete time events were allowed to occur more often.

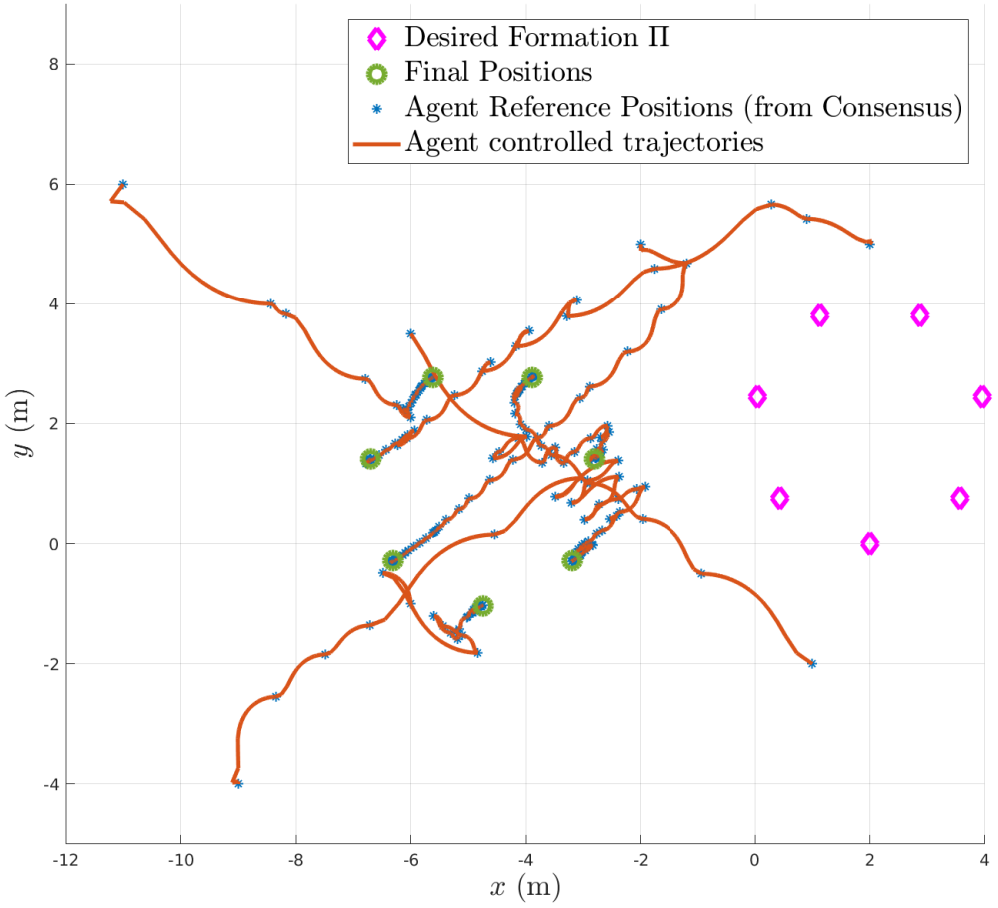


Figure 3.11: A picture showing the robots reaching a circular formation in the plane.

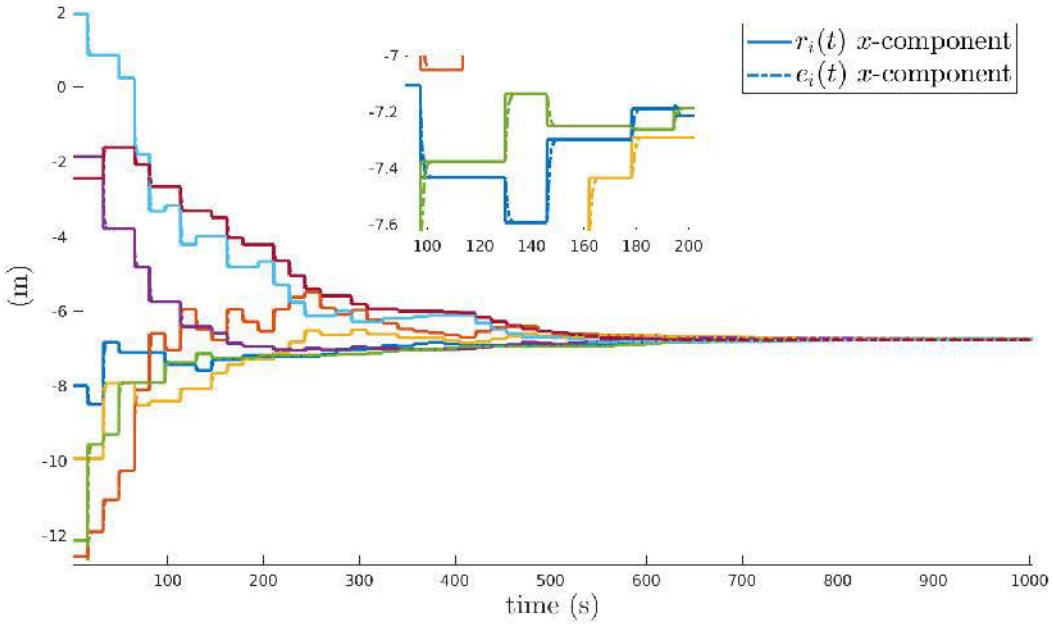


Figure 3.12: Agreement on the x component in the case directed graph case.

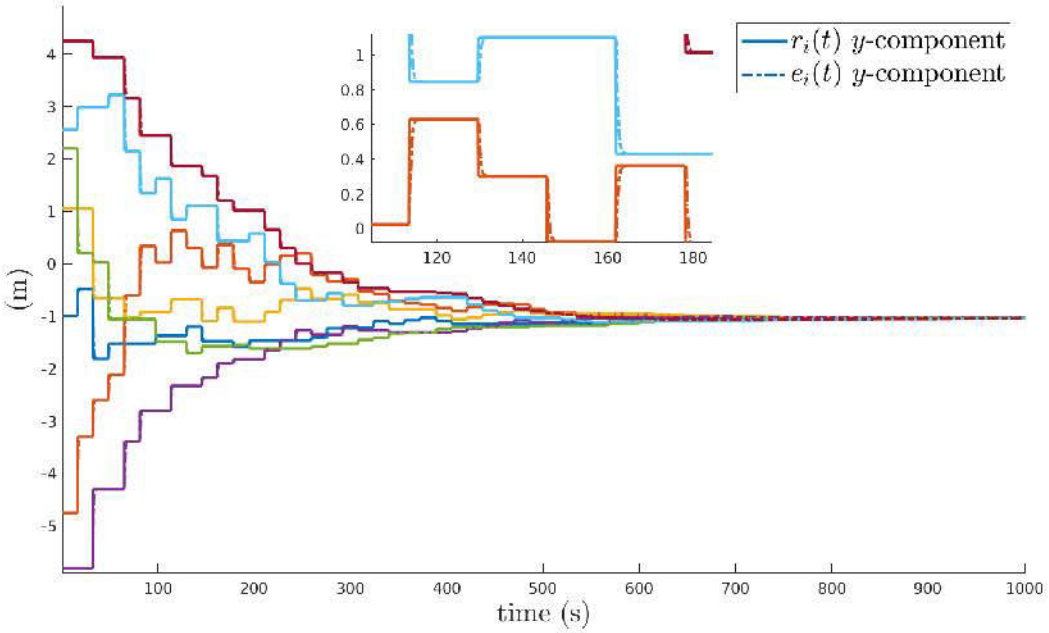


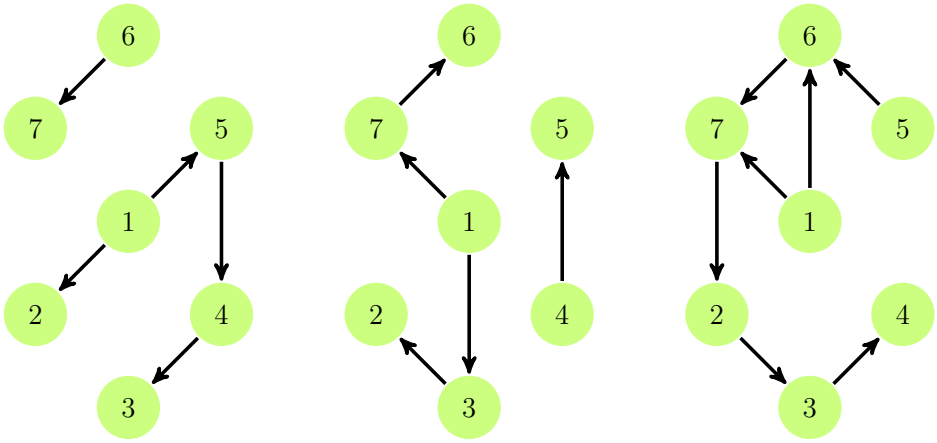
Figure 3.13: Agreement on the y component in the case directed graph case.

3.3.3 Measurements noise

In this last example we consider an exogenous disturbance input δ_i. This disturbance originates, following (2.60), from the noise on the positioning measurements coming

from the localization sensor of the i^{th} vehicle $|\omega_i(t)| \leq \bar{\omega}$ and the noise on the relative positioning measurements $|\omega_{ij}(t)| \leq \bar{\omega}$. For this example we used $\bar{\omega} = 0.1$ m. Using expression (2.61), it is possible to find $|\delta_i(t)| \leq \bar{\delta} = 0.11$ m. As for the desired formation, we choose a triangular formation $\Pi = \{(-3,0),(-2,1),\ (-1,2),(0,3),\ (1,2)(2,1),(3,0)\}$ in meters.

In order to fulfil Assumption 1.3, we enforce once again the intercommunication interval to $L = 4$. The digraph switches between the three directed topologies in Figure 3.14. Moreover, the union graph $\mathcal{G} = \mathcal{G}_1 \bigcup \mathcal{G}_2 \bigcup \mathcal{G}_3$ is connected but not strongly connected. In fact the node 1 corresponds to the centroid of the union graph. From a practical point of view this means that agent 1 influences his neighbors but is not influenced by any of the other nodes. With this configuration, we essentially switch from a leader-less formation realization strategy to a leader-followers one. In this case all the actions performed by the first agents will be re-tracked by the other agents, hence moving vehicle 1 will move the whole formation according to his position. The value of the dwell-time has been calculated once again applying the result of theorem 2.3 obtaining $\underline{\tau}^* = 19.1$ s.



(a) $\sigma(t_k) = 1$. (b) $\sigma(t_k) = 2$. (c) $\sigma(t_k) = 3$.

Figure 3.14: Digraphs used to implement the sensing interaction between the vehicles in the network.

The following weight stochastic matrices $P_{\sigma(t_k)}$ have been chosen:

$$P_1 = \frac{1}{24}\begin{bmatrix} 24 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 22 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 22 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 22 & 2 & 0 & 0 \\ 2 & 0 & 0 & 0 & 22 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 24 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 22 \end{bmatrix}, \quad P_2 = \frac{1}{24}\begin{bmatrix} 24 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 22 & 2 & 0 & 0 & 0 & 0 \\ 2 & 0 & 22 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 24 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 22 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 22 & 2 \\ 2 & 0 & 0 & 0 & 0 & 0 & 22 \end{bmatrix},$$

$$P_3 = \frac{1}{24}\begin{bmatrix} 24 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 22 & 0 & 0 & 0 & 0 & 2 \\ 0 & 2 & 22 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 22 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 24 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 22 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 22 \end{bmatrix}. \tag{3.25}$$

The result of the simulation is represented in Figure 3.15 where the robots are first shown to converge to the desired formation. The first vehicle is then remotely moved north and afterward north-east and the robots are shown to converge to the new location of the formation. From the detailed views of the consensus trajectories in Figure 3.16 and Figure 3.17 it is possible to appreciate the presence of some residual oscillation around the consensus values caused by the noisy measurements coming from the relative positioning sensors.

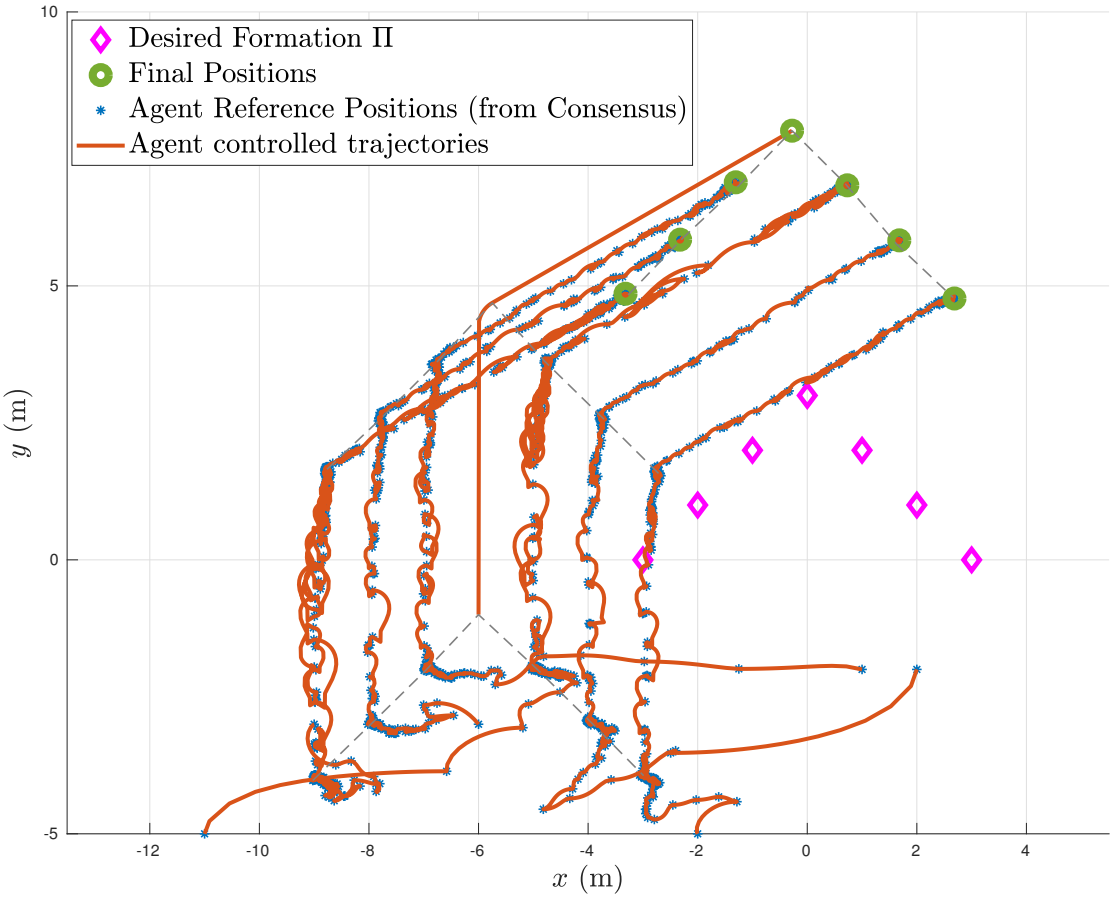


Figure 3.15: A picture showing the robots reaching a triangular formation in the plane.

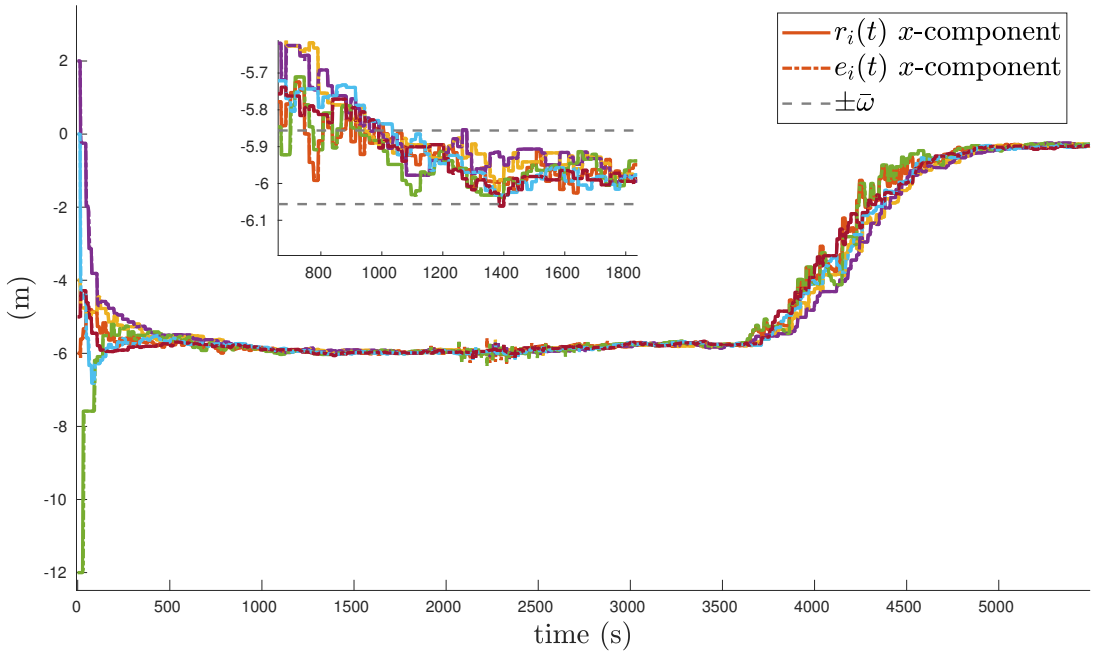


Figure 3.16: Agreement on the x component in presence of exogenous disturbance inputs.

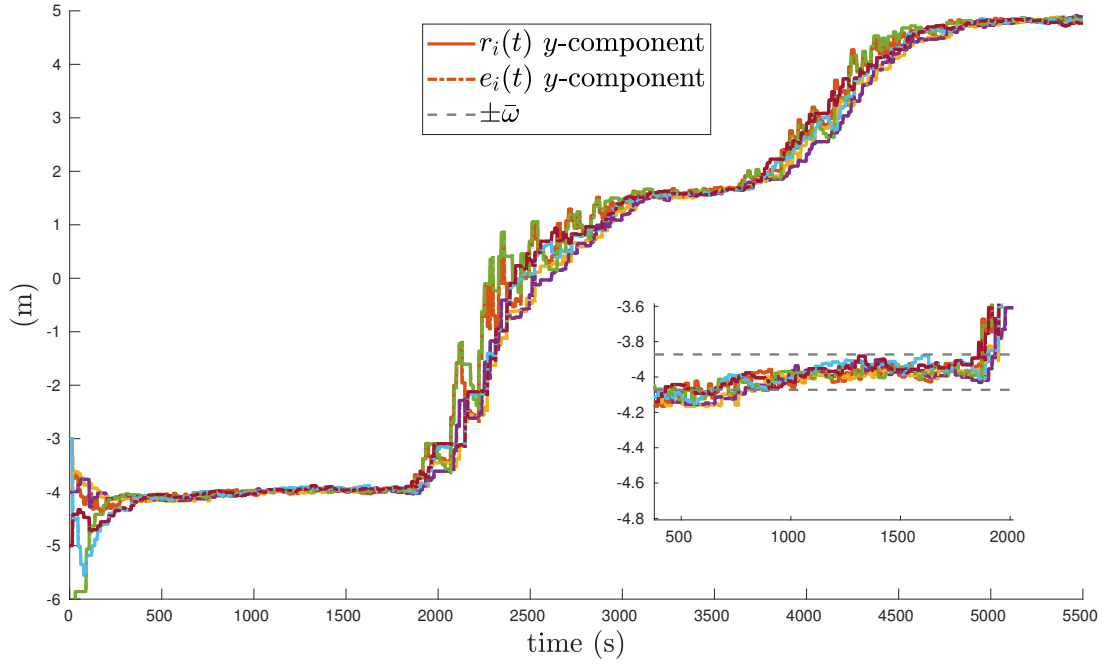


Figure 3.17: Agreement on the y component in presence of exogenous disturbance inputs.

3.4 Conclusions

This Chapter presented the problem of the consensus in fleets of non-holonomic robots as framework of application for the theoretical results discussed in Chapter 2. From this perspective, the local dynamics of each vehicle has been treated as a tracking problem of a piece-wise constant reference originated by a decentralized algorithm using relative positioning measurements. The local tracking problem, or point-stabilization problem, has been addressed in Section 3.1 using an already existent nonlinear piecewise continuous feedback law, which has been proven to comply with the Assumptions introduced in Chapter 2. Afterward, the consensus and the leader-less formation realization problem have been introduced in Section 3.2. Finally the dwell-time stability results have been applied in Section 3.3 in the case of undirected and directed graphs in presence of measurements noise.

Chapter 4

Experimental platform: management of a fleet of Autonomous Mobile Robots (AMR)

In the last years we have witnessed a steady rise in robot-based industrial automation. The industrial robots are capable of carrying out repeated tasks with great accuracy and thus, they are widely deployed in mass production process. On the other hand, the industrial and robotic automation has often been confined to those companies able to afford the elaborate production infrastructure built around these robots.

This scenario is likely to change because of the rise of service robots, which thanks to their versatility, can work in unstructured environment and more effectively react to environmental changes compared to their industrial counterparts. These robots can be easily programmed, intuitively and dynamically combined to carry out a large number of applications.

One such application is the vehicle fleet management system for warehouse and factory environment. The fleet management systems can be seen as a collection of software and hardware components which interact to facilitate the achievement of cooperative tasks and goals. Take for instance the fleet management systems of Kiva mobile robots [Wurman et al., 2008] used for moving goods within the Amazon warehouse facilities [Amazon Robotics, 2019]. Another example is the Asti Easybot autonomous ground vehicles (AGVs) already in use on the FFLOR site as part of the CEA smart logistics research guidelines. In general, those robots are programmed to move autonomously along predefined tracks carrying goods. However, the routes and the timing are provided by a centralized planner which also carries out resource allocation and manages job assignment to individual robots.

The fleet management system studied in this work deals with a more flexible kind of autonomous mobile robots, designed for less repetitive and compulsory tasks such as monitoring and data collection. Such applications have been object of study and development in the last years for indoor public environment [Di Paola et al., 2010] and automated greenhouses [Roldán et al., 2015]. The application of this technology to an industrial environment is motivated either by simple production and performance control purpose or by more risky search and rescue tasks to be performed in a hazardous environment damaged by accidents or disasters [Nourbakhsh et al., 2005].

Those systems require the robots to be able to estimate their own position and to detect moving or fixed obstacles. Those components are used to update the environment map which, in turn, is used by the planner to create new paths for the robots. The user or the operator provides the goals for the fleet assembled in a pre-designed formation. However, such destinations may also come from an ERP (Enterprise Resource Planning) system in an industrial setting. The autonomy of each robot is granted by the navigation module that implements SLAM (Simultaneous localization and mapping) [Thrun et al., 2005]. The existing fleet management solutions focus on system integration, involving various software and hardware components like the one presented by [Coffee et al., 2003], or exploiting the high computational capability offered by some third party cloud engine (see for example the Rapyuta cloud robotics system presented by [Mohanarajah et al., 2015]).

In this work, we are interested in making use of the decentralized control and communication framework offered by the Robot Operating System (ROS) [WillowGarage, 2019c], whose main features and tools are described in the first section. In Section 4.2 we described the details of the algorithm implementation in simulation and on real robotic devices. Finally, in Section 4.3 we give the details of the fleet management system implemented and tested at the Future of Factory Lorraine site in Trémery in collaboration with the CEA nono-Innov laboratory in Paris-Saclay.

4.1 Fleet management system backbone: Robot Operating System

The fleet management system is composed of several software components mainly developed using the ROS client libraries and executed in a Linux environment.

ROS [Quigley et al., 2009] is a collection of software framework for controlling, monitoring and managing single or multiple robots application. ROS uses a peer-to-peer strategy for inter-process or inter-robot communication, it supports multiple programming languages (C/C++, Python, Java, Matlab) and it provides different kinds of tools for robot software development.

For more details about ROS the interested reader can refer to the online wiki [WillowGarage, 2019c]. For the sake of clarity, we list and analyze some of the common concepts related to ROS which will be frequently used in the following. **Nodes** are ROS processes that perform the logic or computational tasks. They communicate with each other by exchanging **messages**. **Topics** are tags over which the network of nodes exchange messages. They can be seen as a channel of communication between different ROS processes, where the nodes can **publish/-subscribe** to a given topic. The nodes can also provide or ask for a **service**. A node which "consumes" a service (i.e. asking for some service to other nodes) is a **service client**, whereas, a node which "produces" the result after a service call is called **service server**. The nodes share a common database to store some relevant quantities or information, called **parameter server**. A special ROS process called **ROS master** sets up the peer-2-peer communication for topics and it controls the services and the parameter server updates.

In order to give a visual representation we can see ROS processes as nodes in a graph topology, connected by edges called topics. Multiple ROS nodes can exchange

messages through topics, but they can also be client of a service (make service calls to other nodes), or server for other services (provide a service for other nodes), or set or retrieve shared data from the parameter server.

ROS functionalities are organized by packages. Those packages contain application-related code which uses one or more ROS client libraries. The ROS community contributes to the development of the ROS framework coding, validating and distributing new packages for new functionalities or new releases of old packages. For instance, three packages that have been extensively used in the experimental platform development are the TF, AMCL and GMapping packages.

TF package: TF is a package that allows the user keep track of multiple coordinate systems transformation. Each time a user creates a new reference frame a new transformation between a parent reference frame to the one just created is published on the buffered TF topic. In this way, the TF packages produce a structure with tree dependencies between different frames. Using the functionalities of the package, the user can transform points, vectors, poses, between any two coordinate frames at any desired point in time.

GMapping package: GMapping package uses the particle filter solution proposed in [Grisetti et al., 2007] to provide SLAM capability via laser-scan measurements. Using this package the robot creates a 2-D occupancy grid map of the environment from laser and pose data collected and at the same time it is capable of locating itself in this newly mapped environment. The GMapping package allows the user to save a newly created map, or to publish a map of the environment with a map server.

Adaptive Monte Carlo Localization package: The AMCL package implements a Monte Carlo localization algorithm (see [Fox et al., 1999] for further reading about this technique) that is used by the mobile agents to localizes themselves in a work space. In this case the mapping and the localization do not take place at the same time as in the SLAM package. The AMCL is purely a localization package so it needs a running map server which provides a previously created map of the environment. The algorithm takes in the environment map, the laser scans data and outputs pose estimates.

Rviz package: The Rviz package provides 3D or 2D visualization tools to represent a wide range of physical measures and quantities. For instance it can be used to display live representations of sensor values coming over ROS topics including camera data, LIDAR distance measurements, sonar data. It can also visualize the physical state of the robot displaying its position orientation and movements. Rviz is most useful since it provides a visualization tools of the ROS perspective of the world, that is how data, measurements, messages and services are behaving in the system.

One of the most important feature that characterizes ROS development flow is the modularity. As represented in Figure 4.1 existent software components or packages can be integrated with the nodes or packages newly developed by the user, simply merging the new nodes in the existent network of topics and services, and registering them to the ROS master.

There are mainly two strategies to implement the fleet management system with ROS: a **single-master** solution or a **multi-master** solution.

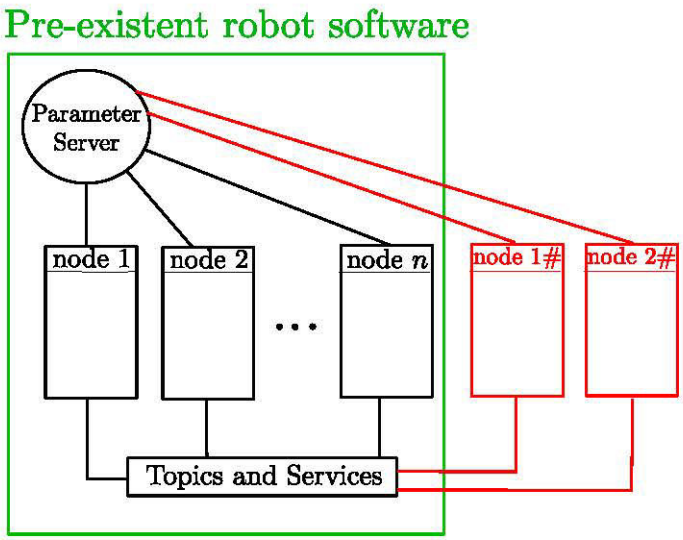


Figure 4.1: The custom functionalities represented by node 1# and node 2# are directly plugged into the existing robot software.

4.1.1 Single Master System

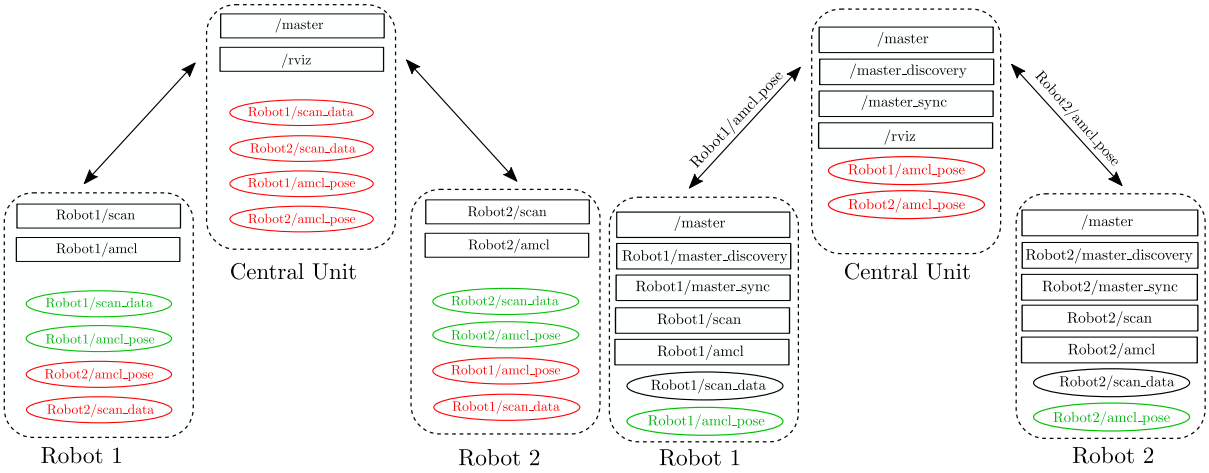
In the single master implementation the ROS master node runs on one machine in the network, while others nodes work in a decentralized manner on different computers. Every node can run on any device in the network with the exception of the hardware interface related processes (driver nodes) that need to be launched on the machine which is directly connected to the hardware interface. In order to let other machines to participate to the distributed network the user needs to specify to each participant the location of the ROS master in the network in terms of the Uniform Resource Identifier (URI) and the IP address that they have to use in the network. This is done simply setting two environment variables *ROS_MASTER_URI* and *ROS_HOSTNAME*.

In order to avoid name conflicts, nodes, topics and services which belong to the same robot need to be grouped under a common name space tag. Take for instance two robots which run at the same time the AMCL nodes. Those nodes will coexist in the network with the same name tags as well as the topics related to them, even though they represent the Monte Carlo localization algorithms of two different robots. To avoid such situation node and topic names are extended specifying the robot which publishes them. The result can be appreciated from the example in Figure 4.2(a) where a network of one central computer and two robots is represented.

With this approach topics messages and services are available over the whole network for subscription and this may require high bandwidth requirement. Moreover, a single master makes the whole system more vulnerable because if the ROS master shuts down on the base station, service based communication between the nodes get stopped. On the other hand, topic based communication can still work because the ROS master establishes the routing for the topic messages just once at the start up of each node. Anyway new topics cannot be created anymore without the ROS master running.

4.1.2 Multi Master System

The multi-master approach is one of the possible solution to overcome the two main drawbacks of the single-master systems. In fact, in the absence of sufficiently high bandwidth or with a poorly reliable base station one can be interested in reducing the amount of topics messages shared in the network as well as removing the dependence on a single ROS master process. There exist several different ROS packages which implement a multi-master network; among them the most used and appreciated by the ROS community is the *multimaster_fkie* package. The package is composed by two nodes the master discover and the master synchronizer. The master discover takes care of making all ROS masters in the network aware of each other. It also monitors and notifies the changes in the network topology (for instance if a ROS master is shut down). The master synchronizer node allows the user to select which topics or services a given ROS master should share in the network with the other masters. In Figure 4.2(b) an example of a multi-master network is represented.



(a) An example of single-master implementation.

(b) A multi-master implementation where only the amcl pose messages circulate in the network, thanks to the master_sync nodes.

Figure 4.2: A schematic view of a single and multi master implementation. The black boxes represent the nodes. The topics subscriptions and publication over the network are represented in red and green respectively, and in black if the topic stays local.

Even though the multi-master approach takes care of several problems encountered in a single master system, it still presents some drawbacks. For instance, it requires an higher demand of computational power for on-board units and it is not well scalable when the number of agents is increased or decreased.

4.2 decentralized software solution

In the following section, we propose an insight of the software solutions and the implementation choices that have been done for this work. The software implementation of the two stage approach presented in Chapter 3 has been first validated in simulation

exploiting a free robotic simulator called **Gazebo** [Koenig and Howard, 2004, Willow-Garage, 2019a] which provides a robust physics engine, high-quality graphics, and convenient programmatic and graphical interfaces. The choice of using Gazebo as first step of the implementation represents a propaedeutic approach to the real implementation. In fact, Gazebo is a robotic simulator which has been developed in symbiosis with ROS, hence all the software tested in Gazebo, can be thereafter embedded in real device for real testing with almost no additional effort. In a second moment, we have been able to adapt and modify the software choices for the implementation on real robotic devices.

4.2.1 Implementation for simulated agents

Our aim is to implement a software solution using the ROS client libraries that matches the block diagram presented in Figure 3.3 and then to simulate the behaviour of the system. The decentralized and the local controller have been developed as separate nodes called *decentralized_controller* and the *local_controller*.

The exchange of the reference position between the two nodes of each robot is done via the usage of the TF package. The TF package serializes coordinate frame transformations composed by a 3D translation vector $v \in \mathbb{R}^3$ and a 3D rotation in quaternion representation $q \in \mathbb{R}^4$ between two coordinate systems; then it sends them over the network through the TF topic. In order to retrieve a coordinate transform each node needs to subscribe to the TF topic and waits for the corresponding coordinate transformation to be available. The decentralized controller publishes a TF coordinates transform $\mathscr{R}_i = \{(r_i^\top, 0)^\top, q_{r_i}\}$ where the quaternion q_{r_i} stands for the orientation of the frame with respect to a common map reference frame; since in our case $r_{i,\theta} = 0$ the quaternion will represent the identity rotation. The local controller subscribes to a frame called *robot_footprint* and denoted as $\mathscr{X}_i = \{(X_i^\top, 0)^\top, q_{X_i}\}$ in the following. This frame is attached to the robot base and corresponds to the actual pose of the vehicle; it is published on the TF topic by the AMCL node which is in charge of the robot localization. The node subscribes also to the newly created $\mathscr{R}_i$ and it uses the TF package tools to retrieve the coordinate transformation $\mathscr{E}_i = \{(e_i^\top, 0)^\top, q_{e_i}\}$ between $\mathscr{R}_i$ and $\mathscr{X}_i$. From $\mathscr{E}_i$, the user can obtain e_i and $e_{i,\theta}$ to implement the control law (3.9) and finally publish the command. The pseudo-code of the local controller node is presented in Algorithm 2.

On the other hand, the relative positioning measurements of the robots have been reproduced with a software expedient rather then actually implemented with simulated sensors. Each robot publishes its own 2D position X_i and its own identity tag (the name that the robot use in the network) on a topic called */dist_msg_i* which is exchanged between the neighbors and circulates over the simulated network, which had been set up using the ROS single-master architecture. The decentralized controller node subscribes to all */dist_msg_j* topics and collects at each iteration those which have been received according to the current network topology, storing them among the neighbors $\mathcal{N}_i$. The robot and the neighbors desired formation positions p_i and p_j are retrieved from the set $\Pi = \left\{(p_1^\top, \dots, p_N^\top)^\top \mid p_i \in \mathbb{R}^2, \quad i = 1, \cdots, N\right\} \in \mathbb{R}^{2 \cdot N}$ simply knowing the identity tag of the sender. Eventually, the new reference is provided using (3.22). The proposed solution is presented as pseudo-code in Algorithm 1.

Algorithm 1 robot_i/distrib_controller

1: get Π, α_i from Param Server
2: subscribe /dist_msg_j, $\forall j \in \mathcal{V}$
3: subscribe TF $\mathscr{X}_i$
4: **repeat** every $\tau_k > \underline{\tau}^*$
5: $X_i \leftarrow \mathscr{X}_i$
6: $\mathcal{N}_i \leftarrow$ /dist_mess_j
7: $\sigma_i = 0$
8: **for** $j \in \mathcal{N}_i$ **do**
9: $p_j \leftarrow \Pi$, /dist_msg_j
10: $X_j \leftarrow$ /dist_msg_j
11: $\sigma_i \mathrel{+}= \alpha_i(X_j - X_i + p_{ij})$
12: **end for**
13: $r_i = X_i + \sigma_i$
14: publish TF $\mathscr{R}_i$
15: publish /dist_msg_i
16: **until** ROS is down

Algorithm 2 robot_i/local_controller

1: get k, γ from Param Server
2: subscribe TF $\mathscr{R}_i$
3: subscribe TF $\mathscr{X}_i$
4: **repeat** every $\tau' \ll \underline{\tau}^*$
5: $\mathscr{E}_i \leftarrow \mathscr{X}_i, \mathscr{R}_i$
6: /cmd_vel_$i \leftarrow \mathscr{E}_i$
7: publish /cmd_vel_i
8: **until** ROS is down

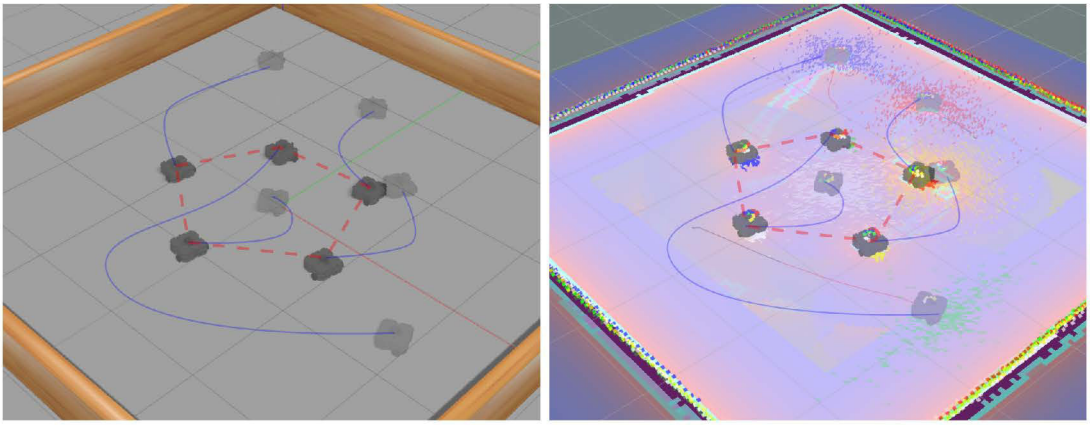
The gain α_i depends on the network topology settings and it is typically chosen in order to fulfil Assumptions 2.2 or 2.4 depending on the case. Note that the decentralized controller iterates once every $\tau_k > \underline{\tau}^*$ in order to fulfil the conditions of Theorems 2.28 or 2.51. In operation 13 we used the cumulative sum of all the neighbors contribution $\sigma_i \in \mathbb{R}^2$ as input to implement (3.22), namely

$$r_i(t_k) = X_i(t_k^-) + \sum_{\substack{j=1 \\ j \neq i}}^{N} P_{i,j}(t_k)(X_j(t_k^-) - X_i(t_k^-) + p_{ij}) = X_i(t_k^-) + \sigma_i(t_k^-), \qquad \text{for } t_k \in \mathcal{T}_i. \tag{4.1}$$

A formation realization simulation has been set up using gazebo, where a set of 5 robotic vehicles need to reach a pentagon formation, using the local controller parameters already discussed in section 3.3. The topology has been simulated as a completely connected undirected graphs. From this information a dwell-time lower bound of $\underline{\tau}^* = 1.7$ seconds has been hard coded in the node implementation. For every robot i four main nodes are executed:

- /robot_i/local_controller,
- /robot_i/decentralized_controller,
- /robot_i/amcl.
- /robot_i/simulated_sensors.

In addition two more important nodes must be run: the map server and the rviz visualization tool. The AMCL node uses the map provided by the map server and the data of a simulated LIDAR sensor to take care of the self localization of the robots (providing $\mathscr{X}_i = \{(X_i^\top, 0)^\top, q_{X_i}\}$ to the TF topics). The scenario is represented in Figure 4.3.



(a) Gazebo view. (b) Rviz view.

Figure 4.3: Pentagon formation realized with ROS implementation in Gazebo simulator and rviz environment.

4.2.2 Implementation for real robotic devices

In order to enable the robotic vehicles to react to the movement of other agents, which are not possibly taking part of the network (think for instance of a worker in a factory) and to move in presence of environmental obstacle, it was mandatory to introduce some sort of collision avoidance and obstacle avoidance policy in our software design.

With regarding to the pose stabilization task, that is the second stage of the proposed decentralized control strategy in Figure 3.3, the objective was achieved by replacing the *local_controller* node with the ROS Navigation Stack Package. The ROS **Navigation Stack** [Marder-Eppstein et al., 2010] is composed by two main nodes, a global planner and a local planner. The navigation stack takes as input a desired goal position given using a particular service call named *move_base* and the map of the environment. In output it provides two planned path, a global one that take takes care of the obstacle avoidance of the fixed mapped obstacle, and a local path to the purpose of avoiding collision with its neighbors or moving obstacle. Finally, the navigation stack provides the */cmd_vel* message necessary for the robot to move on the planned paths.

For what concerns the first step of the two stage approach, in order to ensure collision-free behaviour, we decided to use the **Safety Barrier Certificates** solution as it has been proposed in [Ames et al., 2014] and applied to swarm robotics in [Pickem et al., 2017, Wang et al., 2017]. Considering the same set $\mathcal{V}$ of N AMRs, we denote by $\sigma = (\sigma_1^\top, \dots, \sigma_N^\top)^\top$ and $X = (X_1^\top, \dots, X_N^\top)^\top$ the collection of cumulative inputs and states for dynamics (4.1). To avoid the updated reference r_i and r_j of two robots to be overlapped leading to a potential collision configuration, we impose them to maintain a certain safety distance D_{ij}, namely $\|r_i - r_j\| \geq D_{ij}$. The constraints can be enforced solving in a distributed fashion N Quadratic Programming (QP) problem of the form

$$\sigma_i^* = \underset{\tilde{\sigma}_i}{\operatorname{argmin}} \quad J(\tilde{\sigma}_i) = \|\tilde{\sigma}_i - \sigma_i\| \tag{4.2a}$$

$$\text{s.t.} \quad A_{ij}(X_i, X_j)\tilde{\sigma}_i \leq b_{ij}(X_i, X_j), \qquad \forall j \in \mathcal{N}_i, \tag{4.2b}$$

where matrices $A_{ij}(X_i, X_j) \in \mathbb{R}^{2\times N}$ and $b_{ij}(X_i, X_j) \in \mathbb{R}^2$ encode the inter-agent distance

constraints. The QP problem essentially minimizes the difference between the nominal σ_i and the feasible input $\tilde{\sigma}_i \in \mathbb{R}^2$. The theoretical procedure that is behind the formulation of such QP problem goes behind the scope of the present work. Nevertheless, the interested reader may refer to [Wang et al., 2017] for further details.

The solution proposed in Algorithm 1 is thus modified replacing Operation 14 with a call to the *move_base* service of the navigation stack with pose $(r_i^\top, 0)^\top$. Moreover, Operation 13 is modified using $r_i = X_i + \sigma^*$, i.e. the output of the QP program. The σ^* is computed on line for each iteration using a C/C++ library for QP minimization [Di Gaspero, 2019].

The relative sensing process between the AMRs has been simulated, as previously done for the validation on Gazebo. In order to reproduce the behaviours of a relative positioning sensors, the robots share over the network their *robot_footprint* frame transformation $\mathscr{X}_i = \{(X_i^\top, 0)^\top, q_{X_i}\}$.

4.3 FFLOR fleet management solution

In this section, we analyze the strategies adopted to adapt and implement the decentralized control solution described so far in the framework of the project focused on the CEA FFLOR platform in Trémery. A preliminary study of the application for which the platform was to be built was therefore essential in order to identify the best materials, devices and technologies that could ensure the attainment of the foreshadowed results. In this context, it was decided to implement a deployment application for a fleet of robots, which could realize a formation and then move between obstacles in an industrial environment to the purpose of monitoring, research and collection of data. The application must allow a user connected to a portable device to choose the formation to be realized, check and control its movements and store any data recorded by the agents (thermal data, images, laser scans, etc.).

The designed fleet management system mainly consists of three physical entities: an operator who provides a desired formation shape and who remotely operates the vehicles via a front end Android application; a central control station for monitoring, data collection and analysis purpose and the network of AMR deployed in an unstructured environment. Since the FFLOR facility is equipped with an high bandwidth LAN network and an high reliable central control unit, the solution that has been chosen for the experimental platform is based on a single-master approach. The choice of AMR was made on the basis of the quantity and quality of open source software already available and the ease of integration with sensors of different kinds. The Turtlebot3 (TB3) robotic vehicle [ROBOTIS, 2019] has proven to be a valuable solution for test and experimental purpose. Turtlebot3 is a differential wheels mobile robot with non-holonomic dynamics supplied by the manufacturer with the features resumed in Table 4.1

The final assembled configuration is represented in Figure 4.4. The ROS package *turtlebot bringup* provides the necessary software components to start up the robot and interface it with the embedded sensors and actuators (ROS driver nodes).

The software components described in Section 4.2.2 have been used for the AMRs, equipping the robots with collision and obstacle avoidance capability. One of the

Feature	Model	Details
Chassis		Plate, post, PCB base, ball caster, caster holder.
Motors	3Dynamixel-X series	Current-Based Torque Control (4096 steps, 2.69 mA/step). Built-in encoder.
Embedded board	OpenCR	Open-source Control module, provides low level interfaces to devices.
Computer	Raspberry Pi	Single-board computer for high level task (ROS executions).
LIDAR	360° Laser Distance Sensor LDS-01	120-3,500 mm range, 5% accuracy (500 mm 3,500 mm).
Camera	Raspberry Pi 8 Megapixels	Front Camera.

Table 4.1: Features of the Turtlebot3 robot.

(a) Front view. (b) Rear view.

Figure 4.4: A picture of the assembled robot, complete of the equipment described in Table 4.1.

robots in the network is chosen to be the leader of the fleet, so as to reproduce the situation proposed in Section 3.3.3. The leader AMR does not carry on the formation realization process, whereas it receives the commands directly by an android application operated by the user on a mobile device.

The base station application has been developed with the help of the RQT package. RQT is a Qt-based framework for graphical user interface (GUI) development for ROS [WillowGarage, 2019b]. The desktop application plug-in called *rqt_fflor_server* integrates RVIZ and a data logger and plotter which collects the pose and sensors data coming from the robots. Moreover, the application executes in background the map server and a service server called *fflorsrv*. The latter is in charge of starting up the *decentralized_controller* nodes on the robots. Once an operator selects a shape on the remote mobile application a call to the service *start_formation* is done and the service

server node starts the formation realization node on all the robots, except the leader one.

Finally, the mobile device part of the fleet management system is based on a JAVA application for android which exploits a ROS client node called *android_client*, developed using the ROSJAVA libraries (i.e. the JAVA version of the ROS client libraries) in order to communicate the desired shape of the formation to the base station. Moreover, the same node is also in charge of publishing the velocity command messages for the leader on the corresponding topic.

The software architecture of the management systems can be briefly summarized by the block diagram in Figure 4.5. The performance of the proposed solution have been evaluated in an experiment set up at the FFLOR site in Trémery.

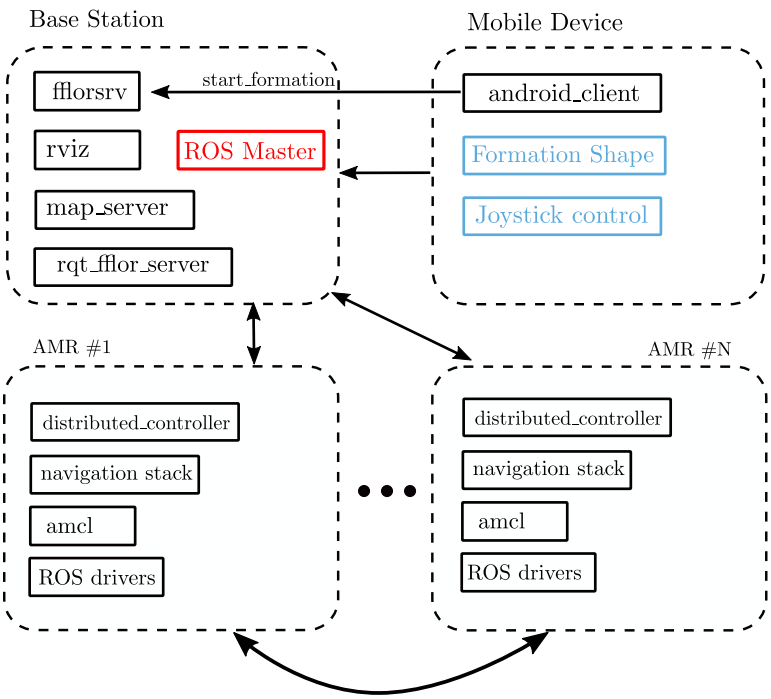
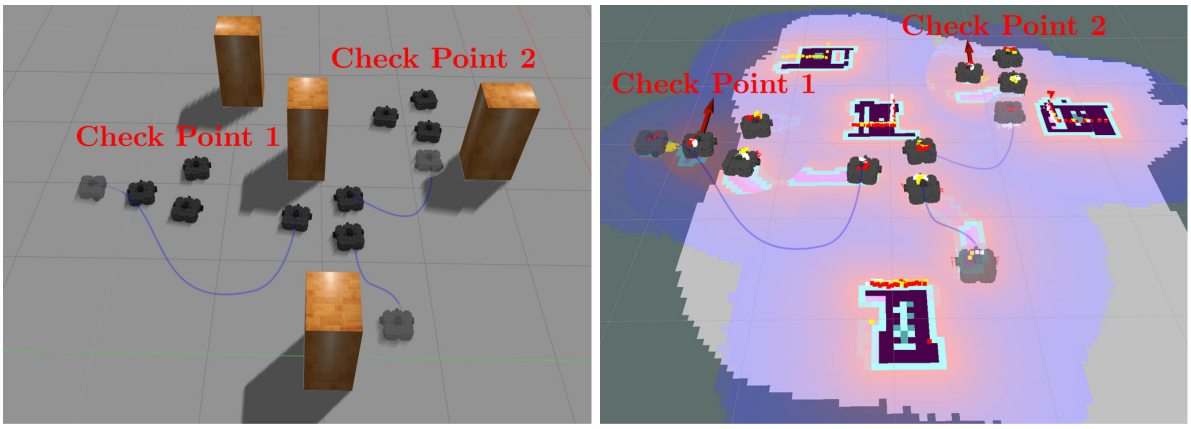


Figure 4.5: The FFLOR fleet management system software. In black are denoted the ROS components, whereas in blue are denoted the JAVA for andorid components of the end user application. The choice of the single-master is highlighted in red in the base station block.

4.3.1 Performance evaluation of the Fleet management System

A typical working scenario in a warehouse was reproduced inside the real facility and in simulated environment using Gazebo. With the use of the remote application, three robot Turtlebot3 connected according to an undirected topology had to realize the triangular formation $\Pi = \{(0.0, 0.0),\ (-0.50, 0.0),\ (-0.25, 0.43)\}$. Once the formation was realized, the fleet was guided along a path through some fixed obstacles reaching two consecutive check-points. The scenario reproduced in the gazebo simulated environment is represented in Figure 4.6, whereas in Figure 4.7 we report the three

configuration obtained with the real platform. The comparison between some performance indicators in the actual implementation and in the simulated environment was collected in Table 4.2. We choose the following performance indicators: the time necessary for the realization of the formation hΔt_1, the time necessary to move the formation to the following check point Δt_{12} and Δt_{23}, the average formation positioning errors $\Delta p = (\Delta p_x, \Delta p_y)^\top$ (which has been evaluated measuring the difference between the desired relatives positions Π and the real ones).



(a) Gazebo view. (b) Rviz view.

Figure 4.6: Simulated reconstruction of the test at the FFLOR site. The AMRs have first to reach the triangular formation. Then the formation is moved using a leader agent to check point 1 and 2. The initial positions are represented in opaque.

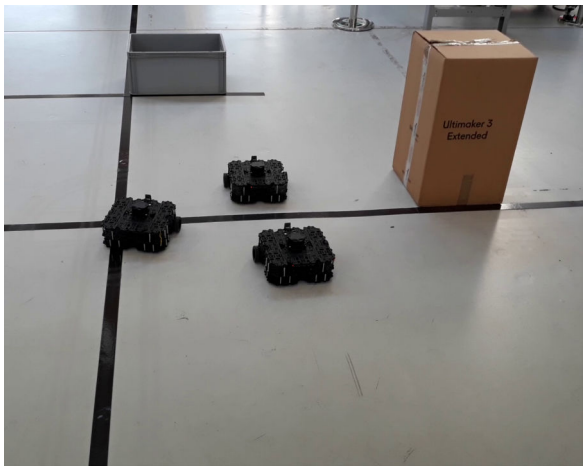
	Δt_1 (s)	Δt_{12} (s)	Δt_{23} (s)	Δp (m)
FFLOR test bench scenario	48	42	51	(0.08, 0.09) $\pm$ 0.03
Gazebo simulated scenario	39	64	59	(0.03, 0.02) $\pm$ 0.02

Table 4.2: Performance indicators comparison between real and simulated scenario.

First of all, it is important to note that the ROS navigation stack runs on both the simulated and the real application with a positioning tolerance of $\pm$ 5 cm, meaning that the final positions are reached by each AMR with this accuracy. Secondly, for what concerns the real test, the slippage of the AMRs wheels caused an accumulation of localization error that the usage of LIDARs sensor is not completely able to fix. This phenomena is to be considered as the main reason of the difference between the positioning errors recorded in the real and the simulated case reported in Table 4.2.

Finally, considering the network aspect, it is worth to notice how the decentralized software approach described in Section 4.2.2 and represented in Figure 4.5 decrease the bandwidth usage. In Figure 4.8, the input-output traffic of the base station wifi interface is represented in the case of a decentralized approach and a fully centralized one (where the robots localization and navigation nodes run on the base station and communicates with the AMR through the network). Even in lack of a multi-master architecture the bandwidth can be considerably reduced thanks to the software decentralization process typical of the ROS implementations.

(a) Formation realization.



(b) First check point.

(c) Second check point.

Figure 4.7: The real experiment scenario represented in its three main phases: when the formation is realized and when the first and the second check point are reached.

4.4 Conclusions

In this chapter we presented the fleet management solution developed in the context of the Future of Factory Lorraine project, in collaboration with the CEA nano-Innov laboratory in Paris-Saclay. The described application meets the requirements of support for monitoring and data collection for indoor use in an industrial environment or for search and rescue purposes. After a brief description of the software technologies used as backbone of the project in Section 4.1, the implementation choices have been introduced in Section 4.2 for what concerned the simulated and the real robotic platform. Finally, in Section 4.3, we analyzed in details the features which characterize the fleet management system, eventually carrying out a performance index comparison between the application with real AMRs at the FFLOR facility and a simulated scenario.

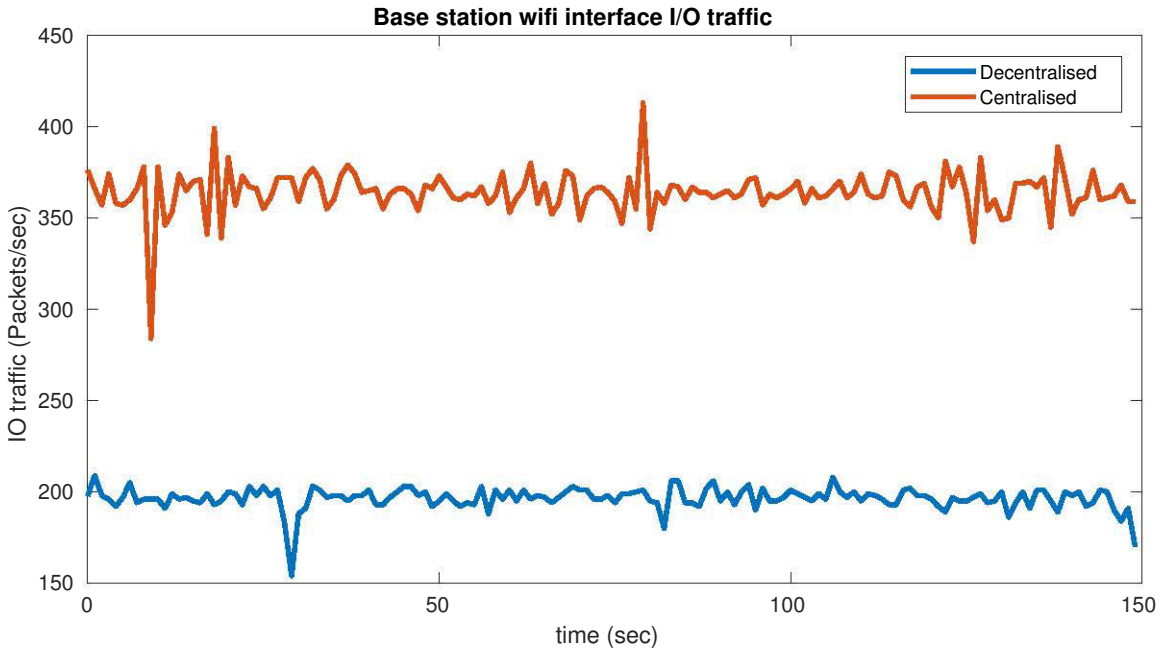


Figure 4.8: The input-output packets traffic of the base station wifi interface. The data have been collected with only one turtlebot and the base station connected to the network. To retrieve the bandwidth usage in the multi robot scenario it is enough to multiply by the number of AMRs in the network.

Chapter 5

Final conclusions

The purpose of the present work was to analyze the agreement behaviour of a class of multi-agent systems taking into account hybrid phenomena that occur when considering sporadic interactions between the agents of the network. The analysis was carried out with particular attention on the topology of the network, with special focus on applications to fleets of robots. This choice of application was motivated by the fact that this work is part of the CEA tech project Future of Factory Lorraine (FFLOR), where innovative digital technologies for industrial automation are developed and tested.

After a brief introduction to the theoretic concept related to this work, we turn our attention to the agreement problem of the piece-wise reference tracking non-linear systems. A two stage approach based on the separation between the decentralized reference agreement task and the local reference tracking task is proposed. The agents update their reference thanks to relative measurements of the state of the neighborhood that are taken in a sporadic manner. The overall system so generated presents hybrid properties that have been considered in the following stability analysis. A sufficient stability condition involving the dwell-time between two updates is proposed in the case of connected undirected topology, switching directed topology and perturbed relative measurements.

Next we applied this framework of analysis, to a network of non-holonomic vehicles. Once the local piece-wise reference tracking dynamic has been shown to fulfil the set of assumptions formulated so far, the two stage approach is applied to a multi-agent system of non-holonomic vehicles for the rendez-vous problem and the formation realization problem.

Finally, we gave an insight of the fleet management system developed for the CEA tech FFLOR platform in Trémery. The system is composed by a remote operator, a base station and a group of Autonomous Mobile Robots to deployed in an industrial hazardous environment.

The results of the work developed in this thesis lead to various possible extensions and new researches. For instance, it could be of interest to try relaxing the exponential convergence assumption and investigate sufficient conditions in case of a less restrictive convergence property of the local tracking dynamics.

Furthermore, the second step of the two stage strategy and the related analysis discussed in Chapter 2 is based on a state piece-wise reference tracking. Inspired by the work of [De Campos et al., 2012] about the agreement of heterogeneous linear

systems and the works about synchronization of non linear heterogeneous systems by [Panteley and Loría, 2017, Isidori et al., 2014] we could think to extend our analysis to the case of the output reference tracking.

An other important aspect which has been overlooked in the theoretic analysis is the impact that an obstacle and collision avoidance policy like the one used in the validation and experimental part of this work (see [Wang et al., 2017] for further details) would have on the stability conditions. These two algorithms may affect the exponential convergence assumption, which is crucial in our theoretical framework. One possible solution, is to consider the additive correction introduced by the collision avoidance algorithm presented in 4.2.2 as a new disturbance and therefore to exploit the ISS properties of the overall dynamics.

Looking at the network properties, it may be interesting to consider the behaviour of the proposed solution with networks whose topology is described by a time varying graph where the nodes are time varying too, namely $\mathcal{G}(t) = (\mathcal{V}(t), \mathcal{E}(t))$. This is typical when one considers multi-agent systems where the agents can be connected or disconnected from the network and it is especially the case for network of robots where battery level or default can compromise the functionality of one of the agent. The other agents should react to those changes according to some pre-determined actions (for instance re-shaping the formation).

Even though the CEA site in Trémery offered an high bandwidth LAN and a reliable base station computer, it would be extremely interesting to extend our implementation to a multi-master configuration in order to achieve a complete decentralization of the software solution. The same results could be reached also changing development framework and using the new released Robotic Operating System 2 (ROS2). In fact, these new client libraries, based on the Data Distributed Service technology, have been specifically designed to be used for multi-agent fully distributed application and eliminate the need to have any ROS master process running in the network.

Appendix A

Instrumental results

For sake of clarity, we recall in the following appendix some mathematical tools and concept used in this work.

Definition A.1 (Positive definite Matrix)**.** *A square symmetric matrix $A \in \mathbb{R}^{n\times n}$ is called positive definite (positive semi-definite) if for all vector $x \in \mathbb{R}^n$, $x^\top Ax$ is strictly positive (non-negative).*

Definition A.2 (Positive Matrix)**.** *A matrix $A \in \mathbb{R}^{n\times m}$ is called positive (non-negative) if all its elements are strictly positive (non-negative), namely $A_{i,j} > 0,\ \forall i,j$ ($A_{i,j} \geq 0,\ \forall i,j$).*

Definition A.3 (Spectral Radius)**.** *Let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of a square matrix $A \in \mathbb{R}^{n\times n}$. Then its spectral radius is defined as*

$$\rho(A) = \max_i |\lambda_i| .$$

Definition A.4 (Kronecker Product)**.** *Given two matrices $A \in \mathbb{R}^{n\times m}$ and $B \in \mathbb{R}^{p\times q}$ the Kronecker product $A \otimes B$ is a $\mathbb{R}^{n\cdot p\times m\cdot q}$ matrix:*

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \ldots & a_{1m}B \\ a_{21}B & a_{22}B & \ldots & a_{2m}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1}B & a_{n2}B & \ldots & a_{nm}B \end{bmatrix}$$

Proposition A.1 ([Lancaster and Farahat, 1972])**.** *Given two matrices $A \in \mathbb{R}^{n\times m}$ and $B \in \mathbb{R}^{p\times q}$ the following holds for the euclidean norm*

$$\|A \otimes B\| = \|A\| \, \|B\|$$

Regarding the last result, we decided to report the result for the euclidean norm, even though in [Lancaster and Farahat, 1972] the Proposition is proven for the more general class of **absolute norms**.

Definition A.5 (Schur Matrix)**.** *A square matrix $A \in \mathbb{R}^{n\times n}$ is called a Schur matrix if the module of all the eigenvalues of A is less then one.*

There are several results in literature which concern Schur matrices. In particular we highlight the following result for non-negative Schur matrices.

Proposition A.2 ([Rantzer, 2011]). *Given a non-negative matrix $A \in \mathbb{R}_+^{n\times n}$, the following statements are equivalent:*

1. *The matrix A is Schur.*
2. *There exists a non-negative vector $z \in \mathbb{R}_+^n$ such that $Az < z$.*
3. *There exists a diagonal positive definite matrix $P \succ 0$ such that $A^\top PA \prec P$.*

Definition A.6 (Vector norm). *Given a vector $x = (x_{(1)}, \dots, x_{(n)}) \in \mathbb{R}^n$ the following norms are defined:*

1. *The Euclidean norm*
$$\|x\| = \left(\sum_{i=1}^{n} \left|x_{(i)}\right|_1^2\right)^{\frac{1}{2}}.$$
2. *The infinity norm*
$$|x| = \max_i \left|x_{(i)}\right|_1.$$
3. *The 1-norm*
$$|x|_1 = \sum_{i=1}^{n} \left|x_{(i)}\right|_1.$$

Definition A.7 (Real Matrix norms). *Given a matrix $A\mathbb{R}^{m\times n}$, the following matrix norms are defined:*

1. *Given a general vector norm that we will denote with $\|\cdot\|$ the corresponding matrix induced norm is*
$$\|A\| = \sup_{\|x\|=1} \frac{\|Ax\|}{\|x\|}$$
2. *The infinity norm*
$$|A| = \max_{1\le i\le m} \sum_{j=1}^{n} \left|A_{(ij)}\right|_1.$$
3. *The 1-norm*
$$|A|_1 = \max_{1\le i\le n} \sum_{i=1}^{m} \left|A_{(ij)}\right|_1.$$
4. *The 2-norm*
$$\|A\| = \sqrt{\rho(A^\top A)}.$$

Proposition A.3 ([Golub and Van Loan, 1996]). *Given a matrix $A \in \mathbb{R}^{n\times m}$ the following holds*
$$\|A\| \le \sqrt{|A|\,|A|_1}.$$

Bibliography

[Aicardi et al., 1995] Aicardi, M., Casalino, G., Bicchi, A., and Balestrino, A. (1995). Closed loop steering of unicycle like vehicles via lyapunov techniques. *IEEE Robotics & Automation Magazine*, 2(1):27–35.

[Ajorlou et al., 2015] Ajorlou, A., Asadi, M. M., Aghdam, A. G., and Blouin, S. (2015). Distributed consensus control of unicycle agents in the presence of external disturbances. *Systems & Control Letters*, 82:86–90.

[Amazon Robotics, 2019] Amazon Robotics (2019). Warehouse automation systems. https://www.amazonrobotics.com/#/. [Online; accessed 29-April-2019].

[Ames et al., 2014] Ames, A. D., Grizzle, J. W., and Tabuada, P. (2014). Control barrier function based quadratic programs with application to adaptive cruise control. In *53rd IEEE Conference on Decision and Control*, pages 6271–6278.

[Astolfi, 1999] Astolfi, A. (1999). Exponential stabilization of a wheeled mobile robot via discontinuous control. *Transactions-American Society of Mechanical Engineers Journal of Dynamic Systems, Measurement, and Control*, 121:121–125.

[Beard and Stepanyan, 2003] Beard, R. and Stepanyan, V. (2003). Information consensus in distributed multiple vehicle coordinated control. In *Proceedings of the 42th IEEE Conference on Decision and Control*, volume 2, pages 2029–2034.

[Bertsekas and Tsitsiklis, 1989] Bertsekas, D. P. and Tsitsiklis, J. N. (1989). *Parallel and distributed computation: numerical methods*, volume 23. Prentice hall Englewood Cliffs, NJ.

[Blondel et al., 2005] Blondel, V. D., Hendrickx, J. M., Olshevsky, A., and Tsitsiklis, J. (2005). Convergence in multiagent coordination, consensus, and flocking. In *Proc. IEEE Conf. on Decision and Control*, pages 2996–3000.

[Borzone et al., 2018a] Borzone, T., Morărescu, I.-C., Jungers, M., Boc, M., and Janneteau, C. (2018a). Hybrid framework for consensus in directed and asynchronous network of non-holonomic agents. *IEEE Control Systems Letters*, 2(4):707–712.

[Borzone et al., 2019] Borzone, T., Morărescu, I.-C., Jungers, M., Boc, M., and Janneteau, C. (2019). Hybrid formalism for consensus of a general class of multi-agent systems with biased measurements. In *2019 European Control Conference (ECC '19)*.

[Borzone et al., 2018b] Borzone, T., Morărescu, I.-C., Jungers, M., Boc, M., and Janneteau, C. (2018b). Hybrid framework for consensus in fleets of non-holonomic robots. In *Proceedings of American Control Conference*, pages 4299–4304.

[Brockett, 1983] Brockett, R. (1983). Asymptotic stability and feedback stabilization. *In Differential Geometric Control Theory*, (R.W. Brockett and R.S. Millman and H.J. Sussmann Eds):181–191.

[Bru et al., 1988] Bru, R., Elsner, L., and Neumann, M. (1988). Models of parallel chaotic iteration methods. *Linear Algebra and its Applications*, 103:175–192.

[Buşoniu and Morărescu, 2014] Buşoniu, L. and Morărescu, I.-C. (2014). Consensus for black-box nonlinear agents using optimistic optimization. *Automatica*, 50(4):1201–1208.

[Bullo et al., 2009] Bullo, F., Cortés, J., and Martinez, S. (2009). *Distributed Control of Robotic Networks. A Mathematical Approach to Motion Coordination Algorithms*. Princeton University Press.

[Cai and Teel, 2009] Cai, C. and Teel, A. R. (2009). Characterizations of input-to-state stability for hybrid systems. *Systems & Control Letters*, 58(1):47–53.

[Campion et al., 1993] Campion, G., Bastin, G., and D'Andrea-Novel, B. (1993). Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. In *[1993] Proceedings IEEE International Conference on Robotics and Automation*, pages 462–469.

[Canudas de Wit et al., 1993] Canudas de Wit, C., Khennouf, H., Samson, C., and Sordalen, O. J. (1993). Nonlinear control design for mobile robots. In *Recent trends in mobile robots*, pages 121–156. World Scientific.

[Canudas de Wit and Sordalen, 1992] Canudas de Wit, C. and Sordalen, O. J. (1992). Exponential stabilization of mobile robots with nonholonomic constraints. *IEEE Transactions on Automatic Control*, 37(11):1791–1797.

[Cao et al., 2013] Cao, Y., Yu, W., Ren, W., and Chen, G. (2013). An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Transactions on Industrial informatics*, 9(1):427–438.

[Carli et al., 2009] Carli, R., Como, G., Frasca, P., and Garin, F. (2009). Average consensus on digital noisy networks. *IFAC Proceedings Volumes*, 42(20):36–41.

[Casadei and Astolfi, 2018] Casadei, G. and Astolfi, D. (2018). Multipattern output consensus in networks of heterogeneous nonlinear agents with uncertain leader: A nonlinear regression approach. *IEEE Transactions on Automatic Control*, 63(8):2581–2587.

[CEA tech, 2019] CEA tech (2019). Fflor — Future of Factory Lorraine. http://www.cea-tech.fr/cea-tech/Pages/en-regions/pfa-usine-du-futur.aspx. [Online; accessed 29-April-2019].

[Coffee et al., 2003] Coffee, J. R., Rudow, R. W., Allen, R. F., Billings, M., Dye, D. A., Kirchner, M. L., Lewis, R. W., Marvin, K. M., Sleeper, R. D., Tekniepe, W. A., et al. (2003). Vehicle tracking, communication and fleet management system. US Patent 6,611,755.

[Cortés, 2006] Cortés, J. (2006). Finite-time convergent gradient flows with applications to network consensus. *Automatica*, 42(11):1993–2000.

[De Campos et al., 2012] De Campos, G. R., Brinón-Arranz, L., Seuret, A., and Niculescu, S.-I. (2012). On the consensus of heterogeneous multi-agent systems: a decoupling approach. *IFAC Proceedings Volumes*, 45(26):246–251.

[Defoort et al., 2008] Defoort, M., Floquet, T., Kokosy, A., and Perruquetti, W. (2008). Sliding-mode formation control for cooperative autonomous mobile robots. *IEEE Transactions on Industrial Electronics*, 55(11):3944–3953.

[Di Gaspero, 2019] Di Gaspero (2019). Quadprog++ — A C++ library for Quadratic Programming. https://github.com/liuq/QuadProgpp. [Online; accessed 29-April-2019].

[Di Paola et al., 2010] Di Paola, D., Milella, A., Cicirelli, G., and Distante, A. (2010). An autonomous mobile robotic system for surveillance of indoor environments. *International Journal of Advanced Robotic Systems*, 7(1):8.

[Dimarogonas and Kyriakopoulos, 2007] Dimarogonas, D. V. and Kyriakopoulos, K. J. (2007). On the rendezvous problem for multiple nonholonomic agents. *IEEE Transactions on Automatic Control*, 52(5):916–922.

[Dong and Farrell, 2008] Dong, W. and Farrell, J. A. (2008). Consensus of multiple nonholonomic systems. In *2008 47th IEEE Conference on Decision and Control*, pages 2270–2275.

[Floquet et al., 2003] Floquet, T., Barbot, J.-P., and Perruquetti, W. (2003). Higher-order sliding mode stabilization for a class of nonholonomic perturbed systems. *Automatica*, 39(6):1077–1083.

[Fossen, 1994] Fossen, T. I. (1994). *Guidance and control of ocean vehicles*. John Wiley & Sons Inc.

[Fox et al., 1999] Fox, D., Burgard, W., Dellaert, F., and Thrun, S. (1999). Monte carlo localization: Efficient position estimation for mobile robots. *AAAI/IAAI*, 1999(343-349):2–2.

[Garin and Schenato, 2011] Garin, F. and Schenato, L. (2011). Distributed estimation and control applications using linear consensus algorithms. In Bemporad, A., Heemels, M., and Johansson, M., editors, *Networked control systems*, chapter 3. Springer.

[Ghabcheloo et al., 2009] Ghabcheloo, R., Aguiar, A. P., Pascoal, A., Silvestre, C., Kaminer, I., and Hespanha, J. (2009). Coordinated path-following in the presence of communication losses and time delays. *SIAM journal on control and optimization*, 48(1):234–265.

[Godsil and Royle, 2013] Godsil, C. and Royle, G. F. (2013). *Algebraic graph theory*, volume 207. Springer Science & Business Media.

[Goebel et al., 2012] Goebel, R., Sanfelice, R., and Teel, A. (2012). *Hybrid Dynamical Systems*. Princeton University Press.

[Golub and Van Loan, 1996] Golub, G. H. and Van Loan, C. F. (1996). *Matrix Computations*. The Johns Hopkins University Press, third edition.

[Grisetti et al., 2007] Grisetti, G., Stachniss, C., Burgard, W., et al. (2007). Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE transactions on Robotics*, 23(1):34.

[Hetel et al., 2013] Hetel, L., Daafouz, J., Tarbouriech, S., and Prieur, C. (2013). Stabilization of linear impulsive systems through a nearly-periodic reset. *Nonlinear Analysis: Hybrid Systems*, 7(1):4–15.

[Huang and Manton, 2009] Huang, M. and Manton, J. H. (2009). Coordination and consensus of networked agents with noisy measurements: stochastic algorithms and asymptotic behavior. *SIAM Journal on Control and Optimization*, 48(1):134–161.

[Isidori et al., 2014] Isidori, A., Marconi, L., and Casadei, G. (2014). Robust output synchronization of a network of heterogeneous nonlinear agents via nonlinear regulation theory. *IEEE Transactions on Automatic Control*, 59(10):2680–2691.

[Jadbabaie et al., 2003] Jadbabaie, A., Lin, J., and Morse, A. S. (2003). Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Trans. on Automatic Control*, 48(6):988–1001.

[Jiang and Nijmeijier, 1997] Jiang, Z.-P. and Nijmeijier, H. (1997). Tracking control of mobile robots: A case study in backstepping. *Automatica*, 33(7):1393–1399.

[Jiang and Wang, 2001] Jiang, Z.-P. and Wang, Y. (2001). Input-to-state stability for discrete-time nonlinear systems. *Automatica*, 37(6):857–869.

[Kanayama et al., 1990] Kanayama, Y., Kimura, Y., Miyazaki, F., and Noguchi, T. (1990). A stable tracking control method for an autonomous mobile robot. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 384–389.

[Khalil, 2001] Khalil, H. (2001). *Nonlinear Systems (Third Edition)*. Prentice Hall.

[Koenig and Howard, 2004] Koenig, N. and Howard, A. (2004). Design and use paradigms for gazebo, an open-source multi-robot simulator. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, volume 3, pages 2149–2154.

[Lancaster and Farahat, 1972] Lancaster, P. and Farahat, H. (1972). Norms on direct sums and tensor products. *mathematics of computation*, 26(118):401–414.

[Léchevin et al., 2006] Léchevin, N., Rabbath, C. A., and Sicard, P. (2006). Trajectory tracking of leader–follower formations characterized by constant line-of-sight angles. *Automatica*, 42(12):2131–2141.

[Liberzon et al., 2014] Liberzon, D., Nešić, D., and Teel, A. R. (2014). Lyapunov-based small-gain theorems for hybrid systems. *IEEE Transactions on Automatic Control*, 59(6):1395–1410.

[Liu et al., 2013] Liu, K., Xie, G., Ren, W., and Wang, L. (2013). Consensus for multi-agent systems with inherent nonlinear dynamics under directed topologies. *Systems & Control Letters*, 62(2):152–162.

[Maghenem et al., 2019] Maghenem, M., Bautista-Castillo, A., Nuño, E., Loría, A., and Panteley, E. (2019). Consensus of multi-agent systems with nonholonomic restrictions via lyapunov's direct method. *IEEE Control Systems Letters*, 3(2):344–349.

[Mahony et al., 2012] Mahony, R., Kumar, V., and Corke, P. (2012). Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *IEEE robotics & automation magazine*, 19(3):20–32.

[Marder-Eppstein et al., 2010] Marder-Eppstein, E., Berger, E., Foote, T., Gerkey, B., and Konolige, K. (2010). The office marathon: Robust navigation in an indoor office environment. In *International Conference on Robotics and Automation*.

[Mesbahi and Egerstedt, 2010] Mesbahi, M. and Egerstedt, M. (2010). *Graph theoretic methods in multiagent networks*. Princeton University Press,Princeton. NY.

[Mohanarajah et al., 2015] Mohanarajah, G., Hunziker, D., D'Andrea, R., and Waibel, M. (2015). Rapyuta: A cloud robotics platform. *IEEE Transactions on Automation Science and Engineering*, 12(2):481–493.

[Morărescu et al., 2015] Morărescu, I.-C., Martin, S., Girard, A., and Muller-Gueudin, A. (2015). Coordination in networks of linear impulsive agents. *IEEE Transactions on Automatic Control*, 61(9):2402–2415.

[Moreau, 2005] Moreau, L. (2005). Stability of multiagent systems with time-dependent communication links. *IEEE Trans. on Automatic Control*, 50(2):169–182.

[Morin and Samson, 2009] Morin, P. and Samson, C. (2009). Control of nonholonomic mobile robots based on the transverse function approach. *IEEE Transactions on robotics*, 25(5):1058–1073.

[Nedic et al., 2009] Nedic, A., Olshevsky, A., Ozdaglar, A., and Tsitsiklis, J. N. (2009). On distributed averaging algorithms and quantization effects. *IEEE Transactions on Automatic Control*, 54(11):2506–2517.

[Nourbakhsh et al., 2005] Nourbakhsh, I. R., Sycara, K., Koes, M., Yong, M., Lewis, M., and Burion, S. (2005). Human-robot teaming for search and rescue. *IEEE Pervasive Computing*, 4(1):72–79.

[Olfati-Saber and Murray, 2004] Olfati-Saber, R. and Murray, R. (2004). Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49:1520–1533.

[Panteley et al., 1998] Panteley, E., Lefeber, E., Loria, A., and Nijmeijer, H. (1998). Exponential tracking control of a mobile car using a cascaded approach. *IFAC Proceedings Volumes*, 31(27):201–206.

[Panteley and Loría, 2017] Panteley, E. and Loría, A. (2017). Synchronization and dynamic consensus of heterogeneous networked systems. *IEEE Transactions on Automatic Control*, 62(8):3758–3773.

[Peng et al., 2015] Peng, Z., Wen, G., Rahmani, A., and Yu, Y. (2015). Distributed consensus-based formation control for multiple nonholonomic mobile robots with a specified reference trajectory. *International Journal of Systems Science*, 46(8):1447–1457.

[Pickem et al., 2017] Pickem, D., Glotfelter, P., Wang, L., Mote, M., Ames, A., Feron, E., and Egerstedt, M. (2017). The robotarium: A remotely accessible swarm robotics research testbed. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1699–1706.

[Quigley et al., 2009] Quigley, M., Faust, J., Foote, T., and Leibs, J. (2009). Ros: an open-source robot operating system. In *IEEE International Conference on Robotics and Automation, Workshop on Open Source Software*.

[Rantzer, 2011] Rantzer, A. (2011). Distributed control of positive systems. In *Proc. of the 50th IEEE Decision and Control and European Control Conference*, pages 6608–6611.

[Reif et al., 1998] Reif, K., Sonnemann, F., and Unbehauen, R. (1998). An EKF-based nonlinear observer with a prescribed degree of stability. *Automatica*, 34(9):1119–1123.

[Ren and Beard, 2005] Ren, W. and Beard, R. W. (2005). Consensus seeking in multiagent systems under dynamically changing interaction topologies. *IEEE Trans. on Automatic Control*, 50(5):655–661.

[ROBOTIS, 2019] ROBOTIS (2019). Turtlebot3 — e-Manual. http://emanual.robotis.com/docs/en/platform/turtlebot3/overview/. [Online; accessed 29-April-2019].

[Roldán et al., 2015] Roldán, J., Joossen, G., Sanz, D., del Cerro, J., and Barrientos, A. (2015). Mini-uav based sensory system for measuring environmental variables in greenhouses. *Sensors*, 15(2):3334–3350.

[Samson and Ait-Abderrahim, 1991] Samson, C. and Ait-Abderrahim, K. (1991). Feedback control of a nonholonomic wheeled cart in cartesian space. In *in Proc. IEEE International Conference on Robotics and Automation*, pages 1136–1141.

[Seuret et al., 2008] Seuret, A., Dimarogonas, D. V., and Johansson, K. H. (2008). Consensus under communication delays. In *2008 47th IEEE Conference on Decision and Control*, pages 4922–4927.

[Siciliano et al., 2010] Siciliano, B., Sciavicco, L., Villani, L., and Oriolo, G. (2010). *Robotics: modelling, planning and control*. Springer Science & Business Media.

[Siegwart et al., 2011] Siegwart, R., Nourbakhsh, I. R., Scaramuzza, D., and Arkin, R. C. (2011). *Introduction to autonomous mobile robots*. MIT press.

[Soares et al., 2012] Soares, J. M., Aguiar, A. P., Pascoal, A. M., and Gallieri, M. (2012). Triangular formation control using range measurements: An application to marine robotic vehicles. *IFAC Proceedings Volumes*, 45(5):112–117.

[Thrun et al., 2005] Thrun, S., Burgard, W., and Fox, D. (2005). *Probabilistic robotics*. MIT press.

[Tsitsiklis, 1984] Tsitsiklis, J. N. (1984). Problems in decentralized decision making and computation. Technical report, Massachusetts Inst of Tech Cambridge Lab for Information and Decision Systems.

[Vicsek et al., 1995] Vicsek, T., Czirók, A., Ben-Jacob, E., Cohen, I., and Shochet, O. (1995). Novel type of phase transition in a system of self-driven particles. *Physical review letters*, 75(6):1226.

[Wang et al., 2017] Wang, L., Ames, A. D., and Egerstedt, M. (2017). Safety barrier certificates for collisions-free multirobot systems. *IEEE Transactions on Robotics*, 33(3):661–674.

[Wang et al., 2015] Wang, Y., Miao, Z., Zhong, H., and Pan, Q. (2015). Simultaneous stabilization and tracking of nonholonomic mobile robots: A lyapunov-based approach. *IEEE Transactions on Control Systems Technology*, 23(4):1440–1450.

[Wieland et al., 2011] Wieland, P., Sepulchre, R., and Allgöwer, F. (2011). An internal model principle is necessary and sufficient for linear output synchronization. *Automatica*, 47:1068–1074.

[WillowGarage, 2019a] WillowGarage (2019a). Gazebo — Robot Simulation Made Easy. http://gazebosim.org/. [Online; accessed 29-April-2019].

[WillowGarage, 2019b] WillowGarage (2019b). Qt-based framework for gui development. http://wiki.ros.org/rqt. [Online; accessed 29-April-2019].

[WillowGarage, 2019c] WillowGarage (2019c). Robot operating system — Wiki. http://wiki.ros.org/. [Online; accessed 29-April-2019].

[Wurman et al., 2008] Wurman, P. R., D'Andrea, R., and Mountz, M. (2008). Coordinating hundreds of cooperative, autonomous vehicles in warehouses. *AI magazine*, 29(1):9–9.

[Xiao et al., 2007] Xiao, L., Boyd, S., and Kim, S.-J. (2007). Distributed average consensus with least-mean-square deviation. *Journal of parallel and distributed computing*, 67(1):33–46.

[Zhong Ping, 2000] Zhong Ping, J. (2000). Robust exponential regulation of nonholonomic systems with uncertainties. *Automatica*, 36(2):189 – 209.

[Zhu et al., 2016] Zhu, L., Chen, Z., and Middleton, R. H. (2016). A general framework for robust output synchronization of heterogeneous nonlinear networked systems. *IEEE Transactions on Automatic Control*, 61(8):2092–2107.

Abstract : Over the last years, multi-agents problems have been extensively studied in the control theory community. One of the most popular multi-agents control topics is the consensus problem where a group of agents reaches an agreement over the value of a certain parameter or variable.

In this work we focus our attention on the consensus problem of networks of non-linear reference tracking agents. In first place, we use sporadic interactions modeled by relative sensing to deal with the decentralized consensus of the references. The reference is therefore fed the tracking dynamics of each agent. Differently from existent works, the stability analysis of the overall system required the usage of hybrid systems theory tools, due to dual nature of the two stages approach. The analysis is carried out considering different scenarios of network topology and interactions. For each case a sufficient condition for the stability is provided in terms of the minimum allowed time between two consecutive reference updates.

The proposed framework is applied to the rendez-vous and formation realisation tasks for non-holonomic mobile robots, which appear among the richest research topics in recent years. The same problem is addressed in the context of a real field application, namely a fleet management system for a group of robotic vehicles deployable in an industrial environment for monitoring and data collection purpose. The development of such application was motivated by the fact that this thesis is part of the Future of Factory Lorraine (FFLOR) project, developed by the technological research department of the *Commissariat à l'énergie atomique et aux énergies alternatives* (CEA tech).

Keywords : Multi-agent systems, non-holonomic systems, consensus, reference tracking systems, differential wheels mobile robots, fleet management systems.

Résumé : Au cours des dernières années, les problèmes multi-agents ont été étudiés de manière intensive par la communauté de la théorie du contrôle. L'un des sujets de contrôle multi-agents les plus populaires est le problème de consensus où un groupe d'agents parvient à un accord sur la valeur d'un certain paramètre ou d'une variable.

Dans ce travail, nous concentrons notre attention sur le consensus des réseaux d'agents avec une dynamique non linéaire de poursuite de référence. En premier lieu, nous utilisons des interactions sporadiques modélisées par la détection relative, pour traiter le consensus décentralisé des références. La référence est donc utilisée pour alimenter la dynamique de poursuite de chaque agent. Contrairement aux travaux existants, l'analyse de stabilité du système globale a nécessité l'utilisation d'outils théoriques propre de la théorie des systèmes hybrides, en raison de la double nature de l'approche en deux étapes. L'analyse est effectuée en tenant compte de différents scénarios de topologie et interactions. Pour chaque cas, une condition suffisante de stabilité est fournie, en termes de temps minimum autorisé entre deux mises à jour de référence consécutives.

Le cadre proposé est appliqué aux missions de rendez-vous et de réalisation de formation pour les robots mobiles non-holonomes, qui figurent parmi les thèmes de recherche les plus riches de ces dernières années. Le même problème est abordé dans le contexte d'une application réelle sur le terrain, à savoir un système de gestion de flotte pour un groupe de véhicules robotisés pouvant être déployés dans un environnement industriel à des fins de surveillance et de collecte de données. Le développement d'une telle application a été motivé par le fait que cette thèse s'inscrit dans le cadre du projet Future of Factory Lorraine (FFLOR), développé par le département de recherche technologique du *Commissariat à l'énergie atomique et aux énergies alternatives* (CEA tech).

Mots clés : Systèmes multi-agents, systèmes non-holonomes, consensus, systèmes à poursuite de référence, système de gestion de flotte.