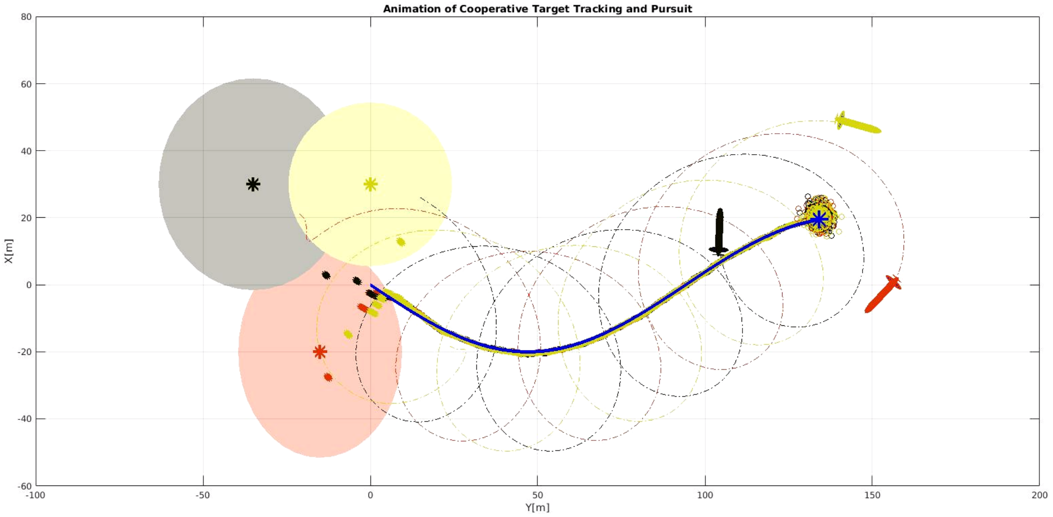
UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO



Cooperative Control and Estimation of Multi-Agent Systems under Communication Constraints

Nguyen Tuan Hung

Supervisor: Doctor António Manuel dos Santos Pascoal

Thesis approved in public session to obtain the PhD Degree in Electrical and Computer Engineering

Jury final classification: Pass with Distinction

2021

UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Cooperative Control and Estimation of Multi-Agent Systems under Communication Constraints

Nguyen Tuan Hung

Supervisor: Doctor António Manuel dos Santos Pascoal

Thesis approved in public session to obtain the PhD Degree in Electrical and Computer Engineering

Jury final classification: Pass with Distinction

Jury:

Chairperson: Doctor João Manuel Lage de Miranda Lemos, Instituto Superior Técnico, Universidade de Lisboa

Members of the Committee:

Doctor Tor Arne Johansen, Faculty of Information Technology and Electrical Engineering, Norwegian University of Science and Technology, Noruega

Doctor Fernando Manuel Ferreira Lobo Pereira, Faculdade de Engenharia, Universidade do Porto

Doctor António Manuel dos Santos Pascoal, Instituto Superior Técnico, Universidade de Lisboa

Doctor João Manuel de Freitas Xavier, Instituto Superior Técnico, Universidade de Lisboa

Funding Institution
Marine UAS project under the Marie Curie Sklodowska grant agreement No 642153 and Bolsa de Investigação (IST)

2021

Acknowledgments

Throughout my PhD journey I have received a great deal of support and assistance.

I am indebted to my supervisor, Professor Antonio Pascoal, for bringing me to IST and being my PhD advisor, teacher, colleague, and friend over the last five years. Your knowledge, experience, and insight in the area of control and robotics was invaluable feedback and pushed me to sharpen my thinking and brought my work to a higher level. I really appreciate for your care and help during writing the thesis and papers for publication. Without your help it would not be possible for me to publish such number of papers and to have this quality of the thesis.

I am also grateful to the my thesis committee members - not only for their time, but for their critical comments and suggestions to improve the quality of the thesis. Among them, I would like to say special thank to Professor Tor Johansen for hosting me during my visit to your group at NTNU, Norway, and for your collaboration in developing and publishing the work. To Professor Xavier, I would say you are the best teacher I ever had. Thank you very much for giving me two wonderful courses in optimization and network science that is essential to build my background for the research work. More than that, I learned so much from you - not only your teaching style but your insight and passion in math and science.

I would also like to thank the members of DSOR group for their help and friendship over the years. Among them I would like to single out Francisco Rego, Joao Quintas, Naveen Crasta, Joao Cruz, Luís Sebastião, Joao Botelho, Manuel Rufino, Miguel Ribeiro, Henrique Ribeiro Jorge Ribeiro, Bahareh Sabetghadam, and Helena Santana.

I would like to send my gratitude to the Marie Sklodowska-Curie Actions funded by the European commission under the scope of MarineUAS project and the Bolsa de Investigação funded by the IST for their generous financial support.

Finally, but not least, I want to thank my family members. To my parents and two younger sisters, thank you very much for always believing in me and your unconditional love. I am also so happy that at the end of the PhD journey I had a nice “joint publication” with my wife, an angel of my family called “Mr. Bill”. To my wife and Bill, thank you so much for being with me to share our happiness at every moment of life.

To each and every one of you – Thank you.

Abstract

This thesis addresses problems in the field of cooperative control and estimation of multi-agent systems (MAS) under stringent communication constraints that include: i) Consensus-synchronization of multi general nonlinear agent system, ii) Cooperative path following (CPF) of multiple constrained autonomous vehicles, and iii) and Range-based cooperative simultaneous target localization and pursuit (SLAP) with multiple autonomous vehicles. Due to the communication constraints imposed by the topology of the inter-agent communication network, where each agent (vehicle) is only capable of exchanging information with a subset of agents in the network, the problems must be addressed in a distributed manner.

To address the first problem, we propose a distributed control strategy with an event-triggered communication (ETC) framework that aims at reducing communications among the agents. The framework possess two important properties: i) practical consensus stabilization, that is, the synchronization error that measures the disagreement among agents' states converges to a ball centered at the origin, with a radius that can be made arbitrarily small and ii) the minimum of the inter-event times for each agent is strictly positive, hence Zeno behavior is excluded.

The CPF problem is defined as steering a group of constrained autonomous vehicles along given spatial paths while holding a desired inter-vehicle formation pattern. The solution proposed involves decoupling the original CPF problem into two sub-problems: i) single path following of input-constrained vehicles and ii) coordination of an input-constrained multi-agent system (MAS). The first is solved by adopting a sampled-data model predictive control (MPC) scheme, whereas the latter is tackled using a novel distributed control law with an (ETC) mechanism. The proposed strategy yields a closed-loop CPF system that is input-to-state-stable (ISS) with respect to the system's state (consisting of the path following error of all vehicles and their coordination errors) and the system's input, which includes triggering thresholds for ETC communications and communication delays.

In order to solve the problem of cooperative target localization and pursuit, we first derive conditions on the motion of the vehicles, called as trackers, under which the target's state is observable. Next, adopting the Fisher information matrix (FIM), a tool to measure the information carried out by range measurements in an estimation-theoretical framework, sufficient conditions on the trackers' motions are derived for the ideal relative geometry between the trackers and the targets for which the range information acquired for estimating the target's state is maximal. We then exploit the aforementioned conditions to plan the motion for the trackers. In this context, each tracker is assigned a curve to track, referred to as a S-T curve, because it admits a hybrid spatial-temporal parametrization. It is guaranteed that if the trackers track the S-T curves then target's state is not only observable but also the range-information acquired for estimating the target's state is maximal. We then propose a distributed estimation and control (DEC) strategy to address the constraints on the communication network among the trackers. For this purpose, a distributed extended Kalman filter (DEKF) is adopted for cooperative estimation of the target's state and a distributed consensus control strategy is adopted for cooperative pursuit of the target. The latter strategy aims at driving all trackers to a desired vicinity of the target while holding an optimal target-trackers relative-geometry that maximizes the range information for estimating the target's state. Finally, we present an event triggered mechanism for the proposed DEC mechanism where the communications among the trackers only take place when found necessary, making the proposed method more efficient for practical implementation.

Keywords

Multi-agent system; Distributed control; Distributed estimation; Event-triggered communications; Synchronization; Consensus; Cooperative path following; Range-based cooperative navigation and target tracking; Fisher information matrix; Model predictive control

Resumo

Esta tese aborda problemas no campo do de controlo cooperativo e estimação de estados de sistemas multiagentes (MAS) sob estritas restrições de comunicação que incluem: i) Sincronização de consenso de sistemas gerais não lineares multiagentes, ii) Seguimento cooperativo de caminhos (CPF) de vários veículos autónomos com constrangimentos, e iii) localização e perseguição simultânea cooperativa baseada em distâncias (SLAP) com vários veículos autónomos veículos. Devido às restrições de comunicação impostas pela topologia da rede de comunicação interagente, onde cada agente (veículo) só é capaz de trocar informações com um subconjunto de agentes na rede, os problemas devem ser tratados de um forma distribuída.

Para resolver o primeiro problema, propomos uma estratégia de controlo distribuído com uma estrutura de comunicação desencadeada por eventos (ETC) que visa reduzir as comunicações entre os agentes. Esta abordagem possui duas propriedades importantes: i) estabilização de consenso prática, ou seja, o erro de sincronização que mede a discordância entre estados dos agentes converge para uma bola centrada na origem, com um raio que pode ser arbitrariamente pequeno e ii) o intervalo de tempo mínimo eventos para cada agente é estritamente positivo, excluindo portanto o comportamento de Zenão.

O problema de CPF é definido como o problema de dirigir um grupo de veículos autónomos com constrangimentos ao longo determinados caminhos espaciais enquanto mantém um padrão de formação interveículo desejado. A solução proposta envolve o desacoplamento do problema original de CPF em dois subproblemas: i) seguimento de caminho de um único veículo com restrições de entrada e ii) coordenação de um sistema multiagente com restrições de entrada (MAS). O primeiro é resolvido pela adoção de um esquema de controlo preditivo com amostragem de dados (MPC), enquanto o último é resolvido usando uma nova lei de controlo distribuído com um mecanismo de comunicações desencadeadas por eventos (ETC). A estratégia proposta produz um sistema CPF de malha fechada que é estável da entrada para do estado (ISS) em relação ao estado do sistema (que consiste nos erros de seguimento de caminho todos os veículos e seus erros de coordenação) e a entrada do sistema, que inclui os limites de desencadeamento de comunicações ETC e

atrasos de comunicação.
Para resolver o problema de localização e perseguição cooperativa de alvos, primeiro derivamos as condições sobre o movimento dos veículos, chamados de rastreadores, sob as quais o estado do alvo é observável. Em seguida, adotando a matriz de informação de Fisher (FIM), uma ferramenta para medir a quantidade de informação obtida por medições de distância sob a perspectiva da teoria de estimação são derivadas condições suficientes dos movimentos dos rastreadores no que diz respeito à geometria relativa ideal entre os rastreadores e os alvos para os quais a informação de distâncias adquirida para estimar o estado do alvo é máxima. Em seguida, exploramos as condições acima mencionadas para o planeamento de movimento para os rastreadores. Neste contexto, cada rastreador recebe uma curva para seguir, referida como curva S-T, porque admite uma parametrização híbrida espaço-temporal. É garantido que se os rastreadores seguirem as curvas S-T, então o estado do alvo é não apenas observável, mas também a informação de distâncias adquirida para estimar o estado do alvo é máxima. Em seguida, propomos uma estratégia de estimação e controlo distribuído (DEC) para abordar as restrições na rede de comunicação entre os rastreadores. Para este propósito, um filtro estendido de Kalman distribuído (DEKF) é adotado para a estimação cooperativa do estado do alvo e uma estratégia de controle de consenso distribuída é adotada para a perseguição cooperativa do alvo. A última estratégia visa direcionar todos os rastreadores para uma vizinhança desejada do alvo, enquanto mantém uma geometria relativa ótima de rastreadores que maximiza a informação de distâncias para estimar o estado do alvo. Finalmente, apresentamos um mecanismo desencadeado por eventos para o mecanismo DEC proposto onde as comunicações entre os rastreadores só ocorrem quando são necessárias, tornando o método proposto mais eficiente para uma implementação prática.

Palavras Chave

Sistemas multiagente; Controlo distribuído; Estimação distribuída; Comunicações desencadeadas por eventos; Sincronização; Consenso; Seguimento de caminho cooperativo; Navegação e seguimento de alvo cooperativo baseado em medições de distâncias; Matriz de informação de Fisher; Controlo preditivo

Contents

List of Figures

List of Tables

List of Algorithms

Acronyms

AMV Autonomous Marine Vehicle. 53, 79, 137

ASV Autonomous Surface Vehicle. xvii, 3, 5, 141

AUV Autonomous Underwater Vehicle. xvii, 3, 141, 144

CPF Cooperative Path Following. ix, 3, 8, 13

CRLB Cramér-Rao Lower Bound. 137

DEC Distributed Estimation and Control. 14, 236

DEKF Distributed Extended Kalman Filter. 14, 236

DVL Doppler Velocity Log. 5

EKF Extended Kalman Filter. 115, 119, 238

ETC Event Triggered Communication. 9, 23, 27

FIM Fisher Information Matrix. 14, 137, 140, 236, 238

GAS Globally Asymptotically Stable. 73

ISS Input-to-State-Stable. 77

KF Kalman Filter. 115, 119

LTV Linear Time Varying System. 114, 126

MAS Multi-Agent System. 1

MPC Model Predictive Control. 14, 181, 238

SLAP Simultaneous Target Localization and Pursuit. x, xi, xii, xiii, 6, 13, 14, 97, 135, 158, 159, 160, 161, 162, 179, 181, 182, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 236, 238

UAV Unmanned Aerial Vehicle. 53, 57

1

Introduction

Contents

Motivated by challenging issues that arise in the field of motion coordination of MAS, this thesis addresses problems in the area of cooperative control of MAS under communication constraints that involve the use of methodologies from networked control systems, state estimation theory, optimization, and event-based communications. Our goal is to develop efficient tools that will allow us to solve cooperative control problems that arise in the operation of multiple autonomous vehicles in numerous missions such as formation keeping, cooperative path following, rendezvous, range-based cooperative navigation, and range-based cooperative target localization and pursuit.

1.1 Multi-agent systems

Over the years, engineers and scientists across diverse fields have been striving to identify the underlying mechanisms of networked systems. Sociologists use networks to predict the behaviour of social systems and to help decision makers [Degroot, 1974,Bullo, 2018,Dong et al., 2018]. Biologists study MAS to understand the behaviors of animal groups such as flocks of birds, schools of fish, swarms of honey bees, and herds of wildebeests, see [Martinez et al., 2007,Okubo, 1986,Seeley and Buhrman, 1999] and the references therein. In the field of control engineering and robotics, researchers study how to bring a number of physical systems to cooperate in a distributed manner, by exchanging information over a support communication network, so as to yield a coherent whole capable of performing an assigned task that might be impossible to be done effectively by a single system. This gives rise to a very active and exciting research field - cooperative distributed control and estimation of multi-agent systems [Ren and Beard, 2008,Bullo, 2018,Li and Duan, 2015]. In the context of the current thesis, we study multi-agent systems that are systems composed of multiple interacting elements, known as agents. Agents are autonomous dynamical systems that are capable of performing given *local tasks* and interacting with other agents via a communication network established among them to achieve a common goal (*global task*). This type of MAS appears in many practical scenarios that involve sensor networks, multiple autonomous vehicles, and embedded robotic systems, to name but a few. In the following section, we will discuss some typical applications of MAS in the area of cooperative control and estimation of multiple autonomous vehicles.

1.2 Motivating examples

The problems addressed in this thesis are strongly motivated by real word applications that require the use of cooperative control techniques, whereby a group of multiple vehicles deployed in a given region, each one carrying different resources, cooperate to achieve a common goal by exchanging relevant data over an underlying communication network. Typical cooperative mission scenarios will be discussed in the subsequent sections, namely, cooperative path following, cooperative range-based navigation, and cooperative target localization and pursuit.

1.2.1 Cooperative path following (CPF)

CPF, a compelling example of formation control, can be defined as the process of steering a group of vehicles along given spatial paths, while holding a desired inter-vehicle formation pattern, without an explicit temporal assignment. In this context, each vehicle is simply required to move along its assigned path at a desired reference speed that may depend on where on the path the vehicle is. A simple example of CPF is illustrated in Fig.1.1. In this setup, each vehicle is assigned a local task, i.e. converge to and follow its preassigned path (a straight line, in case the figure), while exchanging data with all or some of the other vehicles via a wireless communication network in order to position itself relative to other vehicles such that the ensemble will become aligned in a side-by-sided formation. More formally, consider N spatial paths $\{\mathcal{P}_i : \gamma_i \rightarrow \boldsymbol{\eta}_i(\gamma_i); i = 1, ..., N\}$, where each path i is parameterized by the continuous variable $\gamma_i \in \mathbb{R}$, e.g. arc-length. Notice that the variable γ_i is not necessarily path length. In fact, in the context of CPF path parametrization plays a key role in the problem formulation. The components of each path $\boldsymbol{\eta}_i \in \mathbb{R}^3$ typically includes the position of the path, and is specified by the variable γ_i. CPF can be viewed as a two-layer control system: i) *path following* aims at driving each vehicle to converge to and follow its assigned path, i.e. making the vehicle's position $\mathbf{p}_i$ converges to $\boldsymbol{\eta}_i(\gamma_i)$ and ii) *cooperation* along the paths, which is achieved if and only if all path parameters are synchronized, i.e. $\gamma_1 = \gamma_2 = ... = \gamma_N$ and $\dot{\gamma}_i = v_\mathrm{d}$ for all $i = 1, ..., N$ where v_d is the desired speed of the formation, which may depends on the path parameter its self. Using CPF, the desired geometry formation along the paths depends on the way the paths are parameterized. For the example illustrated in Fig.1.1, we can take the path parameters $\gamma_i; i = 1, 2, 3$ as the arc-lengths of paths; and thereby the side-by-side formation is reached when all path parameters equal. Detailed explanations on how to parameterize the paths to achieve different formation patterns can be found in [Ghabcheloo et al., 2009, Hung et al., 2020b, Kaminer et al., 2017].

The concept of CPF is extremely useful in many scientific and commercial applications.

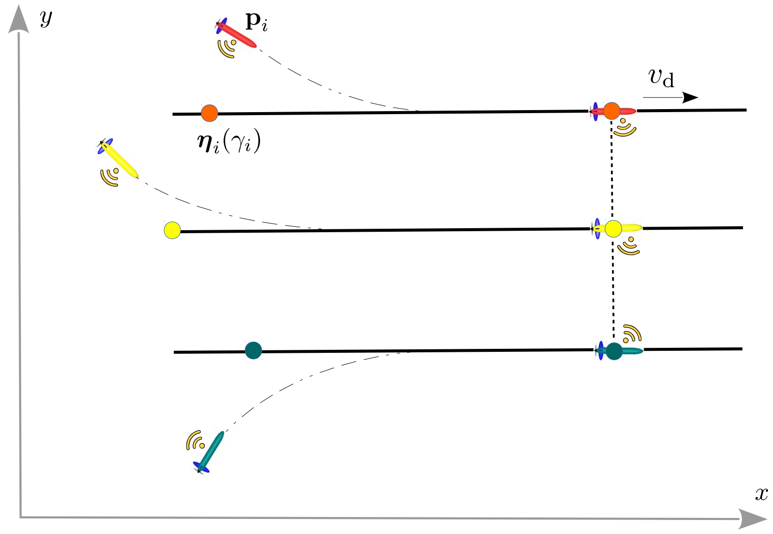


Figure 1.1: An example of CPF in 2D. The position of each vehicle i is denoted by $\mathbf{p}_i$. Each path is parametrized by $\boldsymbol{\eta}_i = [0, \gamma_i]^\mathrm{T} \in \mathbb{R}^2; i = 1, 2, 3$ where γ_i is the arc-length of the path.

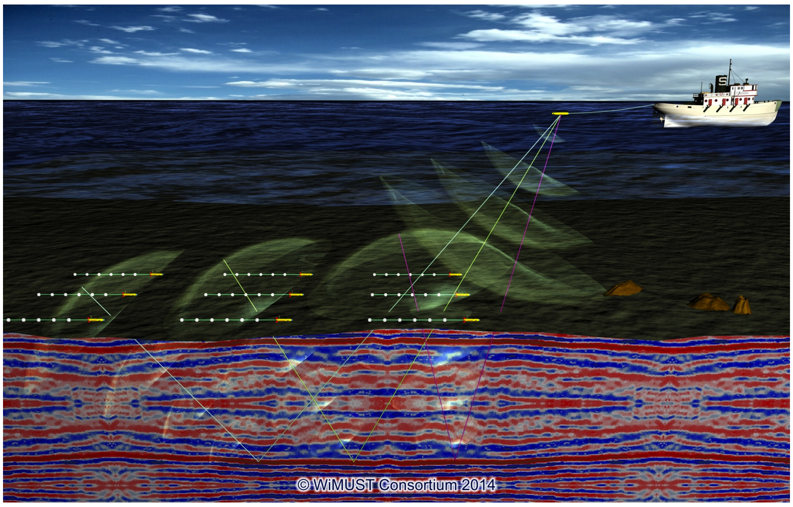


Figure 1.2: Cooperative path following of AUVs in the scope of Wimust project. In this figure, the top vehicles is a ship. In Sines, we actually used three ASVs to generate the acoustic signals, instead of the ship. [Abreu et al., 2016a]

In the scope of WiMUST project, CPF is applied to enable a group of AUVs working in cooperation with ASVs to perform geotechnical acoustic surveys at sea, as illustrated in Fig.1.2 [Abreu et al., 2016a]. In this application, the autonomous vehicles are used as intelligent sensing and communication nodes of a reconfigurable moving acoustic network.

The AUVs are equipped with hydrophone streamers of small aperture, such that the overall system behaves as a distributed acoustic array capable of acquiring acoustic data reflected off the seabed, which is insonified through the use of acoustic sources installed on-board a support ship/boat or ASVs. By actively controlling the geometry of the vehicle formation, it becomes possible to change the shape of the acoustic array according to the needs of a specific application [Pedro Abreu and Silva, 2016]. See also [Klemas, 2015, Kaminer et al., 2017] for application of CPF techniques to coastal monitoring and road search operations.
Although CPF has been studied intensively over the last decade, there are two practical challenges which are the main topics addressed in this thesis. The challenges involve the physical constraints on the vehicles' inputs and the often stringent limitations imposed by the communication network among the vehicles, as formulated in the following problem.

Problem 1.1 (Constrained CPF)**.** *Consider a group of autonomous vehicles and a corresponding set of spatial paths that the vehicles are required to go through at assigned speeds. Assume that the vehicle inputs (e.g. linear and rotational speeds in the case where only the vehicle kinematics are considered) are restricted to take values in compact sets. Further assume each vehicle is only able to exchange information with a subset of vehicles using an underlying communication network. Design a local controller for each vehicle such that the group of vehicles achieves CPF.*

1.2.2 Range-based cooperative navigation and target localization and pursuit

Typically, range-based navigation is defined for an agent, for example, a scuba-diver or an AUV to find his/its own state (position, and possibly velocity and acceleration as well) using the information measured by the agent and ranges to a known single or multiple beacons [Bayat et al., 2016]. If an agent such as AUV can measure its velocity vector (using a Doppler Velocity Log (DVL)), then only the position of the AUV needs to be determined. In other situations, velocity and acceleration need to be determined as well. In cases where beacons are fixed at known positions, the fundamental problem of range-based navigation is to find conditions on the motion of the agent such that the agent's state is observable (or uniquely determined), see [Bayat et al., 2016, Indiveri et al., 2012, Batista et al., 2011] and references therein. In other cases where the beacons are movable and controllable, we are interested in finding the answer to how to control the beacons such that the agent acquires "sufficiently rich" range-information for navigating itself. Fig.1.3(left) illustrates the idea of range-based cooperative navigation where the two ASVs play as cooperative beacons for navigating the scuba-diver and the AUV.

Range-based target localization (or tracking), on the other hand, is defined for one or multiple trackers as the task of finding the state of a target using only ranges from the tracker(s) to the target. The state of the target typically includes the target's position, velocity, and possibly acceleration, depending on the model of the target adopted. Target pursuit is defined as a task of driving the tracker(s) to converge to and stay in the vicinity of the target. This task is crucial when agents are located in an underwater environment where ranges can only be measured up to a relatively short distance. An example of range-based cooperative simultaneous target localization and pursuit (SLAP) is illustrated in Fig.1.3(right) where the ASVs play the roles of trackers.

The cooperative navigation and cooperative target localization above are dual. Thus, in the context of the current thesis, we address the cooperative target localization and pursuit that is stated as follows.

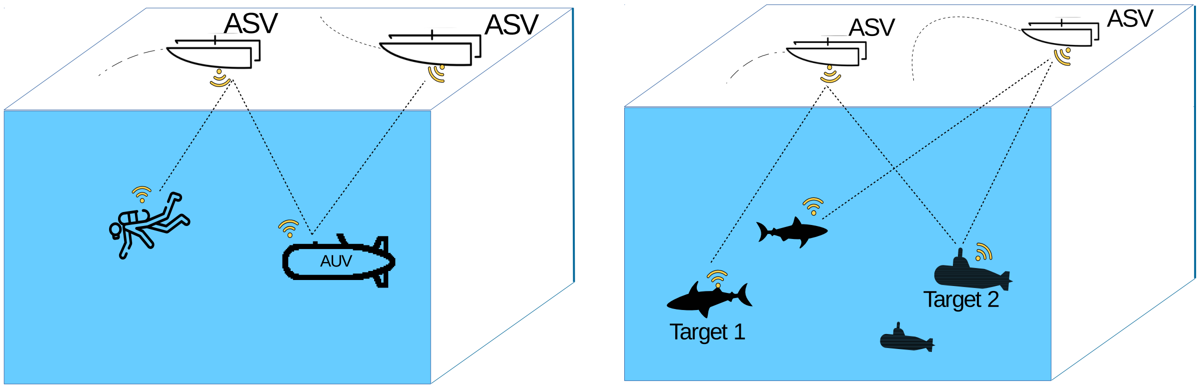


Figure 1.3: Examples of range-based cooperative navigation (left), and range-based cooperative target localization and pursuit (right).

Problem 1.2 (Range-based cooperative SLAP). *Consider a set of multiple autonomous vehicles, called trackers, in charge of localizing and pursuing a set of multiple targets. Each tracker is capable of measuring ranges to the targets, while the information exchange among the trackers has a stringent communication constraints network with each tracker being only able to exchange information with a subset of trackers in the network. Design a distributed estimation and control strategy such that all trackers agree on the character-*

ization of the motion of the targets (e.g. on the evolution of the means and covariances of their states' estimates), while guaranteeing that all trackers converge to and stay in a predefined vicinity of the targets.

1.3 Consensus/synchronization and ETC mechanisms

1.3.1 Consensus/synchronization of multi-agent system

Consensus/synchronization of MAS arises in many control and estimation applications such as in sensor networks, rendezvous maneuvers, formation control, and CPF of autonomous vehicles where a group of agents must reach agreement on a final state (consensus) or trajectory (synchronization). For a clear and intuitive understanding of the consensus/synchronization problem we consider the following simple examples.

Example 1.1 (Rendezvous control, [Bullo, 2018])**.** *Consider a network of N agents (e.g. vehicles) with single integrator dynamics described by*

$$\dot{\mathbf{x}}_i(t) = \mathbf{u}_i(t); \quad i \in 1, ..., N, \tag{1.1}$$

where $\mathbf{x}_i = [x_i, y_i]^{\mathrm{T}} \in \mathbb{R}^2$ and $\mathbf{u}_i \in \mathbb{R}^2$ are respectively the position and the input of the agent (in this case, velocity). Each agent is capable of communicating with a subset of the agents in the network. The distributed control problem is to find inputs $\mathbf{u}_i; i = 1, .., N$ based on local information of the agent itself and that received from its neighbors such that all agent reach a consensus point asymptotically, as illustrated in Fig.1.4. Note that the consensus point is left free.

Example 1.2 (Distributed estimation, [Battistelli and Chisci, 2016])**.** *Consider a wireless sensor network with N sensor nodes, in charge of localizing a moving target (see Fig.1.5). Each sensor node is capable of measuring the distance to the target, performing local computations, and exchanging information with a subset of sensor nodes in the network. The consensus estimation problem aims to find a distributed estimation algorithm for each*

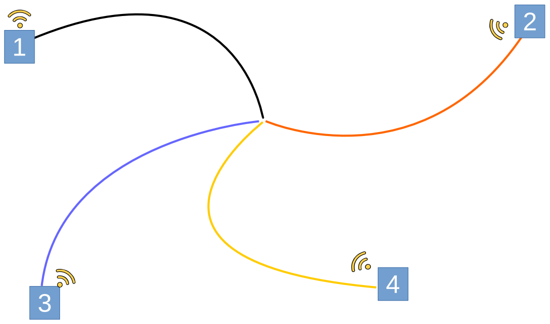


Figure 1.4: Rendezvous: four agents cooperating to reach consensus on a final position

sensor node such that all sensor nodes reach agreement on the distribution (e.g. mean and covariance) of the target's state.

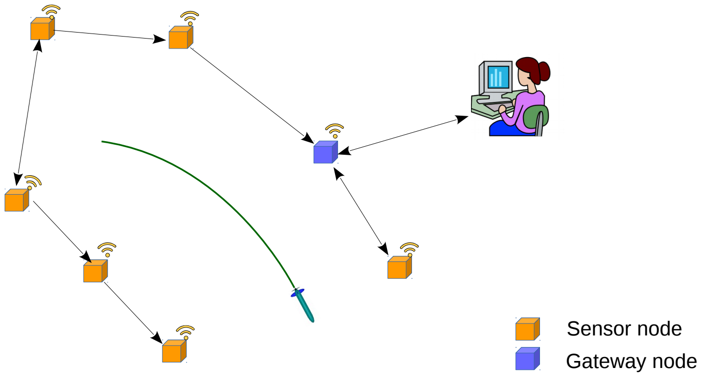


Figure 1.5: Sensor network: multiple sensors cooperate to localize the target (in green)

Example 1.3 (Coordination problem in CPF)**.** *Consider a network of N vehicles where each is assigned a spatial path to follow, parameterized by a variable $\gamma_i, i = 1, ...N$ to follow. The coordination problem consists of controlling the independent motions of the vehicles along the paths so that consensus (synchronization) on the path parameters is achieved asymptotically, that is, $\gamma_i(t) = \gamma_j(t)$ as $t \to \infty; \forall i, j = 1, N$ and all path parameters evolve with a desired speed profile v_d*[1]*, i.e. $\dot{\gamma}_i(t) = v_\text{d}$ as $t \to \infty; \forall i = 1, ..., N$.*

Algebraic graph theory is the essential tool to solve the problem of MAS consensus/synchronization. From a graph theoretical point of view, the network topology established among the agents is considered as a graph. Each agent in the network is represented

[1] In general, v_d can be path dependent, i.e. v_d is a function of the path parameters

by a vertex (node) and a communication link between two agents is represented by an arc (edge). If the communication links are directional, they induce a directed graph (digraph). The graph is undirected if communication links are unidirectional, implying that for any given link, the information can be exchanged in both directions. For this reason, in the literature undirected graphs are also called as bidirectional graphs. A path in a graph is an ordered sequence of nodes such that any pair of consecutive nodes in the sequence is an edge of the graph. An undirected graph is connected if there exists a path between any two nodes; otherwise it is unconnected. The set of in-neighbors of an agent i is represented by $\mathcal{N}_i^{\text{in}}$ and contains all agents $j \in \mathcal{N}_i^{\text{in}}$ from which agent i can receive information. See [Bullo, 2018, Ren and Beard, 2008] for an in depth introduction to graph theory. Fig.1.6 illustrates different types of graphs.

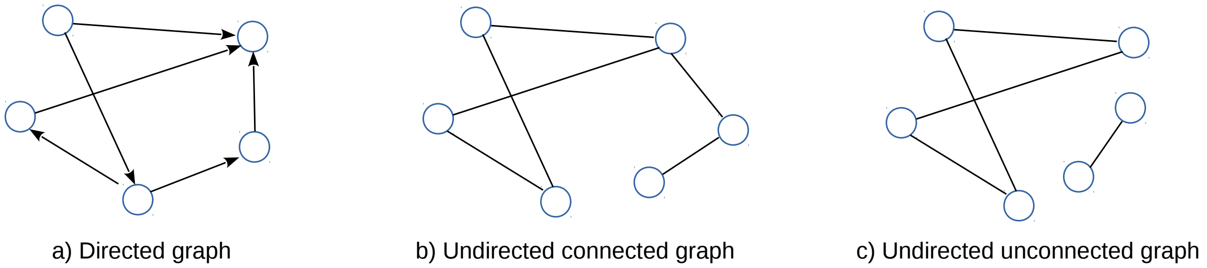

a) Directed graph
b) Undirected connected graph
c) Undirected unconnected graph

Figure 1.6: Examples of different graphs

To respect the flow of information imposed by the topology of the communication network, estimation and control algorithms designed for each agent i must be implemented in a distributed manner, i.e. they may only depend on the local state and on the information received from its in-neighbors as specified by $\mathcal{N}_i^{\text{in}}$. For example, regarding the rendezvous problem stated in **Example 1.1**, it is well known that if the graph is undirected and connected, then the distributed control law given by

$$\mathbf{u}_i(t) = -\sum_{j \in \mathcal{N}_i^{\text{in}}} \mathbf{x}_i(t) - \mathbf{x}_j(t) \tag{1.2}$$

forces all agents to reach a consensus point asymptotically, i.e. $\mathbf{x}_i(t) = \mathbf{x}_j(t)$ for all $i, j = 1, ..., N$ as $t \to \infty$. Consequently, $\mathbf{u}_i(t) \to 0$ as $t \to \infty$ as well. Note that if the agents were to reach the consensus in a finite time, then due to (1.2) $\mathbf{u}_i = \mathbf{0}$ for all i, thereafter, thus maintaining consensus for all times. The proof of this result can be found in the literature, see for examples [Bullo, 2018, Ren and Beard, 2008]. The rendezvous problem considered above is also a special case of the general consensus/synchronization problem addressed in Chapter 2.

1.3.2 Event-triggered communications

In **Example 1.1**, the control law (1.2) solves the rendezvous problem, provided that the communications among vehicles are continuous, that is, the in-neighbors of vehicle i must transmit their states (positions) to vehicle i instantaneously at every instant of time. In practice, it may be impossible to meet this requirement, namely in applications where information is transmitted over a wireless channel, especially if this wireless channel must be shared among other devices. The current standard method is to schedule communications to take place periodically and to set the period as small as possible to obtain adequate performance as continuous communication. However, periodic communication might not be efficient for networks that have low bandwidth (e.g. acoustic signal) or involve large number of agents. In recent years, event-triggered communications (ETC) have come to the fore in order to overcome the above limitations of continuous or periodic communications. Unlike these, ETC are more reactive since in this setup communications only occur when required, as determined by desired objectives, leading to more efficient and cost-effective communication networks.

To grasp the core concepts involved in ETC mechanisms we revisit **Example 1.1** and resolve the Rendezvous problem using a simple but intuitive ETC strategy. Formally, for each generic agent i, let $\{t_{i,k}\}_{k\in\mathbb{N}}$ be the sequence of time instants at which the agent transmits its position to its neighbors, as defined by a mechanism that will be explained later. To avoid continuous or periodic communications, each agent estimates (or predicts) the positions of its in-neighbors and uses these estimates to update control law (1.2). Let $\hat{\mathbf{x}}_j^i$ denote the estimate of $\mathbf{x}_j; j \in \mathcal{N}_i^{\text{in}}$. In what follows we adopt a simple estimation model for the evolution of $\hat{\mathbf{x}}_j^i$ as

$$\hat{\mathbf{x}}_j^i(t) = \mathbf{x}_j(t_{j,k}), \quad t \in [t_{j,k}, t_{j,k+1}). \tag{1.3}$$

for all $j \in \mathcal{N}_i^{\text{in}}$. This model implies that $\hat{\mathbf{x}}_j^i(t)$ holds the last transmitted value of $\mathbf{x}_j$ and is updated whenever agent i receives a new update from agent j. This simple, almost naive strategy, is motivated by the fact that once the agents reached a consensus point, they will remain there thereafter; as a consequence, (1.3) predicts exactly the positions of the in-neighbors of agent i. The distributed control law for each agent i, in context of the ETC mechanism, uses $\hat{\mathbf{x}}_j^i$ rather than $\mathbf{x}_j$, and is given by

$$\begin{aligned}\mathbf{u}_i(t) &= -\sum_{j\in\mathcal{N}_i^{\text{in}}} \mathbf{x}_i(t) - \hat{\mathbf{x}}_j^i(t).\\ &= -\sum_{j\in\mathcal{N}_i^{\text{in}}} \mathbf{x}_i(t) - \mathbf{x}_j(t) + \mathbf{e}_j(t)\end{aligned} \tag{1.4}$$

where $\mathbf{e}_j(t) = \mathbf{x}_j(t) - \hat{\mathbf{x}}_j^i(t); j \in \mathcal{N}_i^{\text{in}}$ is the estimation error of agent j's position, at agent i. Compared with (1.2), (1.4) has the virtue of putting in evidence the contribution of the estimation error. Ideally, one would like for the estimation error to vanish in order achieve the type of performance that is obtained with continuous communications. However, this would only be valid in the rather uninteresting case where all vehicles had already reached the consensus point. Thus, we look for a way to control the estimation error to ensure that $\hat{\mathbf{x}}_j^i$ does not deviate far from the $\mathbf{x}_j$ so that ETC still guarantees an adequate performance when compared with that obtained with continuous communications. Note from (1.3) that

$$\mathbf{e}_j(t) = \mathbf{x}_j(t) - \mathbf{x}_j(t_{j,k}); \quad j \in \mathcal{N}_i^{\text{in}} \quad t \in [t_{j,k}, t_{j,k+1}). \tag{1.5}$$

Thus, $\mathbf{e}_j$ also measures the difference between the current position of agent j ($\mathbf{x}_j(t)$) and the last position broadcast to its in-neighbors ($\mathbf{x}_j(t_{j,k})$). To control $\mathbf{e}_j$, we let agent j broadcast its latest position whenever the size of this error exceeds an arbitrarily small threshold $\epsilon \geq 0$. More formally, we say a communication event for vehicle j is triggered whenever

$$\delta_j(t) \triangleq \|\mathbf{e}_j(t)\| - \epsilon \geq 0$$

where $\delta_j(t); j = 1, ..., N$ is so-called the triggering function. According to (1.5), $\mathbf{e}_j(t_{j,k}) = \mathbf{0}$ for all $k \in \mathbb{N}$, that is, the estimation error will be reset whenever the out-neighbors of agent j receive a new update. As a consequence, the size of the estimation error is alway kept in a ball with radius ϵ, i.e. $\|\mathbf{e}_j(t)\| \leq \epsilon$ for all t.

As an illustrative example, we simulate the ETC mechanism described above for the rendezvous problem with four agents whose network topology is modeled by the undirected graph shown in Fig.1.7. To access the performance of the ETC mechanism, we consider an "energy-like" function that measures the displacement between the positions of the agents, given by $V = \|\mathbf{x}_1 - \mathbf{x}_2\| + \|\mathbf{x}_2 - \mathbf{x}_3\| + \|\mathbf{x}_3 - \mathbf{x}_4\|$. Note that V is always positive and $V = 0$ if and only if all agents reach the consensus point. Later, we will see in Chapter 2 that this type of function plays a role of a Lyapunov function to analyze the convergence of the closed-loop MAS. Numerical simulation results are plotted in Fig.1.8. The performance of the ETC mechanism depends on how the triggering threshold ϵ is selected. The detail of this influence will be discussed in Chapter 2.

Another concern with any event-triggered control or communication mechanism is being able to exclude Zeno behavior, defined as follows.

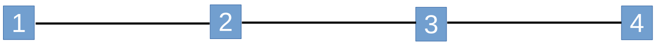
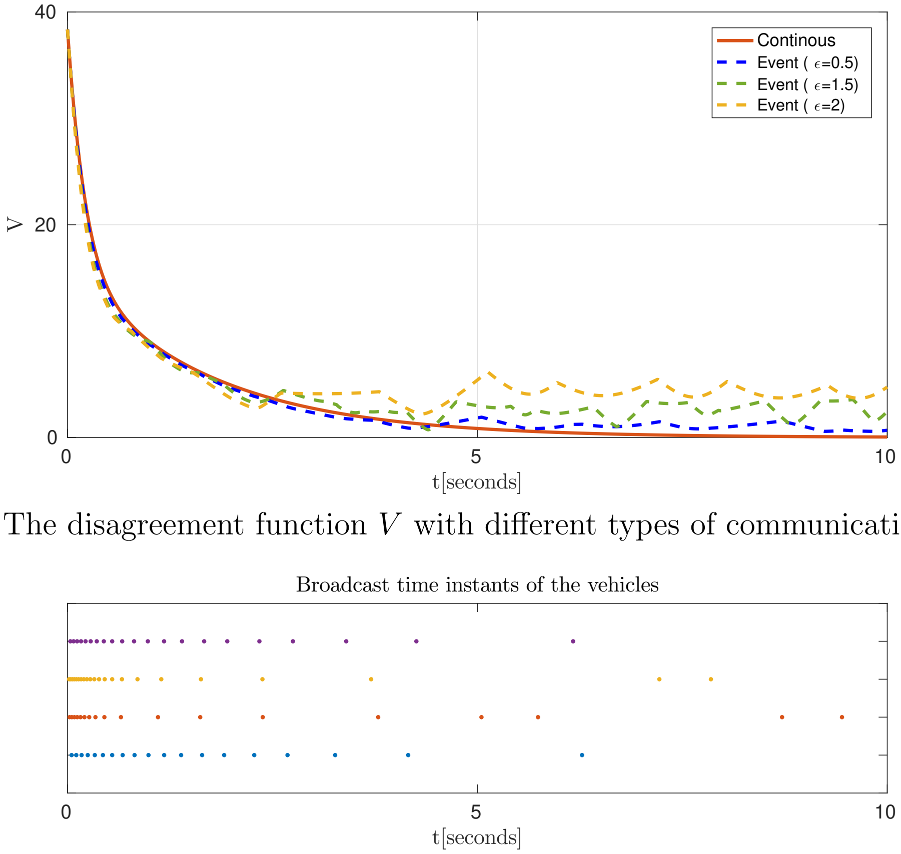


Figure 1.7: The undirected graph models the network of vehicles in the rendezvous problem



a) The disagreement function V with different types of communications.

b) Broadcast time instants with $\epsilon = 0.5$: Rows indicate the time instants $t_{i,k}, i = 1, ..., 4$ at which the agents broadcast their positions to their out-neighbors.

Figure 1.8: Numerical results with the ETC mechanism for the rendezvous problem

Definition 1.1 (Zeno behavior, [Nowzari et al., 2019]). *Consider a resulting closed-loop MAS, in which agents are driven by a distributed control protocol with an ETC mechanism.*

A solution for the closed-loop MAS with a given initial condition exhibits Zeno behavior if there exists $T > 0$ such that $t_{i,k} \leq T$ for all $k \in \mathbb{N}$ and for all $i \in \mathcal{N}$.

Intuitively, if the ETC mechanism generates an infinite number of communication events in a finite time period, the solution exhibits Zeno behavior. The existence of a positive minimum time interval between two consecutive triggering times avoids this impractical situation and is thus critical to the characterization of any event-triggered communication strategy. A rigorous proof showing that the ETC described above for the rendezvous problem excludes Zeno-behavior will be shown in Chapter 2.
In the text before, through a simple example that borrowed from the Rendezvous problems, we introduced the core idea behind the ETC mechanisms proposed in the thesis. In the subsequent chapters, we will show how this idea can be exploited to design ETC mechanisms for a general consensus/synchronization problem (Chapter 2), CPF (Chapter 3), and range-based cooperative SLAP (Chapter 6). In CPF and range-based cooperative SLAP, we will show that the ETC mechanisms proposed have the potential of reducing the number of messages exchanged and the frequency of communications among the vehicles, while guaranteeing an adequate performance as with continuous communications. These characteristics are of the utmost importance in applications where the transmission medium imposes stringent communication constraints (e.g. the vehicles operate and communicate with other vehicles in underwater environment via acoustic communication channels).

1.4 Contributions and publications

The contributions of thesis are described next.
We first propose an event-triggered communication framework for the problem of MAS consensus/synchronization, with the dynamics of the agents given by

$$\dot{\mathbf{x}}_i = A\mathbf{x}_i + \mathbf{f}(\mathbf{x}_i, t) + B\mathbf{u}_i, \qquad \forall i = 1, ..., N \tag{1.6}$$

where $\mathbf{x}_i \in \mathbb{R}^n$ and $\mathbf{u}_i \in \mathbb{R}^m$ are the state and input of the agent, respectively, A, B are matrices with appropriate dimensions, and $\mathbf{f}(\cdot)$ is a general nonlinear function. Compared with existing results in the literature, see for example [Liuzza et al., 2016, Su et al., 2016, Hung et al., 2019], the dynamics of the agents and the network topology are more general, i.e. the agent dynamics include a general linear and a nonlinear Lipschitz term and the underlying communication graphs are directed. In this respect, our results extend those in [Seyboth et al., 2013, Hung et al., 2019]. The exclusion of Zeno behavior is also rigorously proved.

Second, we propose a distributed control strategy to solve the constrained CPF problem stated in **Problem 1.1**. The solution addresses the constraints on the vehicles' inputs and the limitations imposed by the communication network that connects the vehicles [Hung et al., 2020b]. Specifically,

(i) At the *path following* level, we develop an MPC scheme for *path following* that takes into account the vehicle input constraints.

(ii) At the *cooperative* level, we propose a novel distributed control strategy for the *coordination of nonlinear* MAS where the agents' *input constraints* are explicitly taken into account. We also propose an ETC mechanism that is not only capable of reducing the frequency of communications among vehicles but is also robust with respect to time-varying communication delays, making the scheme attractive for scalable networks with limited communication bandwidth.

Third, we propose a systematic approach to solve the range-based cooperative SLAP problem stated in **Problem 1.2**. We identify a few sub-problems where we isolate specific technical challenges that are present in **Problem 1.2**. By studying and solving each sub-problem, we gain insights into a general solution for the original problem. Also, while addressing each sub-problem we find as byproducts interesting results that stand on their own.

(i) The first sub-problem involves characterizing the motion of trackers such that the target's state is globally observable. Using tools from function analysis, a set of conditions are derived for different type of target maneuvers, providing useful guidelines for trackers' motion planning [Hung and Pascoal, 2020b].

(ii) The second sub-problem is to find the optimal motions for the trackers such that the range-information acquired for estimating the target state is maximal. Using a Bayesian Fisher Information Matrix (FIM) approach that borrows from estimation theory, the types of optimal relative motion of the trackers with respect to the target are characterized. We also propose a receding horizon planning, control, and estimation framework for the range-based SLAP problem where the constraints on the trackers' inputs and the uncertainty of the target are taken into account explicitly [Hung et al., 2020a].

(iii) After solving the aforementioned sub-problems, we revisit the original problem. We exploit the knowledge about observability and optimal trajectories for range-based target localization to plan desired motions for the trackers. We then propose a cooperative distributed estimation and control (DEC) strategy to address the constraints on the topology of the inter-tracker communication network. For this purpose, a distributed extended Kalman filter (DEKF) strategy is adopted for cooperative estimation of the target's state, and a distributed consensus control strategy is proposed for cooperative pursuit of the target. The latter aims at driving all trackers to a specified vicinity of the target while holding an optimal target-trackers relative-geometry that maximizes the range information available estimating the target's state. To make the proposed DEC more efficient in terms of communications, we propose event-triggered mechanisms for the DEKF and for the distributed consensus control strategy where communications among the trackers only take place when deemed necessary, according to a well defined mathematical criterion. The stability of the complete closed-loop DEC system is proved rigorously.

Table 1.1: Publications originated by the work presented in this thesis

Topics/Chapters		**Conf.**	**Jour.**
Cooperative control of MAS with ETC mechanism	Ch.2: General framework Ch.3: CPF applications	C1[a] C2[b],C3[c]	J1[e] J2[f]
Cooperative range-based target localization and pursuit	Ch.4: Observability analysis Ch.5: Optimal motion analysis Ch.6: DEC approach	C4[d]	 J3[g] J4[h]

[a] [Hung et al., 2019]
[e] [Hung and Pascoal, 2020a]
[b] [Hung et al., 2018]
[c] [Hung and Pascoal, 2018]
[f] [Hung et al., 2020b]
[d] [Hung and Pascoal, 2020b]
[g] [Hung et al., 2020a]
[h] [Hung et al., 2021]

In Table 1.1, we list the publications originated by the work reported in this thesis, organized by topic (chapter) and types of publications.

1.5 Structure of the thesis

The thesis is divided into two parts and organized as in Fig.1.9. Part I includes Chapters 2 and 3 that describe solutions to problems related to the consensus/synchronization of homogeneous MAS using event-triggered communications. Part II is consisted of Chapters 4-6 that solve problems involving in the range-based cooperative target localization and pursuit.

In Chapter 2, we propose a general framework for the consensus/synchronization of homogeneous MAS using an event triggered communication mechanism. The agents dynamics and the network topology considered are sufficiently general to address a large number of applications, i.e. the agents have linear and Lipschitz nonlinear dynamic terms, and the underlying communication graphs are directed. The convergence of the closed-loop MAS is analyzed and the exclusion of Zeno behavior is also rigorously proved.

Chapter 3 addresses the problem of CPF among multiple autonomous vehicles when the vehicles' input are constrained. To this effect, we decouple the original CPF problem into two sub-problems: coordination of MAS and single path following. The first is tacked by using the results obtained in Chapter 2, whereas the second is solved by a model predictive control (MPC) strategy. An ETC mechanism is also introduced to reduce communications among the vehicles and the communication delay is also addressed. The stability of the closed-loop CPF system is analyzed and the efficacy of the proposed method is illustrated via simulation examples.

In Chapter 4, we address the observability problem of range-based target localization using single or multiple trackers. We consider four cases: i) the target is fixed, ii) the target's velocity vector is known, iii) the target's velocity vector is unknown but constant iv) and, the target's acceleration vector is unknown but constant. Conditions on the motions of the trackers under which the target's state is fully observable are derived.

Chapter 5 continues the work in Chapter 4 that studies further the motions of the trackers. The main objective is to find the optimal motion of the trackers that makes the range-information acquired for estimating the target's state maximal. To this end, we adopt the Bayesian Fisher information matrix (FIM) as a means to quantify the range-information and then find conditions on the motion of the trackers under which the Bayesian FIM is maximized. When the target's state is purely deterministic and tracker's constraints are neglected, the conditions on the optimal motion of the trackers are derived analytically. When the uncertainty of the targets and the constraint on the trackers are taken into account explicitly, a receding horizon optimization framework is proposed to plan and control the motion of the trackers. Several examples are provided to illustrate the efficacy of the proposed method.

Finally, in Chapter 6, we exploit the results derived in Chapter 2-5 to design an efficient distributed estimation and control (DEC) system to address the problem of cooperative target-localization and pursuit using multiple autonomous vehicles. First, we show how to exploit the knowledge on observability and optimal motion of trackers given in Chapters 4-5 to plan the motion of the trackers. We then use the results in Chapters 2-3 to design a distributed cooperative tracking controller for the trackers to track the desired motion. A distributed Extended Kalman Filter algorithm is adopted for the cooperative estimation of the target's state. The stability of the complete closed-loop DEC system is analyzed, showing the robustness of the proposed method.

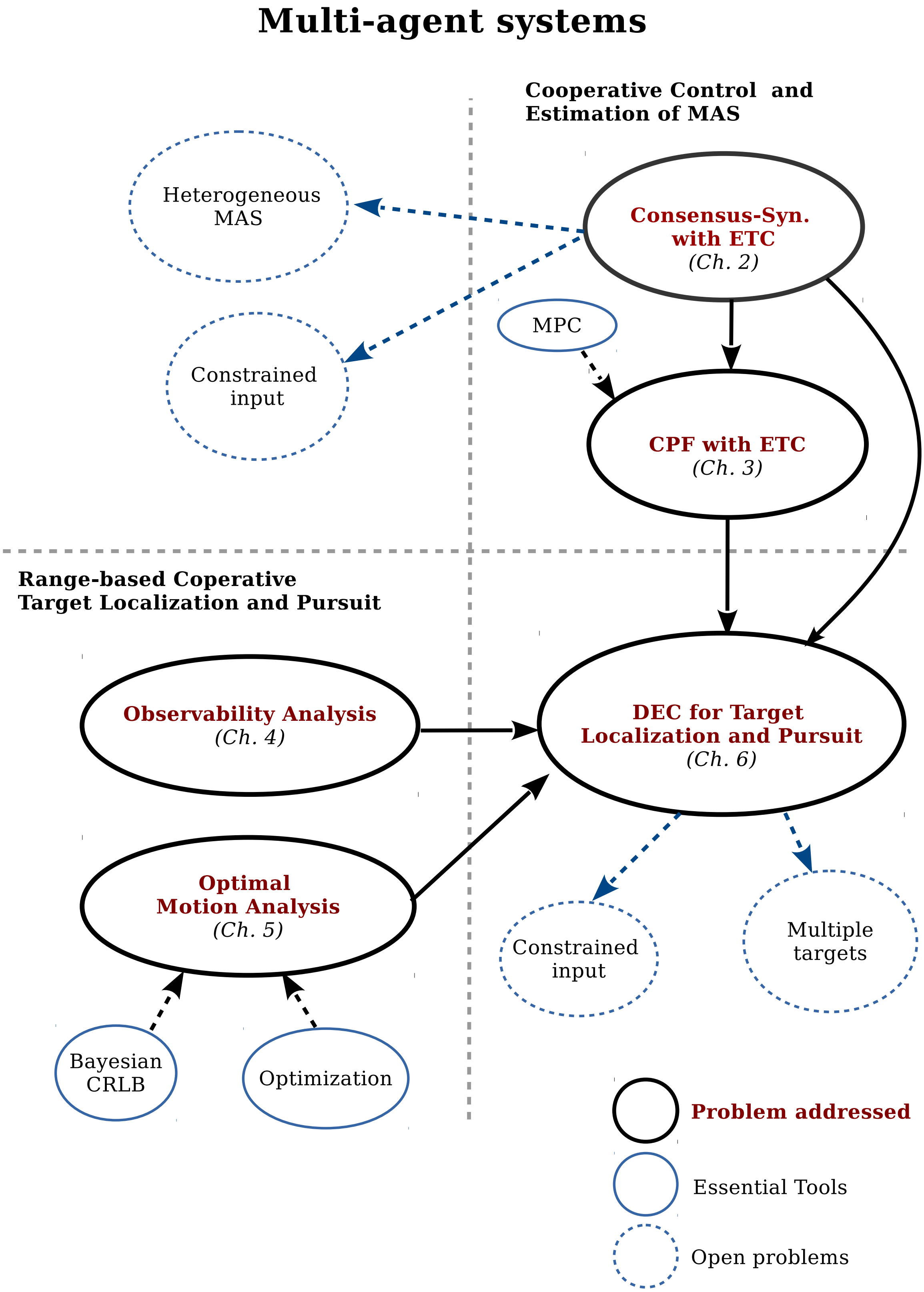


Figure 1.9: Thesis structure

Part I

Cooperative Control of Multi-Agent System with Event-Triggered Communications

2

Consensus/synchronization of nonlinear multi-agent system with event-triggered communications

Contents

This chapter describes a solution to the consensus/synchronization problem for a general class of non-linear multi-agent systems and network topologies that resorts to a distributed control strategy with an event-triggered communication (ETC) mechanism. The strategy proposed has two important properties: i) it yields practical consensus stabilization, that is, the synchronization error that measures the disagreement among agents converges to a ball centered at the origin, with a radius that can be made arbitrarily small and ii) the minimum of the inter-event times for each agent is strictly positive, hence Zeno behavior is excluded. The results obtained are illustrated via simulations.

2.1 Literature review

The problem of consensus/synchronization of multi agent systems (MAS) has attracted widespread interest in the last two decades due to its applications in the areas of sensor networks, mobile systems, autonomous robots, etc., where a group of agents must reach agreement on a final state (consensus) or trajectory (synchronization). Some of the applications are described in [Hung and Pascoal, 2018, Li et al., 2010, Rego et al., 2019a], where consensus techniques were used to achieve desired geometric formations of multiple autonomous vehicles. For background materials on this topic the reader is referred to [Olfati-Saber et al., 2007, Li and Duan, 2015] where communications among the agents are assumed to occur continuously in time.

Driven by the fact that the bandwidth available for communications among multiple agents is severely limited in many practical applications, there has been a flurry of activity in the area of distributed event-triggered control and communications for multi-agent systems, as reported in [Dimarogonas et al., 2012, Garcia et al., 2013, Meng and Chen, 2013, Nowzari and Cortés, 2016, Nowzari et al., 2019] and the references therein. Among such studies, MAS with single integrator dynamics have received a great deal of attention, and many solutions for their coordination have been proposed, see for example the recent survey in [Nowzari et al., 2019]. One of earliest distributed event-triggered control solutions for MAS with single integrator dynamics was proposed in [Dimarogonas et al., 2012], while solutions for both event-triggered control and communications can be found in [Garcia et al., 2013, Nowzari and Cortés, 2016]. The study of the coordination problem of MAS with double integrator dynamics using an event-triggered framework is addressed in [Seyboth et al., 2013], which extends previous results for the case of continuous communications given in [Ren and Beard, 2008]. More recently, a number of authors have addressed the problem of MAS coordination for the case where the agents have more general linear dynamics, [Zhu et al., 2014, Garcia et al., 2014, Garcia et al., 2017, Almeida et al., 2017]. In [Garcia et al., 2014, Zhu et al., 2014], for example, the authors propose solutions for the MAS coordination problem where the triggering function adopted is state dependent, whereas an event-triggered mechanism that is dependent on time is proposed in [Garcia et al., 2017].

Event triggered coordination of MAS with nonlinear dynamics has been less studied and only a few results that consider several particular classes of nonlinear system have appeared recently in the literature [Liuzza et al., 2016, Su et al., 2016, Li et al., 2016, Hung et al., 2019]. A simple class of MAS with nonlinear dynamics was considered in [Liuzza et al., 2016, Hung et al., 2019], where the authors proposed a distributed model-based approach. The authors in [Su et al., 2016] addressed the leader-following multi-agent systems consensus problem with an event-triggered control mechanism. An event-triggered sampling control approach for directed networks was studied in [Li et al., 2016].

Motivated by the above considerations, the main purpose of this chapter is to develop a general framework for distributed control and event-triggered communication mechanism to solve the MAS consensus/synchronization problem where the agents dynamics and the network topology are sufficiently general to address a large number of applications. Compared with existing results in the literature, see for example [Liuzza et al., 2016, Su et al., 2016, Hung et al., 2019], the dynamics of the agents and the network topology are more general, i.e. the agents have linear and Lipschitz nonlinear dynamic terms, and the underlying communication graphs are directed. In this respect, our results extend those in [Seyboth et al., 2013, Hung et al., 2019]. The exclusion of Zeno behavior is also proved rigorously.

2.2 Preliminaries

2.2.1 Notation

In what follows, we let $\mathbb{R}, \mathbb{R}_{>0}$, and $\mathbb{R}_{\geq 0}$ denote the set of real, positive real, and nonnegative real numbers respectively. We shall use the notation $\|\cdot\|$ to denote the Euclidean norm of a vector. We will use the notation $x(t^+) := \lim\limits_{s \to t^+} x(s)$. Given matrices $A, B \in \mathbb{R}^{n \times n}$ the notation $A \succeq B$ implies that $A - B$ is positive semi-definite. A continuous function $\alpha : [0, a) \to [0, \infty)$ is said to be of class $\mathcal{K}$ if it is strictly increasing and $\alpha(0) = 0$. It is said to be of class $\mathcal{K}_\infty$ if $a = \infty$ and $\alpha(r) \to \infty$ as $r \to \infty$. A continuous function $\beta : [0, a) \times [0, \infty) \to [0, \infty)$ is said to be of class $\mathcal{KL}$ if, for each fixed s, the mapping $\beta(r, s)$

is of class $\mathcal{K}$ with respect to r and, for each fixed r, the mapping $\beta(r,s)$ is decreasing with respect to s and $\beta(r,s) \to 0$ as $s \to \infty$. Given a symmetric matrix A, the symbols $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ denote the smallest and the largest eigenvalues of A, respectively.

2.2.2 Algebraic graph theory [Bullo, 2018, Diestel, 2005]

A weight digraph $\mathcal{G} = \mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{A})$ induced by the communication network of a multi-agent system consists of a set of N vertices (nodes) $\mathcal{V} = \{1, 2, ...N\}$, a set of directed edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$, and a weight adjacency matrix $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N\times N}$. The latter satisfies the conditions $a_{ij} > 0$ if $(j,i) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. Here, self-edges (i,i) are not allowed and hence $a_{ii} = 0$. A directed path from vertex i to vertex j is an ordered sequence of vertices such that each immediate pair of vertices is an edge. A digraph is strongly connected if there exists a path from any $i \in \mathcal{V}$ to any $j \in \mathcal{V}$. The set of in-neighbors and the set of out-neighbors of vertex i are defined as $\mathcal{N}_i^{\text{in}} = \{j \in \mathcal{V} : (j,i) \in \mathcal{E}\}$ and $\mathcal{N}_i^{\text{out}} = \{j \in \mathcal{V} : (i,j) \in \mathcal{E}\}$, respectively. The in- and out- degree matrices D^{in} and D^{out} are defined as $D^{\text{in}} = \operatorname{diag}(d_i^{\text{in}})$ and $D^{\text{out}} = \operatorname{diag}(d_i^{\text{out}})$ where

$$d_i^{\text{in}} = \textstyle\sum_{j\in\mathcal{N}_i^{\text{in}}} a_{ij}, \qquad d_i^{\text{out}} = \textstyle\sum_{j\in\mathcal{N}_i^{\text{out}}} a_{ji},$$

respectively. A digraph is balanced if $D^{\text{in}} = D^{\text{out}}$. Any undirected graph is balanced. The Laplacian matrix L of a digraph is defined as $L = D^{\text{in}} - \mathcal{A}$. If $\mathcal{G}$ is strongly connected, then 0 is a simple eigenvalue of L with associated (right) eigenvector $\mathbf{1} := [1]_{N\times 1}$. Further, the digraph $\mathcal{G}$ is balanced if and only if $\mathbf{1}^{\mathrm{T}} L = \mathbf{0}$.

Remark 2.1. *With the graph definition given above, we use the convention that an agent i can receive information from its neighbors in $\mathcal{N}_i^{\text{in}}$ and send information to its neighbors in $\mathcal{N}_i^{\text{out}}$.*

The following lemma and definition will be used in the thesis.

Lemma 2.1. *([Yu et al., 2010, Li and Duan, 2015]) Suppose that the graph $\mathcal{G}$ is strongly connected. Then, there is a positive left eigenvector $\mathbf{r} = [r_1, ..., r_N]^{\mathrm{T}} \in \mathbb{R}^N$ of L associated with the zero eigenvalue of L s.t. $\mathbf{r}^{\mathrm{T}}\mathbf{1} = 1$ and $RL + L^{\mathrm{T}}R \succeq 0$, where $R = \operatorname{diag}(r_1, ..., r_N) \in \mathbb{R}^{N\times N}$.*

Definition 2.1 (Generalized algebraic connectivity, [Yu et al., 2010])**.** *Let L be the Laplacian matrix of a strongly connected digraph $\mathcal{G}$. The generalized algebraic connectivity of the graph is defined as*

$$a(L) = \min_{\mathbf{x} \neq \mathbf{0} \text{ and } \mathbf{x} \perp \mathbf{r}} \frac{\mathbf{x}^{\mathrm{T}}(RL + L^{\mathrm{T}}R)\mathbf{x}}{2\mathbf{x}^{\mathrm{T}}R\mathbf{x}}, \tag{2.1}$$

where R is defined in Lemma 2.1. With the above definition, and if the graph is balanced, $a(L) = \lambda_2(L_s)$, where $L_s \triangleq (L + L^{\mathrm{T}})/2$. For undirected graphs, $a(L) = \lambda_2(L)$, where the latter is called the Fiedler eigenvalue of the graph.

2.3 Problem formulation

We consider the problem of synchronizing the trajectories of multiple networked nonlinear systems (agents). We denote by $\mathbf{x}_i \in \mathbb{R}^n$ and $\mathbf{u}_i \in \mathbb{R}^m$ the state and the input of agent i, respectively. Each agent has the nonlinear dynamics given by

$$\dot{\mathbf{x}}_i = A\mathbf{x}_i + \mathbf{f}(\mathbf{x}_i, t) + B\mathbf{u}_i, \tag{2.2}$$

for all $i \in \mathcal{V}$, where A, B have appropriate dimensions. We assume the nonlinear map $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}_{\geq 0} \to \mathbb{R}^n$ is piecewise continuous in t and Lipschitz in $\mathbf{x}$ with Lipschitz constant $l \in \mathbb{R}_{\geq 0}$, that is, for any $\mathbf{y}, \mathbf{z} \in \mathbb{R}^n$, $\|\mathbf{f}(\mathbf{y}, t) - \mathbf{f}(\mathbf{z}, t)\| \leq l\|\mathbf{y} - \mathbf{z}\|$.
We denote by $\mathcal{G}$ the digraph that describes the inter-agent communications topology and assume that $\mathcal{G}$ is strongly connected. Due to the communication constraints imposed by $\mathcal{G}$, each agent is only able to receive information from its in-neighboring agents. The consensus/synchronization problem is stated next.

Problem 2.1 (Consensus/synchronization of MAS)**.** *Consider a MAS where the dynamics of each agent i are described by* (2.2) *and communications among the agents are imposed by a digraph $\mathcal{G}$. Find a distributed protocol for $\mathbf{u}_i = \mathbf{u}_i(\mathbf{x}_i, \mathbf{x}_j, t); j \in \mathcal{N}_i^{\mathrm{in}}, i \in \mathcal{V}$ such that regardless of the initial states $\mathbf{x}_i(t_0), i \in \mathcal{V}$ the agents reach consensus (synchronized) asymptotically, that is, $\mathbf{x}_1(t) = \mathbf{x}_2(t) = \cdots = \mathbf{x}_N(t)$ as $t \to \infty$ and the agents remain synchronized with identical dynamics described by $\dot{\mathbf{x}}_i = A\mathbf{x}_i + \mathbf{f}(\mathbf{x}_i, t)$ for all $i \in \mathcal{V}$ as $t \to \infty$.*

From (2.2), if the consensus/synchronization problem is solved then the input $\mathbf{u}_i(t) \to \mathbf{0}$ as $t \to \infty$ for all $i \in \mathcal{V}$ as well.
It was shown in [Li et al., 2012] that the distributed protocol given by

$$\mathbf{u}_i = cK \sum_{j \in \mathcal{N}_i^{\text{in}}} a_{ij}(\mathbf{x}_i - \mathbf{x}_j) = cK \sum_{j=1}^{N} a_{ij}(\mathbf{x}_i - \mathbf{x}_j) \tag{2.3}$$

for all $i \in \mathcal{V}$, with proper choices of c and K, solves the consensus/synchronization problem. However, the protocol given by (2.3) relies on continuous communications among the agents. This, in turn, requires that the in-neighbors of agent i transmit their states to agent i continuously to update the input $\mathbf{u}_i$. Unfortunately, in practice the communication bandwidth might be limited as all agents might share the same network. This motivated the development of the distributed control scheme with an ETC mechanism described in this chapter with the purpose of reducing the number of messages exchanged and the frequency of communications among the agents. These characteristics are of the utmost importance in applications where the transmission medium imposes stringent communication constraints (e.g. cooperative control of multiple autonomous underwater vehicles (AUVs) [Rego et al., 2019a]).

2.4 The intuition behind the ETC mechanism

Before describing the ETC mechanism for the general consensus/synchronization problem stated in the previous section, we consider a simple but interesting application of the general setup to illustrate the underlying idea behind the ETC mechanism proposed. This application involves the cooperative control of multiple autonomous vehicles, and is illustrated in Fig.2.1. Let (x_I, y_I, z_I) be unit axes of an inertial frame. For simplicity of exposition, we assume that the vehicles only maneuver along x_I as in Fig.2.1. Let $x_i \in \mathbb{R}$ and $v_i \in \mathbb{R}$ denote the coordinate and speed of the vehicles along x_I, respectively, where $i = 1$ for the UAV, $i = 2$ for the ASV, and $i = 3$ for the AUV. With the above assumptions, each vehicle can be considered as a point mass whose motion along x_I is described by the model

$$\dot{x}_i = v_i, \quad \forall i \in \{1, 2, 3\}. \tag{2.4}$$

The cooperative control problem involves finding v_i to make the vehicles coordinated along the $x_{\mathcal{I}}$ axis and "fly" with a common constant desired speed v_{d}. To facilitate the

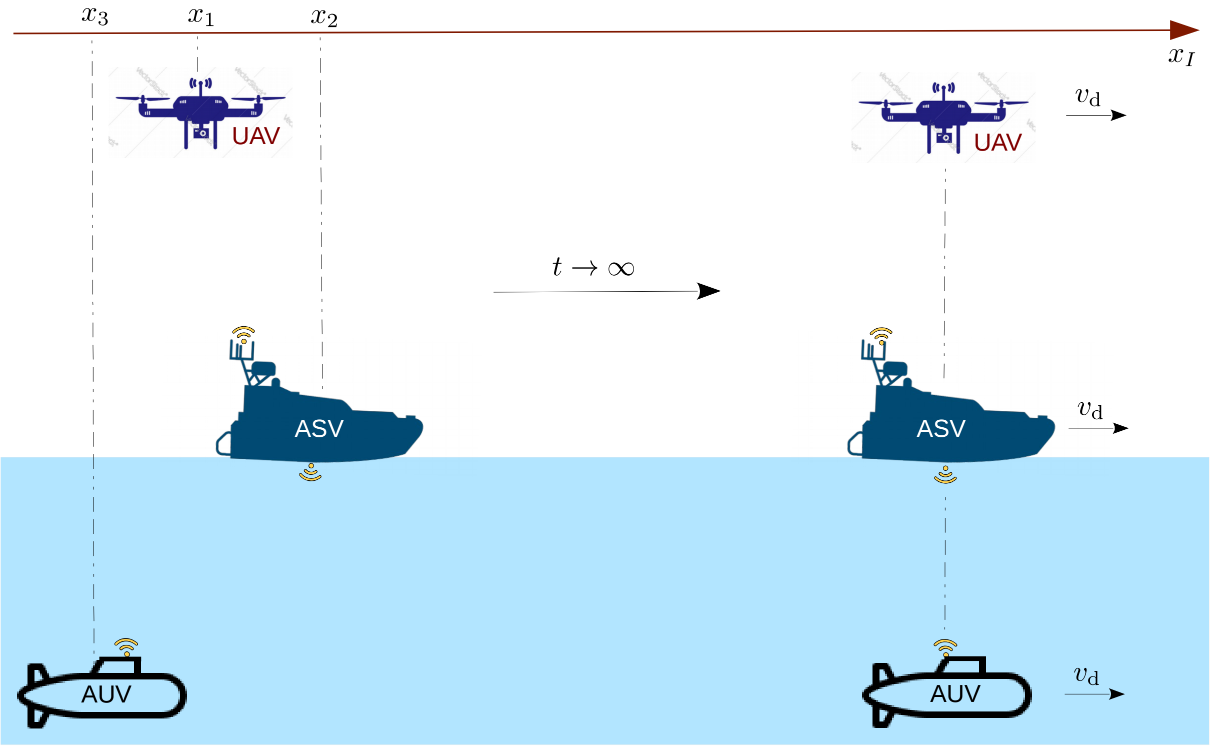


Figure 2.1: Cooperative control of multiple autonomous vehicles

derivation of a control law for v_i the latter can be split in two components as

$$v_i = v_{\mathrm{d}} + u_i \quad \forall i \in \{1, 2, 3\}, \tag{2.5}$$

where $u_i; i \in \{1, 2, 3\}$ are the correction speeds needed to found in order to achieve coordination between the vehicles. Substituting (2.5) in (2.4) yields

$$\dot{x}_i = v_{\mathrm{d}} + u_i, \quad \forall i = 1, 2, 3. \tag{2.6}$$

In this example, the UAV can exchange information with the ASV via a radio link, while the AUV can exchange information with the ASV through an acoustic channel. With this setup, the vehicles' communication network can be modeled by a bidirectional (undirected) graph as in Fig.2.2.
The vehicles' cooperative control problem is stated as follow.

Problem 2.2 (UAV-ASV-AUV cooperative control)**.** *Consider the network of three vehicles where the motion of each is modeled by* (2.6)*. Find a protocol for* $u_i, i \in \{1, 2, 3\}$ *sub-*

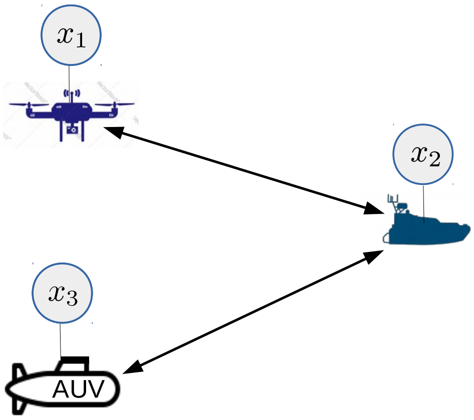


Figure 2.2: The graph induced by the vehicles' network

ject to the constraints imposed by the vehicles' communication network to make the vehicles aligned along x_I and move with a desired constant speed $v_{\rm d}$, i.e. $x_1(t) = x_2(t) = x_3(t)$ and $\dot{x}_1(t) = \dot{x}_2(t) = \dot{x}_3(t) = v_{\rm d}$ as $t \to \infty$, see Fig.2.1.

Clearly, the cooperative control problem stated above is a special case of the consensus/synchronization problem formulated in the previous section. This is also a particular case of the cooperative path following problem described in **Problem 1.1** of Chapter 1, where all the vehicles are already stayed on the desired paths which, in this case, are straight lines parallel to the x_I axis. We will come back and address the general cooperative path following in the next chapter.

The coordination problem stated above can be solved using the protocol (2.3), written explicitly for all vehicles as

$$u_1 = -k(x_1 - x_2), \quad u_2 = -k(2x_2 - x_1 - x_3), \quad u_3 = -k(x_3 - x_2), \tag{2.7}$$

for any $k > 0$. This control protocol implies that in order to compute $u_i; i = \{1, 2, 3\}$ the vehicles must update the state of their neighbors continuously. The ETC mechanism proposed in this chapter aims to relax this strict requirement and also provides an adequate tool to reduce communications among the vehicles while still guarantee the coordination performance.

In the proposed ETC mechanism, the vehicles predict the states of their neighbors and use them in protocol (2.7). This implies that the vehicles are not require to transmit their states to their neighbors continuously; instead, the transmission only happen when found necessary for their neighbors to correct the predictions. Let $\{t_{i,k}\}; i = \{1, 2, 3\}, k \in \mathbb{N}$ denote the sequences of time instants at which the vehicles transmit their states. *The* *main goal of the ETC mechanism is to specify these sequences, i.e. to decide when the vehicles must transmit their states to their neighbors.*

For the simplicity of exposition, we now describe the ETC mechanism involving in the transmission of the ASV's state (x_2), that is, we show how the ETC mechanism specifies the sequence $\{t_{2,k}\}; k \in \mathbb{N}$ for the ASV. For the other vehicles, the ETC mechanism can be derived in a similar manner. To this end, let $\hat{x}_2^1$ and $\hat{x}_2^3$ be the estimate of the ASV's state, predicted by the UAV and the AUV, respectively; see the red variables at the UAV and AUV in Fig.2.3. To predict the ASV's state, the UAV and the AUV can adopt the following simple estimator/predictor:

$$\hat{x}_2^i : \begin{cases} \dot{\hat{x}}_2^i = v_{\mathrm{d}} \\ \hat{x}_2^1(t_{i2,k}^+) = x_2(t_{2,k}) \end{cases} \quad i = 1, 3 \tag{2.8}$$

where $\{t_{i2,k}\}; i = 1, 3, k \in \mathbb{N}$ denote the time instant at which the UAV and AUV receive $x_2(t_{2,k})$. This estimator can be interpreted as follows:

- The first equation of (2.8) is motivated from the fact that if all vehicles get coordinated then all would move with the same desired speed v_{d}.

- The second equation of (2.8) implies that the estimates of the ASV's state are corrected whenever the UAV and the AUV receive the most recent state of the ASV.

Suppose that communication delays are neglected, that is, $t_{i2,k} = t_{2,k}$ for all $i = 1, 3$ and $k \in \mathbb{N}$. Then, (2.8) can be rewritten as

$$\hat{x}_2^i : \begin{cases} \dot{\hat{x}}_2^i = v_{\mathrm{d}} \\ \hat{x}_2^1(t_{2,k}) = x_2(t_{2,k}) \end{cases} \quad i = 1, 3. \tag{2.9}$$

Once the estimates of the ASV's state are available the correction speeds of the UAV and the AUV are updated by

$$u_1 = -k(x_1 - \hat{x}_2^1), \quad u_3 = -k(x_3 - \hat{x}_2^3), \tag{2.10}$$

where, compared with (2.7), the UAV and the AUV use the estimates of the ASV's state,

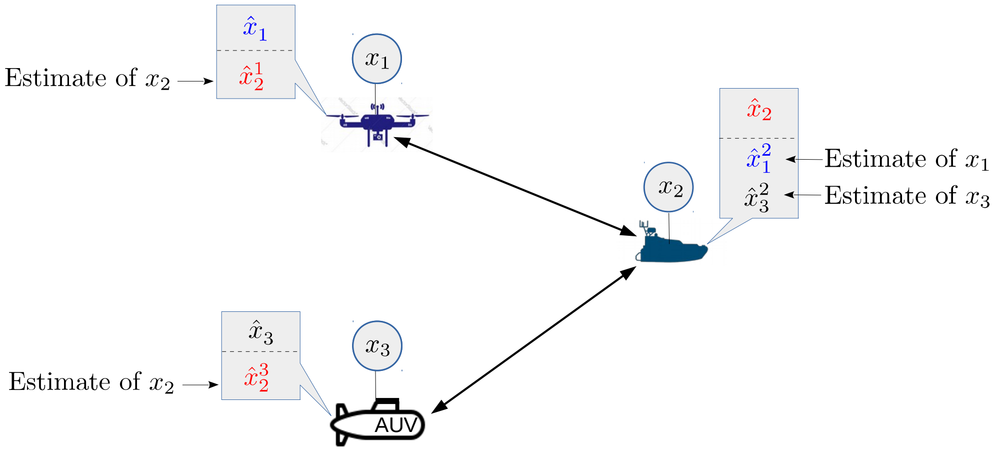


Figure 2.3: The graph induced by the vehicles' network

instead of its true value. To facilitate the analysis u_1 and u_3 in (2.10) can be rewritten as

$$\begin{aligned} u_1 &= -k(x_1 - x_2 - e_2^1) \\ u_3 &= -k(x_3 - x_2 - e_2^3), \end{aligned} \tag{2.11}$$

where

$$e_2^1 \triangleq \hat{x}_2^1 - x_2, \quad \text{and} \quad e_2^3 \triangleq \hat{x}_2^3 - x_2 \tag{2.12}$$

are the estimation errors of the ASV's state at the UAV and the AUV, respectively. The underlying idea behind the ETC mechanism is to find a way to control these estimation errors, forcing them to be bounded in a "small" region in order to guarantee that the UAV and AUV always have "good" predictions of the ASV's state. Notice that due to the second equation of (2.9) these estimation errors are reset to zero whenever the UAV and AUV receive a new state from the ASV. The problem now is how the ASV can "perceive" the estimation errors at the UAV and AUV in order for it to broadcast its state when these errors are large (or exceed a certain threshold). To solve this problem, the ASV runs an estimator identical to the estimators run by the UAV and AUV, given by

$$\hat{x}_2 : \begin{cases} \dot{\hat{x}}_2 = v_{\mathrm{d}} \\ \hat{x}_2(t_{2,k}) = x_2(t_{2,k}), \end{cases} \tag{2.13}$$

where $\hat{x}_2$ is the variable that can be considered as a "copy" of $\hat{x}_2^1$ and $\hat{x}_2^3$; see the red variable at the ASV in Fig. 2.3. Because $\hat{x}_2^1$ and $\hat{x}_2^3$ in (2.9) and $\hat{x}_2$ in (2.13) are initialized at the same value it follows that

$$\hat{x}_2(t) = \hat{x}_2^1(t) = \hat{x}_2^3(t) \tag{2.14}$$

for all t. This implies that $\hat{x}_2$ can help the ASV monitor how good the UAV and the AUV predicts its state. Thus, by letting the ASV broadcast its state whenever

$$|\hat{x}_2 - x_2| \geq \epsilon, \tag{2.15}$$

where $\epsilon \geq 0$ is called as "triggering threshold", then due to (2.12), (2.14), and the reset condition in (2.9), the estimation errors satisfy

$$|e_2^1(t)| = |e_2^3(t)| \leq \epsilon \quad \forall t. \tag{2.16}$$

Thus, by choosing the size of ϵ, we can control the transmission frequency of the ASV's state. Formally, the sequence $\{t_{2,k}\}$, specified by the ETC mechanism is given by

$$t_{2,k+1} = \inf\{t > t_{2,k} : |\hat{x}_2(t) - x_2(t)| \geq \epsilon\}. \tag{2.17}$$

We now apply the ETC mechanism presented above to specify when the UAV and the ASV must transmit their states to the ASV. First, the ASV runs two estimators to predict the states of the UAV and the AUV, given by

$$\hat{x}_1^2 : \begin{cases} \dot{\hat{x}}_1^2 = v_\mathrm{d} \\ \hat{x}_1^2(t_{21,k}^+) = x_1(t_{1,k}) \end{cases} \qquad \hat{x}_3^2 : \begin{cases} \dot{\hat{x}}_3^2 = v_\mathrm{d} \\ \hat{x}_3^2(t_{31,k}^+) = x_3(t_{3,k}), \end{cases} \tag{2.18}$$

where $\hat{x}_1^2$ and $\hat{x}_3^2$ denote the estimates of the UAV and the AUV states, computed by the ASV, respectively; see Fig.2.3. In order to monitor the estimation errors made by these estimations, the UAV and the AUV run two estimators identical to those in (2.18) described by

$$\hat{x}_1 : \begin{cases} \dot{\hat{x}}_1 = v_\mathrm{d} \\ \hat{x}_1(t_{1,k}^+) = x_1(t_{1,k}) \end{cases} \qquad \hat{x}_3 : \begin{cases} \dot{\hat{x}}_3 = v_\mathrm{d} \\ \hat{x}_3(t_{3,k}^+) = x_3(t_{3,k}). \end{cases} \tag{2.19}$$

Recall that without communication delays, (2.19) and (2.18) are identical, that is,

$$\hat{x}_1(t) = \hat{x}_1^2(t) \quad \text{and} \quad \hat{x}_3(t) = \hat{x}_3^2(t), \tag{2.20}$$

for all t. With the ingredients described, the sequences $\{t_{1,k}\}$ and $\{t_{3,k}\}$ can be specified by

$$\begin{aligned} t_{1,k+1} &= \inf\{t > t_{1,k} : |\hat{x}_1(t) - x_1(t)| \geq \epsilon\}, \\ t_{3,k+1} &= \inf\{t > t_{3,k} : |\hat{x}_3(t) - x_3(t)| \geq \epsilon\}. \end{aligned} \tag{2.21}$$

Finally, with the ETC mechanism the input of the ASV is updated as

$$u_2 = -k(2x_2 - \hat{x}_1^2 - \hat{x}_3^2), \tag{2.22}$$

where $\hat{x}_1^2$ and $\hat{x}_3^2$ are given by (2.18).
In the next section, we will generalize the idea presented in this section to design an ETC mechanism for the general consensus/synchronization problem in Section 2.3.

2.5 Consensus/synchronization with ETC mechanism

In this section, we first describe the process of designing an ETC mechanism for the general consensus/synchronization stated in Section 2.3, after which we perform an analysis of the convergence properties of MAS synchronization.

2.5.1 Design of the ETC mechanism

In a ETC mechanism, the control law (2.3) uses, for each agent, the estimates of its in-neighbor states $(\mathbf{x}_j; j \in \mathcal{N}_i^{\text{in}})$, instead of their true states. Let $\hat{\mathbf{x}}_j^i$ be an estimate of $\mathbf{x}_j$ computed by agent i (the procedure to compute this estimate will be explained later). The control law with the ETC mechanism that we propose is given by

$$\mathbf{u}_i = cK \sum_{j \in \mathcal{N}_i^{\text{in}}} a_{ij}(\mathbf{x}_i - \hat{\mathbf{x}}_j^i) \tag{2.23}$$

for all $i \in \mathcal{V}$. The underlying idea in the proposed ETC mechanism is that if $\hat{\mathbf{x}}_j^i$ can provide a "good" estimate of $\mathbf{x}_j$, then the communication among agents does not have to be continuous. We propose the following estimator for $\hat{\mathbf{x}}_j^i$:

$$\hat{\mathbf{x}}_j^i : \begin{cases} \dot{\hat{\mathbf{x}}}_j^i = A\hat{\mathbf{x}}_j^i + \mathbf{f}(\hat{\mathbf{x}}_j^i, t) \\ \hat{\mathbf{x}}_j^i(t_{ij,k}^+) = \mathbf{x}_j(t_{j,k}) \end{cases} \tag{2.24}$$

for $i \in \mathcal{V}$ and $j \in \mathcal{N}_i^{\text{in}}$, where $\{t_{j,k}\}_{k\in\mathbb{N}}$, to be defined, is the sequence of time instants at which agent j broadcasts its state to its out-neighbors, while $\{t_{ij,k}\}_{k\in\mathbb{N}}$ is the corresponding sequence of time instants at which agent i receives updates on the state of agent j; $j \in \mathcal{N}_i^{\text{in}}$. Note that $\{t_{ij,k}\}_{k\in\mathbb{N}}$ is different from $\{t_{j,k}\}_{k\in\mathbb{N}}$ when there is latency in the communications channel. Recall that *the main goal of the ETC mechanism is to specify $\{t_{i,k}\}_{k\in\mathbb{N}}$ for all $i \in \mathcal{V}$, i.e. to decide when the agents must transmit their states to their out-neighbors.* The structure of the estimator (2.24) is motivated by the fact that if synchronization were achieved perfectly, i.e. $\mathbf{x}_i = \mathbf{x}_j$ for all $i, j \in \mathcal{V}$, then the input of each agent would remain at zero, and in this case the estimated variables would be the true states of the agents. Adopting the idea we presented in the simple example in the previous section, in order to control the error between $\mathbf{x}_j$ and $\hat{\mathbf{x}}_j^i$ we define a variable $\hat{\mathbf{x}}_j; j \in \mathcal{V}$ as a "replica" of $\hat{\mathbf{x}}_j^i; i \in \mathcal{N}_j^{\text{out}}$ at agent j. The dynamics of $\hat{\mathbf{x}}_j$ are given by

$$\hat{\mathbf{x}}_j : \begin{cases} \dot{\hat{\mathbf{x}}}_j = A\hat{\mathbf{x}}_j + \mathbf{f}(\hat{\mathbf{x}}_j, t) \\ \hat{\mathbf{x}}_j(t_{j,k}^+) = \mathbf{x}_j(t_{j,k}); \end{cases} \tag{2.25}$$

Clearly, if communication delays are negligible, i.e. $t_{ij,k} = t_{j,k}$ for all k, then it can be seen from (2.24) and (2.25) that $\hat{\mathbf{x}}_j(t) = \hat{\mathbf{x}}_j^i(t)$ for all t. See Fig.2.4 as an illustration of the underlying idea of the ETC mechanism for a network with three agents. Since

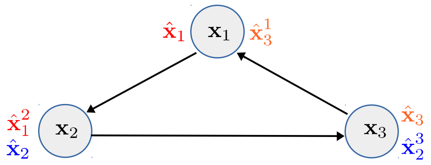


Figure 2.4: An illustrative example: without communication delays, the ETC mechanism ensures that $\hat{\mathbf{x}}_j$ and $\hat{\mathbf{x}}_j^i$ are synchronized, i.e. $\hat{\mathbf{x}}_j(t) = \hat{\mathbf{x}}_j^i(t)$ for all t and $i \in \mathcal{N}_j^{\text{out}}, j \in \mathcal{V}$.

$\hat{\mathbf{x}}_j(t) = \hat{\mathbf{x}}_j^i(t)$, the protocol in (2.23) can be rewritten as

$$\mathbf{u}_i = cK \sum_{j\in\mathcal{N}_i^{\text{in}}} a_{ij}(\mathbf{x}_i - \hat{\mathbf{x}}_j). \tag{2.26}$$

Because $a_{ij} = 0$ if $j \notin \mathcal{N}_i^{\text{in}}$, eq. (2.26) can be rewritten as

$$\mathbf{u}_i = cK \sum_{j=1}^{N} a_{ij}(\mathbf{x}_i - \mathbf{x}_j + \mathbf{e}_j) \tag{2.27}$$

for all $i, j \in \mathcal{V}$, where

$$\mathbf{e}_j \triangleq \mathbf{x}_j - \hat{\mathbf{x}}_j \tag{2.28}$$

for all $j \in \mathcal{V}$. Compared with the protocol for continuous communications given by (2.3), the protocol in (2.27) has the contribution of the estimation error $\mathbf{e}_j$ for all $j \in \mathcal{V}$. The key point in the proposed ETC mechanism is that if $\mathbf{e}_j$ for all $j \in \mathcal{V}$ can be enforced to be bounded then, as we will show later, the synchronization error between agents will also be bounded. This idea was proposed in [Tabuada, 2007], where the author introduced an event-triggered control mechanism to a stabilization problem of nonlinear systems. To ensure that the estimation error $\mathbf{e}_j; j \in \mathcal{V}$ is bounded, we allow agent j to broadcast its state $(\mathbf{x}_j)$ whenever $\|\mathbf{e}_j\|$ reaches a designed bounded *threshold function* $h_j(\cdot)$ that, in general, can be parameterized by time as $h_j(t)$. Formally, for each agent $i; i \in \mathcal{V}$ we define an event-triggering function $\delta_i(t)$ for the communications as

$$\delta_i(t) = \|\mathbf{e}_i(t)\| - h_i(t), \tag{2.29}$$

where $h_i(t)$ belongs to a class of non-negative functions $\mathcal{C}$ defined by $\mathcal{C} := \{f : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0} | c_l \leq f(t) \leq c_u\}$ for all $i \in \mathcal{V}$. For example, $h_i(t) = c_1 + c_2 e^{-\alpha t}$, where $c_1, c_2, \alpha \in \mathbb{R}_{\geq 0}$ are constant parameters, see [Seyboth et al., 2013, Hung et al., 2019]. With the above definition, agent $i; i \in \mathcal{V}$ will transmit its state to its out-neighbors whenever $\delta_i(t) \geq 0$. Formally, the time sequence $\{t_{i,k}\}$ is specfied by the ETC mechanism as

$$t_{i,k+1} = \inf\{t > t_{i,k} : \delta_i(t) \geq 0\}. \tag{2.30}$$

At this point, we have the necessary ingredients to summarize the proposed ETC framework for the consensus/synchronization problem described above. The resulting procedure is summarized in Algorithm 2.1. Inspired from [Li et al., 2012], the gains c and K in protocol (2.23) can be computed using Algorithm 2.2. The rationale behind Algorithm 2.2 will become clear in the next sub-section.

Algorithm 2.1 ETC mechanism for agent i

```
At every time $t$, agent $i$ implements the following procedure:
procedure COORDINATION AND COMMUNICATION
    if $\delta_i(t) \geq 0$ then
        Broadcast $\mathbf{x}_i(t)$
        Reset $\hat{\mathbf{x}}_i$ using (2.25);
    if Receive a new message from agent $j$ then
        if $j \in \mathcal{N}_i^{\text{in}}$ then
            Reset $\hat{\mathbf{x}}_j^i$ using (2.24);
    Run the estimators (2.24) and (2.25);
    Update the protocol for $\mathbf{u}_i$ using (2.23);
return $\mathbf{u}_i$
```

Algorithm 2.2 Selecting the control gains in (2.23)

```
procedure CHOSE $K$
    Solve the following linear matrix inequality (LMI) for variables $P$, $\tau$, and $\mu$:
        $P = P^{\mathrm{T}}, P \succ 0,$    (2.31a)
        $\tau > 0, \mu > 0,$    (2.31b)
        $\begin{bmatrix} AP + PA^{\mathrm{T}} - 2\tau BB^{\mathrm{T}} + \mu l I_n & P \\ P & -\mu I_n/l \end{bmatrix} \prec 0$    (2.31c)
    Chose matrix $K = -B^{\mathrm{T}}P^{-1}$
procedure CHOSE $c$
    Compute $a(L)$ given by (2.1).
    Chose any coupling gain $c \geq \tau/a(L)$.
```

2.5.2 Convergence analysis

We now analyze the synchronization properties of the closed-loop MAS with the ETC algorithm given in Algorithm 2.1. To this end, let

$$\boldsymbol{\xi}_i \triangleq \mathbf{x}_i - \sum_{j=1}^{N} r_j \mathbf{x}_j, \tag{2.32}$$

where r_j is the j^{th} component of the positive left eigenvector $\mathbf{r}$ of the graph Laplacian matrix defined in Lemma 2.1. Note that if the graph is connected and balanced, then $\mathbf{r} = \mathbf{1}/N$. In this case, $\boldsymbol{\xi}_i$ measures the disagreement between agent i's state and the average of all agents' state. Let also $\boldsymbol{\xi} \triangleq [\boldsymbol{\xi}_1^{\mathrm{T}}, ..., \boldsymbol{\xi}_N^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{nN}$ be the synchronization error vector that captures the disagreement among the agents' states. From (2.32), $\boldsymbol{\xi}$ can be rewritten as

$$\boldsymbol{\xi} = W\mathbf{x}, \tag{2.33}$$

where $\mathbf{x} \triangleq [\mathbf{x}_1^{\mathrm{T}}, ..., \mathbf{x}_N^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{nN}$ and

$$W \triangleq (I_N - \mathbf{1}\mathbf{r}^{\mathrm{T}}) \otimes I_n. \tag{2.34}$$

With the above definition, it is clear that all agents are synchronized, that is, $\mathbf{x}_1 = \mathbf{x}_2 = ... = \mathbf{x}_N$ if and only if $\boldsymbol{\xi} = \mathbf{0}$. Therefore, to analyze the synchronization of the MAS, we analyze the convergence of the synchronization error $\boldsymbol{\xi}$ to zero. The main results of the chapter are summarized and stated mathematically next.

Theorem 2.1. *Consider the closed-loop multi-agent system described by* (2.2)*, driven by the distributed control strategy and the ETC mechanism specified in Algorithm 2.1. Suppose further that there exists a solution to the LMI* (2.31)*. Let also* $\mathbf{h} \triangleq [h_1, ..., h_N]^{\mathrm{T}}$ *be the vector-valued function containing all threshold functions. Then, the following statements hold true.*

(i) There exist a $\mathcal{KL}$ *class function* β *and a* $\mathcal{K}$ *class function* γ *such that for any initial state* $\boldsymbol{\xi}(t_0)$ *the synchronization error* $\boldsymbol{\xi}$ *satisfies*

$$\|\boldsymbol{\xi}(t)\| \le \beta(\|\boldsymbol{\xi}(t_0)\|, t - t_0) + \gamma(\sup_{t_0 \le \tau \le t} \|\mathbf{h}(\tau)\|). \tag{2.35}$$

(ii) If the threshold functions are designed such that $\lim_{t\to\infty} h_i(t) = c_l$ *for all* $i \in \mathcal{V}$*, then*

$$\lim_{t\to\infty} \|\boldsymbol{\xi}(t)\| \le r_1 = \frac{2\|F_1\|\lambda_{\max}(R \otimes P^{-1})}{\lambda_{\min}(H_2)\lambda_{\min}(R \otimes P^{-1})}\sqrt{N}c_l, \tag{2.36}$$

where

$$F_1 \triangleq (R \otimes P^{-1})W(\mathcal{A} \otimes cBB^{\mathrm{T}}P^{-1}) \tag{2.37}$$

and

$$H_2 = -(I_N \otimes P)(R \otimes H_1)(I_N \otimes P), \tag{2.38}$$

with

$$H_1 \triangleq AP + PA^{\mathrm{T}} - 2\tau BB^{\mathrm{T}} + \mu l I_n + \mu^{-1} l P^2 \tag{2.39}$$

(iii) If all threshold functions $h_i; i \in \mathcal{V}$ are lower bounded by $c_l > 0$, then the closed-loop MAS system does not exhibit Zeno behavior.

The result stated in *i)* indicates that the synchronization error vector is input-to state stable (ISS) with respect to the input $\mathbf{h}$ (see the definition of ISS in Definition A.1, Appendix A). This also implies that the synchronization error is bounded for any bounded threshold functions that, in the context of the ETC mechanism, are user designed functions used as tuning knobs to trade off communication rate against synchronization performance. The result in *ii)* is a consequence of *i)* and indicates that if the threshold functions converge, then the synchronization error vector converges as well. The size of the ball that the coordination error $\boldsymbol{\xi}$ converges to depends explicitly on the lower bound of the threshold functions. Theorem 2.1 is proved as follows:

i) To show the inequality in (2.35) we analyze the closed-loop dynamics of the synchronization error vector $\boldsymbol{\xi}$. First, substituting the ETC protocol for $\mathbf{u}_i$ given by (2.27) in (2.2) yields

$$\dot{\mathbf{x}} = (I_N \otimes A + cL \otimes BK)\mathbf{x} + \mathbf{g}(\mathbf{x}, t) + \boldsymbol{\eta}, \tag{2.40}$$

where

$$\mathbf{g}(\mathbf{x}, t) \triangleq \begin{bmatrix} \mathbf{f}(\mathbf{x}_1, t) \\ \vdots \\ \mathbf{f}(\mathbf{x}_N, t) \end{bmatrix}, \quad \boldsymbol{\eta} \triangleq \begin{bmatrix} cBK \sum_{j=1}^{N} a_{1j}\mathbf{e}_j \\ \vdots \\ cBK \sum_{j=1}^{N} a_{Nj}\mathbf{e}_j \end{bmatrix}. \tag{2.41}$$

Thus, from (6.55) and (2.40) the closed-loop dynamics of $\boldsymbol{\xi}$ are given by

$$\dot{\boldsymbol{\xi}} = W\dot{\mathbf{x}} = (I_N \otimes A + cL \otimes BK)\boldsymbol{\xi} + W\left(\mathbf{g}(\mathbf{x}, t) + \boldsymbol{\eta}\right), \tag{2.42}$$

where we used the fact that $W(I_N \otimes A + cL \otimes BK) = (I_N \otimes A + cL \otimes BK)W$. We now consider the Lyapunov function candidate defined as

$$V = \boldsymbol{\xi}^{\mathrm{T}}(R \otimes P^{-1})\boldsymbol{\xi}, \tag{2.43}$$

where P is given by Algorithm 2.2 and R is defined in Lemma 2.1. Since $P \succ 0$ and R is a positive diagonal matrix, V is clearly positive definite. Its time derivative along the trajectory of (2.42) is given by

$$\dot{V} = \underbrace{2\boldsymbol{\xi}^{\mathrm{T}}(R \otimes P^{-1})(I_N \otimes A + cL \otimes BK)\boldsymbol{\xi}}_{\triangleq X} + \underbrace{2\boldsymbol{\xi}^{\mathrm{T}}(R \otimes P^{-1})W\mathbf{g}(\mathbf{x}, t)}_{\triangleq Y} + \underbrace{2\boldsymbol{\xi}^{\mathrm{T}}(R \otimes P^{-1})W\boldsymbol{\eta}}_{\triangleq Z} \tag{2.44}$$

The term X in (2.44) can be rewritten as

$$X = 2\boldsymbol{\xi}^{\mathrm{T}}(R \otimes P^{-1}A + cRL \otimes P^{-1}BK)\boldsymbol{\xi}. \tag{2.45}$$

Let $\boldsymbol{\zeta} \triangleq (I_N \otimes P^{-1})\boldsymbol{\xi}$. Substituting $K = -B^{\mathrm{T}}P^{-1}$ defined in Algorithm 2.2 in (2.45) yields

$$X = \boldsymbol{\zeta}^{\mathrm{T}} \left(R \otimes (AP + PA^{\mathrm{T}})\right) \boldsymbol{\zeta} - \boldsymbol{\zeta}^{\mathrm{T}} \left(c(RL + L^{\mathrm{T}}R) \otimes BB^{\mathrm{T}}\right) \boldsymbol{\zeta}. \tag{2.46}$$

Because $(\mathbf{r}^{\mathrm{T}} \otimes I_n)\boldsymbol{\xi} = \mathbf{0}$, $(\mathbf{r}^{\mathrm{T}} \otimes I_n)\boldsymbol{\zeta} = \mathbf{0}$. Therefore, using Definition 2.1, we obtain

$$\boldsymbol{\zeta}^{\mathrm{T}} \left(c(RL + L^{\mathrm{T}}R) \otimes BB^{\mathrm{T}}\right) \boldsymbol{\zeta} \geq 2\boldsymbol{\zeta}^{\mathrm{T}}(R \otimes ca(L)BB^{\mathrm{T}})\boldsymbol{\zeta}.$$

Thus, X in (2.46) is bounded as

$$X \leq \boldsymbol{\zeta}^{\mathrm{T}} \left(R \otimes (AP + PA^{\mathrm{T}} - 2ca(L)BB^{\mathrm{T}})\right) \boldsymbol{\zeta}. \tag{2.47}$$

Furthermore, if c is chosen such that $c \geq \tau/a(L)$ with $\tau > 0$, as given by Algorithm 2.2, then

$$X \leq \boldsymbol{\zeta}^{\mathrm{T}} \left(R \otimes (AP + PA^{\mathrm{T}} - 2\tau BB^{\mathrm{T}}\right) \boldsymbol{\zeta}. \tag{2.48}$$

Let $\bar{\mathbf{x}} \triangleq \sum_{i=1}^{N} r_i \mathbf{x}_i$. The term Y in (2.44) can be expanded as

$$\begin{aligned} Y &= 2\sum_{i=1}^{N} r_i \boldsymbol{\xi}_i^{\mathrm{T}} P^{-1} \left(\mathbf{f}(\mathbf{x}_i, t) - \sum_{j=1}^{N} r_j \mathbf{f}(\mathbf{x}_j, t)\right) \\ &= \underbrace{2\sum_{i=1}^{N} r_i \boldsymbol{\xi}_i^{\mathrm{T}} P^{-1} \left(\mathbf{f}(\mathbf{x}_i, t) - \mathbf{f}(\bar{\mathbf{x}}, t)\right)}_{\triangleq Y_1} + \underbrace{2\sum_{i=1}^{N} r_i \boldsymbol{\xi}_i^{\mathrm{T}} P^{-1} \left(\mathbf{f}(\bar{\mathbf{x}}, t) - \sum_{j=1}^{N} r_j \mathbf{f}(\mathbf{x}_j, t)\right)}_{\triangleq Y_2}. \end{aligned}$$

Because $\sum_{i=1}^{N} r_i\boldsymbol{\xi}_i = 0$, it follows that $Y_2 = 0$. Thus, using the Lipschitz assumption on $\mathbf{f}(\cdot)$ and Young's inequality, we obtain

$$\begin{aligned} Y = Y_1 &\leq 2\sum_{i=1}^{N} r_i l \|\boldsymbol{\xi}_i^{\mathrm{T}} P^{-1}\|\|\boldsymbol{\xi}_i\| \leq \sum_{i=1}^{N} r_i l \boldsymbol{\xi}_i^{\mathrm{T}} \left(\mu (P^{-1})^2 + \mu^{-1} I_n\right) \boldsymbol{\xi}_i \\ &= \boldsymbol{\zeta}^{\mathrm{T}} \left(R \otimes \left(\mu l I_n + \mu^{-1} l P^2\right)\right) \boldsymbol{\zeta} \end{aligned} \tag{2.49}$$

for every $\mu > 0$.

We now compute the upper bound for the term Z in (2.44). Defining $\mathbf{e} = [\mathbf{e}_1^{\mathrm{T}}, ..., \mathbf{e}_N^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{nN}$, $\boldsymbol{\eta}$ can be rewritten from (2.41) as

$$\boldsymbol{\eta} = (\mathcal{A} \otimes cBK)\mathbf{e} = -(\mathcal{A} \otimes cBB^{\mathrm{T}}P^{-1})\mathbf{e}, \tag{2.50}$$

where $\mathcal{A}$ is the adjacency matrix of the graph $\mathcal{G}$. Substituting (2.50) to Z in (2.44) we obtain

$$\begin{aligned} Z &\leq 2\|\boldsymbol{\xi}\|\|(R \otimes P^{-1})W(\mathcal{A} \otimes cBB^{\mathrm{T}}P^{-1})\|\|\mathbf{e}\| \\ &= 2\|\boldsymbol{\xi}\|\|F_1\|\|\mathbf{e}\|, \end{aligned} \tag{2.51}$$

where F_1 is given by (2.37). Thus, from (2.48),(2.49) and (2.51) the time derivative of V in (2.44) is upper bounded as

$$\dot{V} \leq \boldsymbol{\zeta}^{\mathrm{T}}(R \otimes H_1)\boldsymbol{\zeta} + 2\|\boldsymbol{\xi}\|\|F_1\|\|\mathbf{e}\|, \tag{2.52}$$

where H_1 is given by (2.39). Note that H_1 is symmetric. Furthermore, from the LMI (2.31c) in Algorithm 2.2, H_1 is also negative definite (this can be seen by using Schur's complement to rewrite H_1 in the form of LMI (2.31c)). Because $\boldsymbol{\zeta} = (I_N \otimes P^{-1})\boldsymbol{\xi}$, $\boldsymbol{\xi} = (I_N \otimes P)\boldsymbol{\zeta}$. Inserting $\boldsymbol{\xi}$ and H_2 defined in (2.38) in (2.52), we obtain

$$\begin{aligned} \dot{V} &\leq -\boldsymbol{\xi}^{\mathrm{T}} H_2 \boldsymbol{\xi} + 2\|\boldsymbol{\xi}\|\|F_1\|\|\mathbf{e}\| \leq -\lambda_{\min}(H_2)\|\boldsymbol{\xi}\|^2 + 2\|\boldsymbol{\xi}\|\|F_1\|\|\mathbf{e}\| \\ &\leq -\alpha_{\boldsymbol{\xi}}(\|\boldsymbol{\xi}\|) \quad \forall \|\boldsymbol{\xi}\| \geq \rho(\|\mathbf{e}\|) \end{aligned} \tag{2.53}$$

where $\alpha_{\boldsymbol{\xi}} \in \mathcal{K}$ and $\rho \in \mathcal{K}_\infty$ are functions defined by $\alpha_{\boldsymbol{\xi}}(r) = (1-\theta)\lambda_{\min}(H_2)r^2$ and $\rho(r) = \frac{2\|F_1\|}{\theta\lambda_{\min}(H_2)}r$, respectively, and $\theta \in (0, 1)$. Invoking the ISS-Theorem (see Theorem A.1, Appendix A), we conclude that V, given by (2.43), is an ISS-Lyapunov function for the synchronization error vector system given by (2.42) and the system is input to state stable (ISS) with respect to the state $\boldsymbol{\xi}$ and the input $\mathbf{e}$. This implies that there exist functions $\beta \in \mathcal{KL}, \gamma \in \mathcal{K}$ such that for any initial state $\boldsymbol{\xi}(t_0)$ the synchronization error vector satisfies

$$\|\boldsymbol{\xi}(t)\| \leq \beta(\|\boldsymbol{\xi}(t_0)\|, t-t_0) + \gamma(\sup_{t_0 \leq \tau \leq t} \|\mathbf{e}(\tau)\|) \tag{2.54}$$

for all $t \geq t_0$. Furthermore, with the ETC mechanism the error $\mathbf{e}_i$ satisfies $\|\mathbf{e}_i(t)\| \leq h_i(t)$ for all $t \geq t_0$, hence $\|\mathbf{e}(t)\| \leq \|\mathbf{h}(t)\|$ for all $t \geq t_0$. Substituting this result in (2.54), we conclude that the inequality (2.35) holds true.

ii) We now prove the second statement in the theorem. For this purpose, we define two class $\mathcal{K}$ functions $\alpha_1(\|\boldsymbol{\xi}\|) = \lambda_{\min}(R \otimes P^{-1})\|\boldsymbol{\xi}\|^2$ and $\alpha_2(\|\boldsymbol{\xi}\|) = \lambda_{\max}(R \otimes P^{-1})\|\boldsymbol{\xi}\|^2$. Clearly, the Lyapunov function in (2.43) satisfies $\alpha_1(\|\boldsymbol{\xi}\|) \leq V \leq \alpha_2(\|\boldsymbol{\xi}\|)$. Hence, using again Theorem 4.19 in [Khalil, 2002] we obtain

$$\gamma(r) = \alpha_1^{-1}(\alpha_2(\rho(r))) = \frac{2\|F_1\|\lambda_{\max}(R \otimes P^{-1})}{\theta\lambda_{\min}(H_2)\lambda_{\min}(R \otimes P^{-1})} r. \tag{2.55}$$

Furthermore, if $\lim_{t\to\infty} h_i(t) = c_l$ for all $i \in \mathcal{V}$, then $\lim_{t\to\infty} \|\mathbf{h}(t)\| = \sqrt{N}c_l$. Substituting this relation and (2.55) in (2.35) we conclude that the synchronization error $\boldsymbol{\xi}$ satisfies (2.36).

iii) We now prove the third statement of the theorem by showing that the minimum inter-event time for every agent is strictly positive if $c_l > 0$. To this end, let $t_{i,k}$ and $t_{i,k+1}$ be successive triggering times at which agent i sends its state to its out-neighboring agents. We consider the evolution of the estimation error $\mathbf{e}_i(t)$ during the interval $\mathcal{T}_{i,k} \triangleq [t_{i,k}, t_{i,k+1})$ when $\mathbf{e}_i(t)$ is continuous. It follows from (2.28) that $\dot{\mathbf{e}}_i(t) = \dot{\mathbf{x}}_i - \dot{\hat{\mathbf{x}}}_i = A\mathbf{e}_i + (\mathbf{f}(\mathbf{x}_i, t) - \mathbf{f}(\hat{\mathbf{x}}_i, t)) + B\mathbf{u}_i$. Furthermore, $\mathbf{e}_i(t_{i,k}^+) = 0$ because $\hat{\mathbf{x}}_i(t_{i,k}^+) = \mathbf{x}_i(t_{i,k}^+)$ (see (2.25)). Hence,

$$\begin{aligned}\|\mathbf{e}_i(t)\| &\leq \int_{t_{i,k}}^t \|A\mathbf{e}_i(\tau)\|\mathrm{d}\tau + \int_{t_{i,k}}^t \|\mathbf{f}(\mathbf{x}_i(\tau), \tau) - \mathbf{f}(\hat{\mathbf{x}}_i(\tau), \tau)\|\mathrm{d}\tau + \int_{t_{i,k}}^t \|B\mathbf{u}_i(\tau)\|\mathrm{d}\tau \\ &\leq \int_{t_{i,k}}^t (\|A\| + l)\|\mathbf{e}_i(\tau)\|\mathrm{d}\tau + \int_{t_{i,k}}^t \|B\mathbf{u}_i(\tau)\|\mathrm{d}\tau\end{aligned} \tag{2.56}$$

for all $t \in \mathcal{T}_{i,k}$. Note that the last inequality follows from the Lipschitz property of $\mathbf{f}(\cdot)$. To find an upper bound for $\mathbf{e}_i$ we now compute an upper bound for $\mathbf{u}_i(t)$ for all $i \in \mathcal{V}$. Let $\mathbf{u} := [\mathbf{u}_1^{\mathrm{T}}, ..., \mathbf{u}_N^{\mathrm{T}}]^{\mathrm{T}}$. Substituting $K = -B^{\mathrm{T}}P^{-1}$ to (2.27) we obtain

$$\begin{aligned}\mathbf{u} &= -(cL \otimes BB^{\mathrm{T}}P^{-1})\mathbf{x} - (\mathcal{A} \otimes BB^{\mathrm{T}}P^{-1})\mathbf{e} \\ &= -G\boldsymbol{\xi} - T\mathbf{e},\end{aligned} \tag{2.57}$$

where $G \triangleq (cL \otimes BB^{\mathrm{T}}P^{-1})$ and $T \triangleq (\mathcal{A} \otimes BB^{\mathrm{T}}P^{-1})$. Observe that for all $i \in \mathcal{V}$

$$\|\mathbf{u}_i(t)\| \leq \|\mathbf{u}(t)\| \overset{(2.57)}{\leq} \|G\|\|\boldsymbol{\xi}(t)\| + \|T\|\|\mathbf{e}(t)\|. \tag{2.58}$$

Recall also that $\|\mathbf{e}(t)\| \leq \|\mathbf{h}(t)\| \leq \sqrt{N}c_u$ for all $t \geq t_0$, where c_u is the upper bound for $h_i(t)$. Hence, it follows from (2.35) and (2.58) that

$$\|\mathbf{u}_i(t)\| \leq \bar{u} \triangleq \|G\| \left(\beta(\|\boldsymbol{\xi}(t_0)\|, 0) + \gamma(\sqrt{N}c_u)\right) + \|T\|\sqrt{N}c_u$$

for al $t \geq t_0$. Since $\beta \in \mathcal{KL}$, $\bar{u}$ only depends on the initial condition of $\boldsymbol{\xi}$, $\bar{u}$ is the upper bound for $u_i(t)$ for all $t \geq t_0$ and $i \in \mathcal{V}$. Therefore, from (2.56), $\mathbf{e}_i(t)$ can be bounded as

$$\|\mathbf{e}_i(t)\| \leq \int_{t_{i,k}}^{t} (\|A\| + l)\|\mathbf{e}_i(\tau)\|\mathrm{d}\tau + (t - t_{i,k})\|B\|\bar{u}.$$

Let

$$y(t) = \|\mathbf{e}_i(t)\|, \quad \lambda(t) = (t - t_{i,k})\|B\|\bar{u}, \quad \text{and } \mu = \|A\| + l.$$

Applying Gronwall-Bellman inequality (see Lemma A.1 in [Khalil, 2002]) and note that μ is a constant, we obtain

$$y(t) \leq \lambda(t) + \mu \int_{t_{i,k}}^{t} \lambda(s)e^{\mu(t-s)}\mathrm{d}s = \lambda(t) - \int_{t_{i,k}}^{t} \lambda(s)\mathrm{d}e^{\mu(t-s)}.$$

Applying the integration by parts for the last equality, we obtain

$$\begin{aligned} y(t) \leq \lambda(t) - \int_{t_{i,k}}^{t} \lambda(s)\mathrm{d}e^{\mu(t-s)} &= \lambda(t) - \lambda(s)e^{\mu(t-s)}\Big|_{t_{i,k}}^{t} + \|B\|\bar{u}\int_{t_{i,k}}^{t} e^{\mu(t-s)}\mathrm{d}s \\ &= \lambda(t) - \lambda(t) + \frac{\|B\|\bar{u}}{\mu}(e^{\mu(t-t_{i,k})} - 1) \\ &= \|B\|\bar{u}\left(e^{(\|A\|+l)(t-t_{i,k})} - 1\right) / \left(\|A\| + l\right). \end{aligned}$$

Hence,

$$\|\mathbf{e}_i(t)\| \leq \underbrace{\|B\|\bar{u}\left(e^{(\|A\|+l)(t-t_{i,k})} - 1\right) / \left(\|A\| + l\right)}_{\triangleq \Delta(t)}. \tag{2.59}$$

Because a broadcast event for agent i is triggered if and only if $\delta_i(t)$ crosses zero or $\|\mathbf{e}_i(t)\| = h_i(t)$, the next event is triggered not earlier than time $t^* > t_{i,k}$, given by the solution of the equation $\Delta(t) = c_l$. Hence, the minimum inter-event time for any agent is determined by

$$\tau_1 := t^* - t_{i,k} = \frac{\ln\left(1 + c_l(\|A\| + l)/(\|B\|\bar{u})\right)}{\|A\| + l} > 0. \tag{2.60}$$

Since there is a positive lower bound τ_1 on the inter-event intervals, there are no accumulation points in the event sequences and therefore Zeno behavior is excluded. This completes the proof of Theorem 2.1. ■

Remark 2.2. *Note that the LMI* (2.31) *in Algorithm 2.2 is equivalent to the LMI in [Li et al., 2012] (see eq. (6) in [Li et al., 2012]). The feasibility of the LMI in* (2.31) *was discussed in [Li et al., 2012]. For the case of linear MAS systems or when the Lipschitz constant* $l = 0$*, the LMI in* (2.31) *is equivalent to finding* $\tau > 0$ *and* $P \succ 0$ *such that* $AP + PA^{\mathrm{T}} - 2\tau BB^{\mathrm{T}} \prec 0$*. The feasibility of the inequalities is equivalent to the pair* (A, B) *being controllable.*

2.6 Extensions and unified results

In the previous section, the distributed protocol given by (2.23) uses the true state of agent i ($\mathbf{x}_i$) to update $\mathbf{u}_i$. In this section, we analyze the synchronization of MAS by using the following protocol

$$\mathbf{u}_i = cK \sum_{j \in \mathcal{N}_i^{\mathrm{in}}} a_{ij}(\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j^i), \tag{2.61}$$

where $\hat{\mathbf{x}}_i$ and, $\hat{\mathbf{x}}_j^i$ are given by given by (2.25) and (2.24), respectively, for all $i, j \in \mathcal{V}$. With this protocol, we will see that the result of our ETC mechanism in this section generalizes some of the existing results in the literature (for example [Seyboth et al., 2013, Hung et al., 2019]) where the agents' dynamics and the network topologies are special cases of the problem considered in the chapter. We obtain the following result

Theorem 2.2. *Consider the closed-loop MAS described by* (2.2)*, driven by the distributed control strategy and the ETC mechanism specified in Algorithms 2.1 and 2.2, where the distributed protocol given by* (2.61) *is used rather than* (2.23)*. Then, the following statements hold true.*

(i) The properties i) and iii) stated in Theorem 2.1 are satisfied.

(ii) Furthermore, if the threshold functions are designed such that $\lim_{t\to\infty} h_i(t) = c_l$ *for all* $i \in \mathcal{V}$*, then*

$$\lim_{t\to\infty} \|\boldsymbol{\xi}(t)\| \leq r_2 = \frac{2\|F_2\|\lambda_{\max}(R \otimes P^{-1})}{\lambda_{\min}(H_2)\lambda_{\min}(R \otimes P^{-1})}\sqrt{N}c_l, \tag{2.62}$$

where

$$F_2 \triangleq (R \otimes P^{-1})W(L \otimes cBB^{\mathrm{T}}P^{-1}), \tag{2.63}$$

and H_2 *is given by* (2.38)*.*

Proof: The proof is done similarly to the proof of Theorem 2.1. ■

It can be observed from (2.36) and (2.62) that r_1 is computed using the adjacency matrix $\mathcal{A}$, whereas r_2 is computed using the laplacian matrix L (see the difference between F_1 and F_2). Basically, the two bounds are equivalent as both matrices represent the connectivity of the considered digraph.

We next show that the asymptotic bound given by (2.62) generalizes some existing results in the literature.

2.6.1 Average consensus [Seyboth et al., 2013]

In [Seyboth et al., 2013], the authors considered the average consensus problem for undirected graphs where the dynamics of each agent is given by

$$\dot{\mathbf{x}}_i = \mathbf{u}_i \tag{2.64}$$

for all $i \in \mathcal{V}$ and $\mathbf{x}_i, \mathbf{u}_i \in \mathbb{R}$. Compared with the general form given by (2.2), the system (2.64) has $n = m = 1$, $A = 0, B = 1$ and $l = 0$. With these parameters, the LMI given by (2.31) is feasible for every $\tau > 0$ and $P > 0$. Thus, we can chose $\tau = \lambda_2(L) > 0, c = 1$ and $P = 1$ to satisfy Algorithm (2.2). Furthermore, because the graph considered is undirected, it follows from Lemma 2.1 that $R = I_N/N$. Substituting the above parameters in H_2 given by (2.38) and in F_2 given by (2.63) we obtain $H_2 = 2\lambda_2(L)I_N$ and $F_2 = L$. In addition, the authors use the threshold functions $h_i(t) = c_1e^{-\alpha t} + c_0$ for all $i \in \mathcal{V}$, hence

$c_l = c_0$. Inserting the above parameters in (2.62) we obtain

$$r_2 = \|L\|\sqrt{N}c_0/\lambda_2(L), \tag{2.65}$$

which is identical to the bound in Theorem 3.2 (see eq. (8) in [Seyboth et al., 2013]).

2.6.2 A simple nonlinear system [Hung et al., 2019]

In [Hung et al., 2019], the authors considered a synchronization problem for balanced digraphs where the dynamics of each agent are given by

$$\dot{\mathbf{x}}_i = \mathbf{f}(\mathbf{x}_i, t) + \mathbf{u}_i \tag{2.66}$$

for all $i \in \mathcal{V}$ and $\mathbf{x}_i, \mathbf{u}_i \in \mathbb{R}$. Compared with the general form given by (2.2) the system in (2.66) has $n = m = 1$, $A = 0, B = 1$. In this case, the LMI given by (2.31) is equivalent to $\tau, \mu, P > 0$ and $-2\tau + \mu l + \mu^{-1} l P^2 < 0$. By choosing $P = 1$ and $\mu = 1$ any $\tau > l$ will satisfy the LMI. Since the digraph is balanced, $a(L) = \lambda_2(L_s)$, where L_s is defined in Definition 1. Furthermore, by taking $c = \tau/a(L) = \tau/\lambda_2(L_s) > l/\lambda_2(L_s)$ so as to satisfy Algorithm 2.2, this choice of c also satisfies the gain condition given by Theorem 1 in [Hung et al., 2019]. Substituting the above parameters in H_2 given by (2.38) and in F_2 given by (2.63) we obtain $H_2 = 2(c\lambda_2(L_s) - l)I_N$ and $F_2 = cL$. Inserting the above parameters in (2.62) we obtain

$$r_2 = c\|L\|\sqrt{N}c_l/(c\lambda_2(L_s) - l), \tag{2.67}$$

which is identical to the bound given by Corollary 1 in [Hung et al., 2019]). Notice that the coupling gain c in (2.67) plays the same role as k in [Hung et al., 2019].

2.7 Simulation examples

This section illustrates the performance of the proposed ETC mechanism using computer simulations. We consider a network of six agents whose topology is modeled by the digraph illustrated in Fig.2.5. Let $\mathbf{x}_i = [x_{i,1}, x_{i,2}, x_{i,3}, x_{i,4}]^{\mathrm{T}} \in \mathbb{R}^4$ be the state of agent i.

The dynamics of each agent are given by (2.2) with

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -2 & -1 & 2 & 0 \\ 0 & 0 & 0 & 2 \\ 1.95 & 0 & -1.95 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}^{\mathrm{T}},$$

and $\mathbf{f}(\mathbf{x}_i, t) = [0\ 0\ 0\ \text{-}0.333\sin(x_{i3})]^{\mathrm{T}}$. Therefore, the corresponding globally Lipchitz constant for $\mathbf{f}(\cdot)$ is $l = 0.333$. Running Algorithm 2.2 using the CVX tool box ([Grant and Boyd, 2014]), we obtain $c = 2251$ and $K = [\text{-}0.0470,\ \text{-}0.0134,\ 0.0285,\ \text{-}0.0224]$. For the threshold functions, we set $h_i(t) = c_0 + c_1 e^{-c_2 t}$ for all $i \in \mathcal{V} := \{1, ..., 6\}$, with $c_0 = 1e\text{-}3, c_1 = 5$, and $c_2 = 1$.

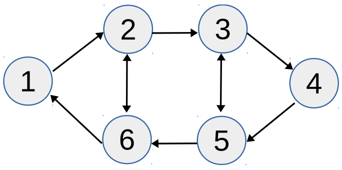


Figure 2.5: Example. Network topology

Fig.2.6 and Fig.2.7 show the simulation results with the ETC mechanism using the control law given by (2.23). It can be observed in Fig.2.6 that the states of all agents are synchronized asymptotically, i.e. the agents' states reach consensus and evolve along a common trajectory. Fig.2.7 shows that the agents broadcast their states at discrete time instants for which the estimation errors $\|\mathbf{e}_i(t)\|$ hit the triggering threshold functions $h_i(t)$; $i \in \mathcal{V}$. The minimum inter-event time and the number of broadcast events for each agent during the simulation period are shown in Table 2.1.

Table 2.1: Minimum inter-event times and number of events

Agent i	$\min\{t_{i,k+1} - t_{i,k}\}(s)$	Total number of events
$i = 1$	0.029	42
$i = 2$	0.117	16
$i = 3$	2.103	4
$i = 4$	0.189	15
$i = 5$	0.081	15
$i = 6$	0.854	5

Fig.2.8 shows the trajectory of the synchronization error with continuous communications (C-C), and with the ETC mechanism using controllers (2.23) and (2.61). According to

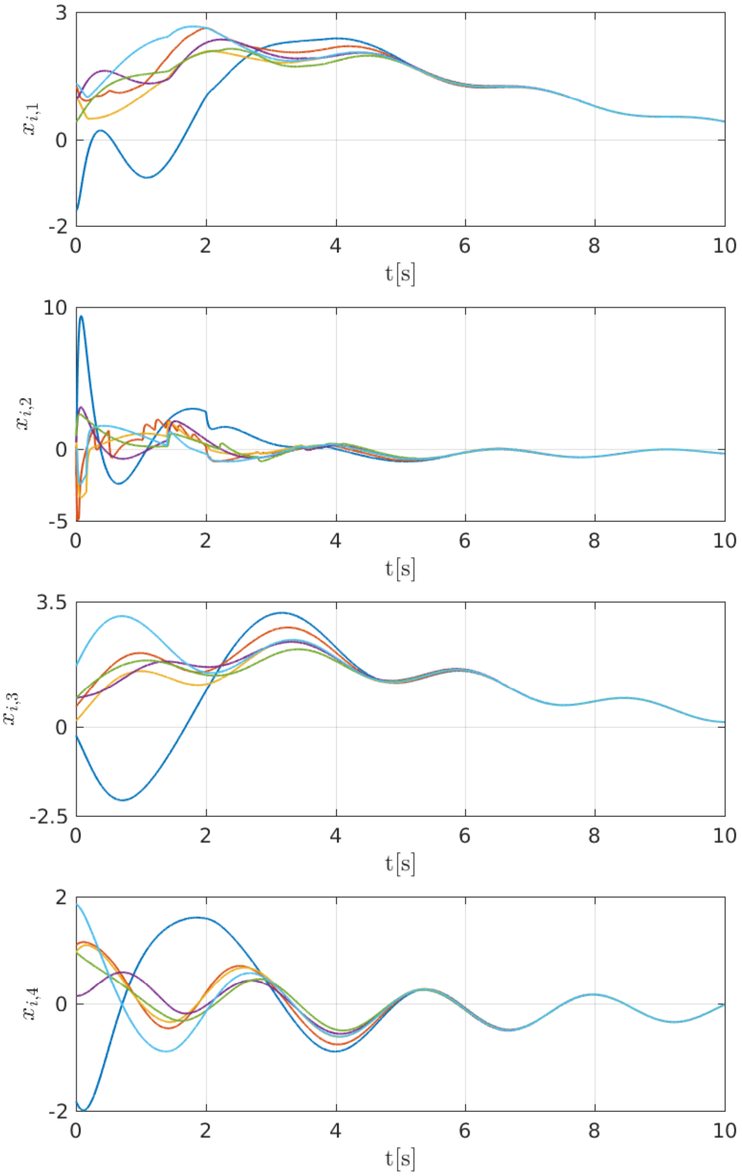
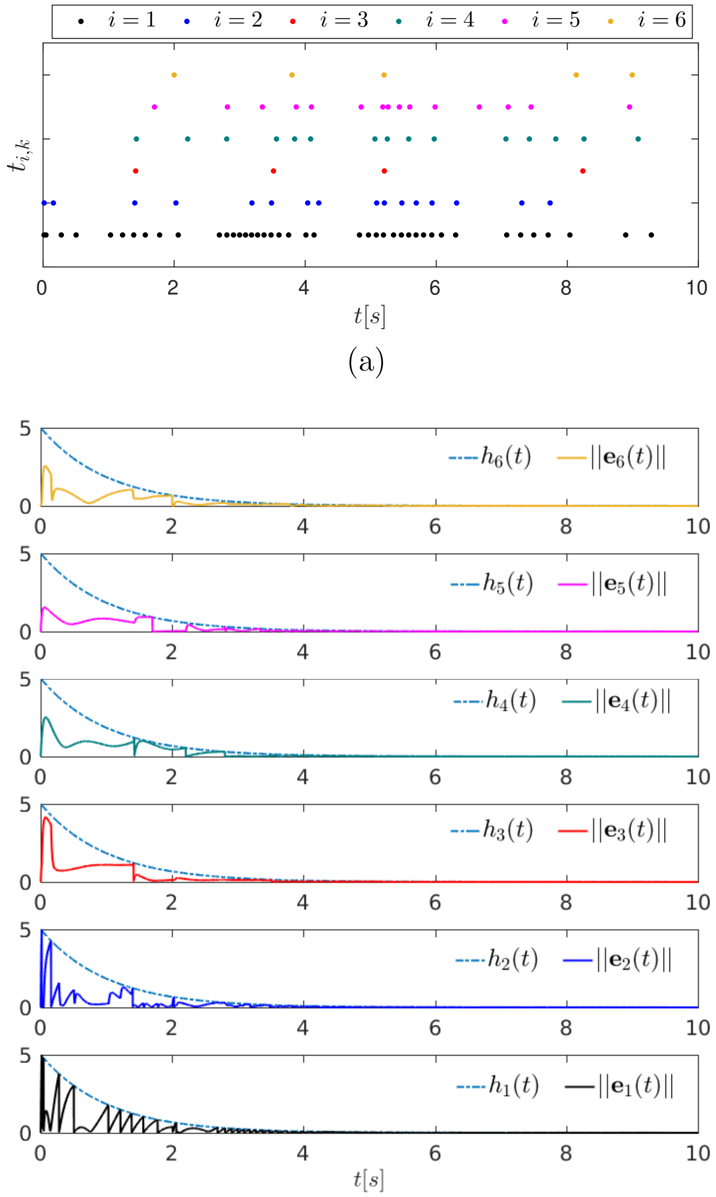


Figure 2.6: ETC mechanism with controller (2.23). Synchronization of agents' states.

Theorem 2.1, using controller (2.23) the asymptotic bound for the synchronization error is computed using (2.36) as $r_1 = 0.1102$. Using controller (2.61), the asymptotic bound is computed using (2.62) as $r_2 = 0.2139$. The figure clearly shows that asymptotically, the synchronization error is upper bounded by the asymptotic bounds. Recall that the asymptotic bounds can be made arbitrarily small by tuning the lower bound on the threshold functions. Hence, the synchronization error would get closer to zero if c_0, the lower bound of the triggering function were reduced. This also would make the synchronization error closer to that obtained using continuous communications. However, this would potentially



(a)

(b)

Figure 2.7: ETC mechanism with controller (2.23). (a) Sequence of broadcast time instants for each agent. (b) Evolution of the estimation errors $\|\mathbf{e}_i(t)\|$ and the threshold triggering functions $h_i(t)$, $i = 1, ..., 6$.

make communications among the agents more frequent.

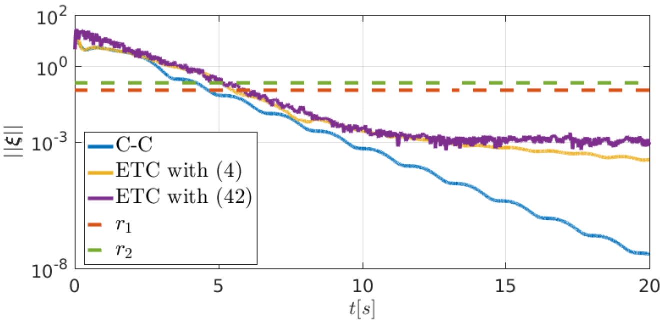


Figure 2.8: Example. Trajectories of the synchronization error $\|\boldsymbol{\xi}\|$ and asymptotic bounds with different communication mechanisms.

2.8 Conclusions

This chapter described a general event triggered communication framework for consensus/synchronization of nonlinear multi-agent systems. The considered agent's dynamics and the network topology are general enough for a large number of applications. We showed how consensus and synchronization can be achieved and how the communications among agents can be reduced by using the proposed event triggered communication mechanism. Further, the minimum inter-event time for every agent was shown to be strictly positive, thus excluding the occurrence of Zeno behavior.

3

CPF of constrained autonomous vehicles with model predictive control and ETC mechanisms

Contents

In this chapter, we present a solution to the problem of multiple vehicle cooperative path following (CPF) that takes explicitly into account vehicle input constraints, the topology of the inter-vehicle communication network, and time-varying communication delays. The objective is to steer a group of vehicles along given spatial paths, at speeds that may be path dependent, while holding a feasible geometric formation. The solution involves decoupling the original CPF problem into two sub-problems: i) single path following of input-constrained vehicles and ii) coordination of an input-constrained multi-agent system (MAS). The first is solved by adopting a sampled-data model predictive control (MPC) scheme, whereas the latter is tackled using a novel distributed control law with an event triggered communication (ETC) mechanism. The proposed strategy yields a closed-loop CPF system that is input-to-state-stable (ISS) with respect to the system's state (consisting of the path following error of all vehicles and their coordination errors) and the system's input, which includes triggering thresholds for ETC communications and communication delays. Furthermore, with the proposed ETC mechanism, the number of communications among the vehicles are significantly reduced.

3.1 Literature review

Cooperative path following, an important class of multiple vehicle formation control, is defined as the problem of steering a group of vehicles along a set of spatial paths, at speeds that may be path dependent, while holding a feasible geometric pattern. Among a myriad of applications related to CPF, we single out those involving unmanned aerial vehicles (UAVs) for coastal monitoring, [Klemas, 2015, Kaminer et al., 2017] and autonomous marine vehicles (AMVs) for marine habitat mapping and geo-technical surveying [Abreu et al., 2016b].

From a control design and analysis standpoint, CPF may be viewed as exhibiting a two-layer control structure: the lower layer, called *path following*, is in charge of making a group of vehicles converge to a set of desired paths parametrized in an appropriately normalized manner, while the upper layer, referred to as networked MAS coordination layer, has the goal of synchronizing the path parameters and making them evolve at the same normalized desired speed profile along the paths. Under these circumstances, proper path parametrization will ensure that the vehicles will reach a desired formation with the assigned individual speed profiles compatible with the paths and the formation (see [Ghabcheloo et al., 2009, Almeida et al., 2012] for an introduction to these concepts). Using this set-up, different approaches to the CPF problem have been proposed in the literature. A simple categorization of the methods used is presented in Table 3.1.

Most approaches assume that the vehicles' inputs (e.g. speed and heading rate in the case of planar kinematic model) are unconstrained. This assumption allows designers to use a wide range of classical nonlinear control methods such as Lyapunov based techniques to design controllers for *path following*, while the *coordination* problem is tackled by resorting to tools from network control theory for *unconstrained* MAS, see for example [Olfati-Saber et al., 2007] for a comprehensive introduction to consensus algorithms and their applications to cooperative control. However, in practice the inputs of the vehicles are always saturated at certain levels due to intrinsically physical limitations. As a consequence, controllers designed for unconstrained vehicles may fail to yield adequate performance. Even worse, stability of the resulting closed loop systems may be seriously compromised if the vehicle constraints are not taken directly into account during the design process.

Due to its ability to handle explicitly input constraints, Model Predictive Control has recently been proposed as a key enabling tool for the solution of CPF problems, see for example [Rucco et al., 2015, Alessandretti and Aguiar, 2017]. In [Rucco et al., 2015], the authors propose an MPC scheme to solve the path following problem, while the coordination problem is solved using a classical consensus law. However, the approach in [Rucco et al., 2015] has two limitations. Firstly, the MPC scheme is designed based on a linearization of the path following error system, which implies that stability of the resulting system is only guaranteed locally. In addition, with the consensus law used in [Rucco et al., 2015] there is no guarantee that the total speed assigned to each vehicle, which is the summation of the nominal desired speed and the correction speed issued by the consensus law satisfy the vehicle's speed constraint. In [Alessandretti and Aguiar, 2017], the authors address the CPF problem using a distributed MPC framework. However, the methodology adopted requires that the speed of vehicles be allowed to be negative, a constraint that is practically impossible to meet for some classes of autonomous vehicles such as fixed-wing UAVs or acrshortamvs.

Another factor that plays a key role in the design of CPF control systems stems from the limitations naturally imposed by the requirement that the agents exchange data over a given communication network. From a purely theoretical standpoint, it is common in the literature to assume that communications occur continuously in time. In this situation, each vehicle has permanent access to the information provided by its neighbors to include it in some form of consensus law. In practice, however this assumption is clearly violated, namely in applications where communication networks exhibit low bandwidth and non-negligible transmission latency. To cope with this situation, it is crucial to explicitly incorporate in the design process the fact that communications do not take place continuously. A possible solution is to consider periodic communications, with the latter taking place at discrete instants of time only [Almeida et al., 2012]. Recently, with the objective of further reducing the rate of inter-agent communications in cooperative MAS control, event-triggered communications have come to the fore. Representative examples include the work in [Aguiar and Pascoal, 2007] and [Rego et al., 2019a] on CPF that exploit the concept of logic-based communications advanced in [Xu and Hespanha, 2004] and that in [Jain et al., 2018], which builds upon an ETC mechanism introduced in [Fan et al., 2015]. Temporary communication losses are taken into account in [Ghabcheloo et al., 2009, Cichella et al., 2015] but only for the case when communications occur continuously.

Table 3.1: CPF categories

Categories		**Literature**
Vehicle inputs	Unconstrained	[Jain et al., 2018, Almeida et al., 2012] [Aguiar and Pascoal, 2007, Ghabcheloo et al., 2009] [Kaminer et al., 2017]
	Constrained	[Rucco et al., 2015, Alessandretti and Aguiar, 2017]
Comm.	Continuous	[Ghabcheloo et al., 2009, Cao et al., 2017] [Klausen et al., 2015, Cichella et al., 2015] [Kaminer et al., 2017]
	Periodic	[Almeida et al., 2012]
	Event-based	[Jain et al., 2018, Aguiar and Pascoal, 2007] [Rego et al., 2019a]
Speed profile	Constant	[Jain et al., 2018, Kaminer et al., 2017]
	Path dependent	[Ghabcheloo et al., 2009, Rego et al., 2019a] [Kaminer et al., 2017]

An important issue in the design of CPF systems is the parametrization of the paths to be followed and the specification of the desired, identical rate of evolution of the path parameters, which can be viewed as a desired normalized speed profile for the agents involved to track. If the desired speed is constant, the coordination problem can be cast in the form of a linear MAS consensus problem [Jain et al., 2018] whereas if the speed is parameterized as a general function of the path parameters (i.e. path dependent), the resulting coordination problem is equivalent to a consensus problem of nonlinear MAS [Ghabcheloo et al., 2009, Rego et al., 2019a]. However, none of methods described in the literature addresses the problem of coordination of *nonlinear* MAS with *input constraints* that arises naturally in the context of CPF.

Motivated by the above considerations, this chapter proposes a CPF control strategy that takes explicitly into account realistic constraints on the vehicles' inputs, the topology of the inter-vehicle communications network, and communication delays. Specifically, the main contributions of this chapter include the following:

(i) At the *path following* level, we develop an MPC scheme for *path following* that takes into account explicitly the fact that in a large number of applications the vehicle's linear speed is strictly positive. When compared with existing MPC-based methods (see for example [Rucco et al., 2015, Alessandretti and Aguiar, 2017]), the proposed MPC scheme has the advantage of avoiding the construction of a terminal set, thus yielding a global region of attraction for single vehicle path following.

(ii) At the *coordination* level, we propose a novel distributed control strategy for the *coordination of nonlinear* MAS where the agents' *input constraints* are explicitly taken into account. We also propose an ETC mechanism that is not only capable of reducing the frequency of communications among vehicles but is also robust with respect to time-varying communication delays, making the scheme attractive for scalable networks with limited communication bandwidth. This result is applicable in the context of CPF for multiple autonomous vehicles, and also in many other applications involving the coordination/synchronization/consensus of nonlinear MAS with input constraints.

3.2 Problem formulation

For simplicity of exposition we consider motions in 2D. In what follows, $\{\mathcal{I}\} = \{x_{\mathcal{I}}, y_{\mathcal{I}}\}$ denotes an inertial frame and $\{\mathcal{B}\}^{[i]} = \{x_{\mathcal{B}}^{[i]}, y_{\mathcal{B}}^{[i]}\}$ denotes a body frame attached to vehicle i. We consider a set of $N \geq 2$ vehicles and the corresponding set of N spatial paths that they are required to follow, described by

$$\{\mathcal{P}^{[i]} : \gamma^{[i]} \to \mathbf{p}_{\mathrm{d}}^{[i]}(\gamma^{[i]}) \in \mathbb{R}^2; i \in \mathcal{N}\}, \tag{3.1}$$

where $\mathcal{N} \triangleq \{1, ..., N\}$ denotes the set of vehicles, $\gamma^{[i]}$ is the variable parameterizing path i, $\mathbf{p}_{\mathrm{d}}^{[i]}(\gamma^{[i]}) = [x_{\mathrm{d}}^{[i]}(\gamma^{[i]}), y_{\mathrm{d}}^{[i]}(\gamma^{[i]})]^{\mathrm{T}}; i \in \mathcal{N}$ is the position vector of a generic point on the path i expressed in the inertial frame. Let $\mathbf{p}^{[i]} = [x^{[i]}, y^{[i]}]^{\mathrm{T}}; i \in \mathcal{N}$ be the position vector of the center of mass of vehicle i expressed in the inertial frame. Assuming that the vehicles have negligible sway speed, their kinematic models are given by

$$\dot{x}^{[i]} = u^{[i]} \cos \psi^{[i]}, \;\; \dot{y}^{[i]} = u^{[i]} \sin \psi^{[i]}, \;\; \dot{\psi}^{[i]} = r^{[i]}, \tag{3.2}$$

where $u^{[i]}, \psi^{[i]}, r^{[i]}; i \in \mathcal{N}$ denote the speed, yaw angle, and yaw rate of vehicle i, respectively. Due to physical limitations of the vehicles, we consider that the speed and the heading rate are constrained, i.e. $(u^{[i]}, r^{[i]}) \in \mathbb{U}^{[i]}$, for all $i \in \mathcal{N}$, where $\mathbb{U}^{[i]}$ is referred as an input constraint set for vehicle i, defined explicitly as

$$\mathbb{U}^{[i]} \triangleq \{(u^{[i]}, r^{[i]}) : u^{[i]}_{\text{min}} \leq u^{[i]} \leq u^{[i]}_{\text{max}}, |r^{[i]}| \leq r^{[i]}_{\text{max}}\}. \tag{3.3}$$

Here, $u^{[i]}_{\text{min}} > 0$ and $u^{[i]}_{\text{max}}$ are lower and upper bounds on the speed, respectively, and $r^{[i]}_{\text{max}}$ is an upper bound on the heading rate.

We note that the kinematics model (3.2) is adequate for a large class of vehicles that include mobile robots [Lapierre et al., 2006], fixed-wing UAVs undergoing planar motion [Rucco et al., 2015], and a wide class of under-actuated acrshortamvs such as Medusa and Delfim [Abreu et al., 2016c]or Charlie [Bibuli et al., 2009], for which the sway speed is in practice so small that it can be neglected. A similar kinematic model with a drift term can be found in [Hung et al., 2018] for the case where the motion is disturbed by constant wind (for AUVs) or constant ocean current (for acrshortamvs). In addition, it is important to remark that in the present work we require the speeds of the vehicles to be non-negative. This is due to the fact that for many autonomous vehicles such as marine robots and fixed-wing UAVs, it is very difficult or even impossible to control the vehicle moving backwards. This strict constraint makes the CPF problem more challenging when compared to the case where this type of constraint is not taken into account, as in [Alessandretti and Aguiar, 2017].

In cooperative path following, vehicle i is assigned path i to follow, see the illustration in Fig.3.1. We consider a scenario where the fleet of vehicles are not only required to follow their assigned paths but also to converge to and maintain a desired geometric formation, while maneuvering with desired speed profiles along the paths in a manner that is compatible with the formation. To solve the constrained CPF problem, the methodology used in this chapter decouples the constrained CPF problem into two sub-problems: *path following of constrained vehicles* to steer the vehicles to converge to their assigned paths and *coordination of constrained MAS* that requires the vehicles to exchange information regarding their progression along the paths (as measured by their path parameters) and

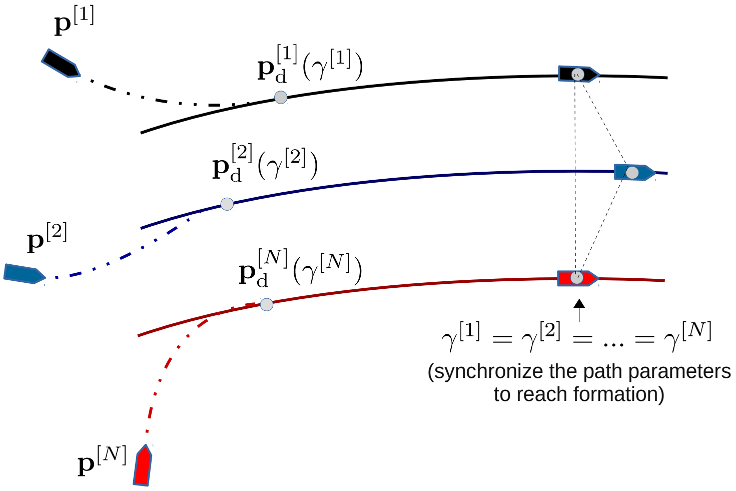

$\mathbf{p}^{[1]}$
$\mathbf{p}_\mathrm{d}^{[1]}(\gamma^{[1]})$
$\mathbf{p}_\mathrm{d}^{[2]}(\gamma^{[2]})$
$\mathbf{p}^{[2]}$
$\mathbf{p}_\mathrm{d}^{[N]}(\gamma^{[N]})$
$\gamma^{[1]} = \gamma^{[2]} = ... = \gamma^{[N]}$
(synchronize the path parameters
to reach formation)
$\mathbf{p}^{[N]}$

Figure 3.1: Illustration of cooperative path following.

negotiate their speeds to reach the desired formation. Using this set-up, we show that in order to control a fleet of vehicles with a desired formation, the vehicles need to exchange very limited information; in this case, a simple scalar path-related parameter that will be used for coordination. From a communication and practical implementation perspective, this is an advantageous feature of the proposed CPF when compared with other formation approaches such as distributed MPC that normally requires more information to be exchanged among the vehicles, see for example [Dunbar and Murray, 2006, Muller et al., 2011].

3.2.1 Single path following of constrained vehicles

In this subsection, we formulate the problem of *single vehicle constrained path following* to make a vehicle converge to a path, while ensuring that the speed of the corresponding path parameter tracks a desired speed profile. To this end, we exploit the concept of "tracking a virtual reference" introduced in [Lapierre et al., 2006]. Because in this section we deal with a single vehicle, for the sake of simplicity we drop the superscript $[i]$ in the variables in equations (3.1)–(3.3). Later, in subsequent sections, we will re-introduce the original notation when necessary.

Consider the path following problem for a single vehicle with the kinematics model given by (3.2), subject to the constraints on the inputs given by (3.3), following a path parameterized by the variable γ given by (3.1). Consider a Parallel Transport frame $\{\mathcal{P}\} = \{x_\mathcal{P}, y_\mathcal{P}\}$ with its origin at an arbitrary point P on the path and its axes defined as follows: $x_{\mathcal{P}}$ is aligned with the tangent to the path and points in the direction of increasing path length and $y_{\mathcal{P}}$ is determined by rotating $x_{\mathcal{P}}$ 90 degrees clock wise (see Fig.3.2). In the set-up adopted for path-following, the Parallel Transport frame moves along the path in a manner to be determined and plays the role of a "*virtual reference*" for both the position and heading angle that the vehicle must track to achieve good path following. We define

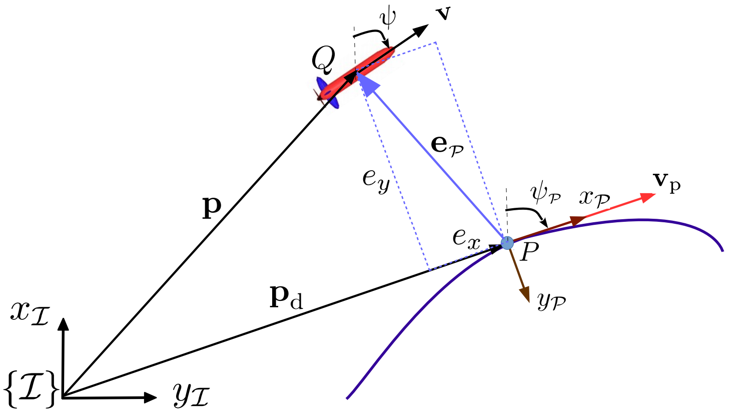


Figure 3.2: Vehicle and reference frames. Velocity vector in the body frame $\mathbf{v} = [u, 0]$. Q is the center of mass of the vehicle and P is the origin of the Parallel Transport Frame at a point on the path.

$$R^{\mathcal{P}}_{\mathcal{I}}(\psi_{\mathcal{P}}) = \begin{bmatrix} \cos(\psi_{\mathcal{P}}) & \sin(\psi_{\mathcal{P}}) \\ -\sin(\psi_{\mathcal{P}}) & \cos(\psi_{\mathcal{P}}) \end{bmatrix} \tag{3.4}$$

as the rotation matrix from the $\{\mathcal{I}\}$ to the $\{\mathcal{P}\}$. Let $\mathbf{e}_{\mathcal{P}} = [e_x, e_y]^{\mathrm{T}}$ be the position error between the vehicle and the *virtual reference*, expressed in the Parallel Transport frame, computed as

$$\mathbf{e}_{\mathcal{P}} = R^{\mathcal{P}}_{\mathcal{I}}(\psi_{\mathcal{P}})(\mathbf{p} - \mathbf{p}_{\mathrm{d}}). \tag{3.5}$$

Note that $R^{\mathcal{P}}_{\mathcal{I}}(\psi_{\mathcal{P}}) = [R^{\mathcal{I}}_{\mathcal{P}}(\psi_{\mathcal{P}})]^{\mathrm{T}}$. Taking time derivative of (3.5) then applying Lemma A.1 in Appendix A yields

$$\begin{aligned} \dot{\mathbf{e}}_{\mathcal{P}} &= [\dot{R}^{\mathcal{I}}_{\mathcal{P}}(\psi_{\mathcal{P}})]^{\mathrm{T}}(\mathbf{p} - \mathbf{p}_{\mathrm{d}}) + R^{\mathcal{P}}_{\mathcal{I}}(\psi_{\mathcal{P}})(\dot{\mathbf{p}} - \dot{\mathbf{p}}_{\mathrm{d}}) \\ &= -S(\boldsymbol{\omega}_{\mathcal{P}})\mathbf{e}_{\mathcal{P}} + R^{\mathcal{P}}_{\mathcal{I}}(\psi_{\mathcal{P}})\dot{\mathbf{p}} - R^{\mathcal{P}}_{\mathcal{I}}(\psi_{\mathcal{P}})\dot{\mathbf{p}}_{\mathrm{d}}, \end{aligned} \tag{3.6}$$

where $S(\boldsymbol{\omega}_{\mathcal{P}}) \in \mathbb{R}^{2\times 2}$ is a skew symmetric matrix and $\boldsymbol{\omega}_{\mathcal{P}} \in \mathbb{R}^2$ is the angular velocity vector of P respect to $\{\mathcal{I}\}$, expressed in $\{\mathcal{P}\}$, given by

$$\boldsymbol{\omega}_{\mathcal{P}} = \begin{bmatrix} \dot{\psi}_{\mathcal{P}} \\ 0 \end{bmatrix} = \begin{bmatrix} \kappa(\gamma)u_{\mathcal{P}} \\ 0 \end{bmatrix} \tag{3.7}$$

with $u_{\mathcal{P}}$ is the inertial speed of the reference point P, expressed in $\{\mathcal{P}\}$ and $\kappa(\gamma)$ is the singed curvature of the path at P, given by

$$u_{\mathcal{P}} = \dot{\gamma}\|\mathbf{p}'_{\mathrm{d}}(\gamma)\|, \qquad \kappa(\gamma) = \frac{x'_{\mathrm{d}}(\gamma)y''_{\mathrm{d}}(\gamma) - x''_{\mathrm{d}}(\gamma)y'_{\mathrm{d}}(\gamma)}{\|\mathbf{p}'_{\mathrm{d}}(\gamma)\|^3}. \tag{3.8}$$

Notice that if γ is the arc-length of the path then $\|\mathbf{p}'_{\mathrm{d}}(\gamma)\| = 1$. In this case, $u_{\mathcal{P}} = \dot{\gamma}$, i.e. the speed of the *virtual reference* equals to the rate of change of the path length. Define

$$e_\psi \triangleq \psi - \psi_{\mathcal{P}} \tag{3.9}$$

as the orientation error between the vehicle's heading and the tangent of the path at P, from which it follows that

$$R_{\mathcal{I}}^{\mathcal{P}}(\psi_{\mathcal{P}})\dot{\mathbf{p}} \overset{(3.2),(3.4)}{=} \begin{bmatrix} u\cos(e_\psi) \\ u\sin(e_\psi) \end{bmatrix}. \tag{3.10}$$

Further, let $\mathbf{v}_{\mathcal{P}} \triangleq [u_{\mathcal{P}}, 0]^{\mathrm{T}} \in \mathbb{R}^2$ be the velocity vector of P respect to $\{\mathcal{I}\}$, expressed in $\{\mathcal{P}\}$, that is,

$$R_{\mathcal{I}}^{\mathcal{P}}(\psi_{\mathcal{P}})\dot{\mathbf{p}}_{\mathrm{d}} = \mathbf{v}_{\mathcal{P}}. \tag{3.11}$$

Substituting (3.10) and (3.11) in (3.6) yields the dynamics of the position error as

$$\dot{\mathbf{e}}_{\mathcal{P}} = -S(\boldsymbol{\omega}_{\mathcal{P}})\mathbf{e}_{\mathcal{P}} + \begin{bmatrix} u\cos(e_\psi) \\ u\sin(e_\psi) \end{bmatrix} - \begin{bmatrix} u_{\mathcal{P}} \\ 0 \end{bmatrix}. \tag{3.12}$$

Furthermore, from (3.9) the dynamics of the orientation error are given by

$$\dot{e}_\psi \overset{(3.2),(3.7)}{=} r - \kappa(\gamma)u_{\mathcal{P}}. \tag{3.13}$$

Collectively, defining $\mathbf{x} = [\mathbf{e}_{\mathcal{P}}^{\mathrm{T}}, e_\psi]^{\mathrm{T}} \in \mathbb{R}^3$ as the path following error vector, its dynamics are given by

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \overset{(3.12),(3.13)}{=} \begin{bmatrix} -g(\gamma)v_\gamma(1 - \kappa(\gamma)e_y) + u\cos(e_\psi) \\ -\kappa(\gamma)g(\gamma)v_\gamma e_x + u\sin(e_\psi) \\ r - \kappa(\gamma)g(\gamma)v_\gamma \end{bmatrix}, \tag{3.14}$$

where $g(\gamma) \triangleq \|\mathbf{p}'_{\mathrm{d}}(\gamma)\|$ and $v_\gamma \triangleq \dot{\gamma}$ is the speed of the path parameter that gives an extra degree of freedom in the process of designing path following controllers and $\mathbf{u} \triangleq [u, v_\gamma, r]^{\mathrm{T}}$ is the input vector of the path following error system.
Notice that we have introduced a new input v_γ to control the evolution of the path parameter γ. Later, for the purpose of designing an input constrained path following controller, v_γ should lie in a constraint set $\mathbb{U}_v$, defined explicitly as

$$\mathbb{U}_v \triangleq \{v_\gamma : |v_\gamma| \leq v_{\max}\}, \tag{3.15}$$

where $v_{\max}$ is a design parameter that will be specified.
We are now in a position to formulate the following constrained path following problem.

Problem 3.1 (Constrained Path Following)**.** *Given a spatial path $\mathcal{P}$ parameterized by γ, a desired positive and bounded speed profile $v_{\mathrm{d}} : \gamma \to v_{\mathrm{d}}(\gamma)$, and the constraint sets for the vehicle's inputs and the speed of the "virtual reference" given by* (3.3) *and* (3.15) *respectively, derive a feedback control law for $(u, r) \in \mathbb{U}$ and $v_\gamma \in \mathbb{U}_v$ to fulfill the following tasks:*

- *Geometric task: drive the path following error $\mathbf{x}$ with the dynamics described in* (3.14) *to zero as $t \to \infty$.*
- *Dynamic task: ensure also that v_γ tracks the desired speed profile $v_{\mathrm{d}}(\gamma)$, that is, $v_\gamma(t) - v_{\mathrm{d}}(\gamma(t)) \to 0$ as $t \to \infty$.*

Stated intuitively, a solution to the input-constrained path following problem consists of adjusting the speed v_γ of the "virtual reference", the speed u, and the heading rate r of the vehicle, subject to given vehicle constraints, to drive the vehicle to the path and keep its velocity vector aligned with the tangent to the path while having the path parameter track the desired speed profile.

Remark 3.1. *If γ is the arc length of the path, then $g(\gamma) \equiv 1$. In this case, the path following error system* (3.14) *resembles the path following error system developed in [Lapierre et al., 2006]. Notice that although parameterizing a path by its arc-length is convenient, the main problem is that it is not always possible to find a closed form expression of the curvature as the function of the arc-length; elliptical and sinusoid paths* *are examples. In our set-up, the path parameter γ in* (3.14) *is not necessarily the arc length, thus making the formulation applicable to any path.*

Remark 3.2. *Since the path parameter γ is not necessarily the arc-length, in general v_γ is not the speed of the "virtual reference" in the inertial frame. In fact, the latter equals $g(\gamma)v_\gamma$. Obviously, if γ is the arc-length of the path, then $g(\gamma) \equiv 1$ and v_γ is truly the speed of the "virtual reference" in the inertial frame.*

3.2.2 Cooperative path following

Before proceeding with the formulation of the multiple vehicle coordination problem, we make the following assumptions.

Assumption 1.

A1.1 Each vehicle is equipped with a path following controller (to be designed later using an MPC scheme) that solves **Problem 3.1** *, where the desired speed profile $v_\mathrm{d}(\cdot)$ for $\dot{\gamma}_i; i \in \mathcal{N}$ along the paths is identical for all vehicles.*

A1.2 The inter-vehicle network topology is time invariant.

In what follows, we assume that the paths that the vehicles must follow are appropriately parameterized to ensure that a given formation is reached when the path parameters, also called coordination states, are equal. For example, to make a number of vehicles follow an equal number of concentric circumferences and be aligned radially along their radii, it suffices to parametrize these paths in terms of their normalized lengths, that is, $\gamma^{[i]} = s^{[i]}/2\pi$, where $s^{[i]}$ is the curvilinear abscissa along path i. Clearly, the vehicles are coordinated and maneuver with a desired normalized path dependent speed $v_\mathrm{d}(\cdot)$ if $\gamma^{[i]}(t) = \gamma^{[j]}(t)$ and $\dot{\gamma}^{[i]}(t) = \dot{\gamma}^{[j]}(t) = v_\mathrm{d}(\gamma^{[i]})$ for all $i, j \in \mathcal{N}$. See [Ghabcheloo et al., 2009] for an introduction to these concepts.

The underlying idea to achieve the coordination is described as follows. Assume for the time being that the vehicles maneuver independently and do not attempt to coordinate their motions. Assume that the path following controller makes each vehicle i converge to the path asymptotically ($\mathbf{x}^{[i]} = \mathbf{0}$) and ensures also that the path parameter evolves with the desired speed profile, that is, $v_\gamma^{[i]} = v_\mathrm{d}(\gamma^{[i]})$. Asymptotically, in this ideal situation we have

$$\dot{\gamma}^{[i]} = v_\gamma^{[i]} = v_\mathrm{d}(\gamma^{[i]}); \qquad i \in \mathcal{N}. \tag{3.16}$$

Replacing $v_\gamma^{[i]} = v_{\mathrm{d}}(\gamma^{[i]})$ and $\mathbf{x}^{[i]} = \mathbf{0}$ in (3.14), and noticing that $\mathbf{x}^{[i]} = \mathbf{0}$ is the equilibrium point of the path following system (3.14), the nominal inertial speeds of the vehicles are given by

$$u^{[i]} = g^{[i]}(\gamma^{[i]})v_{\mathrm{d}}(\gamma^{[i]}); \qquad i \in \mathcal{N}. \tag{3.17}$$

Recall that the desired formation is only achieved when the path parameters reach consensus (or synchronized), that is, $\gamma^{[i]} = \gamma^{[j]}$ for all $i, j \in \mathcal{N}$. This can be accomplished by adjusting the linear speeds of vehicles about the nominal speeds in (3.17) so as to make all vehicles reach agreement in the coordination states (the path parameters) and maneuver with the common normalized desired speed $v_{\mathrm{d}}(\cdot)$. Let $g^{[i]}(\gamma^{[i]})v_{\mathrm{c}}^{[i]}$ be the a correction term for the speed of vehicle i, where $v_{\mathrm{c}}^{[i]}$ is a new input aimed at achieving coordination, to be explained later. The resulting speeds for the vehicles are given by

$$u^{[i]} = g^{[i]}(\gamma^{[i]})(v_{\mathrm{d}}(\gamma^{[i]}) + v_{\mathrm{c}}^{[i]}); \qquad i \in \mathcal{N}. \tag{3.18}$$

Consequently, the dynamics of the path parameters in (3.16) are now extended as

$$\dot{\gamma}^{[i]} = v_{\mathrm{d}}(\gamma^{[i]}) + v_{\mathrm{c}}^{[i]}; \qquad i \in \mathcal{N}. \tag{3.19}$$

At this stage, it is clear that the coordination problem is reduced to finding $v_{\mathrm{c}}^{[i]}; i \in \mathcal{N}$ such that the total speed of each vehicle in (3.18) still satisfies (3.3), that is,

$$u_{\min}^{[i]} \leq g^{[i]}(\gamma^{[i]})(v_{\mathrm{d}}(\gamma^{[i]}) + v_{\mathrm{c}}^{[i]}) \leq u_{\max}^{[i]}; \quad i \in \mathcal{N}, \tag{3.20}$$

and the path parameters are synchronized and evolve with the common speed profile $v_{\mathrm{d}}(\cdot)$. To solve this consensus problem, each vehicle needs to exchange the path parameters (called coordination states) with other vehicles. In this work, we consider that each vehicle is capable of communicating bidirectionally with a set of neighboring vehicles. Let $\mathcal{G}$ be the bidirectional (undirected) graph induced by the interconnection network of the vehicles and $\mathcal{N}^{[i]}$ the set of neighboring vehicles of vehicle i. At the coordination layer, we consider each vehicle to be an agent whose dynamics are given by (3.19). We are now in a position to formulate the coordination problem as follows.

Problem 3.2 (Coordination of input-constrained MAS)**.** *Given a MAS with the dynamics of each agent given by* (3.19) *and the network topology of the MAS modeled by the graph*

$\mathcal{G}$ satisfying Assumption 1, derive a distributed control law for the input $v_{\mathrm{c}}^{[i]}(\gamma^{[i]}, \gamma^{[j]}); j \in \mathcal{N}^{[i]}$, subject to the input constraint (3.20)*, such that $(\gamma^{[i]}(t) = \gamma^{[j]}(t)); \forall i, j \in \mathcal{N}$ and $(\dot{\gamma}^{[i]}(t) = v_{\mathrm{d}}(\gamma^{[i]}(t))); \forall i \in \mathcal{N}$ as $t \to \infty$.*

Note that since the function $v_{\mathrm{d}}(\cdot)$ is common for all agents, reaching consensus in the path parameters and their speeds implies that $v_{\mathrm{c}}^{[i]}$ converges to zero for all $i \in \mathcal{N}$. In the next section, the process of designing controllers to solve the problems defined above shall be illustrated.

3.3 Controller design and main results

Based on the idea of decoupling the constrained CPF problem into the subproblem of *path following* and *MAS coordination*, we propose a distributed CPF control system that, for each vehicle, exhibits the architecture depicted in Fig.3.3. The objective of the "Coordination with ETC Mechanism" block is to compute the correction speed $v_{\mathrm{c}}^{[i]}$. An ETC mechanism is proposed to reduce communications among vehicles so that they will only communicate with its neighbors when found necessary, according to some specific criterion. Once the correction speed has been computed, the MPC controller is used to make the vehicle converge to and follow its assigned path. In other words, the MPC controller is used to stabilize the path following error between the vehicle and its assigned path.

To make the constrained CPF problem solvable, we assume that given the vehicles' input constraints, the planed paths given in (3.1) and the desired speed profile $v_{\mathrm{d}}(\cdot)$ are smooth and satisfy the following conditions.

Condition 3.1.

C1.1 *$v_{\mathrm{d}}(\cdot)$ is bounded, i.e.* $0 < v_{\mathrm{dmin}} \leq v_{\mathrm{d}}(\cdot) \leq v_{\mathrm{dmax}}$.

C1.2 *Condition on the linear speeds:*

There exists a constant $c_{\mathrm{u}} > 0$ such that $u_{\min}^{[i]} + c_{\mathrm{u}} \leq g^{[i]}(\gamma^{[i]}) v_{\mathrm{d}}(\gamma^{[i]}) \leq u_{\max}^{[i]} - c_{\mathrm{u}}$ for all $\gamma^{[i]}$ and $i \in \mathcal{N}$.

C1.3 *Condition on the turning rates:*

There exists a constant c_{r} such that $|\kappa^{[i]}(\gamma^{[i]}) g^{[i]}(\gamma^{[i]}) v_{\mathrm{d}}(\gamma^{[i]})| < r_{\max}^{[i]} - c_{\mathrm{r}}$ for all $\gamma^{[i]}$ and $i \in \mathcal{N}$.

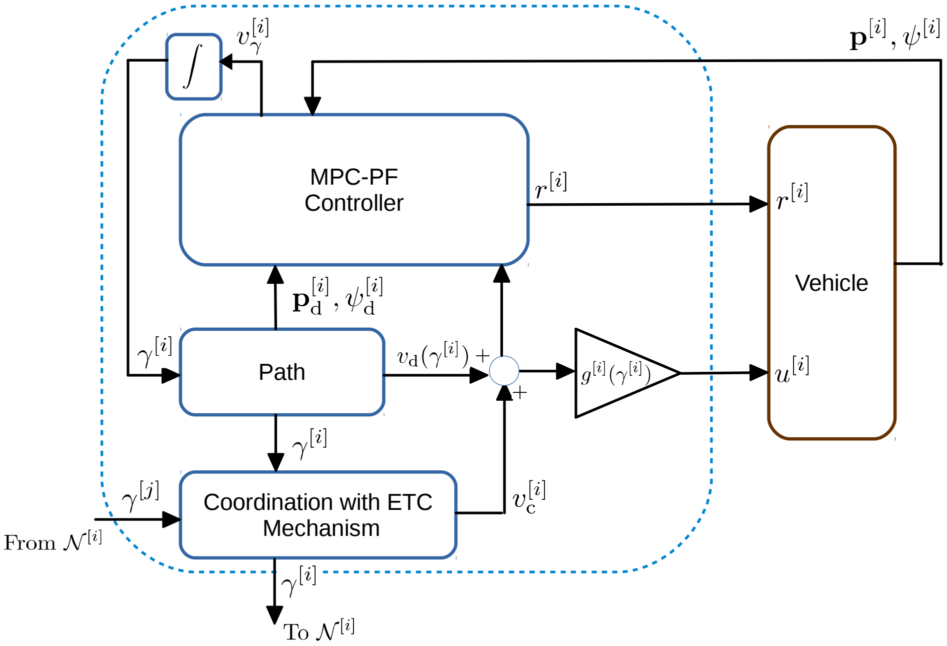


Figure 3.3: CPF control system for vehicle i^{th} with the ETC mechanism.

Remark 3.3. *The above conditions are necessary to ensure that the CPF problem is solvable. To see this, notice in C1.2 that the term $g^{[i]}(\gamma^{[i]})v_\mathrm{d}(\gamma^{[i]})$ is the nominal desired linear speed, computed in the inertial frame, that vehicle i must track, see* (3.17)*. Therefore, c_u gives room for the vehicles to adjust their linear speeds about the nominal ones in order to achieve coordination, see* (3.18)*. Similarly, in C1.3, the left hand side of the inequality is the nominal desired heading rate of vehicle i. Hence, c_r gives room for the vehicles to adjust their heading rates about the nominal ones in order to converge to and follow their assigned paths.*

Example 3.1. *For paths consisting of straight lines and segments of circumferences, such as lawn mowing paths, the above conditions can be significantly simplified. For example, for straight-line paths, if the paths are parameterized by their arc-lengths, then $g^{[i]}(\gamma^{[i]}) \equiv 1$ for all $i \in \mathcal{N}$. Hence, condition C1.3 can be relaxed and C1.2 is equivalent with $u_\mathrm{min}^{[i]} + c_\mathrm{u} \leq v_\mathrm{d}(\gamma^{[i]}) \leq u_\mathrm{max}^{[i]} - c_\mathrm{u}$ for all $i \in \mathcal{N}$. In this case c_u can be simply specified as $c_\mathrm{u} = \min\{v_\mathrm{dmin} - u_\mathrm{min}^{[i]}, u_\mathrm{max}^{[i]} - v_\mathrm{dmax}\}$ for all $i \in \mathcal{N}$.*

In the following subsection we shall propose distributed control laws with different communication scenarios to update the correction speed $v_\mathrm{c}^{[i]}, i \in \mathcal{N}$.

3.3.1 Distributed controllers with an ETC mechanism for the coordination problem

Before introducing distributed control laws to solve the coordination problem (**Problem 3.2**), we define new variables given by

$$z^{[i]} \triangleq \int_0^{\gamma^{[i]}} \frac{1}{v_\mathrm{d}(\gamma)} \mathrm{d}\gamma, \qquad i \in \mathcal{N}. \tag{3.21}$$

Intuitively, $z^{[i]}$ measures the amount of time taken by agent i to travel from 0 to the state $\gamma^{[i]}$. With the above definition, and since $v_\mathrm{d}(\gamma) > 0$ for all γ, it follows that the path parameters $\gamma^{[i]}; i \in \mathcal{N}$ are synchronized (or reach consensus), i.e. $\gamma^{[i]} = \gamma^{[j]}$ for all $i, j \in \mathcal{N}$ iff the variables $z^{[i]}; i \in \mathcal{N}$ are synchronized, i.e. $z^{[i]} = z^{[j]}$ for all $i, j \in \mathcal{N}$. For the sake of convenience, let $\mathbf{z} = [z^{[1]}, z^{[2]}, ..., z^{[N]}]^\mathrm{T} \in \mathbb{R}^N$ and $\bar{z} = \frac{1}{N}\sum_{i=1}^{N} z^{[i]}$ be the average of $\mathbf{z}$. We define the coordination error vector

$$\boldsymbol{\xi} = \mathbf{z} - \bar{z}\mathbf{1} = W\mathbf{z}, \tag{3.22}$$

where $W = I_N - \mathbf{1}\mathbf{1}^\mathrm{T}/N$. Clearly, if the variables $z^{[i]}; i \in N$ reach consensus, then $\mathbf{z}$ spans $\mathbf{1}$. Further, since $W\mathbf{1} = \mathbf{0}$, the variables $z^{[i]}; i \in \mathcal{N}$ reach consensus iff $\boldsymbol{\xi} = \mathbf{0}$. Thus, the problem of driving the variables $z^{[i]}; i \in \mathcal{N}$ to reach consensus amounts to driving the coordination error vector $\boldsymbol{\xi}$ to the origin. For this reason, in what follows we propose distributed control laws for $v_\mathrm{c}^{[i]}$ for all $i \in \mathcal{N}$ under different communication scenarios to drive the error vector $\boldsymbol{\xi}$ to zero.

Remark 3.4. *Notice that the matrix W is similar to the projection matrix Π_ξ defined in [Kaminer et al., 2017], which is popularly used for analyzing consensus of multi agent systems on undirected graphs. The variable $z^{[i]}$ defined in* (3.21) *generalizes the coordination state ξ_i given in [Kaminer et al., 2017], where the paths are parameterized by their arc-lengths.*

Continuous communications

For clarity of presentation of the concepts involved, we start by assuming that communications take place instantaneously and continuously.

Theorem 3.1 (Coordination with continuous communications)**.** *Consider* **Problem 3.2** *. Let Condition 3.1 hold for all $i \in \mathcal{N}$ and let $\mathcal{G}$ be an undirected and connected graph. Then, the distributed control law for $v_{\mathrm{c}}^{[i]}; i \in \mathcal{N}$ given by*

$$v_{\mathrm{c}}^{[i]} = -k_{\mathrm{c}}^{[i]} \tanh(\textstyle\sum_{j \in \mathcal{N}^{[i]}} z^{[i]} - z^{[j]}); \;\; i \in \mathcal{N}, \tag{3.23}$$

where $z^{[i]}$ is given by (3.21) *and $k_{\mathrm{c}}^{[i]}$ are positive gains satisfying the conditions*

$$\begin{aligned} &0 < k_{\mathrm{c}}^{[i]} \leq c_{\mathrm{u}}/g_{\max}^{[i]}; \quad i \in \mathcal{N}, \\ &g_{\max}^{[i]} = \max(g^{[i]}(\gamma^{[i]})) \end{aligned} \tag{3.24}$$

drives all the agents' states (path parameters) to reach consensus asymptotically. In other words, the origin of the coordination error vector $\boldsymbol{\xi}$ is globally asymptotically stable.

Proof: See section 3.7.3.
The next corollary applies to the special case where the desired speed profile $v_{\mathrm{d}}(\cdot)$ is constant.

Corollary 3.1. *Consider* **Problem 3.2** *and let the conditions stated in Theorem 3.1 hold. Further assume that the speed profile is constant, i.e. $v_{\mathrm{d}}(\gamma^{[i]}) \equiv c > 0$ for all $i \in \mathcal{N}$. Then, the distributed control law for $v_{\mathrm{c}}^{[i]}; i \in \mathcal{N}$ given by*

$$v_{\mathrm{c}}^{[i]} = -k_{\mathrm{c}}^{[i]} \tanh(\textstyle\sum_{j \in \mathcal{N}^{[i]}} \gamma^{[i]} - \gamma^{[j]}); \;\; i \in \mathcal{N}, \tag{3.25}$$

where $k_{\mathrm{c}}^{[i]}$ satisfies (3.24)*, drives all the agents' states (path parameters) to reach consensus asymptotically.*

It is interesting to observe that in the case of a constant speed profile, the distributed control law does not depend on the desired speed profiles $v_{\mathrm{d}}(\cdot)$. Further, it follows from (3.25) that the computation of $v_{\mathrm{c}}^{[i]}$ is simplified because there is no need for a block of integrators to compute $z^{[i]}$.

ETC mechanism without communication delays

The distributed control law proposed in subsection 3.3.1 relies on continuous communications among the vehicles. However, this assumption is impossible to meet because practical communication systems require the exchange of data to take place at discrete instants of time. Motivated by this observation, we propose an event triggered communication mechanism in which the vehicles only need to exchange data with their neighbors when necessary, in accordance with an appropriately defined criterion.

In the ETC mechanism, instead of using the true neighboring states ($\gamma^{[j]}; j \in \mathcal{N}^{[i]}$), the control law (3.23) uses their estimates. Following the concepts explained in the previous chapter, the underlying idea is that if any agent can produce "good" estimates of the neighboring states, then there is no need to communicate continuously among the vehicles. Let $\hat{\gamma}^{[ij]}$ be an estimate of $\gamma^{[j]}$ computed by agent i (the procedure to compute this estimate will be explained later). The event triggered distributed control law that we propose is given by

$$v_{\mathrm{c}}^{[i]} = -k_{\mathrm{c}}^{[i]} \tanh\left(\textstyle\sum_{j \in \mathcal{N}^{[i]}} (z^{[i]} - \hat{z}^{[ij]})\right); i \in \mathcal{N}, \tag{3.26}$$

where

$$\hat{z}^{[ij]} \triangleq \int_0^{\hat{\gamma}^{[ij]}} \frac{1}{v_{\mathrm{d}}(\gamma)} \mathrm{d}\gamma, \tag{3.27}$$

and $k_{\mathrm{c}}^{[i]}$ satisfies condition (3.24) for all $i \in \mathcal{N}$.

The control law in (3.26) can be rewritten as

$$v_{\mathrm{c}}^{[i]} = -k_{\mathrm{c}}^{[i]} \tanh\left(\textstyle\sum_{j \in \mathcal{N}^{[i]}} (z^{[i]} - z^{[j]} + e^{[j]})\right); i \in \mathcal{N}, \tag{3.28}$$

where,

$$e^{[j]} = z^{[j]} - \hat{z}^{[ij]} = \textstyle\int_{\hat{\gamma}^{[ij]}}^{\gamma^{[j]}} \frac{1}{v_{\mathrm{d}}(\gamma)} \mathrm{d}\gamma; j \in \mathcal{N}^{[i]}, i \in \mathcal{N}. \tag{3.29}$$

Notice that $v_{\mathrm{d}}(\gamma)$ is bounded below by v_{dmin}, hence

$$|e^{[j]}(t)| \leq |\gamma^{[j]}(t) - \hat{\gamma}^{[ij]}(t)| / v_{\mathrm{dmin}}; j \in \mathcal{N}^{[i]}, i \in \mathcal{N}. \tag{3.30}$$

It can be seen that compared with the control law for continuous communications in (3.23), $v_{\mathrm{c}}^{[i]}$ in (3.28) has the contribution of the estimation error $e^{[j]}$. The key idea in the

proposed ETC mechanism is that if $e^{[j]}; j \in \mathcal{N}^{[i]}, i \in \mathcal{N}$ can be enforced to be bounded then, as we will show later, the coordination error $\boldsymbol{\xi}$ will also be bounded. To bound $e^{[j]}$, we define for every agent the variable $\hat{\gamma}^{[j]}; j \in \mathcal{N}$ as a "replica" of $\hat{\gamma}^{[ij]}; i \in \mathcal{N}^{[j]}$. Thus, if we can enforce the estimation error $\tilde{\gamma}^{[j]} \triangleq \gamma^{[j]} - \hat{\gamma}^{[j]} = \gamma^{[j]} - \hat{\gamma}^{[ij]}$ to be bounded, then from (3.30) $e^{[j]}$ will be bounded for all $j \in \mathcal{N}$.
We now introduce a mechanism to synchronize $\hat{\gamma}^{[i]}$ and $\hat{\gamma}^{[ji]}$ for all $i \in \mathcal{N}$ and $j \in \mathcal{N}^{[i]}$ (note that because the graph is symmetric, this is similar to synchronizing $\hat{\gamma}^{[j]}$ and $\hat{\gamma}^{[ij]}$ for all $j \in \mathcal{N}$ and $i \in \mathcal{N}^{[j]}$). Let $\{t_k^{[i]}\}_{k\in\mathbb{N}}$ be the sequence of time instants at which vehicle i sends its current value of $\gamma^{[i]}(t_k^{[i]}); k \in \mathbb{N}$ to its neighbors $j; j \in \mathcal{N}^{[i]}$. Note that this sequence will be defined by the ETC mechanism. During the interval $\mathcal{T}_k^{[i]} \triangleq [t_k^{[i]}, t_{k+1}^{[i]})$ we propose the following estimator for $\hat{\gamma}^{[i]}$. For $t \in \mathcal{T}_k^{[i]}$:

$$\dot{\hat{\gamma}}^{[i]}(t) = v_{\mathrm{d}}(\hat{\gamma}^{[i]}(t)), \tag{3.31a}$$

$$\hat{\gamma}^{[i]}(t_k^{[i]}) = \gamma^{[i]}(t_k^{[i]}); i \in \mathcal{N}. \tag{3.31b}$$

Equation (3.31b) implies that whenever agent i broadcasts $\gamma^{[i]}$ to its neighbors, the initial condition for $\hat{\gamma}^{[i]}$ will be reset. Similarly, let $\{t_k^{[ji]}\}_{k\in\mathbb{N}}$ be the sequence of time instants at which agent $j; j \in \mathcal{N}^{[i]}$ receives the state of agent i. The estimator for $\hat{\gamma}^{[ji]}; j \in \mathcal{N}^{[i]}, i \in \mathcal{N}$ in the interval $\mathcal{T}_k^{[ji]} \triangleq [t_k^{[ji]}, t_{k+1}^{[ji]})$ is proposed as follows:
For $t \in \mathcal{T}_k^{[ji]}$:

$$\dot{\hat{\gamma}}^{[ji]}(t) = v_{\mathrm{d}}(\hat{\gamma}^{[ji]}(t)), \tag{3.32a}$$

$$\hat{\gamma}^{[ji]}(t_k^{[ji]}) = \gamma^{[i]}(t_k^{[i]}); j \in \mathcal{N}^{[i]}, i \in \mathcal{N}. \tag{3.32b}$$

Equation (3.32b) implies that whenever agent j receives the state of agent i, the initial condition for $\hat{\gamma}^{[ji]}$ will be reset.

Remark 3.5. *The dynamics of the estimators for $\hat{\gamma}^{[i]}$ and $\hat{\gamma}^{[ji]}$, given by* (3.31a) *and* (3.32a)*, respectively, are motivated by the observation that once coordination is achieved, $\gamma^{[i]} = \gamma^{[j]}$ for all $i, j \in \mathcal{N}$, $v_{\mathrm{c}}^{[i]}$ tends to zero for all $i \in \mathcal{N}$, and all path parameters evolve with the same speed profile $v_{\mathrm{d}}(\cdot)$. As a consequence, in this specific situation the estimators truly represent the dynamics of the path parameters. See Fig.3.4 for an illustration of the synchronization between $\hat{\gamma}^{[i]}$ and $\hat{\gamma}^{[ji]}$ for all $i \in \mathcal{N}$ and $j \in \mathcal{N}^{[i]}$.*

To ensure that the estimation error $\tilde{\gamma}^{[i]}; i \in \mathcal{N}$ bounded, we allow agent i to transmit

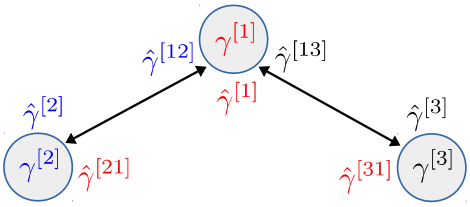


Figure 3.4: The ETC mechanism for the case of negligible delays; $\hat{\gamma}^{[i]}$ and $\hat{\gamma}^{[ji]}$ are synchronized, i.e. $\hat{\gamma}^{[i]}(t) = \hat{\gamma}^{[ji]}(t)$ for all t and $j \in \mathcal{N}^{[i]}, i \in \mathcal{N}$.

$\gamma^{[i]}$ whenever $\tilde{\gamma}^{[i]}$ hits a designed bounded threshold that, in general, can be parameterized by a function of time that we call $\eta^{[i]}(t)$. Formally, we define an event-triggering function $h^{[i]}(t)$ for the communication as

$$h^{[i]}(t) = |\tilde{\gamma}^{[i]}(t)| - \eta^{[i]}(t), \tag{3.33}$$

where $\eta^{[i]}(t)$ belongs to a class of non-negative functions $\mathcal{C}$ defined by $\mathcal{C} \triangleq \{f : R_{\geq 0} \to R_{\geq 0} | 0 \leq f(t) \leq \bar{c}\}$ for all $i \in \mathcal{N}$, where $\bar{c}$ is a uniform upper bound for the function. For example, $\eta^{[i]}(t) = c_1 + c_2 e^{-\alpha t}$ with a proper choice of c_1, c_2 and α is a typical function belonging to $\mathcal{C}$. With the above definition, agent $i; i \in \mathcal{N}$ will send its state to its neighbors whenever $h^{[i]}(t) \geq 0$. Formally, the sequences $\{t_k^{[i]}\}_{k \in \mathbb{N}}$ defining the time instants when the vehicles broadcast their states are given by:

$$t_{k+1}^{[i]} = \inf\{t \geq t_k^{[i]} : h^{[i]}(t) \geq 0\} \tag{3.34}$$

for all $i \in \mathcal{N}$. In summary, with the proposed ETC mechanism the following results hold.

Lemma 3.1. *Suppose there are no communication delays. In this case, the ETC mechanism will ensure that for all t and $i \in \mathcal{N}, j \in \mathcal{N}^{[i]}$*

i) $\hat{\gamma}^{[i]}(t) = \hat{\gamma}^{[ji]}(t)$ *and*

ii) $|\tilde{\gamma}^{[i]}(t)| = |\gamma^{[i]}(t) - \hat{\gamma}^{[i]}(t)| = |\gamma^{[i]}(t) - \hat{\gamma}^{[ji]}(t)| \leq \eta^{[i]}(t)$.

Proof: See section 3.7.4

We show next that with the ETC mechanism proposed above, and in the absence of communication delays, the coordination system satisfies an adequate ISS condition.

Theorem 3.2 (Coordination with ETC and without delays)**.** *Consider* **Problem 3.2** *and let the conditions stated in Theorem 3.1 hold. Further, let the coordination system be* *driven by the proposed ETC mechanism and the distributed control law given in* (3.26)*. Then, the closed-loop coordination error system is (ISS) with respect to the state* $\boldsymbol{\xi}$ *and the input* $\boldsymbol{\eta} \triangleq [\eta^{[1]}, ..., \eta^{[N]}]^{\mathrm{T}}$.

Proof: See section 3.7.5.

Remark 3.6. *We refer the reader to Definition A.1 in Appendix A for the concept of ISS systems. In plain terms, the result in Theorem 3.2 implies that: i) if the input* $\boldsymbol{\eta}$ *is bounded then the state* $\boldsymbol{\xi}$ *is bounded and ii) if* $\boldsymbol{\eta}(t) \to \mathbf{0}$ *as* $t \to \infty$*, then* $\boldsymbol{\xi}(t) \to \mathbf{0}$ *as* $t \to \infty$*, see the convergent input-convergent state property of an ISS system in [Sontag, 2008].*

Remark 3.7. *The above ETC mechanism extends the event triggered mechanism for single integrator MAS described in [Seyboth et al., 2013]. It also generalizes the triggering condition in [Hung and Pascoal, 2018], where the threshold functions* $\eta^{[i]}$ *are constant for all* $i \in \mathcal{N}$*. Compared to [Hung and Pascoal, 2018], this gives more flexibility to reduce the frequency of communications among the vehicles by customizing the triggering threshold function* $\eta^{[i]}(t); i \in \mathcal{N}$*.*

Remark 3.8. *Another concern with any event triggered control or communication mechanism is that if it must guarantee Zeno-free behavior. With the proposed ETC mechanism proposed in this chapter, provided that the lower bound on the threshold function* $\eta^{[i]}$ *is positive for all* $i \in \mathcal{N}$*, then it can be shown that the minimum-inter event time for every agent is strictly positive; which implies that Zeno behavior can be excluded. Intuitively, this implies that if the lower bound of* $\eta^{[i]}$ *is positive then it always takes a period of time for the estimation error* $|\tilde{\gamma}^{[i]}|$ *to reach the triggering threshold* $\eta^{[i]}$*, which is the condition to generate a new event for communication. The proof of this property is lengthy and involved. However, the proof can be done in a similar to that in Theorem 1 of [Hung et al., 2019]. A similar proof can be found in Chapter 2 for the general MAS consensus/synchronization problem.*

Clearly, Theorem 3.1 is a special case of Theorem 3.2 when $\eta^{[i]}(t) \equiv 0$ for all $i \in \mathcal{N}$. That is, $\eta^{[i]}(t) \equiv 0$ implies that the triggering condition (3.33) is satisfied at all times, making the vehicles communicate continuously. To reduce the frequency of communications, the threshold functions $\eta^{[i]}$ can be designed such that they are not necessarily identically equal to zero but $\eta^{[i]}(t) \to 0$ as $t \to \infty$; $i \in \mathcal{N}$. Then, due to the ISS property, the coordination error $\boldsymbol{\xi}$ will converge to zero as $t \to \infty$. In this set-up, the triggering threshold $\eta^{[i]}$ plays the role of a tunning knob to trade off performance of coordination against the cost of communications.

ETC mechanism with communication delays

In this subsection we consider more realistic scenarios where the communication delays are time varying and non-homogeneous. To handle communication delays, we modify slightly the proposed ETC mechanism as follows:
Consider a generic agent i with neighbors $j; j \in \mathcal{N}^{[i]}$. We recall that $t_k^{[i]}$ is the time at which agent i broadcasts its state $(\gamma^{[i]}(t_k^{[i]}))$ to its neighbors and $t_k^{[ji]}$ is the time at which agent j receives that information. Notice that without delay, agent j would receive $\gamma^{[i]}(t_k^{[i]})$ immediately, i.e. $t_k^{[ji]} = t_k^{[i]}$. We now consider the case when agent j can only receive the message broadcast by agent i after a certain time delay denoted $\Delta_k^{[ji]}$. This delay is not known in advance but we assume it can be estimated by agent j. For example, if all agents are equipped with synchronized clocks and, instead of sending only the coordination state $\gamma^{[i]}(t_k^{[i]})$, agent i also sends the tagged time $t_k^{[i]}$, then the time delay can be easily computed as $\Delta_k^{[ji]} = t_k^{[ji]} - t_k^{[i]}$. In general, we define the time delay signal as a function of time, as follows:

$$\Delta_k^{[ji]}(t) = \begin{cases} t - t_k^{[i]}, \text{ if } t_k^{[i]} \le t \le t_k^{[ji]}, \\ 0, \text{ otherwise.} \end{cases} \tag{3.35}$$

Notice how with this definition $\Delta_k^{[ji]}(t_k^{[ji]}) = t_k^{[ji]} - t_k^{[i]}$. We now modify the ETC mechanism proposed in previous section to make it robust against communication delays. To this end, the estimator (3.32) is modified as follows.
For $t \in \mathcal{T}_k^{[ji]}$,

$$\dot{\hat{\gamma}}^{[ji]}(t) = v_{\mathrm{d}}(\hat{\gamma}^{[ji]}(t)), \tag{3.36a}$$

$$\hat{\gamma}^{[ji]}(t_k^{[ji]}) = \gamma^{[i]}(t_k^{[i]}) + \int_{t_k^{[i]}}^{t_k^{[ji]}} v_{\mathrm{d}}(\hat{\gamma}^{[ji]}(\tau))\mathrm{d}\tau. \tag{3.36b}$$

These equations show that when agent $j; j \in \mathcal{N}^{[i]}$ receives $\gamma^{[i]}(t_k^{[i]})$ from agent i, $\hat{\gamma}^{[ji]}$ is reset to the initial value given by (3.36b). Compared to (3.32b), the last term in (3.36b) acts as a "compensation" term for the estimate of $\hat{\gamma}^{[ji]}$ in order to deal with the time delay

$\Delta_k^{[ji]}$.
To see how the modified ETC mechanism is robust against delays, similarly to the case without delays we examine the estimation error $\gamma^{[i]} - \hat{\gamma}^{[ji]}$ which, as we will see later, contributes to the degradation in performance of the coordination error system. To this end, we define $\bar{\eta}^{[i]}(t) \triangleq \sup_{\tau \in [0,t]} \eta^{[i]}(\tau)$ as the upper bound for $\eta^{[i]}(\tau)$ up to time t and $\bar{\Delta}^{[i]}(t) \triangleq \sup_{\tau \in [0,t]}\{\Delta_k^{[ji]}(\tau); j \in \mathcal{N}^{[i]}, t_k^{[i]} \in [0,t]\}$ as the upper bound for the time delays associated with the messages sent by agent i up to time t. We obtain the following result for the estimation error.

Lemma 3.2. *Consider the modified ETC mechanism with time-varying delays. Then, for all $t \geq 0$ and $i \in \mathcal{N}, j \in \mathcal{N}^{[i]}$*

$$|\gamma^{[i]}(t) - \hat{\gamma}^{[ji]}(t)| \leq \bar{\eta}^{[i]}(t) + (v_{\text{dmax}} - v_{\text{dmin}} + k_{\text{max}})\bar{\Delta}^{[i]}(t), \tag{3.37}$$

where $k_{\text{max}} \triangleq \max_{i \in \mathcal{N}} k_{\text{c}}^{[i]}$.

Proof: See section 3.7.6.
Let $\bar{\boldsymbol{\eta}}$ =col$(\bar{\eta}^{[i]}) \in \mathbb{R}^N$ and $\bar{\boldsymbol{\Delta}}$ =col$(\bar{\Delta}^{[i]}) \in \mathbb{R}^N$ and define

$$\boldsymbol{\sigma} = \bar{\boldsymbol{\eta}} + (v_{\text{dmax}} - v_{\text{dmin}} + k_{\text{max}})\bar{\boldsymbol{\Delta}}. \tag{3.38}$$

We obtain the following result for coordination with communication delays.

Theorem 3.3 (ETC mechanism and communication delays). *Consider* **Problem 3.2** *and let the conditions stated in Theorem 3.1 hold. Let the coordination system be driven by the modified ETC mechanism with the distributed control law given in* (3.26)*. Then, the closed-loop coordination error system is ISS with respect to the state $\boldsymbol{\xi}$ and the input $\boldsymbol{\sigma}$.*

Proof: See section 3.7.7.
Clearly, the results stated in Theorem 3.3 generalize the results in Theorem 3.1 and Theorem 3.2. The result in Theorem 3.2 is a special case of that of Theorem 3.3 when the communication delays are zero, that is, $\bar{\boldsymbol{\Delta}}(t) \equiv \mathbf{0}$. In this case, the coordination error system is ISS respect to the input $\bar{\boldsymbol{\eta}}$. Furthermore, if both $\bar{\boldsymbol{\Delta}}(t) \equiv \mathbf{0}$ and $\bar{\boldsymbol{\eta}}(t) \equiv \mathbf{0}$, then $\boldsymbol{\sigma} \equiv \mathbf{0}$. In this case, we recover the result of Theorem 3.1, that is, $\boldsymbol{\xi} = \mathbf{0}$ is GAS.

Remark 3.9. *We notice that having synchronized clocks on the vehicles to compute time delays is not a strong assumption. In fact, with current technology it is neither difficult nor overly expensive to have synchronized clocks (with a drift of less than 200 ns in 24 hours) on-board of all the vehicles that are part of a formation. This solution was recently implemented and tested in the scope of the EU WiMUST project [Kebkal et al., 2017].*

3.3.2 MPC for constrained path following

Section 3.3.1 provided a solution to the computation of the correction speed $v_{\mathrm{c}}^{[i]}$ in order to achieve coordination. With the correction speed, the total speeds assigned to the vehicles are given by

$$u^{[i]} = (v_{\mathrm{d}}(\gamma^{[i]}) + v_{\mathrm{c}}^{[i]})g^{[i]}(\gamma^{[i]}); i \in \mathcal{N}. \tag{3.39}$$

As shown in the proof of Lemma 3.1, the reference speed for $u^{[i]}$, given by (3.39), satisfies the constraint $u_{\min}^{[i]} \leq u^{[i]} \leq u_{\max}^{[i]}$ for all $i \in \mathcal{N}$. Replacing the vehicle speed u in (3.14) by (3.39) for vehicle i, the resulting path following error system for vehicle i is given by

$$\dot{\mathbf{x}}^{[i]} = \mathbf{f}^{[i]}(\mathbf{x}^{[i]}, \mathbf{u}^{[i]}) = \begin{bmatrix} g^{[i]}(\gamma^{[i]}) \left(-v_\gamma^{[i]}(1 - \kappa^{[i]}(\gamma^{[i]})e_y^{[i]}) + (v_{\mathrm{d}}(\gamma^{[i]}) + v_{\mathrm{c}}^{[i]}) \cos(e_\psi^{[i]})\right) \\ g^{[i]}(\gamma^{[i]}) \left(-\kappa^{[i]}(\gamma^{[i]})v_\gamma^{[i]}e_x^{[i]} + (v_{\mathrm{d}}^{[i]} + v_{\mathrm{c}}^{[i]}) \sin(e_\psi^{[i]})\right) \\ r^{[i]} - \kappa^{[i]}(\gamma^{[i]})g^{[i]}(\gamma^{[i]})v_\gamma^{[i]} \end{bmatrix}, \tag{3.40}$$

where $\mathbf{u}^{[i]} = (v_\gamma^{[i]}, r^{[i]})$. It follows from (3.3) and (3.15) that $\mathbf{u}^{[i]}$ is constrained to the set

$$\mathbb{U}_{\mathrm{pf}}^{[i]} \triangleq \{(v_\gamma^{[i]}, r^{[i]}) : |v_\gamma^{[i]}| \leq v_{\max}^{[i]} \text{ and } |r^{[i]}| \leq r_{\max}^{[i]}\}. \tag{3.41}$$

We are now in a position to design an MPC scheme to drive the path following error system (3.40) to zero subject to the input constraint set $\mathbb{U}_{\mathrm{pf}}^{[i]}$ defined by (3.41).
We define a finite horizon open loop optimal control problem (FOCP) $\mathcal{OCP}(t, \mathbf{x}^{[i]}(t), \gamma^{[i]}(t), v_{\mathrm{c}}^{[i]}(t), T_{\mathrm{p}})$ that the sampled-data MPC must solve at every sampling time as follows:

Definition 3.1. $\mathcal{OCP}(t, \mathbf{x}^{[i]}(t), \gamma^{[i]}(t), v_{\mathrm{c}}^{[i]}(t), T_{\mathrm{p}})$

$$\min_{\bar{\mathbf{u}}^{[i]}(\cdot)} J^{[i]}\left(\mathbf{x}^{[i]}(t), \gamma^{[i]}(t), v_{\mathrm{c}}^{[i]}(t), \bar{\mathbf{u}}^{[i]}(\cdot)\right),$$

with

$$J^{[i]}\left(\cdot\right) \triangleq \int_{t}^{t+T_{\mathrm{p}}} l^{[i]}\left(\bar{\mathbf{x}}^{[i]}(\tau), \gamma^{[i]}(\tau), v_{\mathrm{c}}^{[i]}(\tau), \bar{\mathbf{u}}^{[i]}(\tau)\right) \mathrm{d}\tau$$

subject to

$$\dot{\bar{\mathbf{x}}}^{[i]}(\tau) = \mathbf{f}^{[i]}\left(\bar{\mathbf{x}}^{[i]}(\tau), \bar{\mathbf{u}}^{[i]}(\tau)\right), \quad \tau \in [t, t+T_{\mathrm{p}}], \tag{3.42a}$$

$$\bar{\mathbf{x}}^{[i]}(t) = \mathbf{x}^{[i]}(t), \tag{3.42b}$$

$$\bar{v}_{\mathrm{c}}^{[i]}(\tau) = -k_{\mathrm{c}}^{[i]} \tanh\left(\textstyle\sum_{j\in\mathcal{N}^{[i]}} \bar{z}^{[i]}(\tau) - \bar{\hat{z}}^{[ij]}(\tau)\right), \tag{3.42c}$$

$$\dot{\bar{\gamma}}^{[i]}(\tau) = v_{\gamma}^{[i]}(\tau), \tau \in \left[t, t+T_{\mathrm{p}}\right], \bar{\gamma}^{[i]}(t) = \gamma^{[i]}(t), \tag{3.42d}$$

$$\dot{\bar{\hat{\gamma}}}^{[ij]}(\tau) = v_{\mathrm{d}}(\hat{\gamma}^{[ij]}(\tau)), \tau \in \left[t, t+T_{\mathrm{p}}\right], \tag{3.42e}$$

$$\bar{\hat{\gamma}}^{[ij]}(t) = \hat{\gamma}^{[ij]}(t); \quad j \in \mathcal{N}^{[i]}, \tag{3.42f}$$

$$\bar{\mathbf{u}}^{[i]}(\tau) \in \mathbb{U}_{\mathrm{pf}}^{[i]}, \quad \tau \in \left[t, t+T_{\mathrm{p}}\right], \tag{3.42g}$$

$$\tfrac{\partial V}{\partial \mathbf{x}^{[i]}}\mathbf{f}^{[i]}(\mathbf{x}^{[i]}(t), \bar{\mathbf{u}}^{[i]}(t)) \leq \tfrac{\partial V}{\partial \mathbf{x}^{[i]}}\mathbf{f}^{[i]}(\mathbf{x}^{[i]}(t), \mathbf{u}_{\mathrm{n}}(\mathbf{x}^{[i]}(t))). \tag{3.42h}$$

In the constraint equations (3.42), the variables with bar denote predicted variables, to distinguish them from the real variables without a bar. Specifically, $\bar{\mathbf{x}}^{[i]}(\tau)$ is the predicted trajectory of the path following error which is computed using the dynamic model (3.40) and the initial conditions (3.42b); $\bar{\gamma}^{[i]}(\tau)$ is the predicted value of the path parameter $\gamma^{[i]}$ driven by the path following input $\bar{\mathbf{u}}^{[i]}(\tau)$; $\bar{\hat{\gamma}}^{[ij]}$ is the prediction of the state of neighboring agent $j; j \in \mathcal{N}^{[i]}$ by using the estimator (3.36) over the prediction horizon T_{p}; $\bar{z}^{[i]}$ and $\bar{\hat{z}}^{[ij]}$ are computed using (3.21), (3.27) with predicted $\bar{\gamma}^{[i]}$ and $\bar{\hat{\gamma}}^{[ij]}$, respectively. The constraint (3.42h) is referred as a *stability constraint* to guarantee stability. This constraint is constructed based on a Lyapunov function $V : \mathbb{R}^3 \to \mathbb{R}_{\geq 0}$ and its associated stabilizing constrained control law $\mathbf{u}_{\mathrm{n}} : \mathbb{R}^3 \to \mathbb{U}_{\mathrm{pf}}^{[i]}$. This setup is inspired by the result in [de la Pena and Christofides, 2008] to improve the performance of path following. Finally, $l^{[i]} : \mathbb{R}^3 \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}^2 \to \mathbb{R}_{\geq 0}$ is the stage cost of the final horizon cost $J^{[i]}$.

In state feedback sampled-data MPC, the optimal control problem $\mathcal{OCP}(\cdot)$ is repeatedly solved at every discrete sampling instant $t_i = i\delta$, $i \in \mathbb{N}_+$, where δ is a sampling interval. Let $\bar{\mathbf{u}}^{[i]*}(\tau)$ be the optimal solution of the optimal control problem $\mathcal{OCP}(\cdot)$. The MPC control law $\mathbf{u}^{[i]}_{\mathrm{mpc}}(\cdot)$ is then defined as

$$\mathbf{u}^{[i]}_{\mathrm{mpc}}(t) = \bar{\mathbf{u}}^{[i]*}(t) \text{ for } t \in [t_i, t_i + \delta]. \tag{3.43}$$

Before presenting the main result for the path following problem with the proposed MPC scheme, we make the following assumptions.

Assumption 2.

A2.1 The stage cost $l^{[i]}(\cdot)$ is continuous, positive definite, and $l^{[i]}(\cdot) = 0$ when $\bar{\mathbf{x}}^{[i]} = \mathbf{0}$ and $\mathbf{u}^{[i]}_{\mathrm{a}} \triangleq [-v^{[i]}_{\gamma} + (v_{\mathrm{d}}(\gamma^{[i]}) + v^{[i]}_{\mathrm{c}}) \cos e^{[i]}_{\psi}, r^{[i]} - \kappa^{[i]}(\gamma^{[i]}) g^{[i]}(\gamma^{[i]}) v^{[i]}_{\gamma}]^{\mathrm{T}} = \mathbf{0}$.

A2.2 Given the path following error dynamics in (3.40), there exist a Lyapunov function $V : \mathbb{R}^3 \to \mathbb{R}_{\geq 0}$ such that V is positive definite and $V(\mathbf{x}^{[i]}) = 0$ only for $\mathbf{x}^{[i]} = \mathbf{0}$, and an associated nonlinear feedback control law $\mathbf{u}_{\mathrm{n}} : \mathbb{R}^3 \to \mathbb{U}^{[i]}_{\mathrm{pf}}$ that satisfies $\frac{\partial V}{\partial \mathbf{x}^{[i]}} \mathbf{f}(\mathbf{x}^{[i]}, \mathbf{u}_{\mathrm{n}}(\mathbf{x}^{[i]})) \leq 0$ for all $\mathbf{x}^{[i]}$. Further, $\mathbf{u}_{\mathrm{n}}(\mathbf{x}^{[i]})$ globally stabilizes (3.40).

We now state an important result for the constrained path following problem using the proposed MPC scheme.

Theorem 3.4 (Path following with MPC)**.** *Consider the path following error system (3.40) subject to the input constrained set $\mathbb{U}_{\mathrm{pf}}$ given by (3.41), controlled by the proposed MPC scheme, and let Assumption 2 hold true. Then, the origin of the path following error is globally asymptotically stable.*

Proof: See section 3.7.8.
The most important requirement in the proposed MPC scheme is the existence of a stabilizing control law $\mathbf{u}_{\mathrm{n}}(\cdot)$ and an associated Lyapunov function $V(\cdot)$ that satisfies Assumption A2.2. It can be shown that the control law in the following lemma satisfies the assumption.

Lemma 3.3 (Global Constrained Nonlinear PF Controller)**.** *Consider the path following error system (3.40) and let $v_{\max}$ in (3.41) be chosen such that*

$$v_{\mathrm{dmax}} + k_{\mathrm{c}} < v_{\max} < r_{\max} / \max(|\kappa(\gamma) g(\gamma)|). \tag{3.44}$$

Then, the global Lyapunov based control law given by

$$\mathbf{u}_{\mathrm{n}}(\mathbf{x}) = \begin{bmatrix} v_\gamma \\ r \end{bmatrix} = \begin{bmatrix} \frac{1}{g(\gamma)}\left(u\cos(e_\psi) + k_1\tanh(e_x)\right) \\ -\frac{k_3 e_y u \sin(e_\psi)}{(1+e_x^2+e_y^2)e_\psi} - k_2\tanh(e_\psi) + \kappa(\gamma)g(\gamma)v_\gamma \end{bmatrix}, \tag{3.45}$$

where $k_1, k_2, k_3 \in \mathbb{R}_{>0}$ are tuning parameters that satisfy

$$\begin{aligned} &0 < k_1 \le v_{\max}\min(g(\gamma)) - (v_{\mathrm{dmax}} + k_c)g_{\max}, \\ &0.5k_3 u_{\max} + k_2 \le r_{\max} - \max(|\kappa(\gamma)g(\gamma)|)v_{\max} \end{aligned} \tag{3.46}$$

renders the origin of the path following error system GAS. Further, the Lyapunov function associated with the control law (3.45)*, given by*

$$V(\mathbf{x}) = \frac{k_3}{2}\ln(1 + e_x^2 + e_y^2) + \frac{1}{2}e_\psi^2, \tag{3.47}$$

satisfies Assumption A2.2.

Proof: See section 3.7.9.

Remark 3.10. *Notice that for the sake of simplicity we dropped the subscript* [*i*] *in equations* (3.44) - (3.47)*.*

Remark 3.11. *The MPC scheme proposed above is but one possible solution to the problem of stabilizing the path following error system. One can use for example the MPC proposed in [Yu et al., 2015], where terminal constraints are imposed to guarantee recursive feasibility and stability. However, due to the need of a terminal set, the region of attraction in [Yu et al., 2015] is local, while the region of attraction for the path following error in Theorem 3.4 is global. The choice of Lyapunov function in* (3.47) *is inspired by the work of [Jiang et al., 2001] on mobile robot trajectory tracking.*

It is obvious that with the constraint (3.42h) the MPC scheme improves the performance of the closed-loop path following error system compared to the nonlinear control law. A comparative study can be found in [Hung and Pascoal, 2018, Hung et al., 2018].

3.4 Overall closed-loop CPF system

In the previous section, with a view to adopting a decoupling strategy for the design of a cooperative path following system, we proposed a distributed CPF strategy to solve two key problems involved: *i*) multiple agent coordination with an ETC mechanism and *ii*) MPC for input-constrained path following of each agent. The resulting distributed CPF strategy can be implemented using Algorithm 3.1 described below. The algorithm embodies in its structure the decoupling methodology adopted, that is, the CPF control system can be seen as a two-layer control structure. In this context, coordination and communications together play the role of an upper layer whose objective is to coordinate the path parameters to reach a desired formation, while the main objective of the path following layer is to steer the vehicles to their assigned paths. In Theorem 3.3 and Theorem 3.4, we have shown the important facts that if the two layers are considered separately, the path following system of each vehicle is GAS while the coordination error that involves the path parameters is ISS respect to the input $\boldsymbol{\sigma}$ that includes the *trigger threshold* $\boldsymbol{\eta}$ and the communication delays. In this section, we shall state results for the overall closed-loop CPF system where the interaction of the two layers is taken explicitly into account.

Algorithm 3.1 MPC-CPF with the ETC mechanism for vehicle i

```
At every sampled time $t$, vehicle $i$ implements following procedures:
procedure COORDINATION AND COMMUNICATION
    if $h^{[i]}(t) \geq 0$ then
        Broadcast $\gamma^{[i]}(t)$;
        Reset $\hat{\gamma}^{[i]}$ using (3.31b);
    if Receive a new message from vehicle $j$ then
        if $j \in \mathcal{N}^{[i]}$ then
            Reset $\hat{\gamma}^{[ij]}$ using (3.36b);
    Run the estimator (3.31);
    Run the estimator (3.36);
    Update the correction speed $v_{\mathrm{c}}^{[i]}(t)$ using (3.26);
return $v_{\mathrm{c}}^{[i]}(t)$
procedure PATH FOLLOWING
    Update the path following error $\mathbf{x}^{[i]}(t)$;
    Solve the $\mathcal{OCP}(\cdot)$ problem to find $\bar{\mathbf{u}}^{[i]}(\cdot)$;
    Use the MPC control law (3.43) to update $v_{\gamma}^{[i]}(t), r^{[i]}(t)$;
    Update the vehicle's speed $u^{[i]}(t)$ using (3.39);
return $u^{[i]}(t), r^{[i]}(t), v_{\gamma}^{[i]}(t)$
```

Theorem 3.5. *Consider the complete closed-loop CPF system composed by*

- *A set of N vehicles, whose motions are described by* (3.2) *subject to the input constraints given by* (3.3)*.*
- *A set of paths given by* (3.1) *and the desired speed profiles $v_{\mathrm{d}}(\gamma^{[i]})$ satisfying Condition 1, with all $i \in \mathcal{N}$.*

Let the vehicles be controlled by the proposed MPC-CPF and the ETC mechanism given by Algorithm 1. Then, the overall closed-loop system is ISS respect to the state $\mathbf{x}_{\mathrm{cl}} \triangleq [\mathbf{x}_{\mathrm{pf}}^{\mathrm{T}}, \boldsymbol{\xi}^{\mathrm{T}}]^{\mathrm{T}}$ and the input $\boldsymbol{\sigma}$, where $\mathbf{x}_{\mathrm{pf}}$ is the state of the path following layer defined as $\mathbf{x}_{\mathrm{pf}} \triangleq \mathrm{col}(\mathbf{x}^{[i]})$ and recall that $\boldsymbol{\sigma}$, given by (3.38)*, is the disturbance made by the triggering functions and communication delays.*

Proof: See section 3.7.10.

3.5 Simulation examples

We consider a fleet of five Medusa class of AMVs with the input constraints $u^{[i]} \in [0.2, 2]\mathrm{m\,s^{-1}}$ and $r^{[i]} \in [\text{-}0.2, 0.2]\mathrm{rad\,s^{-1}}$ for all $i = \mathcal{N} \triangleq \{1, ..., 5\}$ (see [Abreu et al., 2016b] for the details of the vehicles' specifications). The vehicles are required to execute the two types of CPF missions described in Table 3.2, with the paths parameterized by their normalized arc-lengths. For triangular formations, the vehicles are required to maneuver along parallel paths while adopting the shape of a triangle, see Fig.3.6, left. For circular formations, the vehicles are required to maneuver along nested circumferences and align themselves radially, see Fig.3.6, right. In Table 3.2, for triangular formations, $d^{[i]}$ and $c^{[i]}$ are parameters specifying the desired cross-track and along-track distances between the vehicles, while for circular formations, $a^{[i]}$ are the radii of the circumferences. The communication topology adopted is depicted in Fig.3.5, which shows the indexes of the vehicles and the bidirectional communication links between them (represented by arrows). In the proposed MPC scheme, the Lyapunov-based controller in Lemma 3.3 is used to construct the constraint (3.42h). The tunning parameters for the Lyapunov-based controller, the coordination controller, and the event triggering threshold functions are set in Table 3.3. Notice that the coordination gain $k_{\mathrm{c}}^{[i]}$ is chosen to satisfy conditions (3.24), while the gains

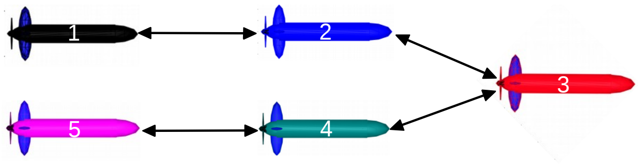


Figure 3.5: Communication topology

Table 3.2: Planed missions

Formation	Planned paths	v_d
Triangular	$\mathbf{p}_\mathrm{d}^{[i]}(\gamma^{[i]}) = [a(\gamma^{[i]} - c^{[i]}), d^{[i]}]^\mathrm{T}$, $a = 50\mathrm{m}, c^{[1]} = c^{[5]} = 0, c^{[2]} = c^{[4]} = 0.1, c^{[3]} = 0.2$, $d^{[1]} =$ -10m$, d^{[2]} =$ -5m$, d^{[3]} = 0\mathrm{m}, d^{[4]} = 5, d^{[5]} = 10\mathrm{m}$	0.02
Circular	$\mathbf{p}_\mathrm{d}^{[i]}(\gamma^{[i]}) = [a^{[i]}\cos(\gamma^{[i]}), a^{[i]}\sin(\gamma^{[i]})]^\mathrm{T}$, $a^{[1]} = 30\mathrm{m}, a^{[2]} = 33\mathrm{m}, a^{[3]} = 36\mathrm{m}$, $a^{[4]} = 39\mathrm{m}, a^{[5]} = 42\mathrm{m}$	0.02

Table 3.3: Controller parameters

Controller	Tunning parameters
Path Following	$k_1^{[1]} = 0.3, k_1^{[2]} = 0.33, k_1^{[3]} = 0.36$, $k_1^{[4]} = 0.39, k_1^{[5]} = 0.42$, $k_2^{[i]} = 0.06, k_3^{[i]} = 0.09, v_\mathrm{max}^{[i]} = 0.05, \forall i = 1, ..., 5$
Coordination	$k_\mathrm{c}^{[i]} = 0.008, i = 1, ..., 5$ $\eta^{[i]}(t) = c_1 e^{-\alpha t} + \epsilon, \forall i = 1, ..., 5$ $c_1 = 0.1, \alpha = 0.2, \epsilon =$ 5e-3

for the Lyapunov-based controller are chosen to satisfy conditions in Lemma 3.3 for all vehicles. The stage cost for the MPC scheme is defined as the quadratic form

$$l^{[i]}(\cdot) = \bar{\mathbf{x}}^{[i]}(\tau)^\mathrm{T} Q \bar{\mathbf{x}}^{[i]}(\tau) + \mathbf{u}_\mathrm{a}^{[i]}(\tau)^\mathrm{T} R \mathbf{u}_\mathrm{a}^{[i]}(\tau),$$

where Q=diag$(1, 1, 2)$ and R=diag$(2, 20)$. The sampling interval is set to $\delta = 0.2$s and the prediction horizon is set to $T_\mathrm{p} = 2$s. To solve the finite optimal control problem $\mathcal{OCP}(\cdot)$, we used Casadi, an open source optimization tool described in [Andersson et al., 2018]. Communication delays are set $\Delta = 2$s for all transmitted messages and for both missions.

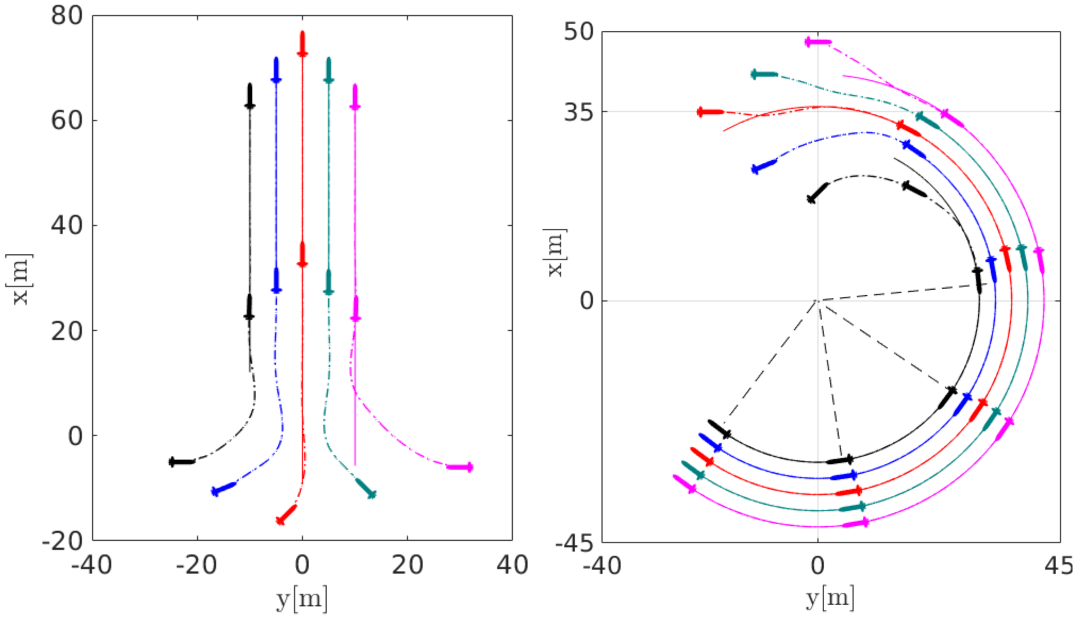


Figure 3.6: Trajectories of the vehicles. Left (triangular formation), right (circular formation). Solid lines are the desired paths, dash-dot lines are the trajectories of the vehicles.

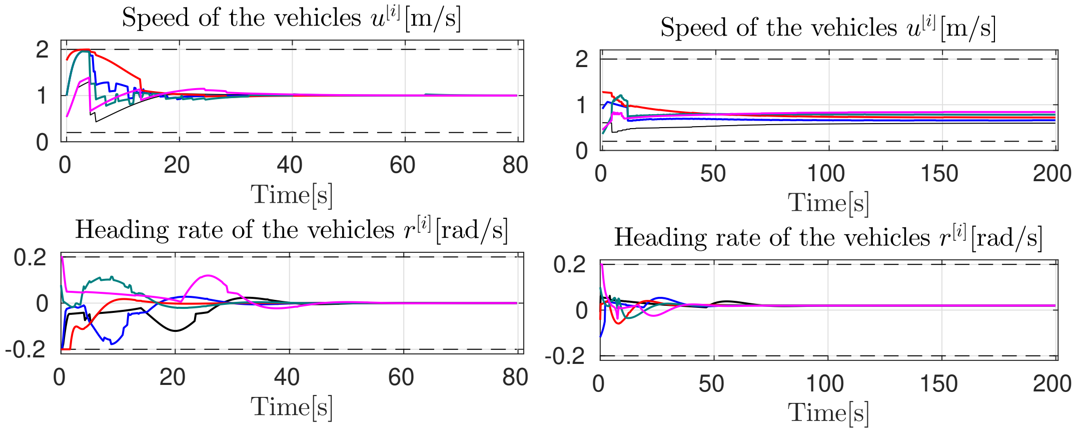


Figure 3.7: Vehicles inputs. Black dash lines are bounds of the inputs.

The trajectories of the vehicles are shown in Fig.3.6. It is visible that the vehicles converge to the desired paths and reach the desired formations in both missions. The performance of the proposed CPF strategy for the two missions is illustrated in Fig.3.7–Fig.3.9. It can be seen from Fig.3.7 that the inputs of the vehicles produced by the proposed CPF strategy satisfy the constraints assigned to them. Notice also in Fig.3.8

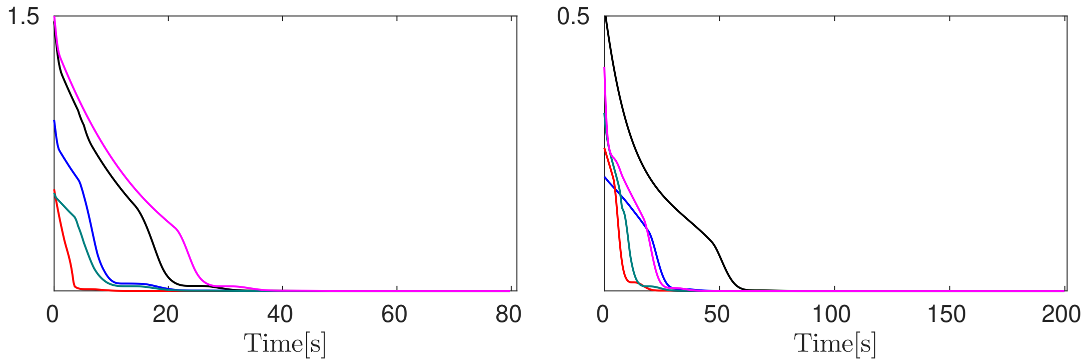


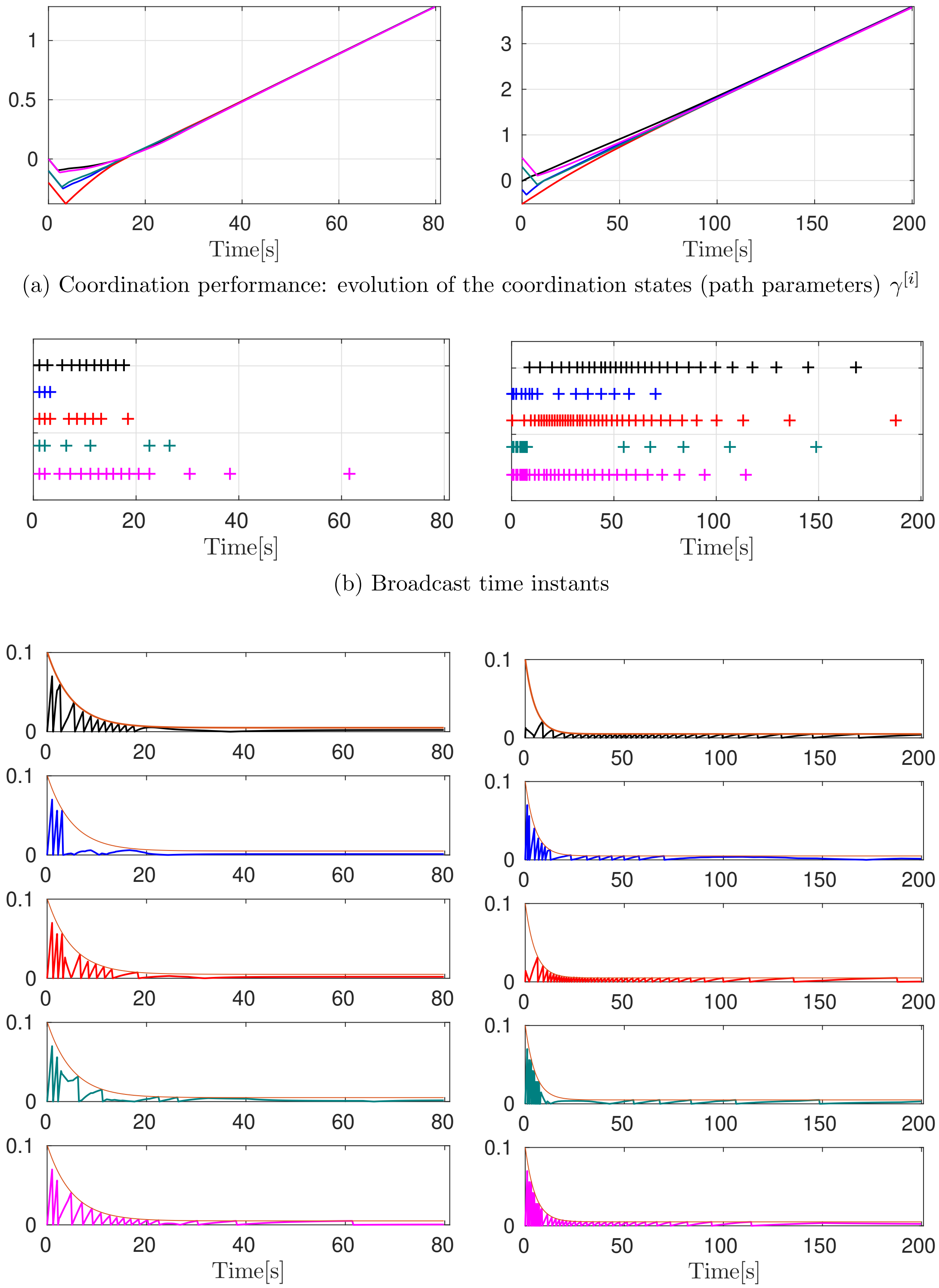
Figure 3.8: Path following performance: evolution of the Lyapunov function V for path following.

how the Lyapunov functions for path following of the vehicles are monotonically decreasing to zero, corroborating the results that the path following errors are asymptotically stable.

Regarding coordination among the vehicles, Fig.3.9(a) shows that the coordination states (path parameters) reach consensus asymptotically and evolve with the desired common speed profile $v_{\rm d}$. In terms of communications between the vehicles, Fig.3.9(b) indicates that at beginning of the simulation, communications take place more frequently. In contrast, when the vehicles reach the desired formations, they no longer need to communicate. This can be explained with the help of Fig.3.9(c) which shows the estimation errors and the triggering threshold functions. At the beginning of the missions, the dynamics of the path parameters are disturbed by the path following system (because the vehicles are away from their paths) and the correction speeds are updated from the coordination system. As a consequence, there are significant errors in the path parameters' estimates. Hence, the estimation errors hit the threshold functions frequently which, in turn, triggers communications more frequently.

3.6 Conclusions

We proposed a solution to the constrained CPF problem that exploits the tools of Model Predictive Control, network theory, and event triggered communications. The main contribution of this work lies in the fact the proposed strategy is not only capable of explicitly handling practical constraints on vehicles' inputs and on the topology of the communications network, but also saves communication bandwidth. We have shown that the path following error of all vehicles error is GAS, which is a strong result for an input-constrained



(a) Coordination performance: evolution of the coordination states (path parameters) $\gamma^{[i]}$

(b) Broadcast time instants

(c) Estimation errors $\tilde{\gamma}^{[i]}$. Solid orange is *threshold triggering functions* $\eta^{[i]}$

Figure 3.9: Performance of coordination and communications. Left (triangular formation), right (circular formation)

system. Practically, this implies that regardless of the initial positions and orientations, the vehicles always converge to and follow their assigned paths. At the coordination level, we proposed a novel distributed control law with an ETC mechanism for the synchronization of multi agent nonlinear system that takes into account the agent input constraints. Future work will aim at implementing the proposed control method in the Medusa vehicles that are property of IST, and assess their performance at sea, [Abreu et al., 2016b].

3.7 Proofs

The following lemmas will be used in the proof of important theorems and corollaries.

3.7.1 Lemma on connectivity of graph

Lemma 3.4. *Let L be the Laplacian matrix of a graph $\mathcal{G}$. Suppose $\mathcal{G}$ is undirected and connected. Then, for any vector $\mathbf{x} \in \mathbb{R}^N$ and $\mathbf{x} \perp \mathbf{1}$, the following inequalities hold:*

$$\lambda_2 \|\mathbf{x}\|^2 \leq \mathbf{x}^{\mathrm{T}} L \mathbf{x} \leq \lambda_N \|\mathbf{x}\|^2, \tag{3.48a}$$

$$\lambda_2 \|\mathbf{x}\| \leq ||L\mathbf{x}|| \leq \lambda_N \|\mathbf{x}\|, \tag{3.48b}$$

where λ_2 and $\lambda_N \in \mathbb{R}_{>0}$ are the second smallest and the largest eigenvalues of L, respectively.

Proof. Let $\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_N \in \mathbb{R}^N$ be the eigenvectors of L associated with the eigenvalues $\lambda_1, \lambda_2, ..., \lambda_N$. Let $\lambda_1 \leq \lambda_2 \leq ... \leq \lambda_N$. Since the graph is undirected and connected, it is well-know that $\lambda_1 = 0$ and $\mathbf{v}_1 = \mathbf{1}$, and $\lambda_i > 0$ for all $2 \leq i \leq N$. From the Courant-Fischer theorem in [Horn and Johnson, 2012], it follows that

$$\lambda_2 = \min_{\mathbf{x} \neq \mathbf{0} \text{ and } \mathbf{x} \perp \mathbf{1}} \frac{\mathbf{x}^{\mathrm{T}} L \mathbf{x}}{\mathbf{x}^{\mathrm{T}} \mathbf{x}}, \quad \lambda_N = \max_{\mathbf{x} \neq \mathbf{0}} \frac{\mathbf{x}^{\mathrm{T}} L \mathbf{x}}{\mathbf{x}^{\mathrm{T}} \mathbf{x}}.$$

Therefore, the inequality (3.48a) holds. We now consider the matrix $B = L^2$. It can be easily checked that B has an eigenvalue at 0 and with an associated eigenvector $\mathbf{1}$. Let $\lambda_i(B)$ be the eigenvalues of B. Clearly, $\lambda_i(B) = \lambda_i^2, i = 1, ..., N$. Applying again the Courant- Fischer theorem, it follows that for any $\mathbf{x} \in \mathbb{R}^N$ and $\mathbf{x} \perp \mathbf{1}$, $\lambda_2(B)\|\mathbf{x}\|^2 = \lambda_2^2 \|\mathbf{x}\|^2 \leq \mathbf{x}^{\mathrm{T}} B \mathbf{x} = \|L\mathbf{x}\|^2 \leq \lambda_N(B)\|\mathbf{x}\|^2 = \lambda_N^2 \|\mathbf{x}\|^2$. Therefore, the inequality (3.48b) holds. ■

3.7.2 Lemma on tan hyperbolic function

Lemma 3.5. *Let $\mathbf{y} \in \mathbb{R}^n$ and $\theta \in (0, 1)$. Then, for all $\mathbf{x} \in \mathbb{R}^n$ such that $\|\mathbf{x}\|_\infty \geq (2n-1)\|\mathbf{y}\|_\infty / \theta$ the following inequality holds*

$$-\mathbf{x}^{\mathrm{T}} \mathbf{tanh}(\mathbf{x} + \mathbf{y}) \leq -\frac{\|\mathbf{x}\|_\infty}{2} \tanh\left((1-\theta)\|\mathbf{x}\|_\infty\right).$$

Proof. In the proof, we will use the following important facts:
Let $a, b \in \mathbb{R}, \alpha > 0$. *If* $|\alpha a| \geq |b|$ *then*

Fact 1: $a \tanh(\alpha a + b) \geq 0$ *and*

Fact 2: $a \tanh(\alpha a + b) \geq |a| \tanh(|\alpha a| - |b|)$.

Fact 1 can be checked by noting that if $|a| \geq |b|$, then a and $\tanh(a + b)$ have the same sign. *Fact 2* holds because tanh is a monotonically increasing function of its argument. The proof of the Lemma proceeds as follows:
Let $\bar{x} \triangleq \|\mathbf{x}\|_\infty$, $\bar{y} \triangleq \|\mathbf{y}\|_\infty$, and $m \triangleq (2n-1)\bar{y}/\theta$ and

$$S \triangleq -\mathbf{x}^{\mathrm{T}}\mathbf{tanh}(\mathbf{x}+\mathbf{y}) = -\textstyle\sum_{i=1}^{n} x_i \tanh(x_i + y_i) \tag{3.49}$$

Recall from *Fact 1* that $x_i \tanh(x_i + y_i) \geq 0$ if $|x_i| \geq |y_i|$ and define the two sets

$$S_1 \triangleq \{x_i : |x_i| \geq \bar{y}\} \text{ and } S_2 \triangleq \{x_i : |x_i| < \bar{y}\}.$$

With the above definition, equation (3.49) can be rewritten as

$$S = \underbrace{-\sum_{x_i \in S_1} x_i \tanh(x_i + y_i)}_{=:C_1} \underbrace{- \sum_{x_i \in S_2} x_i \tanh(x_i + y_i)}_{=:C_2} \tag{3.50}$$

Using Fact 1, we conclude that all the products in the sum of C_1 are negative. Later, we will show that C_2 is bounded. We will henceforth use the condition given in the Lemma that $\bar{x} \geq m$. Note that $m > \bar{y}$ for all $\theta \in (0, 1)$, and therefore $\bar{x} > \bar{y}$. It follows that the set S_1 has at least one element, that is, $|S_1| \geq 1$ and therefore $|S_2| \leq n - 1$. Let i^* be the index such that $x_{i^*} \in S_1$ and $|x_{i^*}| = \bar{x}$. Since $x_i \tanh(x_i + y_i) \geq 0$ for all $x_i \in S_1$, it follows that

$$\begin{aligned} C_1 &\leq -x_{i^*} \tanh(x_{i^*} + y_{i^*}) \\ &= -x_{i^*} \tanh((1-\theta)x_{i^*} + \theta x_{i^*} + y_{i^*}) \\ &= \underbrace{-\frac{x_{i^*} \tanh((1-\theta)x_{i^*})}{\sigma}}_{=:D_1} \underbrace{-\frac{x_{i^*} \tanh(\theta x_{i^*} + y_{i^*})}{\sigma}}_{=:D_2}, \end{aligned} \tag{3.51}$$

where $\sigma \triangleq 1 + \tanh((1-\theta)x_{i^*}) \tanh(\theta x_{i^*} + y_{i^*})$. Because $|\theta x_{i^*}| = \theta \bar{x} \geq \theta m > \bar{y} \geq y_{i^*}$, using *Fact 1* it follows that $0 \leq \tanh((1-\theta)x_{i^*}) \tanh(\theta x_{i^*} + y_{i^*}) \leq 1$. Therefore, $1 \leq \sigma \leq 2$.

Recall that $|x_{i^*}| = \bar{x} = \|\mathbf{x}\|_\infty$ and $1 \le \sigma \le 2$ we can conclude that

$$D_1 \le -\frac{\|\mathbf{x}\|_\infty}{2} \tanh\left((1-\theta)\|\mathbf{x}\|_\infty\right). \tag{3.52}$$

Furthermore, since $|\theta x_{i^*}| \ge y_{i^*}$, using *Fact 2*, it follows that

$$D_2 \le -\frac{|x_{i^*}|\tanh(|\theta x_{i^*}| - |y_{i^*}|)}{2} \le -\frac{(2n-1)}{2\theta}\bar{y}\tanh((2n-1)\bar{y} - \bar{y}) \le -(n-1)\bar{y}\tanh(2(n-1)\bar{y}). \tag{3.53}$$

At this point, we observe that

- For $n = 1$, $D_2 \le 0$. Notice also that $C_2 = 0$ because $|S_2| = 0$.
- For $n \ge 2$, $D_2 \le -(n-1)\bar{y}\tanh(2\bar{y})$. Also, since $|x_i| \le \bar{y}$ for all $x_i \in C_2$ and $|S_2| \le (n-1)$, it follows that $C_2 \le (n-1)\bar{y}\tanh(2\bar{y})$.

We conclude that $D_2 + C_2 \le 0$ for all $n \ge 1$. As consequence, $S = D_1 + D_2 + C_2 \le D_1$. Hence, from (3.52) we conclude that

$$S = -\mathbf{x}^{\mathrm{T}}\mathbf{tanh}(\mathbf{x}+\mathbf{y}) \le -\frac{\|\mathbf{x}\|_\infty}{2}\tanh\left((1-\theta)\|\mathbf{x}\|_\infty\right)$$

for all $\theta \in (0,1)$ and $\|\mathbf{x}\|_\infty \ge (2n-1)\|\mathbf{y}\|_\infty/\theta$. This concludes the proof. ■

3.7.3 Proof of Theorem 3.1

The proof of the theorem is done in two steps.
Step 1: Feasibility. Recall that $u^{[i]} = g^{[i]}(\gamma^{[i]})(v_{\mathrm{d}}(\gamma^{[i]}) + v_{\mathrm{c}}^{[i]})$, Replacing $v_{\mathrm{c}}^{[i]}$ by (3.23) yields

$$u^{[i]} = g^{[i]}(\gamma^{[i]})\Big(v_{\mathrm{d}}(\gamma^{[i]}) - k_{\mathrm{c}}^{[i]}\tanh\Big(\textstyle\sum_{j\in\mathcal{N}^{[i]}} z^{[i]} - z^{[j]}\Big)\Big).$$

Since $g^{[i]}(\gamma^{[i]})$ and $k_c^{[i]}$ are positive for all $\gamma^{[i]}$ and $i \in \mathcal{N}$, it follows that

$$g^{[i]}(\gamma^{[i]})(v_{\mathrm{d}}(\gamma^{[i]}) - k_{\mathrm{c}}^{[i]}) \le u^{[i]} \le g^{[i]}(\gamma^{[i]})(v_{\mathrm{d}}(\gamma^{[i]}) + k_{\mathrm{c}}^{[i]}).$$

Furthermore, because $k_{\mathrm{c}}^{[i]} \le c_u/g_{\max}^{[i]}$ for all $i \in \mathcal{N}$ (see (3.24)), it follows from condition C1.2 that $u^{[i]}$ satisfies the inequality $u_{\min}^{[i]} \le u^{[i]} \le u_{\max}^{[i]}$ for all $i \in \mathcal{N}$, from which it can be concluded that the correction speed (3.23) satisfies the linear speed constraint (3.20).

Step 2: Global Consensus. From (3.21), (3.19) and (3.23) we obtain

$$\begin{aligned}\dot{z}^{[i]} &= \tfrac{1}{v_{\mathrm{d}}(\gamma^{[i]})}(v_{\mathrm{d}}(\gamma^{[i]}) - k_{\mathrm{c}}^{[i]} \tanh\left(\textstyle\sum_{j\in\mathcal{N}^{[i]}} z^{[i]} - z^{[j]}\right)\\ &= 1 - d^{[i]} \tanh\left(\textstyle\sum_{j\in\mathcal{N}^{[i]}} z^{[i]} - z^{[j]}\right)\end{aligned}$$

where $d^{[i]} \triangleq k_{\mathrm{c}}^{[i]}/v_{\mathrm{d}}(\gamma^{[i]}) > 0$ for all $\gamma^{[i]}$ and $i \in \mathcal{N}$. As a consequence, the dynamics of $\mathbf{z}$ are described by

$$\dot{\mathbf{z}} = \mathbf{1} - K\mathbf{tanh}(L\mathbf{z}), \tag{3.54}$$

where $K \triangleq \mathrm{diag}(d^{[1]}, d^{[2]}..., d^{[N]}) \in \mathbb{R}^{N\times N}$. We now consider the Lyapunov function candidate for the closed loop coordination system, defined as

$$V_c(\boldsymbol{\xi}) = \frac{1}{2}\boldsymbol{\xi}^{\mathrm{T}} L\boldsymbol{\xi}. \tag{3.55}$$

Intuitively, V_c measures the disagreement between the agents' states (path parameters). Notice that from the definition in (3.22), $\boldsymbol{\xi} \perp \mathbf{1}$. Using Lemma 3.4 we obtain $V_c(\boldsymbol{\xi}) \geq \lambda_2\|\boldsymbol{\xi}\|^2/2 \geq 0$ for all $\boldsymbol{\xi}$ and $V_c(\boldsymbol{\xi}) = 0$ iff $\boldsymbol{\xi} = \mathbf{0}$. Therefore, V_c is a positive definite function. Computing the time derivative of V_c and using (3.54), we obtain

$$\begin{aligned}\dot{V}_c &= \boldsymbol{\xi} L\dot{\boldsymbol{\xi}} = \mathbf{z}^{\mathrm{T}} L\dot{\mathbf{z}}\\ &= -\mathbf{z}^{\mathrm{T}} LK\mathbf{tanh}(L\mathbf{z}) = -\mathbf{q}^{\mathrm{T}} K\mathbf{tanh}(\mathbf{q}) \leq 0\end{aligned} \tag{3.56}$$

for all $\boldsymbol{\xi}$, where $\mathbf{q} \triangleq L\mathbf{z} = L\boldsymbol{\xi}$. Because $K \succ 0$, $\dot{V}_c = 0$ iff $\mathbf{q} = \mathbf{0}$. Furthermore, $L\mathbf{1} = \mathbf{0}$, this implies $\dot{V}_c = 0$ when either $\boldsymbol{\xi} = \mathbf{0}$ or $\boldsymbol{\xi}$ spans $\mathbf{1}$. However, from the definition in (3.22) $\boldsymbol{\xi}$ is always orthogonal to $\mathbf{1}$, hence $\dot{V}_c = 0$ iff $\boldsymbol{\xi} = \mathbf{0}$. This implies that V_c stops decreasing if and only if $\boldsymbol{\xi} = \mathbf{0}$. Therefore, we conclude that $\boldsymbol{\xi} = \mathbf{0}$ is GAS. This implies that $z^{[i]}(t) = z^{[j]}(t)$ or, equivalently, $\gamma^{[i]}(t) = \gamma^{[j]}(t)$ for all $i, j \in \mathcal{N}$ as $t \to \infty$. ■

3.7.4 Proof of Lemma 3.1

The first relation comes from the fact that without delays the estimators for $\hat{\gamma}^{[i]}$ and $\hat{\gamma}^{[ji]}$ in (3.31) and (3.32), respectively, are always initialized at the same value. Furthermore, since $v_{\mathrm{d}}(\cdot)$ is identical to both estimators, $\hat{\gamma}^{[i]}(t) = \hat{\gamma}^{[ji]}(t)$ for all t (see Fig.3.4 as an example). The second relation stems from the fact with the ETC mechanism $\tilde{\gamma}^{[i]}$ is always enforced to satisfy $|\tilde{\gamma}^{[i]}(t)| \leq \eta^{[i]}(t)$ and, since *i*) holds for all t then *ii*) holds for all t. ■

3.7.5 Proof of Theorem 3.2

The proof is done in three steps:
Step 1: From Lemma 3.1, and (3.30) we conclude that $|e^{[i]}(t)| \leq \eta^{[i]}(t)/v_{\mathrm{dmin}}$ for all t and all $i \in \mathcal{N}$. Letting $\mathbf{e} = [e^{[1]}, e^{[2]}, ..., e^{[N]}]^{\mathrm{T}}$, it follows that

$$\|\mathbf{e}\|_\infty \leq \|\boldsymbol{\eta}\|_\infty / v_{\mathrm{dmin}} \leq \sqrt{N}\|\boldsymbol{\eta}\|/v_{\mathrm{dmin}}. \tag{3.57}$$

Step 2: We show that the closed-loop coordination system is ISS respect to the state $\boldsymbol{\xi}$ and input $\boldsymbol{\eta}$. With the control law (3.28), the dynamics of $\mathbf{z}$ can be rewritten as

$$\dot{\mathbf{z}} = \mathbf{1} - K\mathbf{tanh}(L\mathbf{z} + A\mathbf{e}), \tag{3.58}$$

where A is the adjacency matrix of the graph. Notice that compared with (3.54), for the case continuous communications, the term $A\mathbf{e}$ can be viewed as an external disturbance. It follows from the above that the derivative of Lyapunov function candidate V_c in (3.55) is given by

$$\begin{aligned}\dot{V}_c &= -\mathbf{z}^{\mathrm{T}} LK\mathbf{tanh}(L\mathbf{z} + A\mathbf{e}) \\ &\leq -d_{\min}\mathbf{q}^{\mathrm{T}}\mathbf{tanh}(\mathbf{q} + A\mathbf{e}),\end{aligned}$$

where $d_{\min} \triangleq \min_{i\in\mathcal{N}} d^{[i]} = k_{\min}/v_{\mathrm{dmax}}$ and $k_{\min} \triangleq \min_{i\in\mathcal{N}} k_{\mathrm{c}}^{[i]}$. Now, using Lemma 3.5 (in Appendix A), for any $\theta \in (0,1)$ it follows that

$$\dot{V}_c \leq -d_{\min}\frac{\|\mathbf{q}\|_\infty}{2}\tanh\left((1-\theta)\|\mathbf{q}\|_\infty\right)$$

for all $\|\mathbf{q}\|_\infty \geq (2N-1)\|A\mathbf{e}\|_\infty/\theta$. Recall that $\mathbf{q} = L\boldsymbol{\xi}$. Using Lemma 3.4, we obtain $\|\mathbf{q}\|_\infty = \|L\boldsymbol{\xi}\|_\infty \geq \lambda_2\|\boldsymbol{\xi}\|/\sqrt{N}$. Furthermore, from (3.57), it follows that $\|A\mathbf{e}\|_\infty \leq \|A\|_\infty\|\mathbf{e}\|_\infty \leq \|A\|_\infty\sqrt{N}\|\boldsymbol{\eta}\|/v_{\mathrm{dmin}}$. As a consequence,

$$\dot{V}_c \leq -d_{\min}\frac{\lambda_2\|\boldsymbol{\xi}\|}{2\sqrt{N}}\tanh\left(\frac{(1-\theta)\lambda_2\|\boldsymbol{\xi}\|}{\sqrt{N}}\right) =: -W_1(\boldsymbol{\xi}) \tag{3.59}$$

for all $\|\boldsymbol{\xi}\| \geq \frac{N(2N-1)\|A\|_\infty}{\lambda_2\theta v_{\mathrm{dmin}}}\|\boldsymbol{\eta}\| =: \rho(\|\boldsymbol{\eta}\|)$.

It can bee seen that W_1 is positive definite and ρ is a class $\mathcal{K}$ function. Furthermore,

V_c is bounded according to

$$\alpha_1(\|\boldsymbol{\xi}\|) \leq V_c \leq \alpha_2(\|\boldsymbol{\xi}\|), \tag{3.60}$$

where $\alpha_1(\|\boldsymbol{\xi}\|) \triangleq \lambda_2\|\boldsymbol{\xi}\|^2$ and $\alpha_2(\|\boldsymbol{\xi}\|) \triangleq \lambda_N\|\boldsymbol{\xi}\|^2$ are two $\mathcal{K}$ class functions. Therefore, using the ISS-Theorem (see Theorem A.1 in Appendix A) we conclude that V_c is an ISS-Lyapunov function for the closed-loop coordination error system. Hence, the closed loop coordination system is ISS respect to the state $\boldsymbol{\xi}$ and the input $\boldsymbol{\eta}$. This concludes the proof. ■

3.7.6 Proof of Lemma 3.2

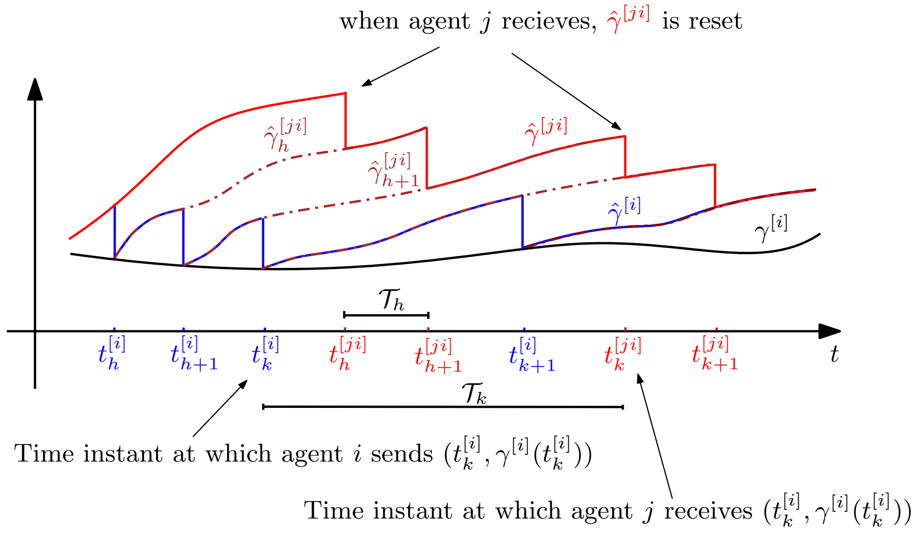
Let $\mathcal{T}_k \triangleq [t_k^{[i]}, t_k^{[ji]})$ be the time interval between the instant $t_k^{[i]}$ when agent i broadcasts a message including $(t_k^{[i]}, \gamma^{[i]}(t_k^{[i]}))$ and instant $t_k^{[ji]}$ when agent $j; j \in \mathcal{N}^{[i]}$ receives this message. It is important to note that the triggering condition for agent i is independent of the communication delays. Therefore, it is possible that agent i may end up sending new messages to agent j before the first message has been received by the latter agent. At the same time, from the point of view of agent j, this agent might also receive different messages from agent i in the interval $\mathcal{T}_k$. These scenarios are illustrated in Fig.3.10.
We now consider the estimation error $\gamma^{[i]} - \hat{\gamma}^{[ji]}$ in the interval $\mathcal{T}_k$. Notice that in this interval $\hat{\gamma}^{[ji]}$ may be discontinuous, because whenever agent j receives a new message from agent i, $\hat{\gamma}^{[ji]}$ will be reset according to (3.36b). Let $t_h^{[ji]}, t_{h+1}^{[ji]}, ..., t_{h+H}^{[ji]} \in \mathcal{T}_k$ be a sequence of time instants at which agent $j; j \in \mathcal{N}^{[i]}$ receives messages broadcast by agent i at the corresponding times $t_h^{[i]}, t_{h+1}^{[i]}, ..., t_{h+H}^{[i]}$. Without loss of generality, we assume that $t_k^{[i]} \leq t_h^{[ji]} \leq t_{h+1}^{[ji]} \leq ... \leq t_{h+H}^{[ji]} \leq t_k^{[ji]}$.

We now consider the estimation error $\gamma^{[i]} - \hat{\gamma}^{[ji]}$ in each interval $\mathcal{T}_h \triangleq [t_h^{[ji]}, t_{h+1}^{[ji]}) \subseteq \mathcal{T}_k$. To this end, we define a new variable $\hat{\gamma}_h^{[ji]}$ as follows

$$\hat{\gamma}_h^{[ji]}(t) = \gamma^{[i]}(t_h^{[i]}) + \textstyle\int_{t_h^{[i]}}^{t} v_{\mathrm{d}}(\hat{\gamma}_h^{[ji]}(\tau))\mathrm{d}\tau \tag{3.61}$$

From (3.36) and (3.61), it can be observed that $\hat{\gamma}^{[ji]}(t) = \hat{\gamma}_h^{[ji]}(t)$ for all $t \in \mathcal{T}_h$. This is also illustrated in Fig.3.10. Therefore, in the interval $\mathcal{T}_h$, instead of examining the error between $\hat{\gamma}^{[ji]}$ and $\gamma^{[i]}$, we examine the error between $\hat{\gamma}_h^{[ji]}$ and $\gamma^{[i]}$. When $t \in \mathcal{T}_h$, from (3.61) we obtain

$$\begin{aligned}\hat{\gamma}_h^{[ji]}(t) &= \hat{\gamma}_h^{[ji]}(t_h^{[ji]}) + \textstyle\int_{t_h^{[ji]}}^{t} v_{\mathrm{d}}(\hat{\gamma}_h^{[ji]}(\tau))\mathrm{d}\tau \\ &= \hat{\gamma}_h^{[ji]}(t_{h+1}^{[i]}) + \textstyle\int_{t_{h+1}^{[i]}}^{t} v_{\mathrm{d}}(\hat{\gamma}_h^{[ji]}(\tau))\mathrm{d}\tau.\end{aligned} \tag{3.62}$$



when agent j recieves, $\hat{\gamma}^{[ji]}$ is reset

Time instant at which agent i sends $(t_k^{[i]}, \gamma^{[i]}(t_k^{[i]}))$

Time instant at which agent j receives $(t_k^{[i]}, \gamma^{[i]}(t_k^{[i]}))$

Figure 3.10: Illustration of the evolution of variables with communication delays. Solid black denotes the true trajectory of $\gamma^{[i]}$. Solid blue denotes the estimate of $\gamma^{[i]}$ at agent i. Solid red denotes the estimate of $\gamma^{[i]}$ at agent j, namely $\hat{\gamma}^{[ji]}$, while dot-brown denotes the auxiliary variable $\hat{\gamma}_h^{[ji]}$.

From (3.19), it follows that

$$\begin{aligned} \gamma^{[i]}(t) &= \gamma^{[i]}(t_h^{[ji]}) + \textstyle\int_{t_h^{[ji]}}^{t} \Big(v_{\mathrm{d}}(\gamma^{[i]}(\tau)) + v_{\mathrm{c}}^{[i]}(\tau) \Big)\, \mathrm{d}\tau \\ &= \gamma^{[i]}(t_{h+1}^{[i]}) + \textstyle\int_{t_{h+1}^{[i]}}^{t} \Big(v_{\mathrm{d}}(\gamma^{[i]}(\tau)) + v_{\mathrm{c}}^{[i]}(\tau) \Big)\, \mathrm{d}\tau. \end{aligned} \tag{3.63}$$

Subtracting both sides of (3.62) from equation (3.63) and taking absolute values, yields

$$|\gamma^{[i]}(t) - \hat{\gamma}_h^{[ji]}(t)| \leq |\gamma^{[i]}(t_{h+1}^{[i]}) - \hat{\gamma}_h^{[ji]}(t_{h+1}^{[i]})| + \underbrace{\textstyle\int_{t_{h+1}^{[i]}}^{t} |v_{\mathrm{d}}(\gamma^{[i]}(\tau)) - v_{\mathrm{d}}(\hat{\gamma}_h^{[ji]}(\tau))| \mathrm{d}\tau + \int_{t_{h+1}^{[i]}}^{t} |v_{\mathrm{c}}^{[i]}(\tau)| \mathrm{d}\tau}_{=:A} . \tag{3.64}$$

Notice that at the time $t_{h+1}^{[i]}$ at which $\gamma^{[i]}$ is reset, $|\gamma^{[i]}(t_{h+1}^{[i]}) - \hat{\gamma}_h^{[ji]}(t_{h+1}^{[i]})| \leq \eta^{[i]}(t_{h+1}^{[i]})$ (see also Fig.3.10). Since $t_{h+1}^{[i]} \leq t$ for all $t \in \mathcal{T}_h$, $\eta^{[i]}(t_{h+1}^{[i]}) \leq \bar{\eta}^{[i]}(t)$. Therefore, $|\gamma^{[i]}(t_{h+1}^{[i]}) - \hat{\gamma}_h^{[ji]}(t_{h+1}^{[i]})| \leq \bar{\eta}^{[i]}(t)$. In addition, from (3.64) we obtain

$$A \leq (v_{\mathrm{dmax}} - v_{\mathrm{dmin}} + k_{\max})(t - t_{h+1}^{[i]}) = (v_{\mathrm{dmax}} - v_{\mathrm{dmin}} + k_{\max})\Delta_{h+1}^{[ji]}(t).$$

Since $t^{[i]}_{h+1} \leq t$ for all $t \in \mathcal{T}_h$, $\Delta^{[ji]}_{h+1}(t) \leq \bar{\Delta}^{[i]}(t)$. We conclude that for all $t \in \mathcal{T}_h$

$$|\gamma^{[i]}(t) - \hat{\gamma}^{[ji]}(t)| \leq \bar{\eta}^{[i]}(t) + (v_{\mathrm{dmax}} - v_{\mathrm{dmin}} + k_{\mathrm{max}})\bar{\Delta}^{[i]}(t). \tag{3.65}$$

Using a similar reasoning, it can be shown that in any time interval $t \in [t^{[ji]}_{h+n}, t^{[ji]}_{h+n+1}) \subseteq \mathcal{T}_k$; $n = 1, ..., H$, inequality (3.65) also holds. Hence, we conclude that the inequality (3.65) holds for all $t \geq 0$. This completes the proof. ■

3.7.7 Proof of Theorem 3.3

The proof is similar to that of Theorem 3.2. Using Lemma 3.2 and (3.30), it follows that

$$|e^{[i]}(t)| \leq \left(\bar{\eta}^{[i]}(t) + (v_{\mathrm{dmax}} - v_{\mathrm{dmin}} + k_{\mathrm{max}})\bar{\Delta}^{[i]}(t)\right)/v_{\mathrm{dmin}}.$$

Hence, from (3.38),

$$\|\mathbf{e}\|_\infty \leq \|\boldsymbol{\sigma}\|_\infty / v_{\mathrm{dmin}} \leq \sqrt{N}\|\boldsymbol{\sigma}\|/v_{\mathrm{dmin}}. \tag{3.66}$$

Proceeding similarly to Step 2 in the proof of Theorem 3.2, we can show that the inequality (3.59) holds for all $\|\boldsymbol{\xi}\| \geq \rho(\|\boldsymbol{\sigma}\|)$. Therefore, the closed loop coordination system is ISS respect to the state $\boldsymbol{\xi}$ and the input $\boldsymbol{\sigma}$. This concludes the proof. ■

3.7.8 Proof of Theorem 3.4

Recursive Feasibility. Clearly, $\mathbf{u}_{\mathrm{n}}(\mathbf{x}^{[i]}(t))$ is one of the feasible solutions of $\bar{\mathbf{u}}^{[i]}(\tau), \tau \in [t, t+\delta]$ satisfying the constraints (3.42g) and (3.42h), while the remaining $\bar{\mathbf{u}}^{[i]}(\tau), \tau \in [t+\delta, t+T_{\mathrm{p}}]$ can be chosen freely in the input space $\mathbb{U}^{[i]}_{\mathrm{pf}}$.

Stability. The proof of globally asymptotic stability relies on the contractive constraint (3.42h) which, together with Assumption A2.2, implies that

$$\dot{V}(t) = \tfrac{\partial V}{\partial \mathbf{x}^{[i]}}\mathbf{f}(\mathbf{x}^{[i]}(t), \mathbf{u}^{[i]}_{\mathrm{mpc}}(t)) \leq \tfrac{\partial V}{\partial \mathbf{x}^{[i]}}\mathbf{f}(\mathbf{x}^{[i]}(t), \mathbf{u}_{\mathrm{n}}(\mathbf{x}^{[i]}(t))) \leq 0.$$

We consider two possible cases for $\mathbf{u}^{[i]}_{\mathrm{mpc}}(t)$. In the first case, the MPC scheme finds $\mathbf{u}^{[i]}_{\mathrm{mpc}}(t) \neq \mathbf{u}_{\mathrm{n}}(\mathbf{x}^{[i]}(t))$, yielding $\dot{V}(t) = \tfrac{\partial V}{\partial \mathbf{x}^{[i]}}\mathbf{f}(\mathbf{x}^{[i]}(t), \mathbf{u}^{[i]}_{\mathrm{mpc}}(t)) < \tfrac{\partial V}{\partial \mathbf{x}^{[i]}}\mathbf{f}(\mathbf{x}^{[i]}(t), \mathbf{u}_{\mathrm{n}}(\mathbf{x}^{[i]}(t))) \leq 0$, that is, V strictly decreasing. In the second case, $\mathbf{u}^{[i]}_{\mathrm{mpc}}(t) = \mathbf{u}_{\mathrm{n}}(\bar{\mathbf{x}}^{[i]}(t))$. Since $\mathbf{u}_{\mathrm{n}}(\mathbf{x}^{[i]})$ globally stabilizes (3.40), we can conclude that $\mathbf{x}^{[i]} \to \mathbf{0}$ as $t \to \infty$. Thus, $\mathbf{u}^{[i]}_{\mathrm{mpc}}(t)$ globally stabilizes (3.40). ■

3.7.9 Proof of Lemma 3.3

The proof is done in two steps:
Feasibility. To show that the heading rate r is feasible, we compute

$$\begin{aligned}|r| &= \left|-\tfrac{k_3 e_y u \sin e_\psi}{(1+e_x^2+e_y^2)e_\psi} - k_2 \tanh(e_\psi) + \kappa(\gamma)g(\gamma)v_\gamma\right| \\ &\leq 0.5k_3 u_{\max} + k_2 + \max(|\kappa(\gamma)g(\gamma)|)v_{\max}.\end{aligned}$$

Clearly, by choosing k_2, k_3 positive such that (3.46) is satisfied, it follows that $|r| \leq r_{\max}$. Next, it is easy to check v is feasible by computing

$$|v_\gamma| = \left|\tfrac{1}{g(\gamma)}\left(u\cos(e_\psi) + k_1 \tanh(e_x)\right)\right| \leq (|u| + k_1)/g(\gamma),$$

Notice that according to (3.39), $|u| \leq (v_{\text{dmax}} + k_c)\max(g(\gamma))$. Hence, Choosing k_1 such that (3.46) is satisfied and using condition (3.44), it follows that $|v_\gamma| \leq v_{\max}$.
Global asymptotic stability. Replacing $\mathbf{u}$ in (3.40) with the control law (3.45) yields the closed-loop path following error system described by

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, u) = \begin{bmatrix} u\cos(e_\psi)e_y + k_1 \tanh(e_x)(1 - e_y) \\ u\left(\sin(e_\psi) - e_x\cos(e_\psi)\right) - k_1\tanh(e_x)e_x \\ -\frac{k_3 e_y \sin(e_\psi)}{(1+e_x^2+e_y^2)e_\psi}u - k_2\tanh(e_\psi) \end{bmatrix}, \tag{3.67}$$

where u is given by (3.39). Notice that system (3.67) is non-autonomous since $u(t)$ is in general a function of time (as v_c depends on the the triggering functions that are time-dependent). To show that $\mathbf{x} = \mathbf{0}$ is GAS we need to show that

i) $\mathbf{x} = \mathbf{0}$ is stable and

ii) $\mathbf{x} = \mathbf{0}$ is globally attractive, i.e. $\lim\limits_{t\to\infty} \mathbf{x}(t) = \mathbf{0}$ for any initial condition $\mathbf{x}(t_0)$.

i). Stability.
Computing the time derivative of the Lyapunov function given in (3.47) along the trajectory of (3.67) yields

$$\dot{V}(\mathbf{x}) = -\frac{k_3 k_1 e_x \tanh(e_x)}{1 + e_x^2 + e_y^2} - k_2 e_\psi \tanh(e_\psi) \leq 0 \tag{3.68}$$

for all $\mathbf{x}$. Using the fact that $\dot{V}$ is a negative semi-definite function and V is radially unbounded, it follows that $\mathbf{x} = \mathbf{0}$ is stable and $\mathbf{x}(t)$ is bounded given any initial condition

$\mathbf{x}(t_0)$ at an arbitrary initial time t_0.
ii). $\mathbf{x} = \mathbf{0}$ is globally attractive.
From (3.68), it can be seen that $\dot{V}$ is negative everywhere except on the line $\Omega \triangleq \{\mathbf{x}|e_x = 0, e_\psi = 0\}$ where $\dot{V}(\mathbf{x}) = 0$. For the system to maintain the $\dot{V}(\mathbf{x}) = 0$ condition, the trajectory of the system must be confined to the line Ω. Unless $e_y = 0$, this is impossible because from the third equation of (3.67)

$$\dot{e}_\psi \equiv 0 \Rightarrow -\frac{k_3 e_y(t)}{1 + e_x^2(t) + e_y^2(t)} u(t) \equiv 0. \tag{3.69}$$

Because $u(t) \neq 0$ for all t, (3.69) holds iff $e_y(t) \equiv 0$. This implies that the system can maintain the $\dot{V}(\mathbf{x}) = 0$ condition only at the origin $\mathbf{x} = \mathbf{0}$. Therefore, $V(\mathbf{x}(t))$ must decrease toward to zero. As a consequence, $\mathbf{x} \to \mathbf{0}$ as $t \to \infty$. This completes the proof.
Regarding this proof, two interesting observations can be made. Firstly, no matter what $u(t)$ is, as long as it does not go through zero the path following error always converges to zero. This means that the update of correction speeds from the coordination layer does not affect to asymptotic stability of the path following error system, i.e. the path following error always converges to zero asymptotically regardless of the correction speeds. Hence, from a stability point of view, the path following control layer is decoupled from the coordination layer. Secondly, the fact that convergence of $\mathbf{x}(t)$ to zero is obtained if $u(t) > 0$ for all t is intuitive, in the sense that forward motion is required to ensure that, by rotating, the vehicle will be able to track the "virtual reference" (the origin of the parallel transport attached to the path).

Remark 3.12. *Recall that the reference speed u assigned for the vehicle, in general, is a function of time due to the ETC mechanism; and therefore the resulting path following error system is non-autonomous. This is the reason why we did not use LaSalle's invariance principle to conclude the stability in the proof. Note that this is different from the single path following studied in [Hung et al., 2018] where the speed of the vehicle depends only on the path parameter, which makes the path following error system autonomous; and therefore the proof of stability can be done using the invariance principle.*

3.7.10 Proof of Theorem 3.5

The proof follows from the results stated in Theorem 3.3 and Theorem 3.4. As stated in Theorem 3.4, the convergence of the path following error of each vehicle to zero is independent of the correction speed computed by the coordination layer. Without loss of generality, the dynamics of $\mathbf{x}_{\text{pf}}$ can be written as

$$\dot{\mathbf{x}}_{\text{pf}} = \mathbf{f}_{\text{pf}}(\mathbf{x}_{\text{pf}}, t). \tag{3.70}$$

From Theorem 3.4, $\mathbf{x}_{\text{pf}} = \mathbf{0}$ is GAS. We now consider the coordination error vector $\boldsymbol{\xi}$ for the overall closed-loop CPF system. In Section 3.2.2, as an intermediate step in the design of a CPF control law, we assumed the vehicles were already on their assigned paths. That is, $\mathbf{x}^{[i]}$ was assumed to be zero for all $i \in \mathcal{N}$. Therefore, we did not take into account the effect of the path following layer on the coordination layer. However, in the overall closed-loop CPF system the dynamics of the path parameters in (3.19) can be rewritten as

$$\dot{\gamma}^{[i]} = v_{\text{d}}(\gamma^{[i]}) + v_{\text{c}}^{[i]} + d_{\text{pf}}^{[i]}, \qquad i \in \mathcal{N}, \tag{3.71}$$

where $d_{\text{pf}}^{[i]} : (\gamma^{[i]}, \mathbf{x}^{[i]}) \rightarrow d_{\text{pf}}^{[i]}(\gamma^{[i]}, \mathbf{x}^{[i]}); i \in \mathcal{N}$ can be viewed as an external disturbance introduced by the path following system. Notice that $d_{\text{pf}}^{[i]}$ is bounded for all $i \in \mathcal{N}$ because $v_{\text{d}}(\cdot)$, $v_{\text{c}}^{[i]}$ are bounded and $\dot{\gamma}^{[i]} = v^{[i]}$, where $v^{[i]}$ is always bounded in the set $\mathbb{U}_{\text{pf}}^{[i]}$ for all $i \in \mathcal{N}$. In addition, it follows from Theorem 3.4 that $\mathbf{x}^{[i]} \rightarrow \mathbf{0}$ as $t \rightarrow \infty$ for all $i \in \mathcal{N}$. This, together with the first equation of (3.40) imply that as $t \rightarrow \infty$, $\dot{\gamma}^{[i]} \rightarrow v_{\text{d}} + v_{\text{c}}^{[i]}$ for all $i \in \mathcal{N}$. From (3.71), this means that $d_{\text{pf}}^{[i]} \rightarrow 0$ as $t \rightarrow \infty$. With the disturbance from the path following layer, the dynamics of $\mathbf{z}$ in (3.58) are rewritten as

$$\dot{\mathbf{z}} = \mathbf{1} - K\mathbf{tanh}(L\mathbf{z} + \boldsymbol{\sigma}) + \mathbf{d}_{\text{pf}}, \tag{3.72}$$

where $\mathbf{d}_{\text{pf}} = [d_{\text{bf}}^{[1]}/v_{\text{d}}(\gamma^{[1]}), ..., d_{\text{bf}}^{[N]}/v_{\text{d}}(\gamma^{[N]})]^{\text{T}} \in \mathbb{R}^N$. As a consequence,

$$\dot{\boldsymbol{\xi}} = W\dot{\mathbf{z}} = -WK\mathbf{tanh}(L\boldsymbol{\xi} + \boldsymbol{\sigma}) + W\mathbf{d}_{\text{pf}} =: \mathbf{f}_{\text{c}}(\boldsymbol{\xi}, \mathbf{x}_{\text{pf}}). \tag{3.73}$$

Since $d_{\text{pf}}^{[i]} \rightarrow 0$ as $t \rightarrow \infty$, $\mathbf{d}_{\text{pf}} \rightarrow \mathbf{0}$ as $t \rightarrow \infty$. Further, $\mathbf{d}_{\text{pf}}$ is always bounded, hence the solution for $\boldsymbol{\xi}$ in (3.73) is always bounded. As a consequence, from [Loría and Panteley, 2005] we conclude that the cascaded system composed by (3.70) and (3.73) is ISS respect to state $\boldsymbol{\xi}_{\text{cl}} \triangleq [\mathbf{x}_{\text{pf}}^{\text{T}}, \boldsymbol{\xi}^{\text{T}}]$ and the input $\boldsymbol{\sigma}$. This completes the proof. ■

Part II

Range-based Cooperative SLAP with Multiple Autonomous Vehicles

4

Observability analysis

Contents

This chapter addresses observability of two problems: i) target localization with a single or multiple mobile trackers (e.g. robots) using range measurements from the trackers to the target, and ii) navigation of an autonomous vehicle under unknown constant disturbance using range measurements from the vehicle to a single beacon. For the first problem, we consider three scenarios: i) the target is fixed, ii) the target's velocity vector is unknown but constant, and iii) the target's acceleration vector is unknown but constant. The main contributions of the chapter are threefold: i) we derive a set of necessary and sufficient conditions on the motion of the trackers under which the target's state, that might include the target's position, velocity and acceleration vectors is globally observable, and ii) we show how the conditions derived lend themselves to an intuitive geometric interpretation that yields valuable guidelines to plan the tracker's motion iii) We compare and suggest suitable fillers for the target localization and single beacon navigation problems.

4.1 Literature review

The problems of range-based navigation and target localization have been studied extensively in recent years. In what follows, by range-based navigation we mean the problem of having a vehicle estimate its own state (position, possibly velocity and acceleration) using measurements of the distances of the vehicle to a single or multiple beacons whose positions are known [Bayat et al., 2016]. Target localization, on the other hand, is defined for one or multiple trackers as the problem of tracking the state of a fixed or moving target using range measurements from the tracker(s) to the target, [Crasta et al., 2018]. The two problems are dual and impose the same fundamental issues on observability analysis. Namely, for the case of target tracking, to find under what conditions on the relative motion of the tracker(s) with respect to the target is the state of the latter observable. This problem is challenging due to the fact that range measurements are nonlinear function of the target's position, thus making the observability of the resulting system hard to analyze.

One of the earliest results on the observability of target localization can be found in [Song, 1999], where the authors conclude that: "the tracker maneuver should include a nonzero jerk motion to track a target with a constant acceleration vector while a nonzero acceleration motion is required to track a target with a constant velocity vector". Although this conclusion sounds logical it can be shown that even if these conditions are satisfied, the target might not be localizable (trackable). To illustrate this, we consider a target starting at an initial position $\mathbf{q}(t_0)$ and moving along a straight line with velocity vector $\mathbf{v}$, as shown in Fig.4.1. We also consider a tracker that can be started anywhere and moves with a velocity vector $\alpha(t)\mathbf{v}$, with $\ddot{\alpha}(t) \neq 0$ for all t, so that it satisfies the conditions stated above. Note that with this velocity vector, the tracker will move along a straight line parallel to the trajectory of the target. With this motion of the tracker, as shown in the figure, there exists a virtual target moving with the same velocity vector $\mathbf{v}$ reflected about the tracker's trajectory, such that the ranges from the tracker to the true target and its mirror image are the same. This, obviously, makes it impossible to distinguish the target and its mirror image and brings attention to the need to study the problem of target observability in a rigorous setting.

Because of the nonlinearity of the map from linear positions to ranges, the range-based observability problem must be addressed in a nonlinear system setting. This can be done by resorting to tools from differential algebraic geometry described in [Hermann and Krener, 1977], where a sufficient condition for local observability of a nonlinear system is given in terms of an observability rank condition. In this context, the local observability for an AUV modeled by an integrator is studied in [Arrichiello et al., 2013]. This work was extended in [Palma et al., 2017] for an AUV modeled as a double integrator system. Another approach to study local observability of a given nonlinear system involves the use of the Fisher information matrix (FIM) to measure the amount of information that the range measurements carry about the target's motion. In contrast to the methods mentioned above this approach yields quantitative results. With this approach, the observability problem is converted into that of finding conditions on the tracker's trajectory so as to guarantee that the FIM is non-singular, thus ensuring that the target state is at least locally observable. See for example [Song, 1999, Masmitja et al., 2018, Ristic et al., 2002, Crasta et al., 2018] and the references therein for related work in the area.

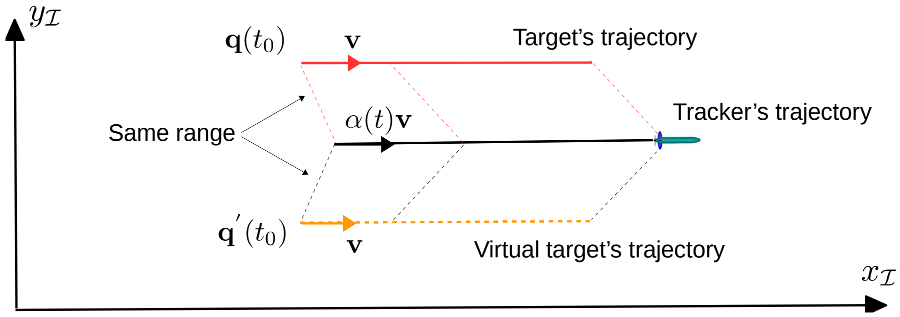


Figure 4.1: A counterexample: the target is not localizable if the tracker moves parallel to the target.

Other interesting results were reported in [Batista et al., 2011], where the authors considered the problem of localizing a source (fixed-target) using range measurements obtained with a single tracker. The underlying idea behind the work was to transform the original nonlinear system into a higher dimensional linear time varying (LTV) system via an appropriate state augmentation. Conditions on the tracker's motion to localize the target were then derived for the LTV system, which were proved to be sufficient for observability of the original nonlinear system. Work along the same lines is reported in [Indiveri et al., 2016] where a different state argumentation is proposed to avoid the singularity that might happen in [Palma et al., 2017, Batista et al., 2011] when the range from the tracker to the target is close to zero.

Motivated by the above considerations, this chapter mainly addresses the observability problem of range-based target localization with one or two trackers. We then extend the proposed method to the problem of range-based navigation with single beacon. With the former, we analyze three scenarios where: i) the target is fixed, ii) the target's velocity vector is unknown but constant, and iii) the target's acceleration vector is unknown but constant. The key contributions of this chapter are the following.

(i) We propose two approaches to the range-based target localization and navigation problems. The first method uses simple mathematical tools to characterize the linear independence of a set of functions of time whereas, the second method borrows the idea from the work in [Batista et al., 2011, Indiveri et al., 2016], which derives observability condition of a nonlinear system from an augmented linear-time varying system. We show that for the single beacon navigation problem our method yields conditions equivalent to that in [Batista et al., 2011]. However, the formulation adopted is much simpler.

(ii) For the target localization problem we also derive results for the case where the target is localized with two trackers. We show that with two range measurements the trackers' motion required for observability of the target is less demanding than in the case of a single tracker.

(iii) Finally, we show how the observability conditions derived lend themselves to intuitive geometric interpretations that yield valuable guidelines to plan the trackers' motions.

4.2 Tools for observability analysis

In this section, we recall some useful definitions and results that will be used for observability analysis in the subsequent sections. Consider the general dynamical system described by

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \\ \mathbf{y}(t) &= \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t)), \end{aligned} \tag{4.1}$$

where $\mathbf{x} \in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^p, \mathbf{y} \in \mathbb{R}^m$ are the state, input, and output of the system, respectively, and $\mathbf{f}(\cdot) \in \mathbb{R}^n, \mathbf{h}(\cdot) \in \mathbb{R}^m$ are (nonlinear) functions.
We use the following definition of observability for the system (4.1) that is an extension of Definition 5-5 in [Chen, 1984] as follows.

Definition 4.1 (Observability)**.** *The dynamical system described by* (4.1) *is said to be (completely state) observable at t_0 if there exists a finite time $t_f > t_0$ such that for any initial state $\mathbf{x}(t_0)$, the knowledge of the input $\mathbf{u}(t_0, t_f)$ and the output $\mathbf{y}(t_0, t_f)$ suffices to determine the initial state $\mathbf{x}(t_0)$. Otherwise, the system is said to be unobservable at t_0.*

In this chapter, in order to solve the observability problem we resort to a simple tool from linear independence of functions over a compact interval of time and is, stated as follows.

Definition 4.2 (Linearly dependent/independent functions - Section 5.2 [Chen, 1984])**.** *Let $\mathbf{f}_i(t)$, for $i = 1, 2, ..., n$ be $p \times 1$ vector real-valued continuous functions of t. Then, $\mathbf{f}_1, ..., \mathbf{f}_n$ are said to be linearly dependent (LI) on time interval $[t_0, t_f]$ if and only if there exist numbers $\alpha_1, ..., \alpha_n$ not all zero such that*

$$\alpha_1 \mathbf{f}_1(t) + \alpha_2 \mathbf{f}_2(t) + \cdots + \alpha_n \mathbf{f}_n(t) = \mathbf{0}.$$

for all $t \in [t_0, t_f]$. Otherwise, they are said to be linearly independent.

Lemma 4.1. *Let $\mathbf{f}_i(t)$, for $i = 1, 2, ..., n$ be $p \times 1$ vector real-valued continuous functions of t defined on the interval $[t_0, t_f]$. Let F be the $p \times n$ matrix with $\mathbf{f}_i$ as its i^{th} columns. Define*

$$W(t_0, t_f) \triangleq \int_{t_0}^{t_f} F^{\mathrm{T}}(t) F(t) \mathrm{d}t.$$

Then, $W(t_0, t_f)$ is non-singular if and only if the columns of $F(t)$ (i.e. $\mathbf{f}_1, \mathbf{f}_2, ..., \mathbf{f}_n$) are linearly independent on $[t_0, t_f]$.

This lemma is equivalent to Theorem 5-1 in [Chen, 1984].

4.3 Problem formulation

For the sake of simplicity and clarity of the notation, in this section we start by formulating the problem of range-based target localization with single tracker. We also formulate the range-based navigation problem which is closely related to the target localization problem. The target localization problem with two trackers will be formulated in Section 4.5.

4.3.1 Range-based target localization

Consider a tracker that is in charge of localizing an unknown fixed or moving target. Let $\{\mathcal{I}\} = \{x_{\mathcal{I}}, y_{\mathcal{I}}, z_{\mathcal{I}}\}$ denotes an inertial frame. In what follows we described target models adopted.

Targets' model: We consider three practical scenarios for the motion of the target.

Scenario A: *Target is fixed.*

Let $\mathbf{q} = [q_x, q_y, q_z]^{\mathrm{T}} \in \mathbb{R}^3$ be the position vector of the target in the inertial frame $\{\mathcal{I}\}$. The target's model is given by

$$\dot{\mathbf{q}} = \mathbf{0}. \tag{4.2}$$

The target state is defined as $\mathbf{x} \triangleq \mathbf{q} \in \mathbb{R}^3$.

Scenario B: *Target moves with unknown constant velocity vector.*

In this case, we assume that the target's velocity vector changes slowly but is *unknown*. An approximate target model in this situation is given by

$$\begin{aligned} \dot{\mathbf{q}} &= \mathbf{v}, \\ \dot{\mathbf{v}} &= \mathbf{0}. \end{aligned} \tag{4.3}$$

In this scenario, the target's state vector is defined as $\mathbf{x} \triangleq \mathrm{col}(\mathbf{q}, \mathbf{v}) \in \mathbb{R}^6$.

Scenario C: *Target moves with unknown constant acceleration vector.*

We now consider the most challenging case where the target's acceleration vector is *un-*

known but changes slowly. An appropriate target model in this scenario is given by

$$\begin{aligned}\dot{\mathbf{q}} &= \mathbf{v}\\ \dot{\mathbf{v}} &= \mathbf{a}\\ \dot{\mathbf{a}} &= \mathbf{0},\end{aligned}\tag{4.4}$$

where $\mathbf{a} \in \mathbb{R}^3$ denotes the target's acceleration vector. In (4.4), the target's state vector is defined as $\mathbf{x} \triangleq \mathrm{col}(\mathbf{q}, \mathbf{v}, \mathbf{a}) \in \mathbb{R}^9$.

Range measurement model: We assume that the tracker is equipped with a sensor unit capable of measuring its range to the target according to the model

$$r(t) = \|\mathbf{p}(t) - \mathbf{q}(t)\|,\tag{4.5}$$

where $\mathbf{p}(t)$ denote tracker's trajectory. The main objective of this chapter is to characterize the type of tracker's trajectory such that the target is localizable (i.e. observable).

Problem 4.1 (Target localization with single tracker)**.** *Consider the target's model given by* (4.2)–(4.4) *depending on the different scenarios considered, and the range measurement model given by* (4.5)*. Derive conditions for the tracker's trajectory under which the target is localizable, i.e. the target's state* $\mathbf{x}$ *is uniquely determined using only the knowledge of range measurements* $r(t)$ *and the tracker's position* $\mathbf{p}(t)$*.*

4.3.2 Range-based single beacon navigation

The range-based target localization problem formulated above is similar to the range-based single beacon navigation problem. That is, given a fixed or moving beacon with known position, find conditions on the trajectory of a vehicle under which the vehicle's position (and possibly velocity) vectors can be determined using only range measurements from the vehicle to the beacon. Formally, consider a vehicle whose motion can be described the model

$$\begin{aligned}\dot{\mathbf{p}} &= \mathbf{u} + \mathbf{v}_c\\ \dot{\mathbf{v}}_c &= \mathbf{0},\end{aligned}\tag{4.6}$$

where $\mathbf{p} \in \mathbb{R}^3$ is the vehicle's position, $\mathbf{u} \in \mathbb{R}^3$ is the vehicle's velocity vector respect to the fluid, and $\mathbf{v}_c \in \mathbb{R}^3$ is the unknown constant disturbance (e.g. ocean current), all expressed in the inertial frame. Suppose that the vehicle can measure range to the beacon using the range measurement model

$$r(t) = \|\mathbf{p}(t) - \mathbf{s}\|, \tag{4.7}$$

where $\mathbf{s} \in \mathbb{R}^3$ is the position of beacon, assumed to be fixed and known. The range-based navigation problem is stated as follow.

Problem 4.2 (Range-based navigation with single beacon)**.** *Consider the vehicle's model given in* (4.6)*, and the range measurement model given by* (4.7)*. Derive conditions for the vehicle's motion (either on* $\mathbf{u}$ *or* $\mathbf{p}$*) under which the vehicle's position* $\mathbf{p}(t)$ *and the unknown disturbance* $\mathbf{v}_c$ *are uniquely determined from the knowledge of the range measurements* $r(t)$*, the vehicle's velocity* $\mathbf{u}(t)$*, and the position of the beacon* $\mathbf{s}$*.*

Although this problem was addressed in [Batista et al., 2011, Indiveri et al., 2016] we will show that the technique we proposed in this chapter can resolve it in a simpler and more efficient manner.

4.4 Target localization with a single tracker

We first describe the solution for the range-based target localization stated in Problem 4.1. We start by considering a simplest case where the target is fixed.

4.4.1 Target is fixed

It follows from (4.5) that

$$\begin{aligned} r^2(t) &= \|\mathbf{p}(t) - \mathbf{q}\|^2 \\ &\overset{(4.2)}{=} \|\mathbf{p}(t)\|^2 - 2\mathbf{p}^{\mathrm{T}}(t)\mathbf{q} + \|\mathbf{q}\|^2. \end{aligned} \tag{4.8}$$

By defining

$$y(t) \triangleq r^2(t) - \|\mathbf{p}(t)\|^2, \tag{4.9}$$

$$F(t) \triangleq [1 \quad -\mathbf{p}^{\mathrm{T}}(t)], \tag{4.10}$$

and $\mathbf{w} \triangleq [\|\mathbf{q}\|^2, \mathbf{q}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^4$, (4.8) can be rewritten as

$$y(t) = F(t)\mathbf{w}. \tag{4.11}$$

Multiplying both sides of the above equation with $F^{\mathrm{T}}(t)$ then integrating it over $[t_0, t_f]$ yields

$$\underbrace{\int_{t_0}^{t_f} F^{\mathrm{T}}(t)y(t)\mathrm{d}t}_{\mathbf{y}(t_0,t_f)\in\mathbb{R}^4} = \underbrace{\left(\int_{t_0}^{t_f} F^{\mathrm{T}}(t)F(t)\mathrm{d}t\right)}_{W(t_0,t_f)\in\mathbb{R}^{4\times 4}}\mathbf{w}. \tag{4.12}$$

In the above equation, $\mathbf{y}(t_0, t_f)$ is known because $r(t)$ and $\mathbf{p}(t)$ are known for all $t \geq t_0$. On the other hand the matrix $W(t_0, t_f)$ captures the information about the tracker's trajectory that decides observability of the target. To be more precisely, define two maps

$$\begin{aligned} &\mathbf{f} : \mathbb{R}^3 \to \mathbb{R}^4, \quad \mathbf{f}(\mathbf{q}) = \begin{bmatrix} \|\mathbf{q}\|^2 \\ \mathbf{q} \end{bmatrix}, \\ &\mathbf{g} : \mathbb{R}^4 \to \mathbb{R}^4, \quad \mathbf{g}(\mathbf{w}) = W(t_0, t_f)\mathbf{w}. \end{aligned} \tag{4.13}$$

By the definition, $\mathbf{g} \circ \mathbf{f} : \mathbb{R}^3 \to \mathbb{R}^4$ maps the target's position $\mathbf{q}$ to the measurement output $\mathbf{y}(t_0, t_f)$. The observability problem now is equivalent with finding condition on the tracker's trajectory under which this map is injective[1], that is, given the measurement output $\mathbf{y}(t_0, t_f)$, under which condition on $\mathbf{p}(t)$ then $\mathbf{q}$ is uniquely determined from $\mathbf{y}(t_0, t_f)$. We obtain the following result.

Theorem 4.1. *Consider the target localization problem with single tracker where the target is fixed. Then, the target's state $\mathbf{x} \triangleq \mathbf{q}$ is uniquely determined if and only if the columns of $F(t)$, given by* (4.10)*, are LI on the interval $[t_0, t_f]$, where $t_f > t_0$.*

Proof:

i. Sufficient: if the columns of $F(t)$ are LI on $[t_0, t_f]$, (i.e. $W(t_0, t_f)$ is non-singular, due to Lemma 4.1) then the map $\mathbf{g}$ is also injective, hence $\mathbf{g} \circ \mathbf{f}$ is injective (because $\mathbf{f}$ is injective), and therefore $\mathbf{q}$ is uniquely determined from $\mathbf{y}(t_0, t_f)$.

ii. Necessary: we now show that if $\mathbf{q}$ is uniquely determined from $\mathbf{y}(t_0, t_f)$ (i.e. $\mathbf{g} \circ \mathbf{f}$ is injective) then the columns of $F(t)$ must be LI on $[t_0, t_f]$. We prove this by

[1] a map (function) $f : X \to Y$ is called injective (also known as one-to-one) if for all $x_1, x_2 \in X$, $f(x_1) = f(x_2)$ implies that $x_1 = x_2$. It is also easy to verify that if f, g are two injective maps then $g \circ f$ is injective as well.

contradiction.

Suppose that $\mathbf{g} \circ \mathbf{f}$ is injective but the columns of $F(t)$ are not LI on $[t_0, t_f]$. Let $\boldsymbol{\alpha} = [\alpha_1, \boldsymbol{\alpha}_\mathbf{2}^\mathrm{T}]^\mathrm{T} \in \mathbb{R}^4$ be a generic nonzero vector with $\alpha_1 \in \mathbb{R}, \boldsymbol{\alpha}_2 \in \mathbb{R}^3$. Since the columns of $F(t)$ are not LI on $[t_0, t_f]$ then there exist some $\boldsymbol{\alpha} \in \mathbb{R}^4, \boldsymbol{\alpha} \neq \mathbf{0}$ s.t. $F(t)\boldsymbol{\alpha} = \alpha_1 + \boldsymbol{\alpha}_2^\mathrm{T}\mathbf{p}(t) = 0$ for all t. There are two possible cases that satisfy the last equation.

(a) $\alpha_1 = 0$, $\boldsymbol{\alpha}_2 \neq \mathbf{0}$, i.e. $\boldsymbol{\alpha}$ has the form $\boldsymbol{\alpha} = [0, \boldsymbol{\alpha}_2^\mathrm{T}]^\mathrm{T}$. An example of this case is $\mathbf{p}(t) = [t, 2t, 5t]$.

(b) $\alpha_1 \neq 0$ and $\boldsymbol{\alpha}_2 = \alpha_1\boldsymbol{\beta}$ with some $\boldsymbol{\beta} \neq \mathbf{0}$, i.e. $\boldsymbol{\alpha}$ has the form $\boldsymbol{\alpha} = \alpha_1[1, \boldsymbol{\beta}^\mathrm{T}]^\mathrm{T}$. An example of this case is $\mathbf{p}(t) = [t+1, 2t+2, t]$.

With the first case, define $\mathbf{q}_1 = \boldsymbol{\alpha}_2/2$ and $\mathbf{q}_2 = -\boldsymbol{\alpha}_2/2$. Clearly $\mathbf{q}_1 \neq \mathbf{q}_2 \neq \mathbf{0}$, however,

$$\mathbf{g} \circ \mathbf{f}(\mathbf{q}_1) - \mathbf{g} \circ \mathbf{f}(\mathbf{q}_2) = \underbrace{W(t_0, t_f) \begin{bmatrix} 0 \\ \boldsymbol{\alpha}_2 \end{bmatrix}}_{=W(t_0,t_f)\boldsymbol{\alpha}} = \mathbf{0}, \tag{4.14}$$

implying that $\mathbf{g}\circ\mathbf{f}$ is non-injective, which contradicts with the beginning assumption that this map is injective.

With the second case, define $\mathbf{q} = \boldsymbol{\beta}/\|\boldsymbol{\beta}\|^2$. Clearly $\mathbf{q} \neq \mathbf{0}$, however

$$\mathbf{g} \circ \mathbf{f}(\mathbf{q}) = \frac{1}{\alpha_1\|\boldsymbol{\beta}\|^2} W(t_0, t_f)\alpha_1 \begin{bmatrix} 1 \\ \boldsymbol{\beta} \end{bmatrix} = \mathbf{0}, \tag{4.15}$$

implying that the map $\mathbf{g}\circ\mathbf{f}$ is non-injective (because $\mathbf{g}\circ\mathbf{f}(\mathbf{0}) = \mathbf{0}$), which contradicts with the beginning assumption that this map is injective.

So we conclude that if $\mathbf{g} \circ \mathbf{f}$ is injective then the columns in $F(t)$ are LI on $[t_0, t_f]$).
■

We now discuss the geometrical intuition behind the condition given in Theorem 4.1. For this purpose, let us examine the following tracker's trajectories for the 2D cases:

$$\mathbf{p}(t) = [t, 2t]^\mathrm{T}, \quad \mathbf{p}(t) = [t+2, 5t]^\mathrm{T}, \quad \mathbf{p}(t) = [t, \sin(t)]^\mathrm{T}. \tag{4.16}$$

With the first trajectory, the matrix $F(t)$ in (4.10) is given by

$$F(t) = [1 \quad t \quad 2t].$$

It is easy to see that for any $\alpha_3 \neq 0$ and $\alpha_2 = -2\alpha_3$ the equation

$$\alpha_1 + \alpha_2 t + \alpha_3 2t = \alpha_1 + (\alpha_2 + 2\alpha_3)t = 0$$

holds true for all t. Hence, according to Definition 4.2 the columns in $F(t)$ are not LI on every interval in $\mathbb{R}$, which violates the conditions in Theorem 4.1, implying that the target is not localizable with the first trajectory. With a similar procedure, we can verify that the target is not localizable with the second trajectory but observable with the third trajectory in (4.16).

The common geometrical interpretation of the first two trajectories is that as time evolves they draw straight-lines on the 2D plane. In fact, the theorem implies that the target's position can not be determined if the tracker moves along any straight line, since every straight-line type trajectory violates the linear independence condition of the columns in $F(t)$. This is not surprising and can be explained intuitively from a geometrical standpoint. Obviously, if the tracker moves along a straight line, then there exists an reflected image (in 2D) or a set of reflected images (in 3D) of the target about that line such that the ranges from tracker to the target and to its reflected image are the same, thus making it impossible to distinguish the true target and its reflected images, see a trajectory in Fig.4.2 (solid-black) as an illustration for the case of 2D.

In practice, it is common to use an under-actuated vehicle (e.g. Medusa class vehicle [Abreu et al., 2016c]) as the tracker to localize the target. In 2D, the $x_\mathcal{I} - y_\mathcal{I}$ plane for example, assuming that the sideslip angle can be neglected, the motion of the vehicle can be simply described as

$$\dot{p}_x = u_x \triangleq v\cos(\psi), \quad \dot{p}_y = u_y \triangleq v\sin(\psi), \tag{4.17}$$

where v is the linear speed and ψ is the heading of the vehicle. For this type of tracker, the condition in Theorem 4.1 implies that it is sufficient to localize the target if the tracker moves with non zero speed ($v \neq 0$) and changes its heading (ψ) at least one time in the interval (t_0, t_f) (see the red curve in Fig.4.2 as an example of this type of trajectory where $\dot{\psi}(t) \neq 0$ for all $t \in [t_0, t_f]$). This can be extended to 3D analogously, and show that it is sufficient for target localization that the tracker moves with non-zero speed and change at least any two Euler angles at any different times in the interval (t_0, t_f).

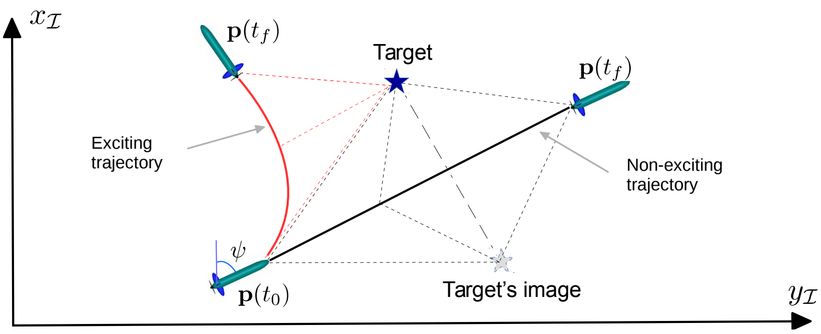


Figure 4.2: Localization of a fixed target with a single tracker. The black trajectory is not "exciting" enough for localizing the target. The target is observable if the tracker performs the red trajectory.

4.4.2 Target moves with unknown constant velocity vector

From the target's model (4.3) and the range measurement model (4.5) we obtain the nonlinear system

$$\begin{aligned} \dot{\mathbf{q}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= \mathbf{0} \\ r(t) &= \|\mathbf{p}(t) - \mathbf{q}(t)\|, \end{aligned} \tag{4.18}$$

where the system's state is the target's state $\mathbf{x} = \text{col}(\mathbf{q}, \mathbf{v}) \in \mathbb{R}^6$ while the tracker's trajectory $\mathbf{p}$ can be viewed as the system's input. Observability of this system, that is, whether or not the target is localizable (observable) depends on the system's input $\mathbf{p}$. In this section we propose two methods to derive conditions on the tracker's trajectory under which the target is observable. The first uses a the methodology proposed in the previous section whereas, the second is along the lines of the work in [Batista et al., 2011,Indiveri et al., 2016], i.e. observability of the nonlinear system (4.18) can be obtained from observability of an augmented LTV system.

Method 1:

It follows from (4.18) that

$$\mathbf{q}(t) = \mathbf{q}_0 + \delta(t)\mathbf{v}, \tag{4.19}$$

where $\mathbf{q}_0 \triangleq \mathbf{q}(t_0)$ and

$$\delta(t) \triangleq t - t_0. \tag{4.20}$$

Expanding the range measurement equation in (4.18) we obtain

$$\begin{aligned} r^2(t) &= \|\mathbf{p}(t) - \mathbf{q}(t)\|^2 \\ &\overset{(4.19)}{=} \|\mathbf{p}(t)\|^2 + \|\mathbf{q}_0\|^2 + 2\delta(t)\mathbf{q}_0^{\mathrm{T}}\mathbf{v} + \delta^2(t)\|\mathbf{v}\|^2 - 2\mathbf{p}^{\mathrm{T}}(t)\mathbf{q}_0 - 2\delta(t)\mathbf{p}^{\mathrm{T}}(t)\mathbf{v}. \end{aligned} \tag{4.21}$$

Define

$$F(t) \triangleq \begin{bmatrix} 1 & 2\delta(t) & \delta^2(t) & -2\mathbf{p}^{\mathrm{T}}(t) & -2\delta(t)\mathbf{p}^{\mathrm{T}}(t) \end{bmatrix} \tag{4.22}$$

and $\mathbf{w} \triangleq [\|\mathbf{q}_0\|^2, \mathbf{q}_0^{\mathrm{T}}\mathbf{v}, \|\mathbf{v}\|^2, \mathbf{q}_0^{\mathrm{T}}, \mathbf{v}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^9$, (4.21) can be rewritten as

$$y(t) = F(t)\mathbf{w}, \tag{4.23}$$

where $y(t)$ is given by (4.9). Multiplying both sides of the above equation with $F^{\mathrm{T}}(t)$ then integrating it over $[t_0, t_f]$ yields

$$\underbrace{\int_{t_0}^{t_f} F^{\mathrm{T}}(t)y(t)\mathrm{d}t}_{\mathbf{y}(t_0,t_f)\in\mathbb{R}^9} = \underbrace{\left(\int_{t_0}^{t_f} F^{\mathrm{T}}(t)F(t)\mathrm{d}t\right)}_{W(t_0,t_f)\in\mathbb{R}^{9\times 9}}\mathbf{w}. \tag{4.24}$$

We now define two maps

$$\begin{aligned} &\mathbf{f} : \mathbb{R}^6 \to \mathbb{R}^9, \quad \mathbf{f}(\mathbf{q}_0, \mathbf{v}) = \mathrm{col}(\|\mathbf{q}_0\|^2, \mathbf{q}_0^{\mathrm{T}}\mathbf{v}, \|\mathbf{v}\|^2, \mathbf{q}_0, \mathbf{v}) \\ &\mathbf{g} : \mathbb{R}^9 \to \mathbb{R}^9, \quad \mathbf{g}(\mathbf{w}) = W(t_0, t_f)\mathbf{w}. \end{aligned} \tag{4.25}$$

By the definition, $\mathbf{g}\circ\mathbf{f} : \mathbb{R}^6 \to \mathbb{R}^9$ maps the initial target's state $(\mathbf{q}_0, \mathbf{v})$ to the measurement output $\mathbf{y}(t_0, t_f)$. The observability problem is equivalent with finding condition on the $W(t_0, t_f)$ under which the map $\mathbf{g} \circ \mathbf{f}$ is injective.

Theorem 4.2. *Consider the nonlinear system* (4.18)*. Then, this system is observable at t_0 i.e. the initial target's state $\mathbf{x}_0 \triangleq \mathrm{col}(\mathbf{q}_0, \mathbf{v})$ is uniquely determined if the columns in $F(t)$, given by* (4.22)*, are LI on the interval $[t_0, t_f]$, where $t_f > t_0$.*

Proof: Clearly, the map $\mathbf{f}$ defined by (4.25) is injective. Also, because the columns of $F(t)$ are LI on $[t_0, t_f]$, (i.e. $W(t_0, t_f)$ is non-singular due to Lemma 4.1) then $\mathbf{g}$ is also injective. As a consequence, $\mathbf{g} \circ \mathbf{f}$ is injective (recall that the composition of two injective maps is an injective map). We conclude that $(\mathbf{q}_0, \mathbf{v})$ is uniquely determined from $\mathbf{y}(t_0, t_f)$, or in other words the nonlinear system is observable at t_0. ■

Remark 4.1. *Note that the condition stated in Theorem 4.2 is sufficient. The necessary condition is an open problem for future work.*

Method 2:

Expanding the range measurement equation in (4.18) we obtain

$$\begin{aligned} r^2(t) &= \|\mathbf{p}(t) - \mathbf{q}(t)\|^2 \\ &\overset{(4.19)}{=} \|\mathbf{p}(t)\|^2 - 2\mathbf{p}^{\mathrm{T}}(t)\mathbf{q}(t) + 2\delta(t)\mathbf{q}_0^{\mathrm{T}}\mathbf{v} + \|\mathbf{q}_0\|^2 + \delta^2(t)\|\mathbf{v}\|^2. \end{aligned} \tag{4.26}$$

Define

$$\mathbf{z} = \mathrm{col}(z_1, z_2, z_3, \mathbf{z}_4, \mathbf{z}_5] \triangleq \mathrm{col}(\|\mathbf{q}_0\|^2, \mathbf{q}_0^{\mathrm{T}}\mathbf{v}, \|\mathbf{v}\|^2, \mathbf{q}, \mathbf{v}) \in \mathbb{R}^9. \tag{4.27}$$

By the definition, it follows from (4.18) that

$$\dot{\mathbf{z}} = \underbrace{\begin{bmatrix} 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times3} & 0_{1\times3} \\ 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times3} & 0_{1\times3} \\ 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times3} & 0_{1\times3} \\ 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & 0_{3\times3} & I_{3\times3} \\ 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & 0_{3\times3} & 0_{3\times3} \end{bmatrix}}_{\triangleq A} \mathbf{z}. \tag{4.28}$$

Furthermore, let

$$C(t) = \begin{bmatrix} 1 & 2\delta(t) & \delta^2(t) & -2\mathbf{p}^{\mathrm{T}}(t) & 0_{1\times3} \end{bmatrix} \in \mathbb{R}^{1\times9}. \tag{4.29}$$

Then, from (4.26), (4.28), and (4.29) we obtain the LTV system, described by

$$\begin{aligned} \dot{\mathbf{z}} &= A\mathbf{z} \\ y(t) &= C(t)\mathbf{z}, \end{aligned} \tag{4.30}$$

where $y(t)$ is given by (4.9). At this point, we have transformed the original system (4.18) with the state $\mathbf{x} \triangleq \mathrm{col}(\mathbf{q}, \mathbf{v}) \in \mathbb{R}^6$ into the LTV system (4.30) with state $\mathbf{z} \in \mathbb{R}^9$. Compared with the original system, the LTV has three additional states, namely, $\|\mathbf{q}_0\|^2, \mathbf{q}_0^{\mathrm{T}}\mathbf{v}$, and $\|\mathbf{v}\|^2$. We now state conditions for observability of the LTV system, from which we draw conclusions for the observability of the original system.

Lemma 4.2. *Consider the LTV system described by* (4.30)*. Then, this LTV system is* *observable at t_0, i.e. $\mathbf{z}(t_0)$ is uniquely determined if and only if the columns of $F(t)$, given by (4.22), are LI on $[t_0, t_f]$, where $t_f > t_0$.*

Proof: In the theory of LTV systems (e.g. Theorem 5-9 in [Chen, 1984]) it is well-known that the system (4.30) is observable on the interval $[t_0, t_f]$ if and only if the Gramian matrix, defined by

$$W(t_0, t_f) = \int_{t_0}^{t_f} e^{A^{\mathrm{T}}(t-t_0)} C^{\mathrm{T}}(t) C(t) e^{A(t-t_0)} \mathrm{d}t \tag{4.31}$$

is non-singular. Note that with matrix A in (4.28), $A^n = 0_{9\times 9}$ for all $n >= 2$ and therefore $e^{A(t-t_0)} = I_{9\times 9} + A\delta(t)$. Thus, it is straightforward to show that

$$C(t)e^{A(t-t_0)} = \underbrace{\begin{bmatrix} 1 & 2\delta(t) & \delta^2(t) & -2\mathbf{p}^{\mathrm{T}}(t) & -2\delta(t)\mathbf{p}^{\mathrm{T}}(t) \end{bmatrix}}_{=F(t), \text{ see } (4.22)}. \tag{4.32}$$

Invoking Lemma 4.1, we conclude that the Grammian matrix given by (4.31) is non-singular if and only if the columns of $F(t)$, given by (4.22), are LI in $[t_0, t_f]$. This concludes the proof. ■

Observability of the original system (4.18) is stated as follow.

Theorem 4.3. *Consider the original nonlinear system (4.18). If the columns of $F(t)$, given by (4.22), are LI on the interval $[t_0, t_f]$, then*

- *the original nonlinear system is observable at t_0, i.e. the initial target's state $\mathbf{x}_0 = \mathrm{col}(\mathbf{q}_0, \mathbf{v})$ is uniquely determined.*
- *a state estimator for the LTV system with global convergence of estimation errors is also a state estimator for the original nonlinear system with global convergence of the estimation errors.*

Proof: we prove this theorem by showing that with the same tracker's trajectories and the range measurement the state of the LTV system matches the state of the original nonlinear system. First, given the LTV system (4.30), we obtain

$$\mathbf{z}(t) = e^{A(t-t_0)}\mathbf{z}(t_0) \overset{A^n=0, \forall n\geq 2}{=} \left(I_{9\times 9} + A\delta(t)\right)\mathbf{z}(t_0). \tag{4.33}$$

Substituting A given by (4.28) and the relation given by (4.27) in the above equation yields

$$\mathbf{z}(t) = \begin{bmatrix} z_1(t_0) \\ z_2(t_0) \\ z_3(t_0) \\ \mathbf{z}_4(t_0) + \delta(t)\mathbf{z}_5(t_0) \\ \mathbf{z}_5(t_0) \end{bmatrix}. \tag{4.34}$$

Thus, the output of the LTV system (4.30) is given by

$$\begin{aligned} y(t) &= C(t)\mathbf{z}(t) \\ &= z_1(t_0) + 2\delta(t)z_2(t_0) + \delta^2(t)z_3(t_0) - 2\mathbf{p}^{\mathrm{T}}(t)\mathbf{z}_4(t_0) - 2\delta(t)\mathbf{p}^{\mathrm{T}}(t)\mathbf{z}_5(t_0). \end{aligned} \tag{4.35}$$

Substituting $y(t)$ given by (4.9) in the above equation yields

$$r(t) = \|\mathbf{p}(t)\|^2 + z_1(t_0) + 2\delta(t)z_2(t_0) + \delta^2(t)z_3(t_0) - 2\mathbf{p}^{\mathrm{T}}(t)\mathbf{z}_4(t_0) - 2\delta(t)\mathbf{p}^{\mathrm{T}}(t)\mathbf{z}_5(t_0). \tag{4.36}$$

Comparing (4.21) - the output of the original nonlinear system with (4.36) yields

$$\begin{aligned} 0 =& (z_1(t_0) - \|\mathbf{q}_0\|^2) + 2\delta(t)(z_2(t_0) - \mathbf{q}_0^{\mathrm{T}}\mathbf{v}) + \delta^2(t)(z_3(t_0) - \|\mathbf{v}\|^2) \\ &- 2\mathbf{p}^{\mathrm{T}}(t)(\mathbf{z}_4(t_0) - \mathbf{p}_0) - 2\delta(t)\mathbf{p}^{\mathrm{T}}(t)(\mathbf{z}_5(t_0) - \mathbf{v}). \end{aligned} \tag{4.37}$$

Since the columns of $F(t)$, given by (4.22), are LI on $[t_0, t_f]$ the relation in (4.37) holds for all $t \in [t_0, t_f]$ if and only if

$$z_1(t_0) = \|\mathbf{q}_0\|^2, \quad z_2(t_0) = \mathbf{q}_0^{\mathrm{T}}\mathbf{v}, \quad z_3(t_0) = \|\mathbf{v}\|^2, \quad \mathbf{z}_4(t_0) = \mathbf{q}_0, \quad \mathbf{z}_5(t_0) = \mathbf{v}. \tag{4.38}$$

This implies that the state of the LTV system matches with the state of the original system. This concludes the proof. ■

Remark 4.2. *Although the conditions on the tracker's trajectory obtained by the two methods are identical the first method is much simpler. However, the second method offers an alternative option to design a filter to estimate the target's state which will be clarified in next remark.*

Remark 4.3. *There are two ways to estimate the target's state. The first is to design an Extended Kalman Filter (EKF) based on the nonlinear model* (4.18). *The second, thanks* *to the second result in the Theorem 4.3, is to design a Kalman Filter (KF) based on the LTV model* (4.30)*. The main advantage of the former over the latter is that it runs upon a lower dimension model. Compared with the first EKF, the KF evolves in a higher dimensional state space* ($\mathbb{R}^9$ *vs* $\mathbb{R}^6$) *but theoretically, it can guarantee global convergence of the estimation errors, provided that the LTV system satisfies the conditions of uniform complete observability [Batista et al., 2011].*

We now discuss several types of trajectory that satisfy the condition given in Theorem 4.3 (recall that it is equivalent with condition in Theorem 4.2). In 2D, it can be checked (using either the definition of linear independence of functions or their Wronskian (Theorem 5-2 in [Chen, 1984])) that every "cycloid-type" trajectory for the tracker given in the form $\mathbf{p}(t) = [p_x(t), p_y(t)]^{\mathrm{T}} = [r_x \sin(\omega t) + c_x t, r_y \cos(\omega t) + c_y t]^{\mathrm{T}}$ with $r_x, r_y, \omega \neq 0$ satisfies the condition in the theorem. See Fig.4.3 as a graphical presentation of several trajectories given by the above formula. Similarly, in 3D, every "helix-type" trajectories of the form $\mathbf{p}(t) = [r_x \sin(\omega_1 t) + c_x t, r_y \cos(\omega_1 t) + c_y t, r_z \sin(\omega_2 t) + c_z t]^{\mathrm{T}}$ with $\omega_1 \neq \omega_2 \neq 0$ satisfies the condition in the theorem. However, "pure helix" trajectories (obtained with $r_z = 0$) do not satisfy the condition. Thus, we can not conclude whether or not the target's states (both the position and velocity vector) are observable since the condition given is sufficient. From a system identification point of view, this can be explained that the full state of the target may not be "completely" observable. That is, if $c_2 = 0$ or $\omega_2 = 0$ then the relative velocity vector between the tracker and the target is given by $\tilde{\mathbf{u}}(t) = \mathbf{u}(t) - \mathbf{v}(t) = [\star, \star, c_1 - v_z]$, where v_z is the third component of $\mathbf{v}$. Since v_z is constant hence the third component of the relative vector $\tilde{\mathbf{u}}$ is constant and, as a consequence, this information may not be "sufficient" to identify the third component of the target's velocity vector $\mathbf{v}$. Therefore, the full target's state might not be completely observable with "pure helix" trajectories.

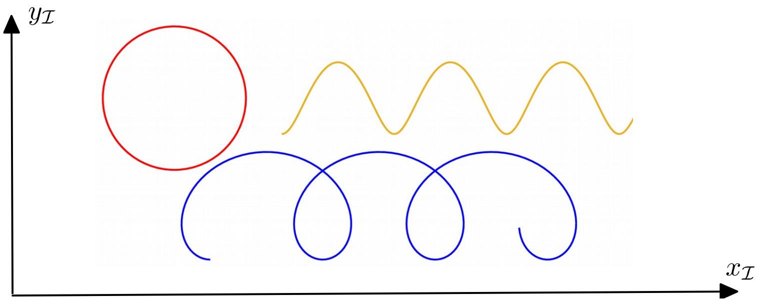


Figure 4.3: Illustration of several "cycloid-type" trajectories.

4.4.3 Target moves with unknown constant acceleration vector

From the target's model (4.4) and the range measurement model (4.5) we obtain the nonlinear system

$$\begin{aligned}\dot{\mathbf{q}} &= \mathbf{v}\\ \dot{\mathbf{v}} &= \mathbf{a}\\ \dot{\mathbf{a}} &= \mathbf{0}\\ r(t) &= \|\mathbf{p}(t)-\mathbf{q}(t)\|,\end{aligned} \tag{4.39}$$

where the system's state is the target's state $\mathbf{x} \triangleq \mathrm{col}(\mathbf{q},\mathbf{v},\mathbf{a}) \in \mathbb{R}^9$ while the tracker's trajectory $\mathbf{p}$ can be viewed as the system's input. In this case, the observability stated in **Problem 4.1** is equivalent to finding conditions for tracker's trajectory under which system (4.39) is observable, i.e. the target is localizable.
Similar to the previous section, we propose two methods to solve this problem.

Method 1:

From (4.39) we obtain

$$\mathbf{q}(t) = \mathbf{q}_0 + \delta(t)\mathbf{v}_0 + 0.5\delta^2(t)\mathbf{a}, \tag{4.40}$$

where $\mathbf{q}_0 \triangleq \mathbf{q}(t_0)$, $\mathbf{v}_0 \triangleq \mathbf{v}(t_0)$ and $\delta(t)$ is given by (4.20). Substituting (4.40) in the range measurement equation (4.39) then extending it yields

$$\begin{aligned} r^2(t) =& \|\mathbf{p}(t)-\mathbf{q}(t)\|^2\\ \overset{(4.40)}{=}& \|\mathbf{p}(t)\|^2 + \|\mathbf{q}_0\|^2 + 2\delta(t)\mathbf{q}_0^{\mathrm{T}}\mathbf{v}_0 + \delta^2(t)\left(\mathbf{q}_0^{\mathrm{T}}\mathbf{a} + \|\mathbf{v}_0\|^2\right) + \delta^3(t)\mathbf{v}_0^{\mathrm{T}}\mathbf{a} + 0.25\delta^4(t)\|\mathbf{a}\|^2\\ &- 2\mathbf{p}^{\mathrm{T}}(t)\mathbf{q}_0 - 2\delta(t)\mathbf{p}^{\mathrm{T}}(t)\mathbf{v}_0 - \delta^2(t)\mathbf{p}^{\mathrm{T}}(t)\mathbf{a}.\end{aligned} \tag{4.41}$$

Define

$$F(t) \triangleq \begin{bmatrix}1 & 2\delta(t) & \delta^2(t) & \delta^3(t) & 0.25\delta^4(t) & -2\mathbf{p}^{\mathrm{T}}(t) & -2\delta(t)\mathbf{p}^{\mathrm{T}}(t) & -\delta^2(t)\mathbf{p}^{\mathrm{T}}(t)\end{bmatrix} \tag{4.42}$$

and $\mathbf{w} \triangleq \mathrm{col}\left(\|\mathbf{q}_0\|^2, \mathbf{q}_0^{\mathrm{T}}\mathbf{v}_0, \mathbf{q}_0^{\mathrm{T}}\mathbf{a} + \|\mathbf{v}_0\|^2, \mathbf{v}_0^{\mathrm{T}}\mathbf{a}, \|\mathbf{a}\|^2, \mathbf{q}_0, \mathbf{v}_0, \mathbf{a}\right) \in \mathbb{R}^{14}$, (4.41) can be rewritten as

$$y(t) = F(t)\mathbf{w}, \tag{4.43}$$

where $y(t)$ is given by (4.9). Proceed similarly to Method 1 in the previous section, we obtain the following result.

Theorem 4.4. *Consider the nonlinear system* (4.39)*. Then, the system is observable at t_0 i.e. the initial target's state $\mathbf{x}_0 \triangleq \mathrm{col}(\mathbf{q}_0, \mathbf{v}_0, \mathbf{a})$ is uniquely determined if the columns in $F(t)$, given by* (4.42)*, are LI on the interval $[t_0, t_f]$, where $t_f > t_0$.*

Proof of this theorem can be done similarly to that of Theorem 4.2.

Method 2:

Expanding the range measurement equation in (4.39) we obtain

$$\begin{aligned} r^2(t) =& \|\mathbf{p}(t) - \mathbf{q}(t)\|^2 \\ \overset{(4.40)}{=}& \|\mathbf{p}(t)\|^2 + \|\mathbf{q}_0\|^2 + 2\delta(t)\mathbf{q}_0^{\mathrm{T}}\mathbf{v}_0 + \delta^2(t)\left(\mathbf{q}_0^{\mathrm{T}}\mathbf{a} + \|\mathbf{v}_0\|^2\right) + \delta^3(t)\mathbf{v}_0^{\mathrm{T}}\mathbf{a} \\ & + 0.25\delta^4(t)\|\mathbf{a}\|^2 - 2\mathbf{p}^{\mathrm{T}}(t)\mathbf{q}(t) \end{aligned} \tag{4.44}$$

Define

$$\mathbf{z} = \begin{bmatrix} z_1, \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ \mathbf{z}_6 \\ \mathbf{z}_7 \\ \mathbf{z}_8 \end{bmatrix} \triangleq \begin{bmatrix} \|\mathbf{q}_0\|^2 \\ \mathbf{q}_0^{\mathrm{T}}\mathbf{v}_0 \\ \mathbf{q}_0^{\mathrm{T}}\mathbf{a} + \|\mathbf{v}_0\|^2 \\ \mathbf{v}_0^{\mathrm{T}}\mathbf{a} \\ \|\mathbf{a}\|^2 \\ \mathbf{q} \\ \mathbf{v} \\ \mathbf{a} \end{bmatrix} \in \mathbb{R}^{14}. \tag{4.45}$$

Then, it follows from (4.39) that

$$\dot{\mathbf{z}} = \underbrace{\begin{bmatrix} 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times3} & 0_{1\times3} & 0_{1\times3} \\ 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times3} & 0_{1\times3} & 0_{1\times3} \\ 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times3} & 0_{1\times3} & 0_{1\times3} \\ 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times3} & 0_{1\times3} & 0_{1\times3} \\ 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times3} & 0_{1\times3} & 0_{1\times3} \\ 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & 0_{3\times3} & I_{3\times3} & 0_{3\times3} \\ 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & 0_{3\times3} & 0_{3\times3} & I_{3\times3} \\ 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \end{bmatrix}}_{\triangleq A} \mathbf{z}. \tag{4.46}$$

Furthermore, let

$$C(t) = \begin{bmatrix} 1 & 2\delta(t) & \delta^2(t) & \delta^3(t) & 0.25\delta^4(t) & -2\mathbf{p}^{\mathrm{T}}(t) & 0_{1\times 3} & 0_{1\times 3} \end{bmatrix} \in \mathbb{R}^{1\times 14}. \tag{4.47}$$

Then, from (4.45), (4.46), and (4.47) we obtain the LTV system, described by

$$\begin{aligned} \dot{\mathbf{z}} &= A\mathbf{z} \\ y(t) &= C(t)\mathbf{z}, \end{aligned} \tag{4.48}$$

where $y(t)$ is given by (4.9). At this point, we have transformed the original nonlinear system (4.39) with the state $\mathbf{x} = \mathrm{col}(\mathbf{q}, \mathbf{v}, \mathbf{a}) \in \mathbb{R}^9$ into the LTV system (4.48) with state $\mathbf{z} \in \mathbb{R}^{14}$. We now state conditions for observability of the LTV system, from which we draw conclusions for the observability of the original nonlinear system.

Lemma 4.3. *Consider the LTV system described by* (4.48)*. Then, the LTV system is observable at* t_0*, i.e.* $\mathbf{z}(t_0)$ *is uniquely determined if and only if the columns of* $F(t)$*, given by* (4.42)*, are LI on* $[t_0, t_f]$*, where* $t_f > t_0$*.*

Proof: The proof can be done similarly to the proof of Lemma 4.2. ■

Theorem 4.5. *Consider the original nonlinear system* (4.39)*. If the columns of* $F(t)$*, given by* (4.42)*, are LI on the interval* $[t_0, t_f]$*, then*

- *the original system is observable at* t_0*, i.e. the initial target's state* $\mathbf{x}_0 = \mathrm{col}(\mathbf{q}_0, \mathbf{v}_0, \mathbf{a})$ *is uniquely determined.*

- *a state estimator for the LTV system with global convergence of estimation errors is also a state estimator for the original nonlinear system with global convergence of the estimation errors.*

Proof of this theorem can be done similarly to that of Theorem 4.3.

Remark 4.4. *There are two ways to estimate the target's state. The first is to design an EKF based on the nonlinear model* (4.39)*. The second, thanks to the second result in Theorem 4.5, is to design an KF based on the LTV model* (4.48)*.*

4.5 Target localization with two trackers

We now consider the cases when the target is localized by two trackers. For the sake of consistency, we keep using the notation in the previous section with an extra subscript $i; i \in \{1, 2\}$ to index the trackers. Specifically, for each tracker i, $\mathbf{p}_i$ denotes its position vector and r_i is the range measurement from the tracker to the target. We now derive the conditions for the motion of the trackers under which the target's state is observable.

4.5.1 Target is fixed

We start by considering the simplest case where the target is fixed. For each tracker i, from the target's model (4.2) and the range measurement model (4.5) we obtain

$$r_i^2(t) = \|\mathbf{p}_i(t) - \mathbf{q}\|^2 = \|\mathbf{p}_i(t)\|^2 - 2\mathbf{p}_i^{\mathrm{T}}(t)\mathbf{q} + \|\mathbf{q}\|^2 \tag{4.49}$$

Subtracting the two ranges we obtain

$$y(t) = 2(\mathbf{p}_2(t) - \mathbf{p}_1(t))^{\mathrm{T}}\mathbf{q}, \tag{4.50}$$

where

$$y(t) \triangleq r_1^2(t) - r_2^2(t) + \|\mathbf{p}_2(t)\|^2 - \|\mathbf{p}_1(t)\|^2. \tag{4.51}$$

Note that $y(t)$ is known for all t because the ranges and the trackers' trajectories are known.

Theorem 4.6. *Consider the target localization problem stated in* **Problem 4.1***, where the target is fixed and two trackers are used. Then, the target is localizable, i.e. the target's position* $\mathbf{q}$ *is uniquely determined if and only if the columns in matrix* $F(t) \in \mathbb{R}^{1\times 3}$*, given by*

$$F(t) = [\mathbf{p}_2(t) - \mathbf{p}_1(t)]^{\mathrm{T}} \tag{4.52}$$

are LI in the interval $[t_0, t_f]$ *for* $t_f > 0$.

Proof: The proof starts with (4.50) and can be done similarly to the proof of Theorem 4.1 with the help of Lemma 4.1. ■

We now discuss the geometrical intuition behind the condition stated in Theorem 4.6. For the sake of clarity, we consider the case where one of the trackers is stationary. Without loss of generality, fix tracker 2, i.e. $\mathbf{p}_2(t) = \mathbf{c}$ for all $t \geq t_0$, where $\mathbf{c}$ is a constant vector. This makes $F(t) = [\mathbf{c} - \mathbf{p}_1(t)]^{\mathrm{T}}$. Hence, for 2D, the necessary and sufficient condition for the columns of $F(t)$ to be LI on $[t_0, t_f]$ implies that the trajectory of tracker 1 must not move along the line that connects two points $\mathbf{p}_1(t_0)$ and $\mathbf{c}$. This is illustrated in Fig.4.4 where it can be seen that the ranges from the trackers to the target and the target's reflected image via the line are the same, making it impossible to distinguish the true target and its reflected image. However, if tracker 1 does not go along with that line (the red line in the figure for example) then the position of target can be uniquely determined.

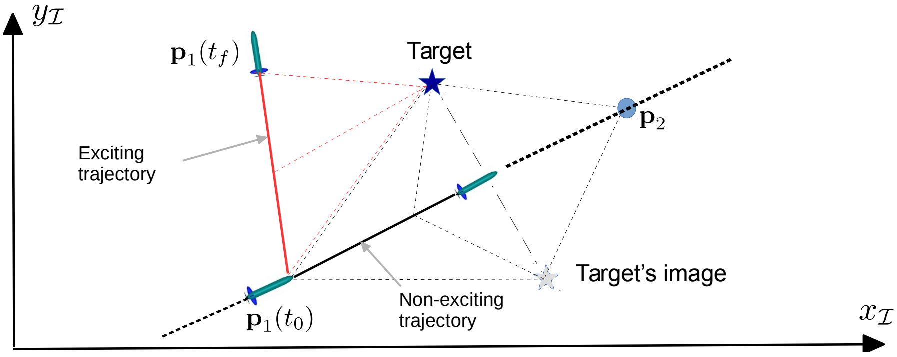


Figure 4.4: Localization of a fixed target using two trackers.

4.5.2 Target moves with unknown constant velocity vector

From the target model (4.3) and the range measurement model (4.5), for each $i \in \{1, 2\}$ we have

$$\begin{aligned} r_i^2(t) =& \|\mathbf{p}_i(t) - \delta(t)\mathbf{v} - \mathbf{q}(t_0)\|^2 \\ =& \|\mathbf{p}_i(t)\|^2 - 2\delta(t)\mathbf{p}_i^{\mathrm{T}}(t)\mathbf{v} - 2\mathbf{p}_i^{\mathrm{T}}(t)\mathbf{q}_0 + \Delta(t), \end{aligned} \tag{4.53}$$

where

$$\Delta(t) \triangleq \|\delta(t)\mathbf{v} + \mathbf{q}_0\|^2. \tag{4.54}$$

Thus, subtracting $r_2^2(t)$ from $r_1^2(t)$ yields

$$\begin{aligned} r_1^2(t) - r_2^2(t) =& \|\mathbf{p}_1(t)\|^2 - \|\mathbf{p}_2(t)\|^2 \\ & -2(\mathbf{p}_1(t) - \mathbf{p}_2(t))^{\mathrm{T}}\mathbf{q}_0 \\ & -2\delta(t)(\mathbf{p}_1(t) - \mathbf{p}_2(t))^{\mathrm{T}}\mathbf{v}. \end{aligned} \tag{4.55}$$

Recall that for this scenario $\mathbf{x}_0 = \operatorname{col}(\mathbf{q}_0, \mathbf{v})$ is the vector of the initial target's state. Equation (4.55) can be rewritten as

$$y(t) = \left[(\mathbf{p}_2(t) - \mathbf{p}_1(t))^{\mathrm{T}} \quad \delta(t)(\mathbf{p}_2(t) - \mathbf{p}_1(t))^{\mathrm{T}}\right] \mathbf{x}_0, \tag{4.56}$$

where $y(t)$ is given by (4.51). We obtain the following result.

Theorem 4.7. *Consider the target localization problem defined in* **Problem 4.1***, where the target moves with unknown velocity vector given by model* (4.4) *and is localized by two trackers. Then, the target is localizable, i.e. the initial target's state* $\mathbf{x}_0 = \operatorname{col}(\mathbf{q}_0^{\mathrm{T}}, \mathbf{v})$ *is uniquely determined if and only if the columns in matrix* $F(t) \in \mathbb{R}^{1\times 6}$*, given by*

$$F(t) = [(\mathbf{p}_2(t) - \mathbf{p}_1(t))^{\mathrm{T}} \quad \delta(t)(\mathbf{p}_2(t) - \mathbf{p}_1(t))] \tag{4.57}$$

are LI in interval $[t_0, t_f]$ *for* $t_f > 0$*.*

Proof: The proof starts with (4.56) and can be done similarly to the proof of Theorem 4.1 with the help of Lemma 4.1. ■

Apparently, the condition for trackers' trajectories given in Theorem 4.7 is less demanding than that in Theorem 4.2, where a single tracker is used to localize the target. Note also that the condition stated in former theorem is necessary and sufficient whereas, that in the latter theorem is sufficient.

4.5.3 Target moves with unknown constant acceleration vector

From the target model (4.4) and the range measurement model (4.5), for each tracker i we obtain

$$\begin{aligned} r_i^2(t) &= \left\|\mathbf{p}_i(t) - \delta(t)\mathbf{v}_0 - 0.5\delta^2(t)\mathbf{a} - \mathbf{q}_0\right\|^2 \\ &= \left\|\mathbf{p}_i(t)\right\|^2 - 2\delta(t)\mathbf{p}_i^{\mathrm{T}}(t)\mathbf{v}_0 - 2\mathbf{p}_i^{\mathrm{T}}(t)\mathbf{q}_0 - \delta^2(t)\mathbf{p}_i^{\mathrm{T}}(t)\mathbf{a} + \Delta(t) \end{aligned} \tag{4.58}$$

for all $i \in \{1, 2\}$ where

$$\Delta(t) \triangleq \left\|\delta(t)\mathbf{v}_0 + 0.5\delta^2(t)\mathbf{a} + \mathbf{q}_0\right\|^2$$

Subtracting $r_2^2(t)$ from $r_1^2(t)$ yields

$$\begin{aligned} r_1^2(t) - r_2^2(t) =& \|\mathbf{p}_1(t)\|^2 - \|\mathbf{p}_2(t)\|^2 \\ & -2(\mathbf{p}_1(t) - \mathbf{p}_2(t))^{\mathrm{T}}\mathbf{q}_0 \\ & -2\delta(t)(\mathbf{p}_1(t) - \mathbf{p}_2(t))^{\mathrm{T}}\mathbf{v}_0 \\ & -\delta^2(t)(\mathbf{p}_1(t) - \mathbf{p}_2(t))^{\mathrm{T}}\mathbf{a}. \end{aligned} \tag{4.59}$$

Recall that $\mathbf{x}_0 = \mathrm{col}(\mathbf{q}_0, \mathbf{v}_0, \mathbf{a})$ is the vector of the initial target's state.

$$y(t) = \left[(\mathbf{p}_2(t) - \mathbf{p}_1(t))^{\mathrm{T}} \quad \delta(t)(\mathbf{p}_2(t) - \mathbf{p}_1(t))^{\mathrm{T}} \quad \delta^2(t)(\mathbf{p}_2(t) - \mathbf{p}_1(t))^{\mathrm{T}}\right] \mathbf{x}_0. \tag{4.60}$$

where $y(t)$ is given by (4.51). We obtain the following result.

Theorem 4.8. *Consider the target localization problem stated in* **Problem 4.1**, *where the target moves with unknown constant acceleration vector given by model* (4.4) *and is localized by two trackers. Then, the target is localizable, i.e. the initial target's state* $\mathbf{x}_0 = \mathrm{col}(\mathbf{q}_0, \mathbf{v}_0, \mathbf{a})$ *is uniquely determined if and only if the columns in matrix* $F(t) \in \mathbb{R}^{1\times 9}$, *given by*

$$F(t) = [(\mathbf{p}_2(t) - \mathbf{p}_1(t))^{\mathrm{T}} \quad \delta(t)(\mathbf{p}_2(t) - \mathbf{p}_1(t)) \quad \delta^2(t)(\mathbf{p}_2(t) - \mathbf{p}_1(t))] \tag{4.61}$$

are LI in interval $[t_0, t_f]$ *for* $t_f > 0$.

Proof: The proof starts with (4.60) and can be done similarly to the proof of Theorem 4.1 with the help of Lemma 4.1. ■

Obviously, the condition for trackers' trajectories given in Theorem 4.8 is less demanding than that in Theorem 4.4, where a single tracker is used to localize the target.

4.6 Range-based navigation with single beacon

In this section we solve the problem of range-based navigation stated in **Problem** 4.2. Let

$$\mathbf{r} = \mathbf{p} - \mathbf{s}. \tag{4.62}$$

Because $\mathbf{s}$ is known, $\mathbf{p}$ is uniquely determined if $\mathbf{r}$ is uniquely determined. From (4.6), (4.7), and (4.62) we obtain the nonlinear system

$$\begin{aligned} \dot{\mathbf{r}} &= \mathbf{u} + \mathbf{v}_c \\ \dot{\mathbf{v}}_c &= \mathbf{0}, \\ r &= \|\mathbf{r}\| \end{aligned} \tag{4.63}$$

We now derive condition on the vehicle input $\mathbf{u}$ such that the system (4.63) is observable. This can be done using either one of the two methods proposed in the previous sections that are presented next.

4.6.1 Method 1:

From (4.62) and the first two equations of (4.63) we obtain the relation

$$\mathbf{r}(t) = \mathbf{r}_0 + \boldsymbol{\lambda}(t) + \delta(t)\mathbf{v}_c, \tag{4.64}$$

where $\mathbf{r}_0 \triangleq \mathbf{r}(t_0) = \mathbf{p}(t_0) - \mathbf{s}$ and

$$\boldsymbol{\lambda}(t) \triangleq \int_{t_0}^{t} \mathbf{u}(\tau)\mathrm{d}\tau. \tag{4.65}$$

Equation (4.64) implies that

$$r^2(t) = \|\mathbf{r}_0\|^2 + \|\boldsymbol{\lambda}(t)\|^2 + \delta^2(t)\|\mathbf{v}_c\|^2 + 2\boldsymbol{\lambda}^{\mathrm{T}}(t)\mathbf{r}_0 + 2\delta(t)\boldsymbol{\lambda}^{\mathrm{T}}(t)\mathbf{v}_c + 2\delta(t)\mathbf{r}_0^{\mathrm{T}}\mathbf{v}_c. \tag{4.66}$$

Define

$$F(t) \triangleq \begin{bmatrix} 1 & 2\delta(t) & \delta^2(t) & 2\boldsymbol{\lambda}^{\mathrm{T}}(t) & 2\delta(t)\boldsymbol{\lambda}^{\mathrm{T}}(t) \end{bmatrix}, \tag{4.67}$$

$$y(t) = r^2(t) - \|\boldsymbol{\lambda}(t)\|^2, \tag{4.68}$$

and $\mathbf{w} \triangleq \mathrm{col}(\|\mathbf{r}_0\|^2, \mathbf{r}_0^{\mathrm{T}}\mathbf{v}_c, \|\mathbf{v}_c\|^2, \mathbf{r}_0, \mathbf{v}_c) \in \mathbb{R}^9$, (4.66) can be rewritten as

$$y(t) = F(t)\mathbf{w}. \tag{4.69}$$

Note that $y(t)$, given by (4.68), is known because $r(t)$ and $\boldsymbol{\lambda}(t)$ are known. Multiplying both sides of the above equation with $F^{\mathrm{T}}(t)$ then integrating it over $[t_0, t_f]$ yields

$$\underbrace{\int_{t_0}^{t_f} F^{\mathrm{T}}(t)y(t)\mathrm{d}t}_{\mathbf{y}(t_0,t_f)\in\mathbb{R}^9} = \underbrace{\left(\int_{t_0}^{t_f} F^{\mathrm{T}}(t)F(t)\mathrm{d}t\right)}_{W(t_0,t_f)\in\mathbb{R}^{9\times 9}} \mathbf{w}. \tag{4.70}$$

Define two maps

$$\begin{aligned} \mathbf{f} : \mathbb{R}^6 \to \mathbb{R}^9, &\quad \mathbf{f}(\mathbf{r}_0, \mathbf{v}_c) = \mathrm{col}(\|\mathbf{r}_0\|^2, \mathbf{r}_0^{\mathrm{T}}\mathbf{v}_{\mathrm{c}}, \|\mathbf{v}_{\mathrm{c}}\|^2, \mathbf{r}_0, \mathbf{v}_{\mathrm{c}}) \\ \mathbf{g} : \mathbb{R}^9 \to \mathbb{R}^9, &\quad \mathbf{g}(\mathbf{w}) = W(t_0, t_f)\mathbf{w}. \end{aligned} \tag{4.71}$$

By the definition, $\mathbf{g} \circ \mathbf{f} : \mathbb{R}^6 \to \mathbb{R}^9$ maps the initial system's state $(\mathbf{r}_0, \mathbf{v}_c)$ to the measurement output $\mathbf{y}(t_0, t_f)$. The observability problem now is equivalent with finding condition on the vehicle's velocity vector $\mathbf{u}$ under which this map is injective.

Theorem 4.9. *Consider the nonlinear system* (4.63)*. Then, the system is observable at* t_0 *i.e. the initial target's state* $\mathbf{x}_0 \triangleq \mathrm{col}(\mathbf{p}_0, \mathbf{v}_c)$ *is uniquely determined if the columns in* $F(t)$*, given by* (4.67)*, are LI on the interval* $[t_0, t_f]$*, where* $t_f > t_0$*.*

Proof: clearly, the map $\mathbf{f}$ in (4.71) is injective. If the columns in $F(t)$ are LI then $W(t_0, t_f)$ is non-singular (due to Lemma 4.1), thus the map $\mathbf{g}$ is injective as well. Consequently, $\mathbf{g} \circ \mathbf{f}$ is injective, which implies that $(\mathbf{r}_0, \mathbf{v}_c)$ is uniquely determined from $\mathbf{y}(t_0, t_f)$. Recall that $\mathbf{r}_0 = \mathbf{p}_0 - \mathbf{s}$ where $\mathbf{s}$ is known, thus the injectivity of $\mathbf{g} \circ \mathbf{f}$ implies that $(\mathbf{p}_0, \mathbf{v}_c)$ is uniquely determined as well. ■

We shall discuss the condition stated the Theorem 4.9 after deriving an identical condition using the second method presented next.

4.6.2 Method 2:

This method is along the lines of the work in [Batista et al., 2011, Indiveri et al., 2016]. The main idea of this approach can be summarized by three steps: i) first, build a LTV system from the original nonlinear system (4.63), ii) second, derive condition on the vehicle's input $\mathbf{u}$ for observability of the LTV system, iii) finally, conclude observability of the original nonlinear system from observability of the LTV system. We go along this line but we will show that our LTV system being built has some advantages compared with the LTV systems in [Batista et al., 2011, Indiveri et al., 2016].

First, (4.64) implies that $\|\mathbf{r}(t) - \boldsymbol{\lambda}(t)\|^2 = \|\mathbf{r}_0 + \delta(t)\mathbf{v}_c\|^2$ which is equivalent to

$$r^2(t) + \|\boldsymbol{\lambda}(t)\|^2 = \|\mathbf{r}_0\|^2 + 2\delta(t)\mathbf{r}_0^{\mathrm{T}}\mathbf{v}_c + \delta^2(t)\|\mathbf{v}_c\|^2 + 2\boldsymbol{\lambda}^{\mathrm{T}}(t)\mathbf{r}(t). \tag{4.72}$$

Define

$$\mathbf{z} = \mathrm{col}(z_1, z_2, z_3, \mathbf{z}_4, \mathbf{z}_5] \triangleq \mathrm{col}(\|\mathbf{r}_0\|^2, \mathbf{r}_0^{\mathrm{T}}\mathbf{v}_c, \|\mathbf{v}_c\|^2, \mathbf{r}, \mathbf{v}_c) \in \mathbb{R}^9. \tag{4.73}$$

Taking the time derivative of $\mathbf{z}$ with respect to (4.63) and (4.62) yields

$$\dot{\mathbf{z}} = \underbrace{\begin{bmatrix} 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times3} & 0_{1\times3} \\ 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times3} & 0_{1\times3} \\ 0_{1\times1} & 0_{1\times1} & 0_{1\times1} & 0_{1\times3} & 0_{1\times3} \\ 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & I_{3\times3} & 0_{3\times3} \\ 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & 0_{3\times3} & 0_{3\times3} \end{bmatrix}}_{\triangleq A} \mathbf{z} + \underbrace{\begin{bmatrix} 0_{1\times3} \\ 0_{1\times3} \\ 0_{1\times3} \\ I_{3\times3} \\ 0_{3\times3} \end{bmatrix}}_{\triangleq B} \mathbf{u}. \tag{4.74}$$

Furthermore, let

$$C(t) = \begin{bmatrix} 1 & 2\delta(t) & \delta^2(t) & 2\boldsymbol{\lambda}^{\mathrm{T}}(t) & 0_{1\times3} \end{bmatrix} \in \mathbb{R}^{1\times9}. \tag{4.75}$$

and

$$y(t) \triangleq r^2(t) + \|\boldsymbol{\lambda}(t)\|^2. \tag{4.76}$$

As a consequence, from (4.72)- (4.76) we obtain a LTV system described by

$$\begin{aligned} \dot{\mathbf{z}} &= A\mathbf{z} + B\mathbf{u} \\ y(t) &= C(t)\mathbf{z}. \end{aligned} \tag{4.77}$$

At this point, we have transformed the original nonlinear system (4.63) with the state $\mathbf{x} \triangleq \mathrm{col}(\mathbf{r}, \mathbf{v}_c) \in \mathbb{R}^6$ into the LTV system (4.77) with state $\mathbf{z} \in \mathbb{R}^9$. Compared with the original system, the LTV has three additional states, namely, $\|\mathbf{r}_0\|^2, \mathbf{r}_0^{\mathrm{T}}\mathbf{v}_c$ and $\|\mathbf{v}_c\|^2$. We now state conditions for observability of the LTV system, from which we draw conclusions for the observability of the original nonlinear system.

Lemma 4.4. *Consider the LTV system described by* (4.77)*. Then, the LTV system is observable at t_0, i.e. $\mathbf{z}(t_0)$ is uniquely determined if and only if the columns of $F(t)$, given by* (4.67)*, are LI on $[t_0, t_f]$, where $t_f > t_0$.*

Proof: In the theory of LTV systems (e.g. Theorem 5-9 in [Chen, 1984]) it is well-known that the LTV system (4.77) is observable at t_0 if and only if the Gramian matrix, defined by

$$W(t_0, t_f) = \int_{t_0}^{t_f} e^{A^{\mathrm{T}}(t-t_0)} C^{\mathrm{T}}(t) C(t) e^{A(t-t_0)} \mathrm{d}t \tag{4.78}$$

is non-singular. Note that with matrix A in (4.74), $A^n = 0_{9\times 9}$ for all $n >= 2$ and therefore $e^{A(t-t_0)} = I_{9\times 9} + A\delta(t)$. Thus, it can be shown that

$$C(t)e^{A(t-t_0)} = \underbrace{\begin{bmatrix} 1 & 2\delta(t) & \delta^2(t) & 2\boldsymbol{\lambda}^{\mathrm{T}}(t) & 2\delta(t)\boldsymbol{\lambda}^{\mathrm{T}}(t) \end{bmatrix}}_{=F(t),\text{see (4.67)}}. \tag{4.79}$$

Invoking Lemma 4.1, we conclude that the Grammian matrix in (4.78) is non-singular if and only if the columns of $F(t)$ are LI in $[t_0, t_f]$. This concludes the proof. ■

Theorem 4.10. *Consider the original nonlinear system* (4.63)*. If the columns of* $F(t)$*, given by* (4.67)*, are LI on the interval* $[t_0, t_f]$*, then*

- *the original nonlinear system is observable at* t_0*, i.e. the initial state* $\mathbf{x}_0 = \mathrm{col}(\mathbf{r}_0, \mathbf{v}_c)$ *is uniquely determined.*

- *a state estimator for the LTV system with global convergence of estimation errors is also a state estimator for the original nonlinear system with global convergence of the estimation errors.*

Proof: the proof of this theorem can be done simply as that of Theorem 4.3. ■

4.6.3 Discussion

As we have shown in the previous section (specifically, in Theorem 4.9 and Theorem 4.10) that the two proposed methods obtain identical sufficient condition for observability of the nonlinear system (4.63). It states that the nonlinear system is observable at any t_0 if the column in $F(t)$, given by (4.67), are LI on $[t_0, t_f]$. This condition is almost equivalent with the sufficient condition stated in Theorem 1 of [Batista et al., 2011] where it states roughly that the nonlinear system (4.63) is observable at t_0 if the columns in

$$H(t) \triangleq \begin{bmatrix} \delta(t) & \delta^2(t) & \boldsymbol{\lambda}^{\mathrm{T}}(t) & \delta(t)\boldsymbol{\lambda}^{\mathrm{T}}(t) \end{bmatrix} \tag{4.80}$$

are LI on the interval $[t_0, t_f]$. In order to reach this conclusion the authors in [Batista et al., 2011] transformed the original nonlinear system (4.63) into a LTV system as the form

$$\begin{aligned} \dot{\mathbf{z}} &= A_z(t, \mathbf{u}(t), y(t))\mathbf{z} + B_z\mathbf{u} \\ y_z(t) &= C_z\mathbf{z}. \end{aligned} \tag{4.81}$$

Then, the observability of the nonlinear system is drawn from observability of the LTV system. Because the matrix A_z in (4.81) depends on the input $\mathbf{u}(t)$ and $y(t)$ this makes the analysis of the Gramian of the LTV system (4.81) much more challenging and involved. In contrast, with our approaches the observability condition can be obtained in a very simple procedure as elaborated in Method 1, without the need of building a LTV system. With our second method the matrix A in our LTV system, described by (4.77), is constant and has many zeros entries, rendering the observability analysis very simple, as shown in the proof of Lemma 4.4. Furthermore, our LTV system does not need to assume $y(t) \neq 0$ to make the matrix A_z well defined as in the work of [Batista et al., 2011]. Observability of system (4.63) was also considered in [Indiveri et al., 2016] and the procedure adopted to build an equivalent LTV system in that reference inspired the formulation in our second method. In this reference, the author built a LTV system with the form

$$\begin{aligned} \dot{\mathbf{z}} &= A\mathbf{z} + B_z\mathbf{u} \\ y(t) &= C(t)\mathbf{z}. \end{aligned} \tag{4.82}$$

where $y(t) = r^2(t) + \|\boldsymbol{\lambda}(t)\|^2 - r^2(t_0)$. The main limitation of this LTV system is that the new measurement output $y(t)$ depends on the measurement of $r(t_0)$ at the initial time t_0, thus the error in measurement of $r(t_0)$ propagates to every measurement of $y(t)$ thereafter. In our formulation, this can be avoided by let $r(t_0)$ as a variable to be estimated. It is also worthy to note that the conditions derived in the same reference are necessary, which is not enough to characterize the type of vehicle's input required for observability of the vehicle's state - the main goal in the context of range-based navigation.

Remark 4.5. *There are two ways to estimate the target's state. The first is to use an EKF for the original nonlinear system* (4.63)*. The second is to use an KF for the LTV system* (4.77)*. The latter can ensure global convergence provided that the LTV system satisfies the conditions of uniform complete observability.*

4.7 Simulation examples

4.7.1 Example 1 - target localization

In this section we present simulation results for the case presented in Section 4.4.2 where a tracker is used localize a target whose velocity is assumed to change slowly so that its motion can be approximated by the target's model (4.3). We assume that the tracker and the target undergo motions in 2D, in the horizontal plane, along trajectories defined by the parameters in Table 4.1. It can be checked that the tracker's trajectory given in the table satisfies the observability conditions in Theorem 4.2. In order to estimate the target's state and for comparison purposes, we run two filters as suggested in Remark 4.3. The first filter, named EKF, was designed based on the nonlinear model (4.18) whereas, the second model named KF-LTV, was constructed based on the LTV model (4.30). The two filters are initialized with the same initial guess for the target's position and velocity, with $\hat{\mathbf{q}}(0) = $ [-30; -60](m) and $\hat{\mathbf{v}}(0) = [0, 0]$(m/s). Random noises are also added to range's and tracker's position measurements. Specially, Gaussian noise with zero mean and standard deviation of 0.5m was added to range measurements, while Gaussian noise with zero mean and covariance of diag$(0.5, 0.5)$m was added to the measurements of the tracker's position. The performance of the two filters is illustrated in Fig.4.5 and Fig.4.6. It can be seen clearly that they estimate relatively well the target's state, i.e. the estimated target's position and velocity given by the filters converge closely to the true target's position and velocity. This implies the trajectory adopted for the tracker not only guarantees observability of target's state but also provides useful range information for target's state

Table 4.1: Simulation setup for Example 1

	Parameters
Target	Trajectory: $\mathbf{q}(t) = [20\sin(0.01t + \pi),\ 0.3t]$ (m) Velocity $\mathbf{v}(t) = [0.2\cos(0.01t + \pi),\ 0.3]$ (m/s)
Tracker	Velocity: $\mathbf{p}(t) = [50\sin(0.1t) + 0.4t + 100,\ \text{-}50\cos(0.1t)]$ m
EKF	$Q = 10e\text{-}6\text{diag}(10, 10, 1, 1)$ $R = 1$
KF-LTV	$Q = 1e\text{-}10\text{diag}(10, 1, 0.1, 1, 1, 10, 10)$ $R = 100$

estimation.

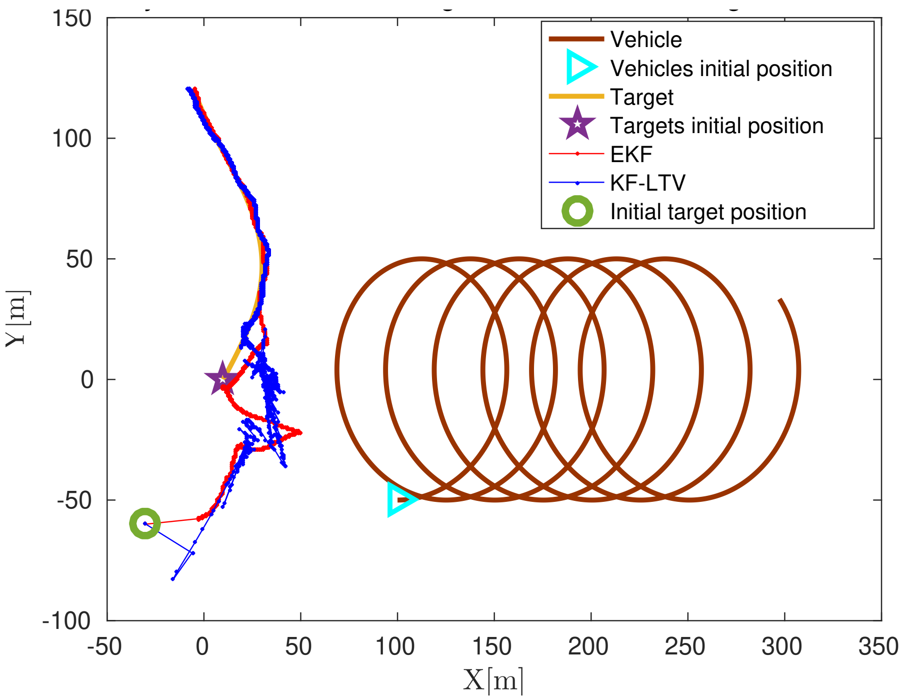


Figure 4.5: Example 1: Trajectories of the trackers, target and target's estimates.

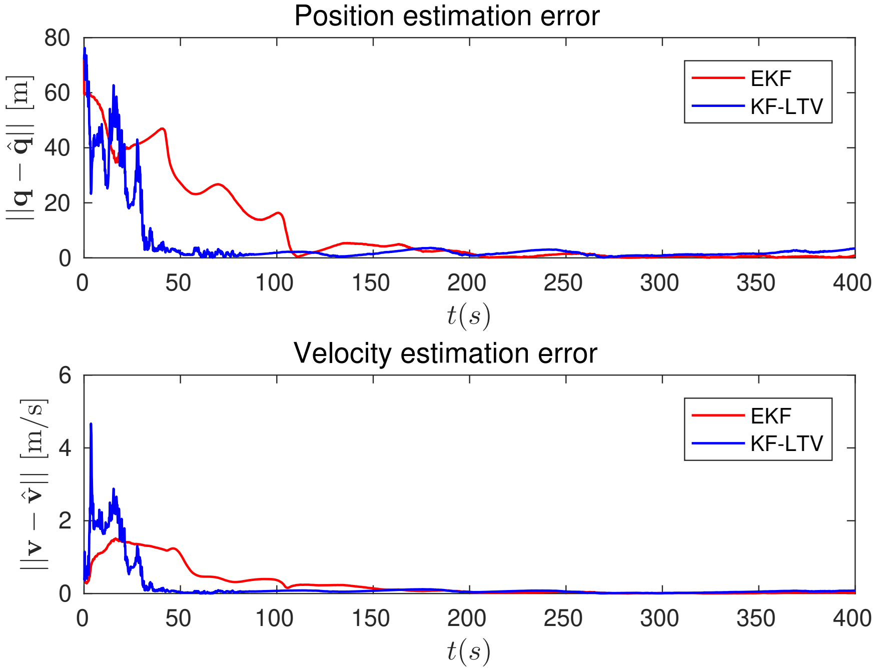


Figure 4.6: Example 1: Estimation errors of target's position (upper) and velocity (lower).

4.7.2 Example 2 - single beacon navigation

In this section, we present simulation results for the single beacon navigation problem. The simulation setup is given in Table 4.2. Note that the vehicle's velocity in the table satisfies observability condition in Theorem 4.9. Thus, the vehicle's position and the ocean current disturbance will be observable.

Table 4.2: Simulation setup for Example 2

	Parameters
Beacon	position $\mathbf{s} = [0, 0, \text{-}50]$ (m)
Vehicle	$\mathbf{p}(0) = [0, 0, 0]$ (m), $\mathbf{u}(t) = [10\cos(0.2t), \text{-}10\sin(0.2t), 2\sin(t) - 0.3]$ m/s
Ocean current	$\mathbf{v}_c = [0.2, 0.3, -0.1]$ m/s
EKF	$Q = 1e\text{-}4\text{diag}(1, 1, 1, 0.1, 0.1, 0.1)$ $R = 1$
KF-LTV	$Q = 1e\text{-}10\text{diag}(1e3, 1e2, 1e\text{-}1, 1e\text{-}2, 1e\text{-}2, 100, 1e\text{-}3, 1e\text{-}3, 100])$ $R = 100$

We assume that the measurements of the vehicle's velocity and ranges from the vehicle to the beacon are disturbed by Gaussian noises with zero mean and covariances of $1e--4I_3$ (m/s) for $\mathbf{u}$ and standard deviations of 0.5(m) for r. The range is taken periodically, at every 0.1 second. We used two filters to estimate the vehicle's position and the ocean current. The first is an EKF based on the nonlinear model (4.63) whereas, the second is a KF based on the LTV model (4.77). The two filters were initialized with $\hat{\mathbf{p}} = [10, \text{-}15, 5]^{\mathrm{T}}$ and $\hat{\mathbf{v}}_c = [0.1, \text{-}0.1, 0.1]^{\mathrm{T}}$.

The simulation results are plotted in Fig.4.7 and Fig.4.8. The figures show that the position and ocean current estimation errors converge to a neighbor of zero, with both filters. This implies that the trajectory of the vehicle is sufficiently exciting for navigation of the vehicle its self and for estimating the ocean current disturbance. Although theoretically, the KF-LTV can guarantee a global convergence of the estimation errors, however we found that it was easier to tune with the EKF. This can be come from the fact that the EKF runs on model (4.63) which does not require to estimate extra coupling variables (i.e. $\|\mathbf{r}_0\|^2, \mathbf{r}_0^{\mathrm{T}}\mathbf{v}_c$ and $\|\mathbf{v}\|_c^2$) as in the KF which builds on the LTV model (4.77)

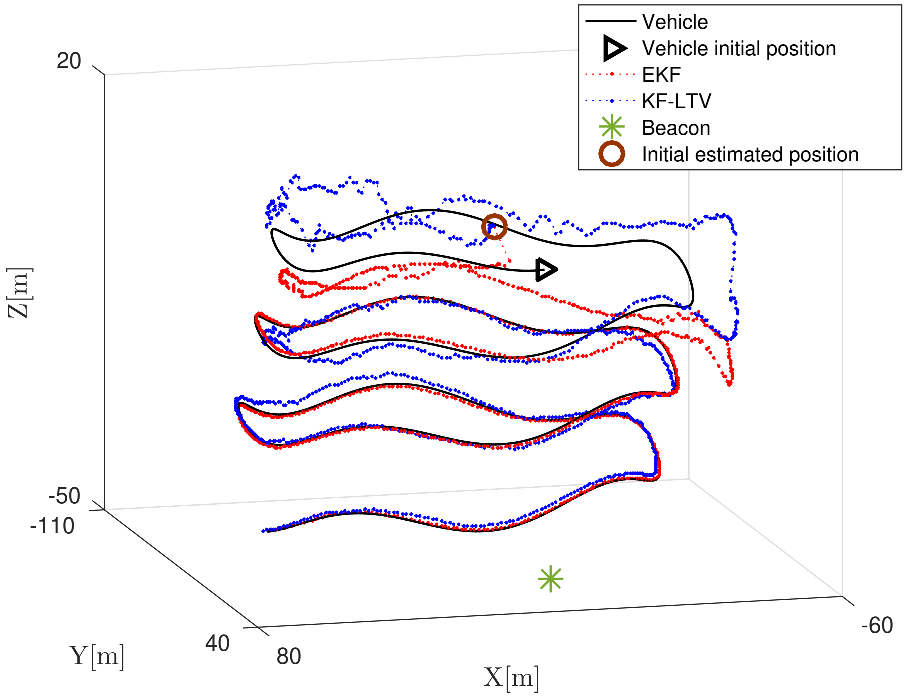


Figure 4.7: Example 2: Trajectories of the vehicle and its estimates.

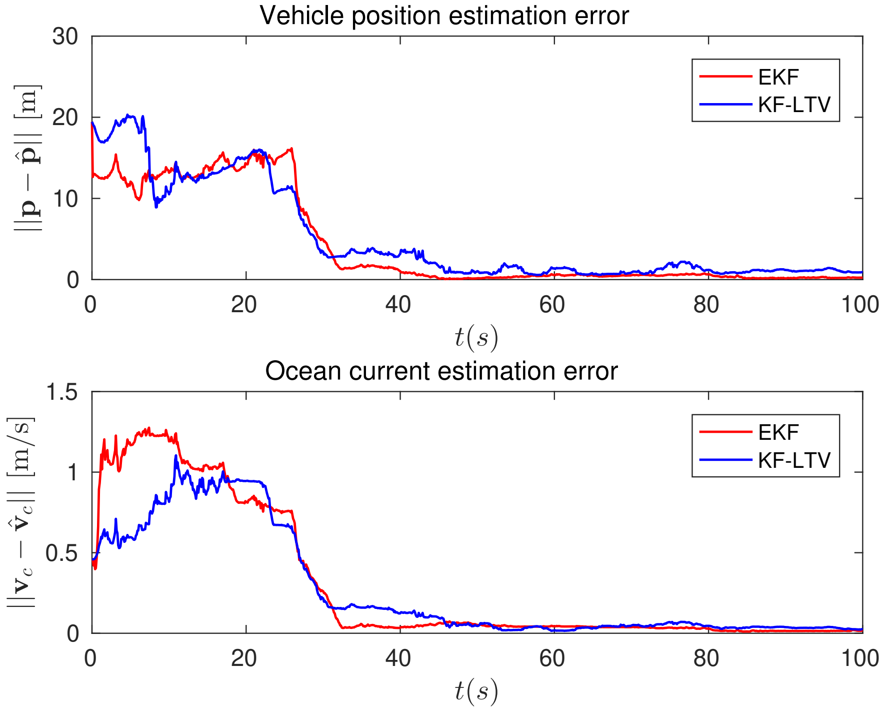


Figure 4.8: Example 2: Estimation errors of vehicle's position (upper) and ocean current (lower).

4.8 Conclusions

We proposed a systematic approach to study the observability problem of range-based navigation and target localization of a moving target using one or two trackers. The approach, which uses simple tools to characterize the linear independence of a set of functions was shown to be very efficient to derive conditions on the motion of tracker(s) to ensure *global observability* of the target state. This is in striking contrast to previous approaches reported in the literature that are more complex in nature. We also gave geometric interpretations of the conditions derived that can be used as guidelines to plan the motions of the trackers. We also show how the proposed method can be extended to the range-based single beacon navigation problem in an analogous manner.

Future work aims to address a challenging question: what type of trajectories that not only ensure global observability of the target but also provide "rich" information for target's state estimation purpose, that is, how characterize trajectories that render the target "strongly observable". Recall that the conditions for observability of the LTI systems obtained in this chapter are also for observability of the target. In the context of LTV systems, degree of observability of a LTV system can be quantified by several measures such as the determinant or condition number of its associated Gramian matrix [Singh and Hahn, 2004]. A higher value of the determinant implies an increased degree of observability whereas, a larger value of the condition number implies an decreased degree of observability. Therefore, a promising research direction to address the above question is to investigate the Gramian matrix of the LTV systems obtained in this chapter, with the main objective of characterizing trajectories that render the LTV system highly observable.

In next chapter we proposed an alternative (quantitative) approach to the range-based target problem, adopting another tool from estimation theory called Fisher Information Matrix.

5

Range-based SLAP using posterior CRLB and model predictive control

Contents

In this chapter, we address the general problem of multiple target localization and pursuit using measurements of the ranges from the targets to a set of autonomous pursuing vehicles, referred to as trackers, the positions of which are all known at all times. We develop a general framework for targets with models exhibiting uncertainty in the initial state, process, and measurement noise. The main objective is to compute optimal motions for the trackers that maximize the range-based information available for target localization and at the same time yield good target pursuit performance. The solution proposed is rooted in an estimation-theoretical setting that involves the computation of an appropriately defined Bayesian Fisher Information Matrix (FIM). The inverse of the latter yields a posterior Cramér-Rao Lower Bound (CRLB) on the covariance of the targets' state estimation errors that can be possibly achieved with any estimator. Using the FIM, sufficient conditions on the trackers' motions are derived for the ideal relative geometry between the trackers and the targets for which the range information acquired is maximal. This allows for an intuitive understanding of the types of ideal tracker trajectories.
To deal with realistic constraints on the trackers' motions and the requirement that the trackers pursue the targets, we then propose a model predictive control (MPC) framework for optimal tracker motion generation with a view to maximizing the predicted range information for target localization while taking explicitly into account the trackers' dynamics, strict constraints on the trackers' states and inputs, and prior knowledge about the targets' states. The efficacy of the MPC is assessed in simulation through the help of representative examples motivated by operational scenarios involving single and multiple targets and trackers.

5.1 Literature review

The problem of target localization and pursuit has received widespread attention due to its importance in a vast number of applications in the areas of marine science, surveillance and reconnaissance, search-and-rescue, and military operations, see [Clark et al., 2013, Zolich et al., 2017, Philippe Lacomme, 2001, Ristic et al., 2004]. In the literature, *localization* usually refers to the problem of finding the location of an unknown stationary target, while tracking refers to the task of estimating the trajectory of a moving target. In this chapter, localization includes both tasks. *Pursuit*, in what follows, has the more control-oriented meaning of ensuring that in the course of their motion the entities in charge of estimating the trajectories of the targets stay in pre-defined neighborhoods of the targets. In this context, target pursuit is a secondary task that aims to enhance the "visibility" of the targets so as to yield better quality of the information available for target localization. In this chapter, this information consists of measurements of the ranges between the trackers and the targets.

The simplest and most classical problem of range-based target localization involves computing the position of a single fixed target using a network of fixed sensors equipped with range-measuring devices. A fundamental problem arising in this context is that of deciding on the number of sensors and how to best place them so that the position of the target can be uniquely estimated with a desired level of accuracy (optimal sensor placement). A solution to this problem can be obtained by adopting an estimation theoretical framework that involves the computation of an appropriately defined Fisher information matrix (FIM). See for example [Martinez and Bullo, 2006, Bishop et al., 2010, Moreno-Salinas et al., 2013] for a discussion of these issues. See also [Moreno-Salinas et al., 2016] for optimal sensor placement solutions in the case of multiple static targets and sensors.

In recent years, there has been growing interest in exploiting the use of single or multiple mobile sensors (called trackers) for target localization, focusing on applications with unmanned aerial vehicles (UAVs) and autonomous marine vehicles (AMVs), see for example [Masmitja et al., 2018, Masmitja et al., 2017, Mandić et al., 2016, Clark et al., 2013, Indiveri et al., 2012, Crasta et al., 2018] and the references therein. In this setup, the trackers carry range measuring devices to acquire successive ranges to the targets of interest and use the range information to estimate the state of each target. Obviously, this approach has many advantages when compared with the traditional method of using a fixed sensor network. First, thanks to the mobility of the trackers, ranges can be acquired from a large number of positions relative to each target, thus potentially providing more information for target localization [Masmitja et al., 2018]. Second, the visibility of the targets can be facilitated by controlling the trackers to be in a given neighborhood of them. Lastly, the use of trackers in the form of AMVs avoids the cumbersome and costly deployment of long baseline (LBL) systems that are classically used in underwater target localization.

This, however leads to the challenging problem of how to plan the motion of the trackers such that their maneuvers ensure that the measured ranges from the trackers to the targets yield, in a well defined mathematical sense, the information needed to estimate the position of the target with a given level of accuracy through the use of an appropriately designed estimator. Technically, the answer to this problem should provide conditions on the motions of trackers that yield target motion observability, given measurements of the ranges between trackers and targets. A classical approach to this problem reported in the literature is to utilize tools from observability analysis, see for example [Pillon et al., 2016, Jauffret et al., 2017, Arrichiello et al., 2013, Indiveri et al., 2012, Hung and Pascoal, 2020b] and references therein. For instance, in [Pillon et al., 2016], the author shows that it is impossible to observe a target moving at a constant velocity vector if the tracker (observer) moves at a constant velocity as well. However, if the tracker's trajectory is composed of at least two straight line segments with different orientations, the target's motion is observable if certain conditions on the bearing between the tracker and the target are satisfied. This analysis is further investigated in [Jauffret et al., 2017] where the trajectory of the tracker is considered to be smooth. In [Arrichiello et al., 2013], the author analyzes the relative motion between the target and tracker using observability rank condition given in the work of [Hermann and Krener, 1977]. Recent work in [Hung and Pascoal, 2020b] provides a set of necessary and sufficient conditions on the motions of one or two trackers under which different type's of target's motions are observable. In general, the above approaches provide conditions on the tracker's trajectories that yield target state observability. However, these are essentially qualitative result that do not provide a good measure of how observable the target motions, a key requisite for adequate tracker motion planning.

A quantitative approach to motion planning can be derived using the celebrated Fisher Information Matrix (FIM), as a means to quantify the amount of information that range measurements carry about the motions of a target. Using this approach, the target localization problem is converted into that of finding conditions on the trackers' trajectories that maximize range-related information available to estimate the target's state. We recall that in an estimation theoretical framework, the inverse of the FIM yields a lower bound (the celebrated Cramér-Rao lower bound, abbreviated CRLB) on the covariance of the target's state estimation error that can possibly be achieved with any practical estimator [Tichavsky et al., 1998, Van Trees, 2004, Ristic et al., 2004]. In the context of range-based target localization, the CRLB is mostly used to access the performance that can be achieved with target state estimators [Ristic et al., 2002, Ristic et al., 2004]. A number of studies that exploit the use of the FIM for tracker motion planning have been published in the literature, see [Masmitja et al., 2018, Masmitja et al., 2019, Crasta et al., 2018] and the reference therein for the case of one tracker one target. More recently, similar tools have been used to tackle the tracker motion planning problem in the case of multiple tracker-multiple target configurations [Crasta et al., 2018]. Notice, however that in [Crasta et al., 2018, Masmitja et al., 2019] the authors resort to the use of the so-called parametric FIM. In this context, the targets evolve in a deterministic manner and the initial conditions of the targets are viewed as parameters to be estimated. As such, the work eschews the far more realistic case where the initial state of the targets (prior information) is described by a random variable that captures the uncertainty in their location, a fact that mandates the use of so-called Bayesian FIM and the computation of posterior CRLBs [Tichavsky et al., 1998], explained later in this chapter. Most approaches to tracker localization, including those reported in [Crasta et al., 2018, Masmitja et al., 2019], also fail to address explicitly the constraints introduced by the limitations in the maneuverability of the trackers. In addition, the guidance law proposed in [Crasta et al., 2018, Masmitja et al., 2019] for the trackers is based on the assumption that the motions of the targets are known in advance, which is unrealistic in the context of target localization. Still, the results in [Crasta et al., 2018] are valuable in terms of understanding at an intuitive level, what kinds of optimal tracker trajectories are suited to selected types of target motion patterns.

Motivated by the above considerations, in this chapter we provide an answer to the question of "how to plan optimal motions" for a set of trackers so as to maximize the range information available to localize and pursue multiple unknown targets by exploiting the properties of an appropriately defined Bayesian FIM. Specifically, the main contributions of this chapter include the following:

(i) We construct a Bayesian FIM using the range measurements from multiple trackers to multiple targets as a means to quantify the range information for the estimation of targets' states. The formalism adopted allows for the study of problems far more appropriate and realistic than those addressed in [Crasta et al., 2018]. First, we consider a more general scenario (later denoted *Scenario B*) where the velocity vectors of the targets are considered to be unknown and must be estimated. Second, the motion of targets are considered to be a dynamical system with given prior information, allowing us to incorporate explicitly the prior knowledge of the targets' states. Third, the FIM in this chapter is the Bayesian FIM [Tichavsky et al., 1998] which is more suitable and appropriate for the problem considered than the parametric FIM studied in [Crasta et al., 2018]. The Bayesian FIM takes into account the dynamics of the trackers and the targets systematically and is computed sequentially, making its computation is simpler, clean and more transparent then the method adopted in [Crasta et al., 2018]. Forth, the depths of the trackers and targets are explicitly taken into account. Lastly, we derive sufficient conditions on the relative geometry between the trackers and the targets trajectory under which the range information computed by the determinant of the FIM is maximum.

(ii) We also propose an MPC-based tracker motion planning, control, and estimation strategy that takes into account the trackers' constraints explicitly, in order to plan optimal motions for the trackers to localize and pursue the targets. In the MPC framework adopted, the control and planing processes are based on the estimated information about the target, thus making the approach more realistic than the guidance law given in [Crasta et al., 2018] where the targets' motions are assumed to be known in advance.

5.2 Notation

We denote by I_n the identity matrix of size n and by $0_{m\times n}$ the zero matrix of size $m \times n$. For two matrices $A, B \in \mathbb{R}^{n\times n}$ the notation $A \succeq B$ implies that $A - B$ is a positive semi-definite matrix. We further let $\det(\cdot)$ and $\mathrm{Tr}(\cdot)$ denote the determinant and the trace of a square matrix, respectively. The symbol $\|\cdot\|$ denotes the Euclidean norm of a vector in $\mathbb{R}^n$. Given $\mathbf{x} \in \mathbb{R}^n$ and a symmetric positive-definite matrix $D \in \mathbb{R}^{n\times n}$, $\|\mathbf{x}\|_D^2 \triangleq \mathbf{x}^{\mathrm{T}} D\mathbf{x}$. Given a set of matrices $W_1, ..., W_p \in \mathbb{R}^{n\times m}$, the symbol $\mathrm{Diag}(W_1, ..., W_p)$ means the block diagonal matrix whose diagonal blocks are the matrices $W_k; k \in \{1, .., p\}$. Given a set of vectors $\mathbf{x}_1, ..., \mathbf{x}_p, \mathrm{col}(\mathbf{x}_i) = [\mathbf{x}_1^{\mathrm{T}}, ..., \mathbf{x}_p^{\mathrm{T}}]^{\mathrm{T}}$.

5.3 Problem formulation

5.3.1 System model

Consider a group of trackers charged with the task of localizing and pursuing a group of moving targets whose motions are partially unknown. As an example, Fig.5.1 illustrates the situation where two autonomous surface vehicles (ASVs) localize and pursue three autonomous underwater vehicles (AUVs) using acoustic range measurements. In general, given $p, q \in \mathbb{N}$, we define $i \in \mathcal{S} \triangleq \{1, ..., p\}$ and $\mathcal{S}_{\mathrm{T}} \triangleq \{1, ..., q\}$, where p and q denote the number of trackers and targets, respectively. In what follows, $\{\mathcal{I}\} = \{x_{\mathcal{I}}, y_{\mathcal{I}}, z_{\mathcal{I}}\}$ denotes an inertial coordinate frame and $\{\mathcal{B}\}^{[i]} = \{x_{\mathcal{B}}^{[i]}, y_{\mathcal{B}}^{[i]}, z_{\mathcal{B}}^{[i]}\}$ denotes a body coordinate frame attached to tracker i; $i \in i \in \mathcal{S}$. We now discuss the tracker and target models considered in this chapter.

Tracker dynamics: For each $i \in i \in \mathcal{S}$, let $z^{[i]}$ be the $z_{\mathcal{I}}$ coordinate of tracker i in frame $\{\mathcal{I}\}$. For the sake of simplicity, when analyzing the geometry of the motion of trackers,

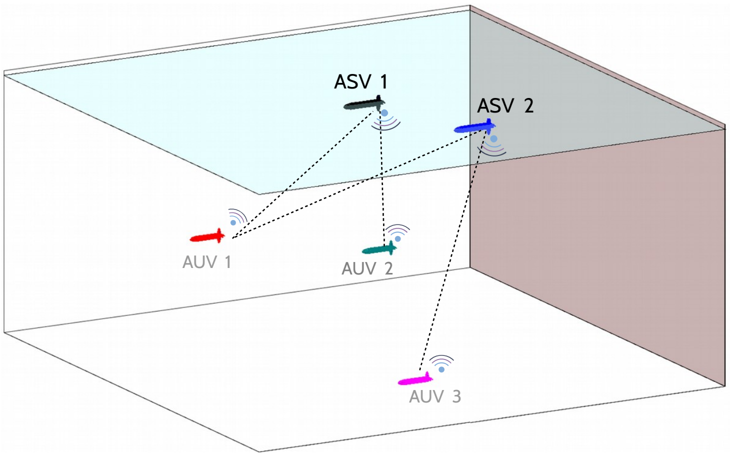


Figure 5.1: An example of two trackers (ASVs) localizing three targets (AUVs) using acoustic range measurements.

we assume that all trackers operate at known constant but possibly different depths, that is, $z^{[i]}(t) = \bar{z}^{[i]}$ for all $t \geq 0$ and $i \in S$. With the above assumptions, the planar motion of tracker i; $i \in i \in \mathcal{S}$, can be described by the simplified kinematic model

$$\dot{\mathbf{p}}^{[i]} = v^{[i]}[\cos(\chi^{[i]}), \sin(\chi^{[i]})]^{\mathrm{T}}, \quad \dot{\chi}^{[i]} = r^{[i]}, \tag{5.1}$$

where $\mathbf{p}^{[i]} = [x^{[i]}, y^{[i]}]^{\mathrm{T}} \in \mathbb{R}^2$ is the horizontal position of tracker i in $\{\mathcal{I}\}$; $v^{[i]} = \left\|\boldsymbol{v}^{[i]}\right\|$ is its total linear speed; $\chi^{[i]}$ is the course angle (see Fig.5.2), and $r^{[i]}$ is the course angle rate. Notice that if the sideslip angle of the tracker is sufficiently small to be ignored, than course angle and course angle rate are equivalent to heading angle and yaw rate, respectively. Our main objective is to find *smooth* linear speed and course rate references for an autopilot to drive the trackers along trajectories that yield rich range-information for target localization. We introduce the constraints

$$\dot{v}^{[i]} = a_v^{[i]}, \quad \dot{r}^{[i]} = a_r^{[i]}, \tag{5.2}$$

where $a_v^{[i]}$ and $a_r^{[i]}$ denote the linear and angular acceleration of the tracker, respectively, which we assume are bounded. In state-space form, we let $\mathbf{z}^{[i]} = [x^{[i]}, y^{[i]}, \psi^{[i]}, v^{[i]}, r^{[i]}]^{\mathrm{T}} \in \mathbb{R}^5$ be the state vector and $\mathbf{v}^{[i]} = [a_v^{[i]}, a_r^{[i]}]^{\mathrm{T}} \in \mathbb{R}^2$ the input vector. The tracker's model (5.1) can now be rewritten as

$$\begin{cases} \dot{\mathbf{z}}^{[i]} = \mathbf{g}(\mathbf{z}^{[i]}, \mathbf{v}^{[i]}), \\ \mathbf{p}^{[i]} = C\mathbf{z}^{[i]} \end{cases} \tag{5.3}$$

where $\mathbf{g}\colon \mathbb{R}^5 \times \mathbb{R}^2 \to \mathbb{R}^5$ and $C \in \mathbb{R}^{2\times 5}$ are given by

$$\mathbf{g}(\mathbf{z}^{[i]}, \mathbf{v}^{[i]}) = \begin{bmatrix} v^{[i]}\cos(\chi^{[i]}) \\ v^{[i]}\sin(\chi^{[i]}) \\ r^{[i]} \\ a_v^{[i]} \\ a_r^{[i]} \end{bmatrix} \quad \text{and} \quad C = \begin{bmatrix} I_2 & 0_{2\times 3} \end{bmatrix}.$$

respectively. Later, for the purpose of system design, (5.3) will be discretized in time, yielding

$$\begin{cases} \mathbf{z}_{k+1}^{[i]} = \mathbf{g}_{\mathrm{d}}(\mathbf{z}_k^{[i]}, \mathbf{v}_k^{[i]}), \\ \mathbf{p}_k^{[i]} = C\mathbf{z}_k^{[i]}, \end{cases} \tag{5.4}$$

where $k \in \mathbb{N}$ indexes discrete time instants and $\mathbf{g}_{\mathrm{d}}(\cdot)$ is a nonlinear function that depends on the chosen discretization procedure. Due to physical limitations, the linear and the rotational speeds and the accelerations of the trackers are bounded. For this reason, we will impose the state and input constraints

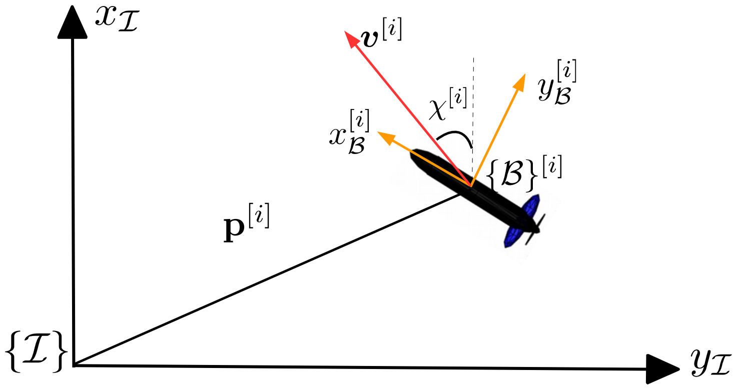


Figure 5.2: Illustration of the planar motion of vehicle i .

$$\mathbf{v}^{[i]} \in \mathcal{V}^{[i]}, \quad \mathbf{z}^{[i]} \in \mathcal{Z}^{[i]}, \tag{5.5}$$

where $\mathcal{V}^{[i]} \subseteq \mathbb{R}^2$ and $\mathcal{Z}^{[i]} \subseteq \mathbb{R}^5$ are input and state constraint sets, respectively, for trackers i; $i \in i \in \mathcal{S}$.

Target model: Let $z_{\mathrm{T}}^{[\alpha]}$ be the $z_{\mathcal{I}}$ coordinate of target α in $\{\mathcal{I}\}$. We assume that all targets move at known and constant depths, that is, $z_{\mathrm{T}}^{[\alpha]}(t) = \bar{z}_{\mathrm{T}}^{[\alpha]}$ for all $t \geq 0$ and $\alpha \in \mathcal{S}_{\mathrm{T}}$. Let also $\mathbf{q}_k^{[\alpha]} = [x_{\mathrm{T},k}^{[\alpha]}, y_{\mathrm{T},k}^{[\alpha]}]^{\mathrm{T}} \in \mathbb{R}^2$ be the projection of the target position vector at discrete time k on the horizontal plane $x_{\mathcal{I}} - y_{\mathcal{I}}$. We consider each target α as a point mass whose motion is described by the discrete model

$$\mathbf{x}_{k+1}^{[\alpha]} = \mathbf{f}(\mathbf{x}_k^{[\alpha]}, \mathbf{u}_k^{[\alpha]}), \tag{5.6}$$

where $\mathbf{x}_k^{[\alpha]} \in \mathbb{R}^n, (n \geq 2)$ is the target's state vector that needs to be estimated, $\mathbf{u}_k^{[\alpha]} \in \mathbb{R}^2$ is the input vector. Note that $\mathbf{x}_k^{[\alpha]}$ includes $\mathbf{q}_k^{[\alpha]}$ and possibly $\dot{\mathbf{q}}_k^{[\alpha]}$. In this chapter, we consider two instances of (5.6) corresponding to the following practical scenarios.

Scenario A: *Target velocity vectors are known*

To justify the assumption, consider a fleet of targets (AUVs) performing a pre-defined mission underwater, for example, cooperative path following [Rego et al., 2019a], with known pre-defined velocity vectors. Under this assumption, we adopt the following target model:

Target model A:

$$\mathbf{x}_{k+1}^{[\alpha]} = \mathbf{x}_k^{[\alpha]} + T_{\text{s}}\mathbf{u}_k^{[\alpha]}, \tag{5.7}$$

where $\mathbf{x}_k^{[\alpha]} = \mathbf{q}_k^{[\alpha]} \in \mathbb{R}^2$; $\mathbf{u}_k^{[\alpha]} \in \mathbb{R}^2, \alpha \in \mathcal{S}_{\text{T}}$, is the target velocity vector in the inertial frame that is *known* to all the trackers, and T_{s} is the sampling interval. Consequently, in this particular case only the positions of the targets need to be estimated. We also assume that prior information on the initial target's state vector $\mathbf{x}_0^{[\alpha]}$ is given in terms of a Gaussian probability density function (PDF) described as

$$\mathbf{x}_0^{[\alpha]} \sim \mathcal{N}(\mathbf{c}_{\text{A},0}^{[\alpha]}, P_{\text{A},0}^{[\alpha]}) \tag{5.8}$$

for some $\mathbf{c}_{\text{A},0}^{[\alpha]} \in \mathbb{R}^2$ and $P_{\text{A},0}^{[\alpha]} \in \mathbb{R}^{2\times 2}; \alpha \in \mathcal{S}_{\text{T}}$.

Scenario B: *Targets' velocity vectors are unknown*

In this case, the trackers need to estimate both the position and velocity vectors of each target. We also consider the case where the target's velocity vector changes slowly, so that it can be assumed to be approximately constant over the observation window, i.e. $\dot{\mathbf{q}}_k^{[\alpha]} = \mathbf{0}$ for all $\alpha \in \mathcal{S}_{\text{T}}$. We thus let $\mathbf{x}_k^{[\alpha]} = [\mathbf{q}_k^{[\alpha]}; \dot{\mathbf{q}}_k^{[\alpha]}] \in \mathbb{R}^4$ be the state vector of the target α; $\alpha \in \mathcal{S}_{\text{T}}$, that must be estimated. The following model for each target, named *Target model B*, can be rewritten explicitly from (5.6) as follows:

Target model B:

$$\mathbf{x}_{k+1}^{[\alpha]} = A_{\text{B}}\mathbf{x}_k^{[\alpha]}, \tag{5.9}$$

where

$$A_{\text{B}} = \begin{bmatrix} I_2 & T_{\text{s}}I_2 \\ 0_{2\times 2} & I_2 \end{bmatrix} \in \mathbb{R}^{4\times 4}. \tag{5.10}$$

Assume further that prior information on the initial target's state $\mathbf{x}_0^{[\alpha]}$ is given by the Gaussian PDF

$$\mathbf{x}_0^{[\alpha]} \sim \mathcal{N}(\mathbf{c}_{\mathrm{B},0}^{[\alpha]}, P_{\mathrm{B},0}^{[\alpha]}) \tag{5.11}$$

for some $\mathbf{c}_{\mathrm{B},0}^{[\alpha]} \in \mathbb{R}^4$ and $P_{\mathrm{B},0}^{[\alpha]} \in \mathbb{R}^{4\times 4}; \alpha \in \mathcal{S}_{\mathrm{T}}$.

Measurement model: We assume that each tracker is equipped an acoustic-based system that measures the distances to all targets at approximately the same discrete instants of time. At each time k, let $d_k^{[i,\alpha]}$ be the true distance from tracker $i; i \in i \in \mathcal{S}$, to target $\alpha; \alpha \in \mathcal{S}_{\mathrm{T}}$, defined as

$$d_k^{[i,\alpha]} = \sqrt{\left\|\mathbf{p}_k^{[i,\alpha]}\right\|^2 + \left(\bar{z}^{[i]} - \bar{z}_{\mathrm{T}}^{[\alpha]}\right)^2}, \tag{5.12}$$

where

$$\mathbf{p}_k^{[i,\alpha]} \triangleq \mathbf{p}_k^{[i]} - \mathbf{q}_k^{[\alpha]}. \tag{5.13}$$

Further, let $y_k^{[i,\alpha]}$ denote the range measurements which we assume are corrupted by Gaussian white noise according to the range measurement model

$$y_k^{[i,\alpha]} = d_k^{[i,\alpha]} + \eta_k^{[i,\alpha]}, \tag{5.14}$$

where $\eta_k^{[i,\alpha]} \sim \mathcal{N}(0, \sigma^2), i \in i \in \mathcal{S}$ and $\alpha \in \mathcal{S}_{\mathrm{T}}$. In practice, range measurements can only be obtained up to a certain distance that depends on the type of range-measuring device used and the environmental conditions (see [Moreno-Salinas et al., 2016]). Therefore, we make the following assumption.

Assumption 3. *We assume that the farthest distance that can be measured reliably by any range-measuring devices is* $d_{\max} > 0$. *We further assume that all range measurements are taken within this distance, i.e.* $d_k^{[i,\alpha]} \leq d_{\max}$ *for all* $k \in \mathbb{N}$, $i \in i \in \mathcal{S}$ *and* $\alpha \in \mathcal{S}_{\mathrm{T}}$. *These constraints will be addressed explicitly in Section 5.6.*

5.3.2 Problem statement

The multiple target localization and pursuit problem can now be formally defined as follows.

Problem 5.1 (Target localization and pursuit)**.** *Consider a set of multiple trackers and a set of multiple targets. Assume that the trackers' dynamics are given by* (5.1) *subject* *to input and state constraints* (5.5)*, and the targets' model is given by* (5.6) *where the targets' states* $\mathbf{x}_k^{[\alpha]}; \alpha \in \mathcal{S}_{\mathrm{T}}$ *are unknown. Further assume that the range measurement model is given by* (5.14)*. Under these conditions, design inputs* $\mathbf{v}^{[i]}; i \in i \in \mathcal{S}$ *for each tracker so that the following tasks are fulfilled*

- *Localization task: Ensure that the range measurements provide "sufficiently rich range information" to estimate the targets' states, in a well defined mathematical sense.*
- *Target pursuit: In addition to the localization task, guarantee that the trackers are in a desired vicinity of the targets, that is, ensure that the distance from any tracker to any target does not exceed* $r^* \leq d_{\max}$*, where* r^* *is a design parameter.*

A natural question that arises in this context is how to quantify the range information required to estimate the states of the targets with a desired level of accuracy. With this objective in mind we adopt an estimation theoretical setting that involves the computation of the FIM, whose definition and construction in the context of target localization are presented in the next section.

Remark 5.1. *Notice that in the target localization and pursuit problem, target localization is the primary task while target pursuit is secondary. The objective of the latter is to keep the trackers close to the targets so as to be able to acquire useful range measurements.*

5.4 The Bayesian FIM in the context of range-based target localization

5.4.1 The Bayesian FIM for a general target model

We start by recalling the concept and construction of the Bayesian FIM that arises in the context of estimation of dynamical systems [Tichavsky et al., 1998]. Consider the problem of estimating the state of a discrete nonlinear system described by

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k \\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k, \boldsymbol{\phi}_k) + \boldsymbol{\eta}_k, \end{aligned} \tag{5.15}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ is the state, $\mathbf{u}_k \in \mathbb{R}^m$ is a known deterministic input, $\mathbf{y}_k \in \mathbb{R}^l$ is the measurement output at discrete time k, $\boldsymbol{\phi}_k$ is a known deterministic trajectory (to be defined later) while $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, Q)$ and $\boldsymbol{\eta}_k \sim \mathcal{N}(\mathbf{0}, R)$ are independent Gaussian random processes that describe the state and measurement noises, respectively. In the context of range-based target localization, $\mathbf{h}(\cdot)$ are the distances from the trackers to the targets and $\boldsymbol{\phi}_k$ are the positions of the trackers (see (5.12)). Let $\hat{\mathbf{x}}_k$ be an estimate of $\mathbf{x}_k$ based on a set of k measurements samples $\{\mathbf{y}_i, \boldsymbol{\phi}_i, \mathbf{u}_{i-1}\}_{i=1}^{k}$ and the prior knowledge of the initial probability density function $p(\mathbf{x}_0)$. According to [Van Trees, 2004], the covariance matrix of $\hat{\mathbf{x}}_k$, denoted P_k, given by any estimator is lower bounded as

$$P_k = \mathbb{E}\{(\hat{\mathbf{x}}_k - \mathbf{x}_k)(\hat{\mathbf{x}}_k - \mathbf{x}_k)^{\mathrm{T}}\} \succeq \mathcal{I}_k^{-1}, \tag{5.16}$$

where $\mathcal{I}_k$ is the so-called Fisher information matrix associated with the estimation of the state $\mathbf{x}_k$ and its inverse is the posterior Cramér-Rao Lower Bound (CRLB). Applying the methodology described in [Tichavsky et al., 1998], the Bayesian FIM is given by the recursive formula

$$\mathcal{I}_{k+1} = D_k^{22} - D_k^{21}(\mathcal{I}_k + D_k^{11})^{-1} D_k^{12}, \tag{5.17}$$

where

$$D_k^{11} = \mathbb{E}\{\left[\nabla_{\mathbf{x}_k}\mathbf{f}^{\mathrm{T}}(\mathbf{x}_k, \mathbf{u}_k)\right] Q^{-1} \left[\nabla_{\mathbf{x}_k}\mathbf{f}^{\mathrm{T}}(\mathbf{x}_k, \mathbf{u}_k)\right]^{\mathrm{T}}\}, \tag{5.18}$$

$$D_k^{12} = -\mathbb{E}\{\nabla_{\mathbf{x}_k}\mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)\}Q^{-1} = [D_k^{21}]^{\mathrm{T}}, \tag{5.19}$$

$$D_k^{22} = Q^{-1} + \Omega_{k+1}, \tag{5.20}$$

with

$$\Omega_{k+1} = \mathbb{E}\{H_{k+1}(\mathbf{x}_{k+1}, \boldsymbol{\phi}_{k+1})\} \tag{5.21}$$

and

$$\begin{aligned} &H_{k+1}(\mathbf{x}_{k+1}, \boldsymbol{\phi}_{k+1}) \\ &= \left[\nabla_{\mathbf{x}_{k+1}}\mathbf{h}^{\mathrm{T}}(\mathbf{x}_{k+1}, \boldsymbol{\phi}_{k+1})\right] R^{-1} \left[\nabla_{\mathbf{x}_{k+1}}\mathbf{h}^{\mathrm{T}}(\mathbf{x}_{k+1}, \boldsymbol{\phi}_{k+1})\right]^{\mathrm{T}}. \end{aligned} \tag{5.22}$$

In the above equations, $\mathbb{E}$ denotes the expectation operator and, for a given $\mathbf{x} = [x_1, ..., x_n]^{\mathrm{T}} \in \mathbb{R}^n$, $\nabla_{\mathbf{x}} \triangleq [\frac{\partial}{\partial x_1}, ..., \frac{\partial}{\partial x_n}]^{\mathrm{T}}$. Note that the expectation in (5.18) and (5.19) is with respect to the distribution of $\mathbf{x}_k$ while in (5.21) it is computed with respect to the distribution of $\mathbf{x}_{k+1}$. The recursion in (5.17) is initialized with the prior information of the initial state

$\mathbf{x}_0$ as

$$\mathcal{I}_0 = \mathbb{E}\{[\nabla_{\mathbf{x}_0} \log p(\mathbf{x}_0)][\nabla_{\mathbf{x}_0} \log p(\mathbf{x}_0)]^{\mathrm{T}}\}. \tag{5.23}$$

We now consider a special case of (5.15) where the state equation is linear, given by

$$\begin{aligned} \mathbf{x}_{k+1} &= A\mathbf{x}_k + B\mathbf{u}_k + \mathbf{w}_k \\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k, \boldsymbol{\phi}_k) + \boldsymbol{\eta}_k. \end{aligned} \tag{5.24}$$

With this model, it follows from (5.18) and (5.19) that $D_k^{11} = A^{\mathrm{T}}Q^{-1}A$ and $D_k^{12} = A^{\mathrm{T}}Q^{-1}$. Inserting the latter in (5.17) yields

$$\begin{aligned} \mathcal{I}_{k+1} =& Q^{-1} + \Omega_{k+1} \\ & -Q^{-1}A(\mathcal{I}_k + A^{\mathrm{T}}Q^{-1}A)^{-1}A^{\mathrm{T}}Q^{-1}. \end{aligned} \tag{5.25}$$

Applying the matrix inversion lemma[1], (5.25) can be simplified as

$$\mathcal{I}_{k+1} = (Q + A\mathcal{I}_k^{-1}A^{\mathrm{T}})^{-1} + \Omega_{k+1}. \tag{5.26}$$

If A is non-singular and the process noise is absent, that is, if $Q = 0$ in (5.26), then

$$\mathcal{I}_{k+1} = [A^{-1}]^{\mathrm{T}}\mathcal{I}_k A^{-1} + \Omega_{k+1}. \tag{5.27}$$

Since $\mathbf{h}(\cdot)$ is a nonlinear function, in general it may be impossible to compute Ω_{k+1} given by (5.21) analytically. In practice, a Monte Carlo simulation method can be used to approximate the expectation operator. The key idea behind the Monte Carlo method is that, given $\mathbf{x}_0^{(j)} \sim p(\mathbf{x}_0); j = 1, ..., M$ and a sequence of $\{\mathbf{u}_i, \boldsymbol{\phi}_{i+1}\}_{i=0}^{k}$, we carry out M simulations of the system (5.24) to obtain M realizations of the trajectory $\{\mathbf{x}_{i+1}^{(j)}\}_{i=0}^{k}$. Then, the expectation in (5.21) can be computed approximately as

$$\Omega_{k+1} \approx \frac{1}{M}\sum_{j=1}^{M} H_{k+1}(\mathbf{x}_{k+1}^{(j)}, \boldsymbol{\phi}_{k+1}). \tag{5.28}$$

5.4.2 The Bayesian FIM for Target Model A

For each $i \in \mathcal{S}$, let $\mathcal{I}_{\mathrm{A},k}^{[i,\alpha]} \in \mathbb{R}^{2\times 2}$ be the Bayesian FIM at time k, associated with the estimation of the state $\mathbf{x}_k^{[\alpha]}$ of target α with motion described by *Target model A*, using

[1] $(A + BCD)^{-1} = A^{-1} - A^{-1}B(DA^{-1}B + C^{-1})^{-1}DA^{-1}$

the range measurements from tracker i. Notice that the system that results from the combination of *Target model A* described by (5.7) and the output model given by (5.12) is a special case of (5.24) with $A = I_2, B = T_\mathrm{s} I_2, R = \sigma$, $\boldsymbol{\phi}_k = \mathbf{p}_k^{[i]}, Q = 0$ and $\mathbf{h}(\cdot) = d_k^{[i,\alpha]}$, where $d_k^{[i,\alpha]}$ is given by (5.12). Substituting the above parameters in (5.27), $\mathcal{I}_{\mathrm{A},k}^{[i,\alpha]}$ is given by

$$\mathcal{I}_{\mathrm{A},k+1}^{[i,\alpha]} = \mathcal{I}_{\mathrm{A},k}^{[i,\alpha]} + \Omega_{k+1}^{[i,\alpha]} \tag{5.29}$$

where $\Omega_{k+1}^{[i,\alpha]}$ is computed as in (5.21), yielding

$$\Omega_{k+1}^{[i,\alpha]} = \mathbb{E}\left\{ \frac{1}{\sigma^2} \left(\frac{\mathbf{p}_{k+1}^{[i,\alpha]}}{d_{k+1}^{[i,\alpha]}} \right) \left(\frac{\mathbf{p}_{k+1}^{[i,\alpha]}}{d_{k+1}^{[i,\alpha]}} \right)^\mathrm{T} \right\} \tag{5.30}$$

with $\mathbf{p}_k^{[i,\alpha]}$ as in (5.13). Note that the expectation in (5.30) is taken over the distribution of $\mathbf{q}_{k+1}^{[\alpha]}$. We now consider $\mathcal{I}_{\mathrm{A},k}^{[\alpha]}$, the FIM for estimating $\mathbf{x}_k^{[\alpha]}$, using the range measurements from all trackers, collectively. Compared with the case of a single tracker, more range measurements are augmented to the measurement output vector, i.e. $\mathbf{h}(\cdot) = [d_k^{[1,\alpha]}, ..., d_k^{[p,\alpha]}]^\mathrm{T} \in \mathbb{R}^p$ and $R = \mathrm{Diag}(\sigma, ..., \sigma) \in \mathbb{R}^{\mathrm{p}\times\mathrm{p}}$. Inserting the above in (5.27) yields

$$\mathcal{I}_{\mathrm{A},k+1}^{[\alpha]} = \mathcal{I}_{\mathrm{A},k}^{[\alpha]} + \sum_{i=1}^{p} \Omega_{k+1}^{[i,\alpha]}. \tag{5.31}$$

It follows from (5.8) and (5.23) that the recursions (5.29) and (5.31) start with the prior information

$$\mathcal{I}_{\mathrm{A},0}^{[i,\alpha]} = \mathcal{I}_{\mathrm{A},0}^{[\alpha]} = [P_{\mathrm{A},0}^{[\alpha]}]^{-1}, \tag{5.32}$$

that is, the FIM iterations start with the prior information on the target's positions. Let $\mathbf{x}_k = [\mathbf{x}_k^{[1]}, ..., \mathbf{x}_k^{[q]}] \in \mathbb{R}^{2q}$ be the states of all targets that need to be estimated. Also, let $\mathcal{I}_{\mathrm{A},k} \in \mathbb{R}^{2q\times 2q}$ denote the total information available to estimate $\mathbf{x}_k$ using all range measurements from all the trackers to each of the targets. Using the methodology mentioned in the previous cases, it can be shown that

$$\mathcal{I}_{\mathrm{A},k} = \mathrm{Diag}(\mathcal{I}_{\mathrm{A,k}}^{[1]}, ..., \mathcal{I}_{\mathrm{A,k}}^{[\mathrm{q}]}), \tag{5.33}$$

where each $\mathcal{I}_{\mathrm{A},k}^{[\alpha]}; \alpha = 1, ..., q$ is computed using (5.31).

5.4.3 The Bayesian FIM for Target Model B

For every $i \in \mathcal{S}$, let $\mathcal{I}_{\mathrm{B},k}^{[i,\alpha]} \in \mathbb{R}^{4\times4}$ be the FIM associated with the estimation of the state $\mathbf{x}_k^{[\alpha]} \in \mathbb{R}^4$ of target α, with motion described by *Target model B*, using the range measurements from tracker i. Notice that *Target model B* given by (5.9) is a special case of (5.24) with $A = A_{\mathrm{B}}$, where A_{B} is given by (5.10), $R = \sigma$, $\boldsymbol{\phi}_k = \mathbf{p}_k^{[i]}, Q = 0$ and $\mathbf{h}(\cdot) = d_k^{[i,\alpha]}$. Substituting the above parameters in (5.27), $\mathcal{I}_{\mathrm{B},k}^{[i,\alpha]}$ is given by

$$\mathcal{I}_{\mathrm{B},k+1}^{[i,\alpha]} = [A_{\mathrm{B}}^{-1}]^{\mathrm{T}} \mathcal{I}_{\mathrm{B},k}^{[i,\alpha]} A_{\mathrm{B}}^{-1} + \begin{bmatrix} \Omega_{k+1}^{[i,\alpha]} & 0_{2\times2} \\ 0_{2\times2} & 0_{2\times2} \end{bmatrix}. \tag{5.34}$$

Let $\mathcal{I}_k^{[\alpha]}$ be the FIM associated with the estimation of $\mathbf{x}_k^{[\alpha]}$ using the range measurements from all trackers, collectively. Similar to the case of *Target model A*, $\mathcal{I}_{\mathrm{B},k}^{[\alpha]}$ can be computed using the recursion formula

$$\mathcal{I}_{\mathrm{B},k+1}^{[\alpha]} = [A_{\mathrm{B}}^{-1}]^{\mathrm{T}} \mathcal{I}_{\mathrm{B},k}^{[\alpha]} A_{\mathrm{B}}^{-1} + \sum_{i=1}^{p} \begin{bmatrix} \Omega_{k+1}^{[i,\alpha]} & 0_{2\times2} \\ 0_{2\times2} & 0_{2\times2} \end{bmatrix}. \tag{5.35}$$

It follows from (5.11) and (5.23) that the recursions (5.34) and (5.35) start with the prior information

$$\mathcal{I}_{\mathrm{B},0}^{[i,\alpha]} = \mathcal{I}_{\mathrm{B},0}^{[\alpha]} = [P_{\mathrm{B},0}^{[\alpha]}]^{-1}. \tag{5.36}$$

Let $\mathbf{x}_k = [\mathbf{x}_k^{[1]}, ..., \mathbf{x}_k^{[q]}]^{\mathrm{T}} \in \mathbb{R}^{4q}$ be the state of all targets that needs to be estimated. Also, let $\mathcal{I}_{\mathrm{B},k} \in \mathbb{R}^{4q\times4q}$ denote the total information for estimating $\mathbf{x}_k$. Clearly, $\mathcal{I}_{\mathrm{B},k}$ is given by

$$\mathcal{I}_{\mathrm{B},k} = \mathrm{Diag}(\mathcal{I}_{\mathrm{B,k}}^{[1]}, ..., \mathcal{I}_{\mathrm{B,k}}^{[\mathrm{q}]}), \tag{5.37}$$

where each $\mathcal{I}_{\mathrm{B},k}^{[\alpha]}; \alpha = 1, ..., q$ is computed by (5.35). Note that by construction, the FIM is symmetric and positive semidefinite. In the context of the present chapter, the information carried by the FIM for estimation purposes is measured by its determinant, the metric adopted in [Moreno-Salinas et al., 2013, Bishop et al., 2010].

5.5 Preliminary analysis: ideal geometries for maximum range-related information

In the previous section, a FIM was constructed as a means to quantify the range information available to estimate the targets' states. In this section, we will analyze a special case of the Bayesian FIMs derived for *Target Model A* and *B* where there is no prior information on the initial state of the target[2]. We will show that this special case leads to simplified FIMs that allows us to derive analytically "ideal" condition on the trackers' trajectories that yield maximum achievable range-information acquired to estimate the targets' states. The results in this section helps understand at a very intuitive level the types of optimal relative tracker-target geometries that the trackers should reach and maintain to maximize the range-related information. Furthermore, they play an important role in benchmarking the types of solutions that will be obtained numerically using the far more realistic approach to target localization and pursuit introduced in Section 5.6.

To analyze the Bayesian FIMs with the above assumptions, their formulas in recursions (5.31) and (5.35) can be rewritten in compact form as follows. Firstly, let

$$a_n^{[i,\alpha]} = \left(x_n^{[i]} - x_{\mathrm{T},n}^{[\alpha]}\right) / d_n^{[i,\alpha]} \tag{5.38a}$$

$$b_n^{[i,\alpha]} = \left(y_n^{[i]} - y_{\mathrm{T},n}^{[\alpha]}\right) / d_n^{[i,\alpha]}, \tag{5.38b}$$

for all $n \in \{1, ..., k\}$. Define also two vectors $\mathbf{a}_{i,\alpha} = [a_1^{[i,\alpha]}, ..., a_k^{[i,\alpha]}]^{\mathrm{T}} \in \mathbb{R}^k$ and $\mathbf{b}_{i,\alpha} = [b_1^{[i,\alpha]}, ..., b_k^{[i,\alpha]}]^{\mathrm{T}} \in \mathbb{R}^k$.

Lemma 5.1. *Consider the Bayesian FIMs related to the problem of estimating the state of target α computed by recursions* (5.31) *and* (5.35)*, corresponding to Target model A and Target model B, respectively. If there is no prior information on the initial target state, i.e. $\mathcal{I}_{\mathrm{A},0}^{[i,\alpha]}$ in* (5.32) *and $\mathcal{I}_{\mathrm{B},0}^{[i,\alpha]}$ in* (5.36) *are zero then,*

$$\mathcal{I}_{\mathrm{A},k}^{[\alpha]} = \sum_{i=1}^{p} \frac{1}{\sigma^2} \begin{bmatrix} \|\mathbf{a}_{i,\alpha}\|^2 & \mathbf{a}_{i,\alpha}^{\mathrm{T}}\mathbf{b}_{i,\alpha} \\ \mathbf{a}_{i,\alpha}^{\mathrm{T}}\mathbf{b}_{i,\alpha} & \|\mathbf{b}_{i,\alpha}\|^2 \end{bmatrix} \tag{5.39}$$

[2] Equivalent with $\mathcal{I}_{\mathrm{A},0}^{[i,\alpha]}$ in (5.32), $\mathcal{I}_{\mathrm{B},0}^{[i,\alpha]}$ in (5.36) are zero

and

$$\mathcal{I}_{\text{B},k}^{[\alpha]} = \begin{bmatrix} A_\alpha & B_\alpha \\ B_\alpha & C_\alpha \end{bmatrix}, \tag{5.40}$$

where $A_\alpha = \mathcal{I}_{\text{A},k}^{[\alpha]}$ *with* $\mathcal{I}_{\text{A},k}^{[\alpha]}$ *given by* (5.39),

$$\begin{aligned} B_\alpha &= -\sum_{i=1}^{p} \frac{1}{\sigma^2} \begin{bmatrix} \|\mathbf{a}_{i,\alpha}\|_{D_1}^2 & \mathbf{a}_{i,\alpha}^{\text{T}} D_1 \mathbf{b}_{i,\alpha} \\ \mathbf{a}_{i,\alpha}^{\text{T}} D_1 \mathbf{b}_{i,\alpha} & \|\mathbf{b}_{i,\alpha}\|_{D_1}^2 \end{bmatrix}, \\ C_\alpha &= \sum_{i=1}^{p} \frac{1}{\sigma^2} \begin{bmatrix} \|\mathbf{a}_{i,\alpha}\|_{D_2}^2 & \mathbf{a}_{i,\alpha}^{\text{T}} D_2 \mathbf{b}_{i,\alpha} \\ \mathbf{a}_{i,\alpha}^{\text{T}} D_2 \mathbf{b}_{i,\alpha} & \|\mathbf{b}_{i,\alpha}\|_{D_2}^2 \end{bmatrix}, \end{aligned} \tag{5.41}$$

$D_1 = \text{Diag}(\tau_1, ..., \tau_k) \in \mathbb{R}^{k\times k}$, $D_2 = \text{Diag}(\tau_1^2, ..., \tau_k^2) \in \mathbb{R}^{k\times k}$ *and*

$$\tau_n = (k-n)T_{\text{s}},\ n = \{1, ..., k\}.$$

Proof: For *Target model A*, (5.39) is obtained by substituting $\mathcal{I}_{\text{A},0}^{[i,\alpha]} = 0$ and the relation in (5.38) in (5.31), while for *Target model B*, (5.40) is obtained by substituting $\mathcal{I}_{\text{B},0}^{[i,\alpha]} = 0$ and the relation in (5.38) in (5.35). Notice also that since the process noise is zero the target moves in a deterministic manner; thus, the expectation in (5.30) was dropped to obtain (5.39) and (5.40). ■

Before we proceed we introduce the variables

$$c^{[i,\alpha]} = \left(\frac{(\bar{z}^{[i]} - \bar{z}_{\text{T}}^{[\alpha]})}{d_{\max}} \right)^2 \tag{5.42}$$

and $\gamma^{[i,\alpha]}$, with the latter defined by

$$\cos(\gamma_n^{[i,\alpha]}) = \left(x_n^{[i]} - x_{\text{T},n}^{[\alpha]} \right) / \|\mathbf{p}_n^{[i,\alpha]}\| \tag{5.43}$$

for all $n \in \{1, ..., k\}$. By definition, $\gamma_n^{[i,\alpha]}$ is the angle between the projection on the $x_{\mathcal{I}} - y_{\mathcal{I}}$ plane of the relative position vector from tracker i to target α and the $x_{\mathcal{I}}$-axis (see Fig.5.3). The main results in this section are presented next.

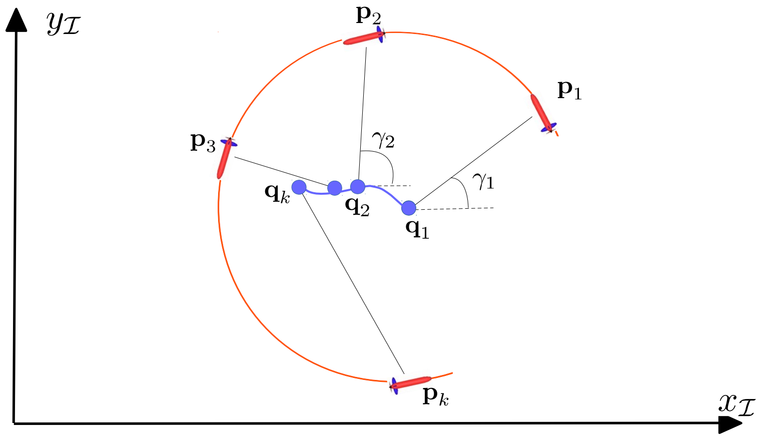


Figure 5.3: Illustration of an ideal tracker-target geometry that maximizes the range information. Successive positions and respective trajectories of target (blue) and tracker (red). This is the case of one tracker - one target so for the simplicity the superscript $[i, \alpha]$ in γ is dropped.

5.5.1 Single tracker - single target

We start by considering the simple case of a single tracker localizing a single target. In this configuration, $i = p = 1$ and $\alpha = q = 1$. Therefore, for simplicity of notation we drop the superscripts (subscripts) i and α wherever they appear in this subsection. The following result provides a measure of the range-related information available in this scenario.

Theorem 5.1. *Consider the case of a single tracker localizing a single target. Let assumptions in Lemma 5.1 hold. Then, the following statements hold true.*

i. For Scenario A, the range information quantified by the determinant of $\mathcal{I}_{\text{A},k}$ is maximal when $\mathcal{I}_{\text{A},k} = \mathcal{I}_{11\text{A}}$, where

$$\mathcal{I}_{11\text{A}} = (1 - c)\sigma^{-2}\mathcal{I}_{\text{A}}^{o}, \tag{5.44}$$

c is given by (5.42)*, and*

$$\mathcal{I}_{\text{A}}^{o} = kI_2/2. \tag{5.45}$$

ii. For Scenario B, the determinant of $\mathcal{I}_{\text{B},k}$ is maximal when $\mathcal{I}_{\text{B},k} = \mathcal{I}_{11\text{B}}$, where

$$\mathcal{I}_{11\text{B}} = (1 - c)\sigma^{-2}\mathcal{I}_{\text{B}}^{o}, \tag{5.46}$$

c is given by (5.42),

$$\mathcal{I}^o_{\mathrm{B}} = \begin{bmatrix} kI_2/2 & \Delta_1 I_2/2 \\ \Delta_1 I_2/2 & \Delta_2 I_2/2 \end{bmatrix}, \tag{5.47}$$

with $\Delta_1 \triangleq -\sum_{n=1}^{k} \tau_n$*, and* $\Delta_2 \triangleq \sum_{n=1}^{k} \tau_n^2$.

The matrices $\mathcal{I}_{11\mathrm{A}}$ *and* $\mathcal{I}_{11\mathrm{B}}$ *are called the optimal range information matrices for Scenarios A and B, respectively.*

Proof: See section 5.9.1.

Remark 5.2. *For Target model A, if the tracker and the target are at the same depth, that is,* $c = 0$*, then the optimal FIM in* (5.44) *recovers the result in [Crasta et al., 2018] (see Lemma 2 in [Crasta et al., 2018] for the case one tracker-one target). This happens because, for this particular case, we assumed that there is no prior information on the initial state of the target and therefore the target can be viewed as a deterministic process with unknown initial target's state. Thus, under these assumptions the simplified Bayesian FIM associated with the estimation of the target state at current time* ($\mathbf{x}_k$) *in this chapter is equivalent to the parametric FIM associated with the estimation of the initial target's state* ($\mathbf{x}_0$) *in [Crasta et al., 2018]. However, it is important to stress that the parametric FIM in [Crasta et al., 2018] is only applicable to deterministic targets, whereas as shown in the previous section the Bayesian FIM in this chapter is applicable to all types of target motion. Furthermore, the method used to compute the Bayesian FIM in the present chapter is a recursive approach, which is simpler and more transparent than that used to compute the parametric FIM in [Crasta et al., 2018].*

We now study possible target-tracker geometries that maximize the range-related information. We obtain the following result.

Proposition 5.1. *Consider the case of a single tracker localizing a single target. In order to reach the maximal range information, as characterized in Theorem 5.1, an optimal trajectory for the tracker is obtained by encircling the projection of the target on the* $x_{\mathcal{I}} - y_{\mathcal{I}}$ *plan such that any two successive range measurements are taken at positions that satisfy*

$$\gamma_{n+1} - \gamma_n = \omega \triangleq \pm 2\pi/N \tag{5.48}$$

for all $n \in \{1, ..., k\}$*, where* γ_n *defined in* (5.43) *and some natural number* $N; N \geq 3$*. Furthermore, if the tracker and the target are at different depths, in addition to* (5.48) *the tracker encircles the target along a circumference of radius* $r \triangleq \sqrt{d_{\max}^2 - (\bar{z} - \bar{z}_{\mathrm{T}})^2}$*.*

Proof: See section 5.9.2.
In (5.48), the symbol ± indicate the direction of the tracker's motion (" + " is counter-clockwise and " − " is clockwise). Proposition 5.1 implies that if the target is fixed (stationary), an ideal trajectory for the tracker is obtained by having the tracker follow a circumference centered at the target with a constant linear speed and a constant course rate, see Fig.5.3.

5.5.2 Multiple trackers - single target

We now consider the case when more than one tracker is used to localize a single moving target and derive the following result.

Theorem 5.2. *Consider the case of* p *trackers localizing a single target, say* α*. Let assumptions in Lemma 5.1 hold. Then, the following statements hold true.*

i. For Scenario A, the determinant of $\mathcal{I}_{\mathrm{A},k}^{[\alpha]}$ *is maximal when* $\mathcal{I}_{\mathrm{A},k}^{[\alpha]} = \bar{\mathcal{I}}_{\mathrm{A},k}^{[\alpha]}$*, where*

$$\bar{\mathcal{I}}_{\mathrm{A},k}^{[\alpha]} = \left(p - \sum_{i=1}^{p} c^{[i,\alpha]} \right) \sigma^{-2} \mathcal{I}_{\mathrm{A}}^{o}, \tag{5.49}$$

$c^{[i,\alpha]}$ *is given by* (5.42)*, and* $\mathcal{I}_{\mathrm{A}}^{o}$ *is given by* (5.45)*.*

ii. For Scenario B, the determinant of $\mathcal{I}_{\mathrm{B},k}^{[\alpha]}$ *is maximal when* $\mathcal{I}_{\mathrm{B},k}^{[\alpha]} = \bar{\mathcal{I}}_{\mathrm{B},k}^{[\alpha]}$*, where*

$$\bar{\mathcal{I}}_{\mathrm{B},k}^{[\alpha]} = \left(p - \sum_{i=1}^{p} c^{[i,\alpha]} \right) \sigma^{-2} \mathcal{I}_{\mathrm{B}}^{o}, \tag{5.50}$$

$c^{[i,\alpha]}$ *is given by* (5.42)*, and* $\mathcal{I}_{\mathrm{B}}^{o}$ *is given in* (5.47)*.*

Proof: See section 5.9.3.
We now discuss possible geometries that maximize the range information. To this end, we shall study the angle $\beta^{[i,j;\alpha]}$ formed by the two relative position vectors from the trackers i and j to the target α in the $x_{\mathcal{I}} - y_{\mathcal{I}}$ plane (see Fig.5.4). We obtain the following result.

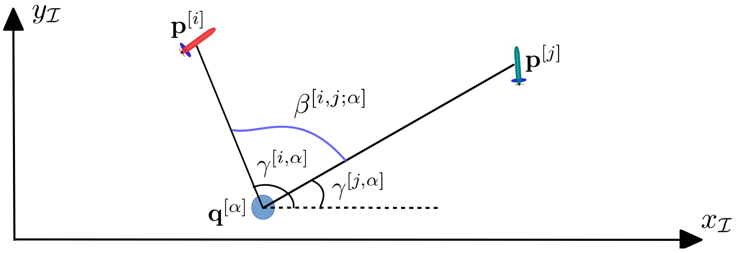


Figure 5.4: Angle formed by the relative vectors between trackers and target α in the $x_{\mathcal{I}} - y_{\mathcal{I}}$ plane.

Geometry for two trackers- single target (p=2, q=1)

Proposition 5.2. *Consider the case of two trackers localizing a single target α. In order to obtain maximal range information, as characterized in Theorem 5.2, an ideal trajectory for the trackers is to maintain the relative position vectors from the two trackers to the target orthogonal, that is,*

$$\beta_n^{[1,2;\alpha]} = \pi/2 + l\pi \tag{5.51}$$

for all $n \in \{1, ..., k\}$ and $l \in \mathbb{N}$. Furthermore, if the trackers and the target are at different depths, in addition to the orthogonality condition above the condition given by $\left\|\mathbf{p}_n^{[i]} - \mathbf{q}_n^{[\alpha]}\right\| = \sqrt{d_{\max}^2 - (\bar{z}^{[i]} - \bar{z}_{\mathrm{T}}^{[\alpha]})^2}$ for all $i \in \{1, 2\}$ and $n \in \{1, ..., k\}$ applies.

Proof: See section 5.9.4.

Fig.5.5 illustrates possible trackers-target trajectories that maximize the range information when the trackers and the targets are at the same depth.

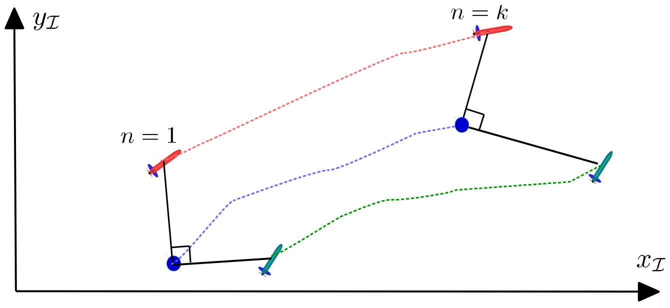


Figure 5.5: Example of an ideal relative trackers-target geometry for the case of two trackers that yields maximum achievable range information. Trackers' trajectories: tracker 1 (red), and tracker 2 (green).

Geometry for p trackers - single target ($p \geq 3, q = 1$)

Proposition 5.3. *Consider the case of p trackers localizing a single target α. In order to obtain maximal range information, as characterized in Theorem 5.2, an ideal trajectory for the trackers is to keep them around the target in such a way that*

$$\beta_n^{[i,j;\alpha]} = \beta_n^{[j,i;\alpha]} = 2\pi/p \tag{5.52}$$

for all $i, j \in \{1, ..., p\}$ and $n \in \{1, ..., k\}$. Furthermore, if the trackers and the target are at different depths, in addition to the condition in (5.52) *the condition given by $\left\| \mathbf{p}_n^{[i]} - \mathbf{q}_n^{[\alpha]} \right\| = \sqrt{d_{\max}^2 - (\bar{z}^{[i]} - \bar{z}_{\mathrm{T}}^{[\alpha]})^2}$ for all $n \in \{1, ..., k\}$ and $i \in \{1, ..., p\}$ applies.*

Proof: See section 5.9.5.

Fig.5.6 illustrates, for the case of three trackers and one target, possible tracker-target trajectories that yield the maximum range information when all trackers and the target are at the same depth. In this case, the trackers move in such a way as to keep the target at the in-center of the equilateral triangle formed by their positions, viewed as vertices of the triangle.

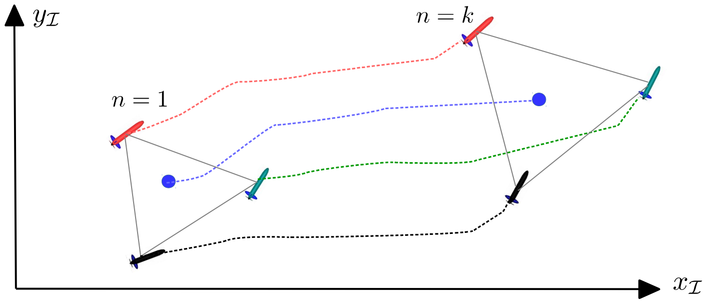


Figure 5.6: Example of a relative trackers-target geometry that maximizes range information. Trackers and target trajectories: tracker 1 (red), tracker 2 (green), tracker 3 (black), and target (blue).

5.5.3 Multiple trackers - multiple targets

We now consider the case of multiple trackers and multiple targets. We first characterize the optimal range-related FIM, as follows.

Theorem 5.3. *Consider the situation of p trackers localizing q targets. Let assumptions in Lemma 1 hold. Then, the following statements hold true.*

i. For Scenario A, the determinant of $\mathcal{I}_{\mathrm{A},k}$ is maximal when $\mathcal{I}_{\mathrm{A},k} = \bar{\mathcal{I}}_{\mathrm{A},k}$, where

$$\bar{\mathcal{I}}_{\mathrm{A},k} = \mathrm{Diag}(\bar{\mathcal{I}}^{[1]}_{\mathrm{A},k}, ..., \bar{\mathcal{I}}^{[q]}_{\mathrm{A},k}), \tag{5.53}$$

$\bar{\mathcal{I}}^{[\alpha]}_{\mathrm{A},k}; \alpha = \{1, ..., q\}$ *given by* (5.49).

ii. For Scenario B, the determinant of $\mathcal{I}_{\mathrm{B},k}$ is maximal when $\mathcal{I}_{\mathrm{B},k} = \bar{\mathcal{I}}_{\mathrm{B},k}$, where

$$\bar{\mathcal{I}}_{\mathrm{B},k} = \mathrm{Diag}(\bar{\mathcal{I}}^{[1]}_{\mathrm{B},k}, ..., \bar{\mathcal{I}}^{[q]}_{\mathrm{B},k}), \tag{5.54}$$

$\bar{\mathcal{I}}^{[\alpha]}_{\mathrm{B},k}; \alpha = \{1, ..., q\}$ *given by* (5.50).

Proof: See section 5.9.6.

We now discuss the ideal geometry for the case of two trackers localizing multiple targets.

Proposition 5.4. *Consider the case of two trackers localizing q targets. In order to obtain maximal range information, as characterized in Theorem 5.3, the ideal trajectories for the trackers correspond to maintaining the relative position vectors from them to each target orthogonal. Furthermore, if the trackers and the targets are at different depths, in addition to the orthogonality condition the condition given by $\left\|\mathbf{p}^{[i]}_n - \mathbf{q}^{[\alpha]}_n\right\| = \sqrt{d^2_{\max} - (\bar{z}^{[i]} - \bar{z}^{[\alpha]}_{\mathrm{T}})^2}$ for all $n \in \{1, ..., k\}, i \in \{1, 2\}$ and $\alpha \in \{1, ..., q\}$ applies.*

Proof: The proof is similar to that of Proposition 5.2. ■

Fig.5.7 illustrates a possible tracker-target trajectory that maximizes the range information for the case of two trackers and two targets. It can be seen that, in order to obtain maximal range information, the trackers move so that the circumscribed circumference that is centered at the middle of a line joining the two trackers goes through the positions the targets.

5.6 MPC framework for range-based SLAP

The previous section addressed the problem of multiple target localization using multiple trackers by characterizing the types of possible target-tracker geometries that yield max-

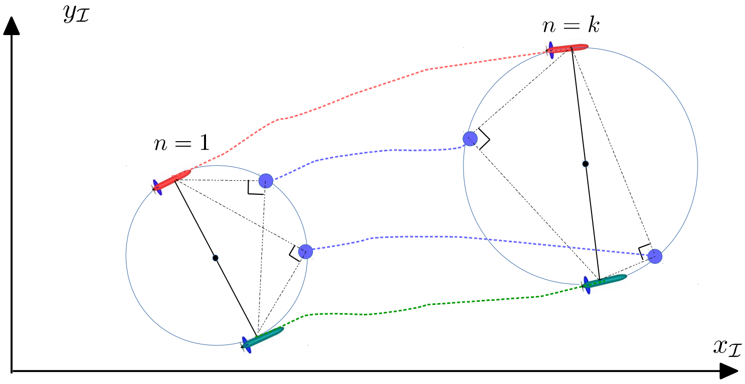


Figure 5.7: Illustration of ideal trackers-targets trajectories that maximizes the range information. Targets trajectories (blue). Tracker 1 (red), tracker 2 (green).

imum range-based information. The results obtained characterize the ideal positions of the trackers with respect to the foreseeen motion of the targets. The analysis provided valuable insight into the types of ideal tracker trajectories required. However, further work is required to bring these theoretical advances to bear on the development of effective target localization and pursuit systems. In fact, the analysis eschewed four key issues that occur in real situations: i) The target's motion can not be known completely in advance, ii) The motions of the trackers may be severely restricted due to their dynamics and state/input constraints, iii) the trackers should maneuver in the vicinity of the targets in order to ensure that range measurements can be obtained using appropriate acoustic sensors, and iv) the optimal target-tracker geometries must be defined with respect to the estimated positions of the targets (obtained with a properly designed estimator), since the real states of the latter are unknown.

To address the above challenges, we propose a scheme that involves the execution of three different phases: optimal tracker motion planning based on prior information about the

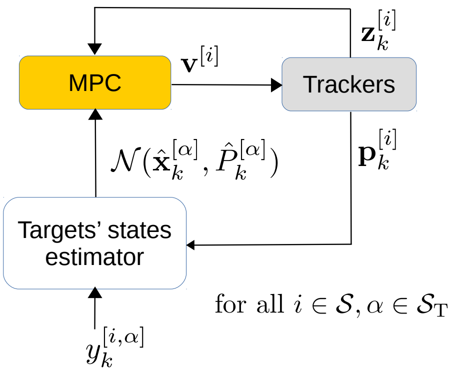

MPC
$\mathbf{v}^{[i]}$
Trackers
$\mathbf{z}_k^{[i]}$
$\mathcal{N}(\hat{\mathbf{x}}_k^{[\alpha]}, \hat{P}_k^{[\alpha]})$
$\mathbf{p}_k^{[i]}$
Targets' states estimator
for all $i \in \mathcal{S}, \alpha \in \mathcal{S}_{\mathrm{T}}$
$y_k^{[i,\alpha]}$

Figure 5.8: Receding horizon strategy for target localizing and pursuit.

targets, motion control of the trackers based on the planned motion, and range- based target state estimation. To implement the first two, an MPC-like framework is adopted whereby the motions of the trackers are planned by resorting to a receding horizon framework. In this set-up, the cost criterion adopted includes in its structure a measure of the FIM constructed in Section 5.4. The latter uses the prior knowledge of the targets to predict the range-related information available for target localization over a short prediction horizon, with the objective of maximizing it by proper choice of the inputs to the trackers and, as a consequence, of the tracker trajectories. Only the first term in the optimal sequence of tracker inputs is used to drive the trackers, after which the procedure is repeated.

The MPC scheme is illustrated in Fig.5.8 and is described as follows. Let $\mathbf{z}_k^{[i]}$ be the state of the tracker $i; i \in \mathcal{S}$ at discrete time k. Assume that at each time instant k the estimates of the targets' states, denoted $\hat{\mathbf{x}}_k^{[\alpha]}$ and their covariance matrices $\hat{P}_k^{[\alpha]} : \alpha \in \mathcal{S}_{\mathrm{T}}$ can be provided by an estimator (e.g. EKF). Given this initial information at time k, the FIM based on N range measurements ahead taken at instants $k+1, ..., k+N$ can be predicted using (5.33) for *Target model A* and (5.37) for *Target model B*, where N is called the prediction horizon. From (5.33) and (5.37), it can be seen that the predicted FIM depends on the trackers' inputs $\mathbf{v}^{[i]}; i \in \mathcal{S}$ and the initial information about the targets. Therefore, the predicted FIM, denoted $\mathcal{I}_{\mathrm{p}}$ is defined explicitly as

$$\mathcal{I}_{\mathrm{p}}(\mathbf{z}_k^{[i]}, \hat{\mathbf{x}}_k^{[\alpha]}, \hat{P}_k^{[\alpha]}, \bar{\mathbf{v}}^{[i]}) \triangleq \begin{cases} \mathcal{I}_{\mathrm{A},N} \text{ for } \textit{Scenario A}, \\ \mathcal{I}_{\mathrm{B},N} \text{ for } \textit{Scenario B}, \end{cases}$$

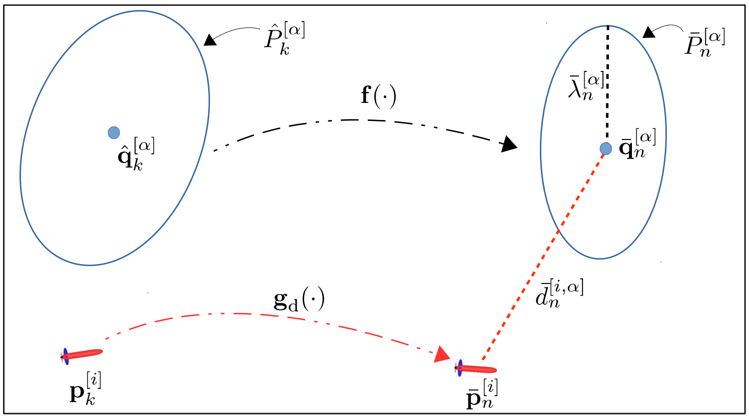

$\hat{P}_k^{[\alpha]}$
$\bar{P}_n^{[\alpha]}$
$\mathbf{f}(\cdot)$
$\bar{\lambda}_n^{[\alpha]}$
$\hat{\mathbf{q}}_k^{[\alpha]}$
$\bar{\mathbf{q}}_n^{[\alpha]}$
$\bar{d}_n^{[i,\alpha]}$
$\mathbf{g}_{\mathrm{d}}(\cdot)$
$\mathbf{p}_k^{[i]}$
$\bar{\mathbf{p}}_n^{[i]}$

Figure 5.9: Illustration of the predicted uncertainty of the target's position for *Target model A*. Recall that in this case the target's state only contains the target's position.

where $\mathcal{I}_{\text{A},N}$ and $\mathcal{I}_{\text{B},N}$ are computed using (5.33) and (5.37), respectively while $\bar{\mathbf{v}}^{[i]} = [[\bar{\mathbf{v}}_k^{[i]}]^{\text{T}}, ..., [\bar{\mathbf{v}}_{N-1}^{[i]}]^{\text{T}}]^{\text{T}}; i \in \mathcal{S}$, are the trackers' inputs over the prediction horizon. It is important to note that at every sampled time k, $\mathcal{I}_{\text{A},N}$ and $\mathcal{I}_{\text{B},N}$ are initialized with the prior information $\hat{P}_k^{[\alpha]}$. Based on the discussion in the previous section, the primary objective is to find an optimal input to maximize the range-based information that is defined by the cost

$$J_{\text{FIM}} = -\ln\det\left(\mathcal{I}_{\text{p}}\left(\mathbf{z}_k^{[i]}, \hat{\mathbf{x}}_k^{[\alpha]}, \hat{P}_k^{[\alpha]}, \bar{\mathbf{v}}^{[i]}\right)\right). \tag{5.55}$$

where $\ln\det(\cdot)$ and not simply $\det(\cdot)$ is adopted due to computational advantages [Boyd et al., 2004]. Let $\bar{P}_n^{[\alpha]}; n = k+1, ..., N$ be the covariance of the target' state predicted over the prediction horizon, computed by using the initial covariance $\hat{P}_k^{[\alpha]}$ and the *Target models (A or B)*. Let $\bar{\lambda}_n^{[\alpha]}$ be the length of the major axis of the ellipse representing the predicted uncertainty region of target's position. Note that $\bar{\lambda}_n^{[\alpha]}$ is computed from $\bar{P}_n^{[\alpha]}$ (for *Target model A*, this is illustrated in Fig.5.9). For the purpose of ensuring that the trackers pursue the targets and remain in the vicinity of the latter, as stated in task 2 (see sub-section 5.3.2), we propose the tracking cost

$$J_{\text{Track}} = \sum_{i=1}^{p}\sum_{\alpha=1}^{q}\sum_{n=k+1}^{k+N} -\log(r^* - \bar{d}_n^{[i,\alpha]} - \bar{\lambda}_n^{[\alpha]} - \sigma), \tag{5.56}$$

where $\bar{d}_n^{[i,\alpha]}$ denotes the predicted distance from tracker i to the estimated position of target α, which is computed using (5.12) over the prediction horizon and r^* is the upper bound for the distance from each tracker to each target.

Let $\bar{\mathbf{e}}^{[i]} = [\bar{v}^{[i]}, \bar{r}^{[i]}]$, where $\bar{v}^{[i]}$ and $\bar{r}^{[i]}$ are the computed linear and angular speeds of the i^{th} tracker, respectively over the prediction horizon. As a means to limit the above values, collectively taken as a proxy for tracker energy consumption, we consider the energy-related cost

$$J_{\text{Energy}} = \sum_{i=1}^{p}\sum_{n=k}^{k+N-1} \left\|\bar{\mathbf{e}}_n^{[i]}\right\|_{E_i}^2,$$

where $E_i \in \mathbb{R}^{2\times 2}$ is a diagonal matrix and $E_i \succeq 0$ for all $i \in \mathcal{S}$. Finally, to control the smoothness of the linear speed $\bar{v}^{[i]}$ and the angular speed $\bar{r}^{[i]}$ that can be used as references for autopilots on board the trackers, we define the cost

$$J_{\text{Input}} = \sum_{i=1}^{p}\sum_{n=k}^{k+N-1} \left\|\bar{\mathbf{v}}_n^{[i]}\right\|_{K_i}^2,$$

where $K_i \succeq 0$ for all $i \in \mathcal{S}$.

With the above ingredients, the problem of computing the trackers' inputs over a given time-horizon with the purpose of yielding good target localization and pursuit can be cast in the form of the following optimal control problem:

Definition 5.1. *The optimal control problem, denoted* $\mathcal{OCP}(\mathbf{z}_k^{[i]}, \hat{\mathbf{x}}_k^{[\alpha]}, \hat{P}_k^{[\alpha]}, \bar{\mathbf{v}}^{[i]}(\cdot))$*, is stated as follows:*

$$\min_{\bar{\mathbf{v}}^{[i]}(\cdot); i \in \mathcal{S}} \quad J_{\text{FIM}} + \rho_1 J_{\text{Track}} + \rho_2 J_{\text{Input}} + \rho_3 J_{\text{Energy}}, \tag{5.57}$$

subject to

$$\begin{aligned}
\bar{\mathbf{z}}_{n+1}^{[i]} &= \mathbf{g}_{\text{d}}(\bar{\mathbf{z}}_n^{[i]}, \bar{\mathbf{v}}_n^{[i]}), && i \in \mathcal{S}, && \text{(5.58a)}\\
\bar{\mathbf{p}}_n^{[i]} &= C\bar{\mathbf{z}}_n^{[i]}, && i \in \mathcal{S}, && \text{(5.58b)}\\
\bar{\mathbf{z}}_k^{[i]} &= \mathbf{z}_k^{[i]}, && i \in \mathcal{S}, && \text{(5.58c)}\\
\bar{\mathbf{z}}_n^{[i]} &\in \mathcal{Z}^{[i]}, \quad \bar{\mathbf{v}}_n^{[i]} \in \mathcal{V}^{[i]}, && i \in \mathcal{S}, && \text{(5.58d)}\\
\bar{\mathbf{x}}_{n+1}^{[\alpha]} &= \mathbf{f}(\bar{\mathbf{x}}_n^{[\alpha]}, \bar{\mathbf{u}}_n^{[\alpha]}), && \alpha \in \mathcal{S}_{\text{T}}, && \text{(5.58e)}\\
\bar{\mathbf{x}}_k^{[\alpha]} &= \hat{\mathbf{x}}_k^{[\alpha]} && \alpha \in \mathcal{S}_{\text{T}} && \text{(5.58f)}\\
\bar{d}_n^{[i,\alpha]} &= \left\|\bar{\mathbf{p}}_n^{[i]} - \bar{\mathbf{q}}_n^{[\alpha]}\right\|, && \alpha \in \mathcal{S}_{\text{T}}, && \text{(5.58g)}
\end{aligned}$$

for $n \in \{k, ..., k+N-1\}$

where $\rho_1, \rho_2, \rho_3 \geq 0$ are weighting factors.

In the constraint equations (5.58), the variables with a bar denote predicted variables, to distinguish them from the actual variables, which are without bars. Equations (5.58a)-(5.58d) are associated with the trackers' dynamics and the trackers' state and input constraints, while (5.58e)-(5.58f) are the constraints associated with the targets' models. Notice how the optimal solution for the trackers' inputs depends on the initial conditions (5.58c) and (5.58f), which are updated at very time k. This implies that the trajectory of the trackers need to be re-planned due to the changes in the initial conditions and justifies our approach of using a receding horizon scheme (MPC) to solve the target localization and pursuit problem. In the MPC scheme, the optimal control problem $\mathcal{OCP}(\cdot)$ is repeatedly solved at every discrete sampling instant k. Let $\bar{\mathbf{v}}^{[i]*}(\cdot); i \in \mathcal{S}$ be the optimal solution of the optimal control problem. The MPC control law for each tracker's input is then defined as

$$\mathbf{v}^{[i]}(t) := \bar{\mathbf{v}}_k^{[i]*} \quad \text{for } t \in [k, k+1) \tag{5.59}$$

for all $i \in \mathcal{S}$.

In summary, the proposed receding horizon planing, control and estimation for the target localization and pursuit problem can be implemented using Algorithm 5.1.

Algorithm 5.1 Receding horizon planing, control and estimation strategy for target localization and pursuit

```
**Initialization** ($k = 0$):
For target model A: $\hat{\mathbf{x}}_0^{[\alpha]} = \mathbf{c}_{\mathrm{A},0}^{[\alpha]}, \hat{P}_0^{[\alpha]} = P_{\mathrm{A},0}^{[\alpha]}$
For target model B: $\hat{\mathbf{x}}_0^{[\alpha]} = \mathbf{c}_{\mathrm{B},0}^{[\alpha]}, \hat{P}_0^{[\alpha]} = P_{\mathrm{B},0}^{[\alpha]}$
At every sampled time $k$, repeat the following procedure:
**procedure** PLANING, CONTROL AND ESTIMATION
    Solve the $\mathcal{OCP}(\cdot)$ defined by (5.57) and (5.58).
    Collect all ranges from trackers to targets.
    Run estimators (e.g. EKF) to update $\hat{\mathbf{x}}_k^{[\alpha]}, \hat{P}_k^{[\alpha]}; \alpha \in \mathcal{S}_{\mathrm{T}}$
**return** $\mathbf{v}^{[i]}$ for all $i \in \mathcal{S}$ using the MPC law (5.59) and $\hat{\mathbf{x}}_k^{[\alpha]}, \hat{P}_k^{[\alpha]}$ for all $\alpha \in \mathcal{S}_{\mathrm{T}}$
```

5.7 Simulation examples

In this section, we present and discuss simulation results with the objective of illustrating the performance of the proposed MPC framework for localization and pursuit of underwater targets (AUVs) using surface trackers (ASVs). We consider two situations. In the first situation, an ASV is used for single target localization and pursuit, while in the second situation two ASVs are used for localization and pursuit of two targets. The simulation parameters are given in Table 5.1. Ranges measurement are available every $T_s = 2$s. Furthermore, the ASVs are required to pursue the AUVs and stay inside each of the AUV's vicinity, with $r^* = 100$m. The length of the prediction window is set as $N = 6$. To solve the optimal control problem in the MPC scheme, we use Casadi, an open source optimization tool [Andersson, 2013]. To estimate the target states, an extended Kalman filter (EKF) was employed. The design of the EKF is straightforward, thus we omit its description.

To assess the performance of EKF for the localization task, we define the position estimation and velocity estimation errors as follows:

$$
\begin{aligned}
\text{PosErr} &= \sum_{\alpha=1}^{q} \left\| \mathbf{q}_k^{[\alpha]} - \hat{\mathbf{q}}_k^{[\alpha]} \right\|, \\
\text{VelErr} &= \sum_{\alpha=1}^{q} \left\| \mathbf{u}_k^{[\alpha]} - \hat{\mathbf{u}}_k^{[\alpha]} \right\|,
\end{aligned}
\tag{5.60}
$$

where $\hat{\mathbf{q}}^{[\alpha]}$ and $\hat{\mathbf{u}}^{[\alpha]}$ are estimated position and velocity vectors of the targets obtained from the EKF. The simulation results are shown next.

Table 5.1: Simulation setup

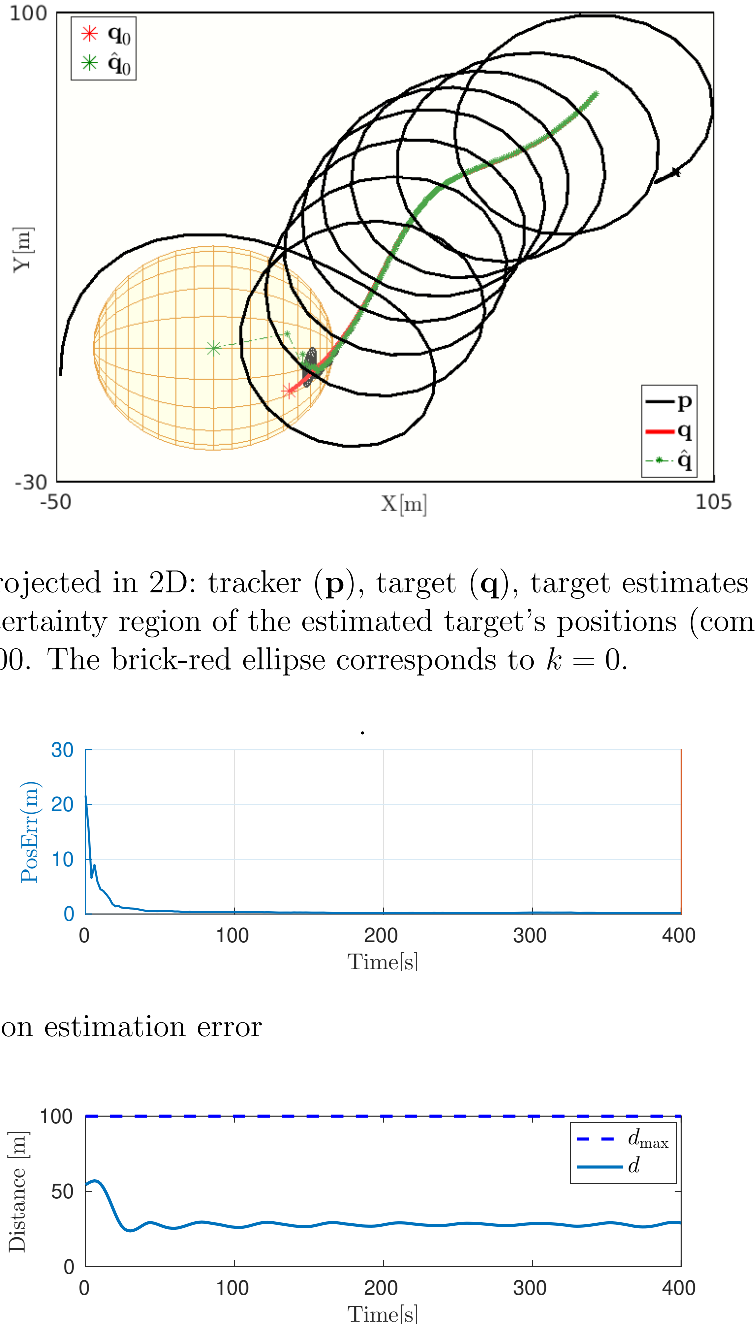
	Parameters	
Trackers (ASVs)	Depths	$\bar{z}^{[1]} = 0\text{m}; \bar{z}^{[2]} = 0\text{m}$
	Vel. constraints	$v^{[i]} \in [0, 4]\text{m/s}, r^{[i]} \in [\text{-}0.2, 0.2]\text{rad/s}$
	Acc. constraints	$a_v^{[i]} \in [\text{-}0.1, 0.1]\text{m/s}^2$, $a_r^{[i]} \in [\text{-}0.01, 0.01]\text{rad/s}^2$
		for $i \in \{1, 2\}$
Targets (AUVs)	Depth	$\bar{z}_{\text{T}}^{[1]} = 5\text{m}; \bar{z}_{\text{T}}^{[2]} = 8\text{m}$
	Velocity vector	$\mathbf{u}^{[\alpha]} = \begin{bmatrix} 0.2 + 0.1\cos(0.1x_{\text{T}}^{[\alpha]}) \\ 0.2 + 0.1\sin(0.1x_{\text{T}}^{[\alpha]}) \end{bmatrix} \text{m/s}$
		for $\alpha = \{1, 2\}$
	Initial positions	$\mathbf{q}_0^{[1]} = [5, \text{-}5]^{\text{T}}, \mathbf{q}_0^{[2]} = [0, 0]^{\text{T}}\text{m}$
RMN[a]	Standard deviation	$\sigma = 0.5\text{m}$

[a] Range measurement noise

5.7.1 Simulation 1: single tracker - single target (p=1,q=1)

In this case, we use one ASV to localize and pursue a single target, that is, $p = q = 1$ and we take the values corresponding to $i = \alpha = 1$ in Table 5.1. For the MPC scheme, the weighting parameters are set as $\rho_1 = 0.01$ and $\rho_2 = \rho_3 = 0$. The EKF is initialized with the prior information of the target given by Table 5.2.

The performance of the MPC scheme for target localization and pursuit under with *Scenarios A* and *B* are plotted in Fig.5.10 and Fig.5.11, respectively. It can be clearly seen



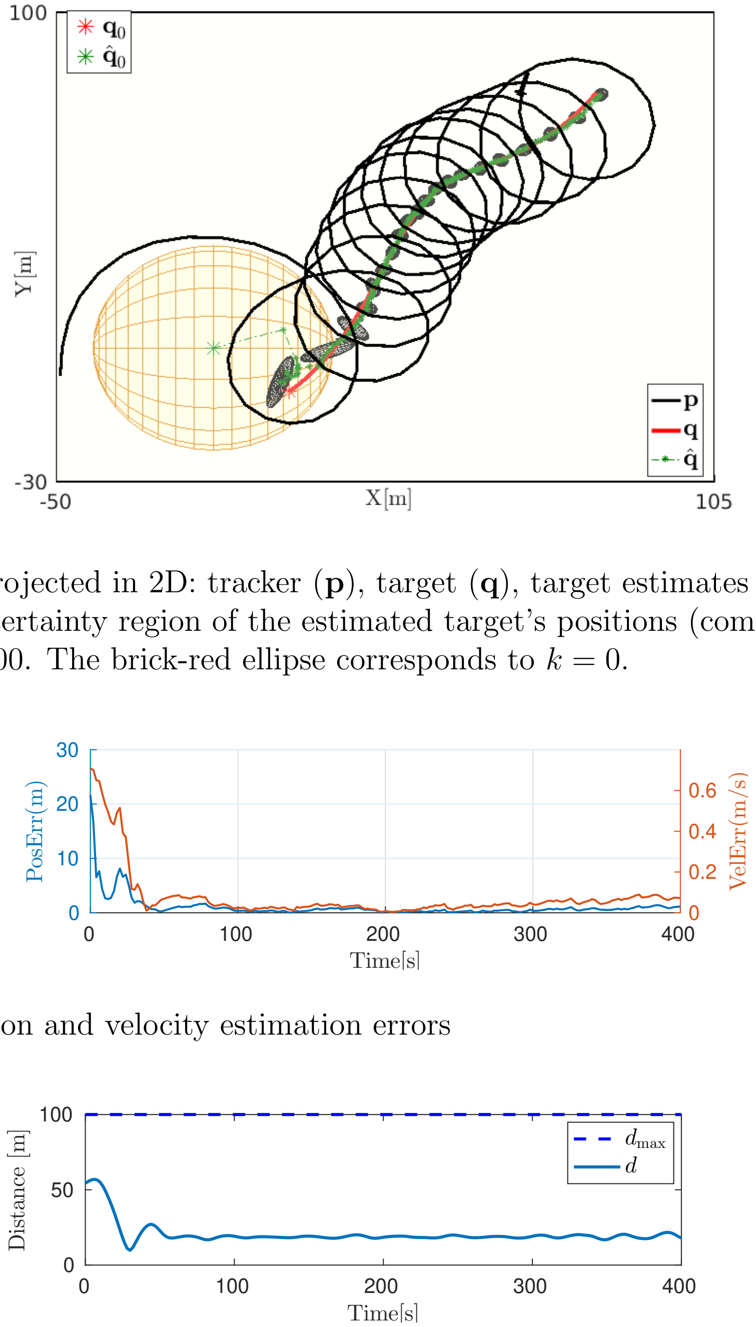
a) Trajectories projected in 2D: tracker ($\mathbf{p}$), target ($\mathbf{q}$), target estimates ($\hat{\mathbf{q}}$). Ellipses represent the uncertainty region of the estimated target's positions (computed from $\hat{P}_k$) at $k = 0, 10, ..., 400$. The brick-red ellipse corresponds to $k = 0$.

b) Target's position estimation error

c) Distance from the tracker to the target

Figure 5.10: Single tracker-single target for the case where the target's velocity vector is known (*Target model A*).

from the figures that the proposed MPC scheme performs well in this simulation set-up. That is, the ASV's trajectories generate "sufficiently rich" range information to estimate



a) Trajectories projected in 2D: tracker ($\mathbf{p}$), target ($\mathbf{q}$), target estimates ($\hat{\mathbf{q}}$). Ellipses represent the uncertainty region of the estimated target's positions (computed from $\hat{P}_k$) at $k = 0, 10, ..., 400$. The brick-red ellipse corresponds to $k = 0$.

b) Target's position and velocity estimation errors

c) Distance from the tracker to the target

Figure 5.11: Single tracker-single target for the case where the target's velocity vector is unknown (*Target model B*).

the target's state. This can be verified by observing Fig.5.10 (a,b) and Fig.5.11 (a,b), where it is evident that the target's state estimation errors converge to a small neigh-

Table 5.2: Prior information on the initial target's sate (Gaussian PDF)

	Scenario A[a]	**Scenario B**[b]
Mean	$\mathbf{c}^{[1]}_{\mathrm{A},0} = [\text{-}25, \text{-}20]^{\mathrm{T}}$	$\mathbf{c}^{[1]}_{\mathrm{B},0} = [\text{-}25, \text{-}20, \text{-}0, 2, 0, 75]^{\mathrm{T}}$
Cov.	$P^{[1]}_{\mathrm{A},0} = \mathrm{Diag}(200, 200)$	$P^{[1]}_{\mathrm{B},0} = \mathrm{Diag}(200, 200, 0.5, 0.5)$

[a] See (5.8)
[b] See (5.11)

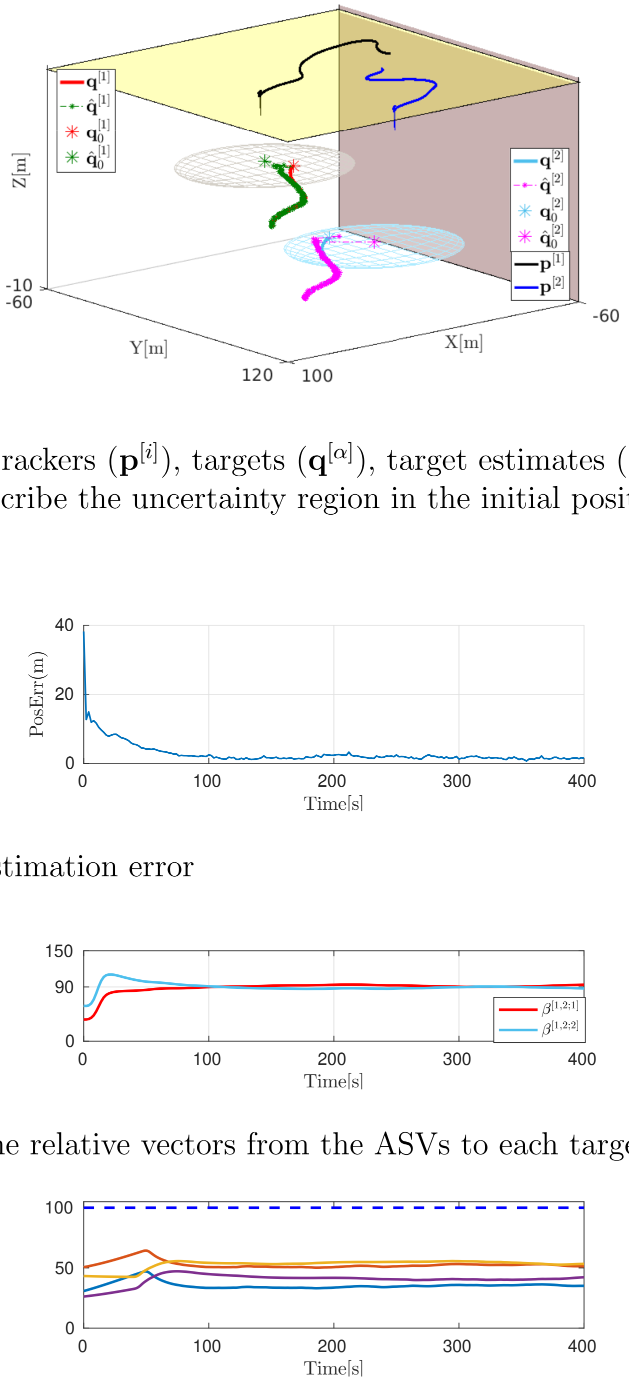
borhood of zero quickly. Comparing Fig.5.10(a,b) with Fig.5.11(a,b) it can be seen that in Scenario A, where the target's velocity vector is known, the estimated target's position converges to the small neighborhood of target's position faster with higher accuracy. Fig.5.10(c) and Fig.5.11(c) show that the distances from the tracker to the target are kept below r^*, implying that the pursuit task is also fulfilled.

Fig.5.10(a) and Fig.5.11(a) also show that, regardless of the target model A or B, the ASV follows and encircles the target's uncertainty regions, which are represented by ellipses in the figures. These trajectories are similar to the ideal trajectories stated in Proposition 5.1 that maximize the range information. Recall that the analysis in Proposition 5.1 neglected the tracker's constraints and dynamics and the target state is considered to be deterministic. Nevertheless, the analysis provides an intuitive understanding of the trajectories obtained in this simulation when the tracker's constraints and dynamics, and the uncertainty of the target are taken explicitly into account.

5.7.2 Simulation 2: multiple trackers - multiple targets (p=2,q=2)

In this simulation, two trackers (ASVs) are deployed to localize and track two targets. For the MPC scheme, the weighting parameters are set as $\rho_1 = 0.01, \rho_2 = 0, \rho_3 = 1$, and $E_i = \mathrm{Diag}(0.001, 0.01)$ for $i = \{1, 2\}$. In this simulation, prior information about the initial targets state is used to construct the predicted FIM.

The performance of the MPC scheme for target localization and pursuit under *Scenarios A* and *B* are plotted in Fig.5.12 and Fig.5.13, respectively. It can be clearly seen in Fig.5.12(a,b) and Fig.5.13(a,b) that the estimated targets' states quickly converge to the targets' states in both scenarios. It is also interesting to observe from Fig.5.12(c) and Fig.5.13(c) that the angles between the relative position vectors from the ASVs to each target converge to 90 degree, thus recovering the behavior predicted in Proposition 5.4 for trajectories that maximize range-information.



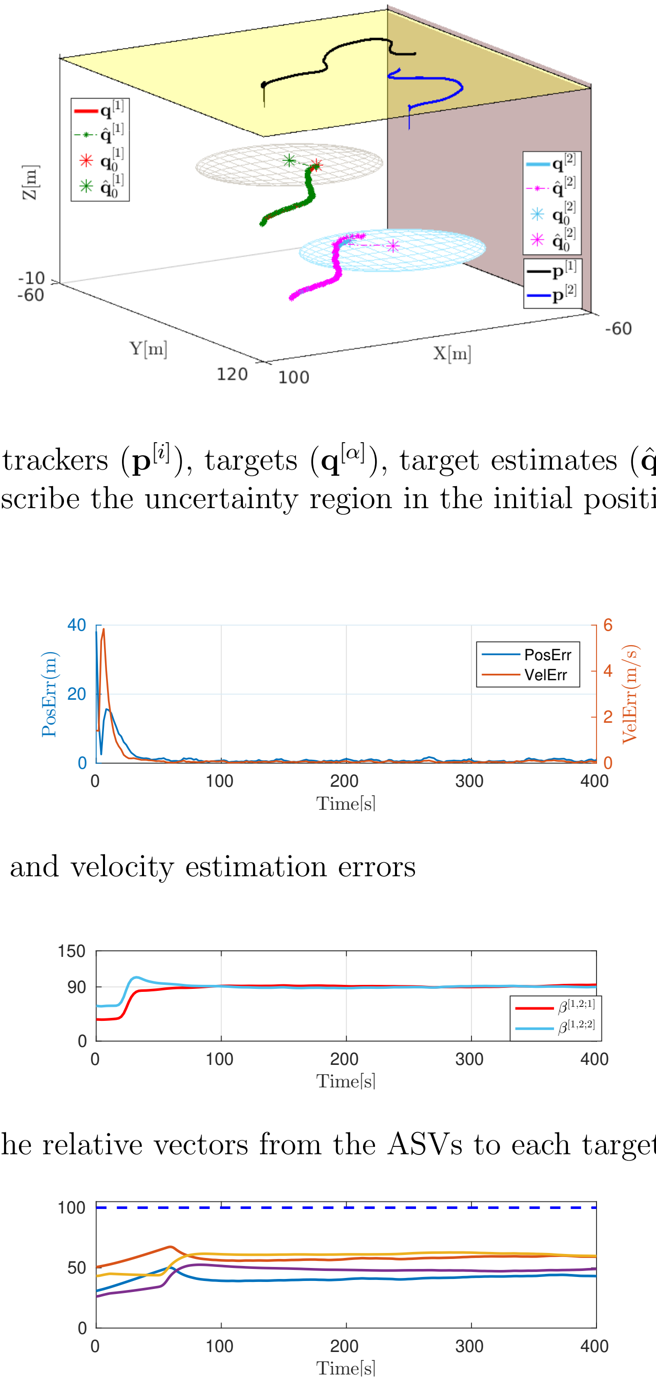
a) 3D-Trajectories: trackers ($\mathbf{p}^{[i]}$), targets ($\mathbf{q}^{[\alpha]}$), target estimates ($\hat{\mathbf{q}}^{[\alpha]}$); $i, \alpha = 1, 2$. Dark and cyan ellipses describe the uncertainty region in the initial positions of target 1 and 2, respectively.

b) Target position estimation error

c) Angles between the relative vectors from the ASVs to each target (in degrees)

d) Distances $d^{[i,\alpha]}; i, \alpha = 1, 2$ from the ASVs to the targets (in m). Dash-blue is r^*.

Figure 5.12: Two trackers and two targets for the case where the targets' velocity vector is known (*Target model A*).



a) 3D-Trajectories: trackers ($\mathbf{p}^{[i]}$), targets ($\mathbf{q}^{[\alpha]}$), target estimates ($\hat{\mathbf{q}}^{[\alpha]}$); $i, \alpha = 1, 2$. Dark and cyan ellipses describe the uncertainty region in the initial positions of target 1 and 2, respectively.

b) Targets' position and velocity estimation errors

c) Angles between the relative vectors from the ASVs to each target (in degrees)

d) Distances $d^{[i,\alpha]}; i, \alpha = 1, 2$ from the ASVs to the targets (in m). Dash-blue is r^*.

Figure 5.13: Two trackers and two targets for the case where the targets' velocity vector is unknown (*Target model B*).

Fig.5.12(d) and Fig.5.13(d) show that the distances from the ASVs to the targets are kept smaller than $r^* = 100$m, thus implying that the pursuit task is fulfilled in both scenarios. Finally, we point out that unlike the case where a single ASV is used, the trajectories of the ASVs are far less demanding in terms of the types of maneuvers executed.

5.8 Conclusions

We proposed an optimization-based approach to the problem of multiple target localization and pursuit using measurements of the ranges between the trackers and the targets. The underlying idea of the proposed method is to find optimal trajectories for the trackers that maximize the range-related information embodied in an appropriately defined Bayesian FIM associated with the problem of target state estimation. Analytically, we showed that for the ideal case where the trackers' motion constraints are neglected and there is no prior information on the initial target's state, there exists an ideal relative geometry of the trackers and the targets for which the range information acquired is maximal. This geometry lends itself to a simple intuitive interpretation. To deal with practical constraints and to consider the fact that the targets' states are random and only estimated on-line, we proposed an MPC framework for optimal tracker motion generation with a view to maximizing the predicted range information for target localization, while taking explicitly into account the trackers' motion constraints, the prior knowledge of the targets' states, and the requirement that the pursuing trackers remain in the vicinities of the targets. By defining appropriate cost and constraints, the MPC scheme is also capable of tackling more challenging cases such as: i) the trackers must avoid obstacles and ii) collision avoidance between the trackers. Future work will aim at decentralizing the MPC scheme, and implementing it in a distributed manner, making the proposed method more scalable for the case where a larger numbers of trackers may be used.

5.9 Proofs

The following lemmas will be used in the proof:

Lemma 5.2. *Let $U, V \in \mathbb{R}^{n\times n}$ and $U, V \succeq \mathbf{0}$. Then,* $\det(U) + \det(V) \leq \det(U+V)$.

The lemma is implied by Minkowski's determinant inequality (see [Horn and Johnson, 2012], p. 510).

Lemma 5.3. *If the trackers and the targets operate at the same depth, that is, $\bar{z}^{[i]} = \bar{z}_{\mathrm{T}}^{[\alpha]}$ for all $i \in \mathcal{S}, \alpha \in \mathcal{S}_{\mathrm{T}}$, then $a_n^{[i,\alpha]} = \cos(\gamma_n^{[i,\alpha]})$ and $b_n^{[i,\alpha]} = \sin(\gamma_n^{[i,\alpha]})$ for all $i \in \mathcal{S}, \alpha \in \mathcal{S}_{\mathrm{T}}$ and $n \in \{1, ..., k\}$.*

The lemma follows directly from (5.12), (5.38) and (5.43).

5.9.1 Proof of Theorem 5.1

i) We first prove the result for *Scenario A*. In this case, $i = p = 1$ and $\alpha = q = 1$, thus (5.39) can be rewritten as

$$\mathcal{I}_{\mathrm{A},k} = \frac{1}{\sigma^2}\begin{bmatrix} \|\mathbf{a}\|^2 & \mathbf{a}^{\mathrm{T}}\mathbf{b} \\ \mathbf{a}^{\mathrm{T}}\mathbf{b} & \|\mathbf{b}\|^2 \end{bmatrix}. \tag{5.61}$$

Recall that in this simple case we dropped the superscript (subscript) α in (5.39) for simplicity in notation. We define $z_n = (\bar{z} - \bar{z}_{\mathrm{T}})^2/d_n^2$ for all $n \in \{1, ..., k\}$. It follows from (5.42) and Assumption 3 that

$$z_n \geq c \tag{5.62}$$

for all n. Let

$$\mathbf{z} = [z_1, ..., z_k]^{\mathrm{T}} \in \mathbb{R}^k \tag{5.63}$$

and also define

$$Z_{\mathrm{A}} = \sigma^{-2}\mathrm{Diag}(\|\mathbf{z}\|^2/2, \|\mathbf{z}\|^2/2). \tag{5.64}$$

Because of (5.62), it can be seen that

$$\det(Z_{\mathrm{A}}) \geq \det(Z_{\mathrm{A}}^*) \tag{5.65}$$

for all $\mathbf{z}$, where $Z_{\text{A}}^* \triangleq c\sigma^{-2}I_2k/2 = c\sigma^{-2}\mathcal{I}_{\text{A}}^o$. The equality holds when $Z_{\text{A}} = Z_{\text{A}}^*$, that is, when $z_n = c$ for all $n \in \{1, ..., k\}$. We now consider the matrix

$$X \triangleq \mathcal{I}_{\text{A},k} + Z_A = \frac{1}{\sigma^2}\begin{bmatrix} \|\mathbf{a}\|^2 + \|\mathbf{z}\|^2/2 & \mathbf{a}^{\text{T}}\mathbf{b} \\ \mathbf{a}^{\text{T}}\mathbf{b} & \|\mathbf{b}\|^2 + \|\mathbf{z}\|^2/2 \end{bmatrix}. \tag{5.66}$$

By definition, X is symmetric and has a constant trace, that is, $\text{Tr}(X) = \sigma^{-2}(\|\mathbf{a}\|^2 + \|\mathbf{b}\|^2 + \|\mathbf{z}\|^2) = \sigma^{-2}k$ for all $\mathbf{a}, \mathbf{b}$ and $\mathbf{z}$. Using Theorem 1.2 in [Popescu et al., 2004] it follows that

$$\det(X) \leq \det(X^*), \tag{5.67}$$

for all $\mathbf{a}, \mathbf{b}$ and $\mathbf{z}$, where $X^* \triangleq \sigma^{-2}\text{Diag}(k/2, k/2) = \sigma^{-2}\mathcal{I}_{\text{A}}^o$. The equality holds when $X = X^*$. Because $\mathcal{I}_{\text{A},k}, Z_{\text{A}} \succeq 0$, it follows from Lemma 1 and (5.65)-(5.67) that $\det(\mathcal{I}_{\text{A},k}) \leq \det(X) - \det(Z_{\text{A}}) \leq \det(X^*) - \det(Z_{\text{A}}^*)$. The equality holds when $X = X^*$ and $Z = Z_{\text{A}}^*$. In other words, $\mathcal{I}_{\text{A},k}$ is maximal when $\mathcal{I}_{\text{A},k} = X^* - Z_{\text{A}}^* = (1-k)\sigma^{-2}\mathcal{I}_{\text{A}}^o = \mathcal{I}_{11\text{A}}$. This concludes the proof for *Scenario A*.

ii) We now present the proof for *Scenario B*. To this end, define

$$Z = \begin{bmatrix} Z_{\text{A}} & Z_{\text{B}} \\ Z_{\text{B}} & Z_{\text{C}} \end{bmatrix} \in \mathbb{R}^{4\times 4}, \tag{5.68}$$

where Z_{A} is given by (5.64) and

$$\begin{aligned} Z_{\text{B}} &= -\sigma^{-2}\text{Diag}(\|\mathbf{z}\|_{\text{D}_1}^2/2, \|\mathbf{z}\|_{\text{D}_1}^2/2), \\ Z_{\text{C}} &= \sigma^{-2}\text{Diag}(\|\mathbf{z}\|_{\text{D}_2}^2/2, \|\mathbf{z}\|_{\text{D}_2}^2/2), \end{aligned} \tag{5.69}$$

where $\mathbf{z}$ is given by (5.63). We now consider the matrix

$$Y \triangleq \mathcal{I}_{\text{B},k} + Z = \begin{bmatrix} A + Z_{\text{A}} & B + Z_{\text{B}} \\ B + Z_{\text{B}} & C + Z_{\text{C}} \end{bmatrix} \triangleq \begin{bmatrix} Y_{\text{A}} & Y_{\text{B}} \\ Y_{\text{B}} & Y_{\text{C}} \end{bmatrix},$$

where A, B, C are given by (5.41). To show that $\det(\mathcal{I}_{\text{B},k})$ is maximal when $\mathcal{I}_{\text{B},k} = \mathcal{I}_{11\text{B}}$, we suppose that the following hypotheses hold true (this will be shown later).

- *Hypothesis 1*: $\det(Z) \geq \det(Z^*)$ for all $\mathbf{z}$, where $Z^* = c\sigma^{-2}\mathcal{I}_{\text{B}}^o$. The equality holds when $Z = Z^*$.
- *Hypothesis 2*: $\det(Y) \leq \det(Y^*)$ for all $\mathbf{a}, \mathbf{b}$ and $\mathbf{z}$, where $Y^* = \sigma^{-2}\mathcal{I}_{\text{B}}^o$. The equality holds when $Y = Y^*$.

Using the hypotheses and arguments similar to those presented for *Scenario A*, we can show that $\det(\mathcal{I}_{\mathrm{B},k})$ is maximal when $\mathcal{I}_{\mathrm{B},k} = Y^* - Z^* = (1-c)\sigma^{-2}\mathcal{I}^o_{\mathrm{B}} = \mathcal{I}_{11\mathrm{B}}$. To complete the proof for *Scenario B* we now prove the hypotheses as follows.

Proof of Hypothesis 1. If $\bar{z} = \bar{z}_{\mathrm{T}}$, it follows that $Z = Z^* = 0$. Thus, *Hypothesis 1* holds trivially. Now consider the case $\bar{z} \neq \bar{z}_{\mathrm{T}}$. This implies that $Z_{\mathrm{A}} \succ 0$. Using the Schur's complement of matrix Z yields

$$\det(Z) = \det(Z_{\mathrm{A}})\det(Z_{\mathrm{C}} - Z_{\mathrm{B}}Z_{\mathrm{A}}^{-1}Z_{\mathrm{B}}). \tag{5.70}$$

From (5.69), we obtain

$$Z_{\mathrm{C}} - Z_{\mathrm{B}}Z_{\mathrm{A}}^{-1}Z_{\mathrm{B}} = \sigma^{-2}\mathrm{Diag}(\lambda/2, \lambda/2), \tag{5.71}$$

where $\lambda = \|\mathbf{z}\|^2_{D_2} - (\|\mathbf{z}\|^2_{D_1})^2/\|\mathbf{z}\|^2$. Expanding λ, we have

$$\begin{aligned}\lambda\|\mathbf{z}\|^2 &= \left(\sum_{n=1}^{k} z_n^2\tau_n^2\right)\left(\sum_{n=1}^{k} z_n^2\right) - \left(\sum_{n=1}^{k} z_n^2\tau_n\right)^2\\ &= \underbrace{\frac{1}{2}\sum_{i,j=1}^{k} z_i^2 z_j^2(\tau_i-\tau_j)^2}_{\triangleq M} > 0.\end{aligned}$$

From (5.64), (5.70), and (5.71) it follows that

$$\det(Z) = \left(\frac{\sigma^{-4}\|\mathbf{z}\|^4}{4}\right)\left(\frac{\sigma^{-4}\lambda^2}{4}\right) = \frac{\sigma^{-8}}{16}M^2 > 0.$$

It can be easily seen that when the sampling interval is fixed, that is τ_n, $n \in \{1, ..., k\}$ is constant, then M is lower bounded by the lower bound of z_n for $n \in \{1, ..., k\}$. Because of (5.62), $\det(Z)$ is smallest when $z_n = c$. Substituting $z_n = c$ for all $n \in \{1, ..., k\}$ in (5.68) we conclude that $\det(Z)$ is smallest when $Z = c\sigma^{-2}\mathcal{I}^o_{\mathrm{B}} = Z^*$.

Proof of Hypothesis 2. Consider the matrix Y. It can be checked that the trace of each block of Y is always constant. Specifically, $\mathrm{Tr}(Y_{\mathrm{A}}) = \mathrm{Tr}(A + Z_{\mathrm{A}}) = \|\mathbf{a}\|^2 + \|\mathbf{b}\|^2 + \|\mathbf{z}\|^2 = k/\sigma^2$. Similarly, $\mathrm{Tr}(Y_{\mathrm{B}}) = \mathrm{Tr}(B + Z_{\mathrm{B}}) = \Delta_1/\sigma^2$ and $\mathrm{Tr}(Y_{\mathrm{C}}) = \mathrm{Tr}(C + Z_{\mathrm{C}}) = \Delta_2/\sigma^2$. Applying Theorem 1.2 in [Popescu et al., 2004] it follows that $\det(Y)$ is maximized if and only if each block is a scaled identity matrix of the form $Y_{\mathrm{A}} = \mathrm{Tr}(Y_{\mathrm{A}})I_2/2 = \sigma^{-2}kI_2/2, Y_{\mathrm{B}} = \mathrm{Tr}(Y_{\mathrm{B}})I_2/2 = \sigma^{-2}\Delta_1 I_2/2$ and $Y_{\mathrm{C}} = \mathrm{Tr}(Y_{\mathrm{C}})I_2/2 = \sigma^{-2}\Delta_2 I_2/2$. Comparing with (5.47), this proves *Hypothesis 2*, thus completing the proof for *Scenario B* and Theorem 5.1. ■

5.9.2 Proof of Proposition 5.1

The proof is done for Scenario A. The proof for Scenario B is identical. Theorem 5.1 implies that the range information is maximal when $\mathcal{I}_{\mathrm{A},k} = \mathcal{I}_{11\mathrm{A}}$. This, together with (5.44), (5.45) and (5.61) implies that

$$\|\mathbf{a}\|^2 = \|\mathbf{b}\|^2 = (1-c)k/2, \tag{5.72a}$$

$$\mathbf{a}^{\mathrm{T}}\mathbf{b} = 0. \tag{5.72b}$$

We now show that the tracker's trajectory stated in Proposition 5.1 satisfies (5.72).
i) We first consider the case where the tracker and the target are at the same depth, that is, $z = z_{\mathrm{T}}$ and therefore $c = 0$. Using Lemma 5.3, (5.72) can be rewritten as

$$\sum_{n=1}^{k} \cos^2(\gamma_n) = \sum_{n=1}^{k} \sin^2(\gamma_n) = k/2, \tag{5.73a}$$

$$\sum_{n=1}^{k} \cos(\gamma_n)\sin(\gamma_n) = 0. \tag{5.73b}$$

For $k \geq 3$, it is well-known that to satisfy (5.73) the displacement between any two successive angles must be equal and satisfies $(\gamma_{n+1} - \gamma_n)k = 2\pi l$, where $l = 1, 2, ...,$ see [Bishop et al., 2010, Moreno-Salinas et al., 2013]. If the tracker can be controlled such that (5.48) is satisfied then at discrete time instants $k = 2\pi l/\omega = Nl$, $\mathcal{I}_{\mathrm{A},k} = \mathcal{I}_{11\mathrm{A}}$, where $l = 1, 2, ...$
ii) We now consider the case where the tracker and the target are at different depths. A solution to (5.72a) is $a_n^2 + b_n^2 = 1 - c$ for all $n \in \{1, ..., k\}$. Further, as shown in the proof of Theorem 5.1, in order to obtain the maximal range information, $d_n = d_{\max}$ for all $n \in \{1, ..., k\}$. Hence, it follows from (5.38) that the trajectory of the tracker must satisfy $(x_n - x_{\mathrm{T},n})^2 + (y_n - y_{\mathrm{T},n})^2 = d_{\max}^2 - (z - z_{\mathrm{T}})^2$ for all $n \in \{1, ..., k\}$. This implies that the tracker must encircle the target with radius $r \triangleq \sqrt{d_{\max}^2 - (z - z_{\mathrm{T}})^2}$. This completes the proof. ■

5.9.3 Proof of Theorem 5.2

i) We first consider *Scenario A*. To this end, let

$$z_n^{[i,\alpha]} = (\bar{z}^{[i]} - \bar{z}_{\mathrm{T}}^{[\alpha]})^2/(d_n^{[i,\alpha]})^2 \tag{5.74}$$

and define vector $\mathbf{z}^{[i,\alpha]} = [z_1^{[i,\alpha]}, ..., z_k^{[i,\alpha]}]^{\mathrm{T}} \in \mathbb{R}^k$. Define also the matrix

$$Z_\alpha = \sum_{i=1}^{p} \frac{1}{\sigma^2} \begin{bmatrix} \left\|\mathbf{z}^{[i,\alpha]}\right\|^2/2 & 0 \\ 0 & \left\|\mathbf{z}^{[i,\alpha]}\right\|^2/2 \end{bmatrix}.$$

Consider now the matrix $F \triangleq \mathcal{I}_{\mathrm{A},k}^{[\alpha]} + Z_\alpha$, where $\mathcal{I}_{\mathrm{A},k}^{[\alpha]}$ given by (5.39). It can be checked that F has a constant trace, that is, $\mathrm{Tr}(F) = \sum_{i=1}^{p}(\|\mathbf{a}_{i,\alpha}\|^2 + \|\mathbf{b}_{\mathbf{i},\alpha}\|^2 + \left\|\mathbf{z}^{[i,\alpha]}\right\|^2) = \sum_{i=1}^{p}\sigma^{-2}k = p\sigma^{-2}k$. Using Theorem 1.2 in [Popescu et al., 2004], we conclude that $\det(F)$ is maximal when $F = F^* \triangleq p\sigma^{-2}kI_2/2 = p\sigma^{-2}\mathcal{I}_{\mathrm{A}}^{o}$. Furthermore, because $z_n^{[i,\alpha]} \geq c^{[i,\alpha]}$ for all $d_n^{[i,\alpha]}$ it is obvious that $\det(Z_\alpha)$ is smallest when $Z_\alpha{=}Z_\alpha^* \triangleq \sum_{i=1}^{p} c^{[i,\alpha]}\sigma^{-2}\mathcal{I}_{\mathrm{A}}^{o}$. Similar to the proof of Theorem 1 for *Scenario A*, we conclude that $\det(\mathcal{I}_{\mathrm{A},k}^{[\alpha]})$ is maximal when $\mathcal{I}_{\mathrm{A},k}^{[\alpha]} = F^* - Z_\alpha^* = \bar{\mathcal{I}}_{\mathrm{A},k}^{[\alpha]}$. This concludes the proof for *Scenario A*.
ii) The proof for *Scenario B* follows using the methodology adopted in the proof for *Scenario A*. ■

5.9.4 Proof of Proposition 5.2

We prove this proposition by showing that the conditions on the trackers' trajectories given in Proposition 5.2 yield the optimal range information matrices introduced in Theorem 5.2 for the case of two trackers ($p = 2$) and a single target. The proof is done for *Scenario A*. The methodology adopted carrying over to the proof of *Scenario B*.
Theorem 5.2 implies that in order to maximize the range information, one must have $\mathcal{I}_{\mathrm{A},k}^{[\alpha]} = \bar{\mathcal{I}}_{\mathrm{A},k}^{[\alpha]}$. From (5.39) and (5.49), this implies that

$$\sum_{i=1}^{2} \frac{1}{\sigma^2} \begin{bmatrix} \|\mathbf{a}_{i,\alpha}\|^2 & \mathbf{a}_{i,\alpha}^{\mathrm{T}}\mathbf{b}_{\mathbf{i},\alpha} \\ \mathbf{a}_{i,\alpha}^{\mathrm{T}}\mathbf{b}_{\mathbf{i},\alpha} & \|\mathbf{b}_{\mathbf{i},\alpha}\|^2 \end{bmatrix} = \left(2 - \sum_{i=1}^{2} c^{[i,\alpha]}\right)\sigma^{-2}\mathcal{I}_{\mathrm{A}}^{o}. \tag{5.75}$$

Equation (5.75) is equivalent to

$$\sum_{i=1}^{2} \left\|\mathbf{a}_{i,\alpha}^{2}\right\| = \sum_{i=1}^{2} \left\|\mathbf{b}_{i,\alpha}\right\|^2 = (2-\sum_{i=1}^{2} c^{[i,\alpha]})k/2, \tag{5.76a}$$

$$\sum_{i=1}^{2} \mathbf{a}_{i,\alpha}^{\mathrm{T}} \mathbf{b}_{i,\alpha} = 0 \tag{5.76b}$$

i) We now consider the case when the trackers and the target are at the same depth, that is, $\bar{z}^{[i]} = \bar{z}^{[\alpha]}$ for all $i \in \{1, 2\}$. Using Lemma 5.3, (5.76b) can be rewritten as

$$\sum_{n=1}^{k} \cos(\gamma_n^{[1,\alpha]}) \sin(\gamma_n^{[1,\alpha]}) + \cos(\gamma_n^{[2,\alpha]}) \sin(\gamma_n^{[2,\alpha]}) = 0 \tag{5.77}$$

for all $n \in \{1, ..., k\}$. It can be easily checked that $\beta_n^{[1,2;\alpha]} \triangleq \gamma_n^{[1,\alpha]} - \gamma_n^{[2,\alpha]} = \pi/2 + l\pi$ for all $n \in \{1, ..., k\}$ and $l \in \mathbb{Z}$ satisfies (5.77). This concludes the case where the trackers and the target are at the same depth .

ii) We now consider the case when the trackers and the target are at different depths. A solution to (5.76a) is $(a_n^{[i,\alpha]})^2 + (b_n^{[i,\alpha]})^2 = 1 - c^{[i,\alpha]}$ for all $n \in \{1, ..., k\}$. Further, as shown in the proof of Theorem 5.2, $d_n^{[i,\alpha]} = d_{\max}$ for all $i \in \mathcal{S}$ and $n \in \{1, ..., k\}$ in order to obtain the maximal range information. Hence, it follows from (5.38) that the trajectory of the trackers must satisfy $\left\|\mathbf{p}_n^{[i]} - \mathbf{q}_n^{[\alpha]}\right\| = \sqrt{d_{\max}^2 - (z^{[i]} - z_{\mathrm{T}}^{[\alpha]})^2}$ for all $i \in \{1, 2\}$ and $n \in \{1, ..., k\}$. This completes the proof. ■

5.9.5 Proof of Proposition 5.3

i) For the cases when the trackers and the targets have the same depth, the proof can be found in Proposition 3 in [Bishop et al., 2010].

ii) If the depths are different, the proof can be done analogously. ■

5.9.6 Proof of Theorem 5.3

We start the proof for *Scenario A*, then followed for *Scenario B*.

i) For *Scenario A*, from (5.33), it follows that $\det(\mathcal{I}_{\mathrm{A},k}) = \prod_{\alpha=1}^{q} \det(\mathcal{I}_{\mathrm{A},k}^{[\alpha]}))$. Hence, $\det(\mathcal{I}_{\mathrm{A},k})$ is maximal when each $\det(\mathcal{I}_{\mathrm{A},k}^{[\alpha]})); \alpha \in \{1, ..., q\}$ is maximal. In Theorem 5.2, it was shown that for each α, $\det(\mathcal{I}_{\mathrm{A},k}^{[\alpha]})$ is maximal when $\mathcal{I}_{\mathrm{A},k}^{[\alpha]} = \bar{\mathcal{I}}_{\mathrm{A},k}^{[\alpha]}$ for all $\alpha \in \{1, ..., q\}$.

This implies that $\det(\mathcal{I}_{\text{A},k})$ is maximal when $\mathcal{I}_{\text{A},k} = \text{Diag}(\bar{\mathcal{I}}^{[1]}_{\text{A},k}, ..., \bar{\mathcal{I}}^{[q]}_{\text{A},k})$. This concludes the proof for *Scenario A*.

ii) The proof for *Scenario B* follows similar arguments. ■

6

A distributed estimation and control strategy for range-based cooperative SLAP

Contents

In the previous chapter, we proposed a solution to the range-based SLAP problem that builds upon the idea of maximizing range-related information for estimation of the targets' state using an MPC like strategy. This approach is "optimal" in the sense that it optimizes trackers' trajectories to acquire maximal range-related information for the estimating the targets' states. The approach is also "universal" because it can be extended easily to integrate other tasks such as targets' pursuit, obstacle avoidance, and even collision avoidance among the trackers. Furthermore, with the MPC strategy the trackers' constraints on their inputs (velocities, accelerations) are taken into account explicitly in the process of planning trackers' trajectories. However, as any optimization based control method, the main drawback of this approach is on its requirement in solving an on-line nonlinear optimization problem, which demands high computation burden and resource. We recall that in Chapter 5, for the use of multiple trackers, the control and estimation algorithms are implemented in a centralized manner, which might be not efficient in applications where the communications among the trackers are constrained by a network topology. Solving the above problems of Chapter 5 is the main subject of this chapter.

The results obtained in the previous chapter provide a useful understanding of the types of optimal trajectories that the trackers need to perform to acquire maximal range information for the estimation of the targets' states. Exploiting this understanding and the knowledge on observability gained in Chapter 4, this chapter proposes an efficient strategy (in terms of computation and implementation) to the range-based SLAP problem resorting to tools from distributed estimation and control. We first address the case of a single tracker, as a means to introduce the basic techniques upon which the solution for the multiple tracker problem builds. In the single tracker case, at the motion planning level, a trajectory is assigned to the tracker that yields global target observability and maximizes the range-based information available for target estimation purposes. Stated simply, the trajectory consist of the composition of motions along the target's trajectory and along a closed path around the target. Borrowing from concepts of path following, the latter is parameterized by a free spatial parameter (e.g. path length), the dynamics of which can be chosen to shape the approach of the tracker to the closed path. The resulting trajectory can be viewed as a general spatial-temporal (S-T) curve with a hybrid curve parametrization. At the control level, we derive control laws for robust trajectory tracking and show that under mild assumptions on the convergence of the target's state estimation, the tracker converges to and remains in a desired vicinity of the target.

For the case of multiple trackers, we propose an efficient distributed estimation and control (DEC) strategy for the trackers that takes into account explicitly the constraints on the inter-tracker communication network. To this end, a distributed extended Kalman filter (DEKF) and a distributed control law for cooperative S-T curves tracking are developed to cooperatively pursue and localize the target. Using this set-up, all trackers converge to a vicinity of the target while keeping an optimal tracker-target relative geometry that maximizes the range information acquired to estimate the target's state. We then propose event-triggered communication mechanisms for the DEC strategy in which the vehicles only need to communicate with their neighbors when found necessary; thus, reducing number of communications and communication frequency among the vehicles. The stability of the complete closed-loop DEC systems (with and without the ETC mechanisms) are analyzed rigorously and the efficacy of the proposed strategy is illustrated with intensive simulations for 2-D and 3-D cases.

6.1 Literature review

Range-based SLAP using autonomous vehicles is a challenging but interesting topic that has been receiving a widespread interest recent years. We refer the reader to Section 4.1 of Chapter 4 and Section 5.1 of Chapter 5 for a summary on the state of the art of this topic. There are two fundamental coupled problems in the context of range-based SLAP. The first involves the localization task, that is, how plan and control the vehicles, called tracker to perform "exciting" maneuver in order to acquire "sufficiently" rich range information for the estimation of target's state. Most work reported in the literature address with this problem, using classical tools for observability analysis ([Batista et al., 2011, Pillon et al., 2016, Jauffret et al., 2017, Arrichiello et al., 2013, Indiveri et al., 2012, Masmitja et al., 2018, Ristic et al., 2002, Hung and Pascoal, 2020b]), or estimation theory like Fisher information matrix ([Crasta et al., 2018, Hung et al., 2020a]). The second problem relates to the pursuit task, that is, how to control the vehicles such that they converge to a vicinity of the target while holding a desired relative geometric formation with access to range measurements only. Several solutions to this problem are presented in the literature [Chaudhary et al., 2016, Nguyen et al., 2018, Hung et al., 2020a]. For example, a possible solution is to estimate the position of the target and then use this information to design a tracking controller to pursue the target [Nguyen et al., 2018, Hung et al., 2020a]. With this approach, the localization and pursuit tasks are tightly coupled. A solution that does not requires an explicit estimation step is reported in [Chaudhary et al., 2016]. However, this approach requires that the time derivative of the range measurement be available to the tracker.

From a theoretical standpoint, the range-based SLAP problem can be solved with only one tracker or a set of multiple trackers, see some preliminary results on observability analysis in [Hung and Pascoal, 2020b, Crasta et al., 2018, Hung et al., 2020a]. Each scenario has its own advantages and disadvantages and technical challenges. It is obvious that using only one tracker is more cost-effective and easier to implement. However, using multiple trackers potentially enhances both the performance and robustness of the localization and pursuit system. In addition, as shown in [Hung and Pascoal, 2020b, Crasta et al., 2018], using multiple trackers makes the motions of the latter smoother. The intrinsic problem when using multiple trackers is how to make them cooperate in an efficient manner in order to localize and pursue a target. To address this problem, a receding horizon planing, control and estimation framework was proposed in [Hung et al., 2020a] to solve the problem in a centralized manner. However, it might be difficult and inefficient to implement this approach in practice when the trackers are distributed over a large spatial region. A more efficient approach in terms of data exchanged among the trackers consists of using a distributed state estimation framework, where each tracker estimates independently the state of the target using its own measurements and reduced additional information received from the neighboring trackers. Due to the growing interest in wireless sensor networks, the problem of distributed state estimation has received great attention in recent years. This led to the development of many methods to tackle the nonlinear distributed estimation problem based on classical techniques such as extended Kalman filtering [Battistelli and Chisci, 2016], moving horizon estimation [Farina et al., 2012], and particle filtering [Manuel and Bishop, 2014]. For a survey of solutions to this problem the reader is refereed to [Rego et al., 2019b].

Motivated by the above considerations, we propose a systematic approach to solve the problem of range-based SLAP for 2-D and 3-D cases with *i) a single autonomous vehicle and ii) multiple autonomous vehicles.* Specifically, the main contributions include the following:

(i) Inspired by recent results on observability analysis in [Hung and Pascoal, 2020b] and optimal tracker motion planing in [Hung et al., 2020a] in the context of range-based target localization, a trajectory is computed for each tracker to go through. An ideal trajectory consists of the composition of two types of motion: along the target's trajectory and on a spatial path encircling the target. For this reason, the trajectory can be viewed as a spatial-temporal (S-T) curve with a hybrid parameterization. By parameterizing the curves appropriately it can be guaranteed that if the trackers track the desired S-T curves accurately, then the target motions become observable and, at the same time, the trackers converge to an optimal desired relative formation that, in a well defined mathematical sense, renders the range-information acquired "maximal" for target state estimation purposes.

(ii) Two types of trajectory tracking controllers that use the estimated target's state are derived for the trackers to track the desired S-T curves. The controllers are shown to be robust against bounded errors in the estimate of the target's sate.

(iii) For the case of multiple trackers, we propose an efficient distributed estimation and control (DEC) strategy for the trackers to localize and pursue the target in a cooperative manner. The aim of the DEC strategy is twofold: i) to coordinate the trackers along the planed S-T curves in order for them to reach and maintain a desired relative formation and ii) to fuse the information about the target's state estimated by each tracker so that all trackers reach consensus on the estimate of the target's state. Exploiting tools from network sciences and algebraic graph theory, the DEC strategy takes into account explicitly the constraints of the communication network topology established among the trackers, where each tracker is only capable of exchanging information with a sub-set of trackers in the network. The proposed DEC ensures that asymptotically, all trackers reach consensus on the estimates of the target's state and maintain a desired geometrical formation in a vicinity of the target. We then propose event-triggered communication mechanisms for the DEC strategy in which the vehicles only need to communicate with their neighbors when found necessary; thus, reducing number of communications and communication frequency among the vehicles. The proposed strategy requires very limited information to be exchanged among the trackers, thus making it very efficient for practical implementation.

6.2 Notation and algebraic graph theory

6.2.1 Notation

In this chapter, we use the notation defined in Section 2.2.1 of Chapter 2. We also use the following notation. Given vectors $\mathbf{x}_i \in \mathbb{R}^n; i = 1, ..., N$, $\mathrm{col}(\mathbf{x}_1, ..., \mathbf{x}_N) \triangleq [\mathbf{x}_1^{\mathrm{T}}, ..., \mathbf{x}_N^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{n \times N}$. The notation $\mathcal{N}(\boldsymbol{\mu}, P)$ denotes a Gaussian (normal) distribution with mean $\boldsymbol{\mu}$ and covariance matrix P.

6.2.2 Algebraic graph theory

We refer to Section 2.2.2 of Chapter 2 for an introduction to algebraic graph theory and some important results that will be used in this chapter.

6.3 Problem formulation and background results

6.3.1 Problem formulation

Trackers' model

Consider a group of $N (N \geq 1)$ autonomous vehicles, henceforth called trackers, charged with the task of localizing and pursuing a moving target. In what follows, $\{\mathcal{I}\} = \{x_{\mathcal{I}}, y_{\mathcal{I}}, z_{\mathcal{I}}\}$ denotes a inertial frame and $\{\mathcal{B}\}^{[i]} = \{x_{\mathcal{B}}^{[i]}, y_{\mathcal{B}}^{[i]}, z_{\mathcal{B}}^{[i]}\}$ denotes a body frame attached to tracker i; $i \in \mathcal{V}$. For each $i \in \mathcal{V}$, let $\mathbf{p}^{[i]} \in \mathbb{R}^3$ be the inertial position vector of the tracker expressed in $\{\mathcal{I}\}$, and $\mathbf{v}^{[i]} = [v_1^{[i]}, v_2^{[i]}, v_3^{[i]}]^{\mathrm{T}}$ its velocity vector expressed in the body frame $\{\mathcal{B}\}^{[i]}$, consisting of surge ($v_1^{[i]}$), sway ($v_2^{[i]}$), and heave ($v_3^{[i]}$) speed components. The tracker' kinematic model is given by

$$\dot{\mathbf{p}}^{[i]} = R(\boldsymbol{\eta}^{[i]})\mathbf{v}^{[i]}, \tag{6.1}$$

where $R(\boldsymbol{\eta}^{[i]}) \in SO(3)^1$ is the rotation matrix from $\{\mathcal{B}\}^{[i]}$ to $\{\mathcal{I}\}$, locally parameterized by a vector $\boldsymbol{\eta}^{[i]} \triangleq [\phi^{[i]}, \theta^{[i]}, \psi^{[i]}]^{\mathrm{T}}$ that contains the Euler angles of roll ($\phi^{[i]}$), pitch ($\theta^{[i]}$) and yaw ($\psi^{[i]}$). Note that the time derivative of the rotation matrix satisfies

$$\dot{R}(\boldsymbol{\eta}^{[i]}) = R(\boldsymbol{\eta}^{[i]})S(\boldsymbol{\omega}^{[i]}), \tag{6.2}$$

where $\boldsymbol{\omega}^{[i]} = [p^{[i]}, q^{[i]}, r^{[i]}]^{\mathrm{T}}$ is the body-fixed angular velocity vector and S is a skew-symmetric matrix defined as

$$S(\boldsymbol{\omega}^{[i]}) = \begin{bmatrix} 0 & \text{-}r^{[i]} & p^{[i]} \\ r^{[i]} & 0 & \text{-}q^{[i]} \\ \text{-}p^{[i]} & q^{[i]} & 0 \end{bmatrix}. \tag{6.3}$$

[1] $SO(3) \triangleq \{R \in \mathbb{R}^{3\times 3} : RR^{\mathrm{T}} = I_3, \det(R) = 1\}$

In the present chapter, for simplicity of presentation, we consider that the trackers are under-actuated vehicles for which the sway and heave speeds are negligible, i.e. $\mathbf{v}^{[i]} = [v_1^{[i]}, 0, 0]; i \in \mathcal{V}$. The input of each tracker i is therefore defined as

$$\mathbf{u}^{[i]} = [v_1^{[i]}, p^{[i]}, q^{[i]}, r^{[i]}]^{\mathrm{T}} \in \mathbb{R}^4, \tag{6.4}$$

consisting of surge speed $v_1^{[i]}$ and angular velocity $\boldsymbol{\omega}^{[i]}$.

Remark 6.1. *For the* **2-D** *case, e.g. when the trackers are restricted to move in the $x_{\mathcal{I}} - y_{\mathcal{I}}$ horizontal plane, the tracker's model* (6.1) *applies with $\boldsymbol{\eta}^{[i]} = \psi^{[i]}$ and $\boldsymbol{\omega}^{[i]} = r^{[i]}$. In this case, the tracker's control inputs include the surge speed and the yaw rate, i.e. $\mathbf{u}^{[i]} = [v_1^{[i]}, r^{[i]}]^{\mathrm{T}} \in \mathbb{R}^2$ and $S(\boldsymbol{\omega}^{[i]}) = \begin{bmatrix} 0 & \text{-}r^{[i]} \\ r^{[i]} & 0 \end{bmatrix}$. In practice, the tracker's model* (6.1) *is adequate for a wide class of under-actuated marine vehicles of which Medusa and Delfim ([Abreu et al., 2016c]) and Charlie ([Bibuli et al., 2009]) are representative examples.*

Target's model

Let $\mathbf{q}(t) \in \mathbb{R}^3$ be the target's trajectory to be tracked, and $\boldsymbol{v}(t) = \dot{\mathbf{q}}(t)$ be the target velocity vector, both expressed in $\{\mathcal{I}\}$. Even though the target's trajectory is unknown, we assume that the target changes its velocity slowly that its motion can be described by a quasi-steady model of the form

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + \mathbf{w}(t), \tag{6.5}$$

where $\mathbf{x}(t) = [\mathbf{q}^{\mathrm{T}}(t), \boldsymbol{v}^{\mathrm{T}}(t)]^{\mathrm{T}} \in \mathbb{R}^6$ is the state of the target, $\boldsymbol{v}(t)$ is slowly varying, $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, Q_t)$ is a zero mean Gaussian process noise with covariance Q_t and $A = \begin{bmatrix} 0 & I_3 \\ 0 & 0 \end{bmatrix}$. Let $\mathbf{x}_k = [\mathbf{q}_k^{\mathrm{T}}, \boldsymbol{v}_k^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^6$ be the state of the target, at the discrete time $k; k \in \mathbb{N}$. For target state estimation purposes (e.g. using a discrete EKF), (6.5) can be discretized using Euler's method, yielding

$$\mathbf{x}_{k+1} = F\mathbf{x}_k + \mathbf{w}_k, \tag{6.6}$$

where

$$F = \begin{bmatrix} I_3 & T_s I_3 \\ 0_{3\times 3} & I_3 \end{bmatrix}, \tag{6.7}$$

T_s is the sampling period, and $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, Q); Q = T_s^2 Q_t$.

Remark 6.2. *For the 2-D case, the target model* (6.6) *applies with* $\mathbf{x} \in \mathbb{R}^4$ *and* $F = \begin{bmatrix} I_2 & T_s I_2 \\ 0_{2\times 2} & I_2 \end{bmatrix}$.

Range measurement model

Assume that each tracker i is equipped with an acoustic unit that measures its distance to the target. Let also $d_k^{[i]}$ be the true distance between tracker $i; i \in \mathcal{V}$ and the target at discrete time k, defined as

$$d_k^{[i]} = \left\| \mathbf{p}_k^{[i]} - \mathbf{q}_k \right\|, \tag{6.8}$$

where $\mathbf{p}_k^{[i]}$ denotes the position of tracker i at discrete time instant k. Further, let $y_k^{[i]}$ denote the range measurements which, we assume, are corrupted by white Gaussian noise according to the range measurement model

$$y_k^{[i]} = d_k^{[i]} + \eta_k^{[i]}, \tag{6.9}$$

where $\eta_k^{[i]} \sim \mathcal{N}(0, \sigma); i \in \mathcal{V}$ is Gaussian measurement noise. The cooperative target localization and pursuit problem that is the main topic of this chapter can now be formally defined as follows.

Problem 6.1 (Cooperative target localization and pursuit)**.** *Consider a group of* $N (N \geq 1)$ *trackers, with models described by* (6.1) *that are in charge of localizing and pursuing an unknown target whose dynamics are described by* (6.5). *Assume the trackers' inter-communication network is modeled by a digraph* $\mathcal{G}$. *Suppose that the trackers' positions and orientations are known. Assume further that the trackers can measure ranges to the target according to the measurement equation* (6.9). *Let* $\hat{\mathbf{x}}^{[i]}$ *denote an estimate of the target's state* $\mathbf{x}$ *computed by tracker* i. *Design a distributed control law for* $\mathbf{u}^{[i]}; i \in \mathcal{V}$ *and a distributed estimation algorithm for* $\hat{\mathbf{x}}^{[i]}; i \in \mathcal{V}$ *to fulfill the following tasks:*

- *Cooperative pursuit: ensure that asymptotically all trackers stay in a given vicinity of the target, i.e.*

$$\lim_{t\to\infty} \left\| \mathbf{p}^{[i]}(t) - \mathbf{q}(t) \right\| \leq r_{\mathrm{c}}, \tag{6.10}$$

for a given r_{c}.

- *Cooperative localization (estimation): ensure that all estimates* $\hat{\mathbf{x}}^{[i]}$ *of the target's sate reach consensus, that is,*

$$\lim_{k\to\infty} \left\| \hat{\mathbf{x}}_k^{[i]} - \mathbf{x}_k \right\| \leq r_{\mathrm{e}} \tag{6.11}$$

for all $i = 1, \ldots N$, *where* r_{c}, *and* r_{e} *are given positive values.*

Remark 6.3. *The problem formulation above can be easily extended to the cases where the trackers and the target move in different horizontal planes, e.g. when the trackers are at the surface while the target is underwater with a known depth. In this scenario,* (6.10) *implies that the trackers should remain in a desired 2-D region centered at the projection of the target on the surface horizontal plane.*

6.3.2 Background results

In this subsection, we briefly recall some preliminary results on target motion observability and optimal tracker trajectory generation for range-based target localization. The results described will be used to plan the motion for the trackers in the next section.

Result 1: single tracker-single target

Consider the scenario where one tracker is used to localize a target moving with unknown constant velocity i.e. $\boldsymbol{v}(t)$ is constant. As shown in [Hung and Pascoal, 2020b], in a 2-D setting the target's state (position and velocity) is observable, i.e. the target's state is uniquely determined if the trajectory of the tracker belongs to a class of "cycloid-type" trajectories defined by $\mathbf{p}(t) = [p_{\mathrm{x}}(t), p_{\mathrm{y}}(t)]^{\mathrm{T}} = [r_{\mathrm{x}} \sin(\omega t) + c_{\mathrm{x}} t, r_{\mathrm{y}} \cos(\omega t) + c_{\mathrm{y}} t]^{\mathrm{T}}$ with $r_{\mathrm{x}}, r_{\mathrm{y}}, \omega \neq 0$. Furthermore, using the Fisher Information Matrix (FIM) as a means to quantifying the range information available for target estimation, it was shown that a class of optimal trajectories that maximize range information consists of having the tracker encircle the target, see [Hung et al., 2020a, Crasta et al., 2018]. In 3-D, any "helix-type" trajectory defined by $\mathbf{p}(t) = [r_{\mathrm{x}} \sin(\omega_1 t) + c_{\mathrm{x}} t, r_{\mathrm{y}} \cos(\omega_1 t) + c_{\mathrm{y}} t, r_{\mathrm{z}} \sin(\omega_2 t) + c_{\mathrm{z}} t]^{\mathrm{T}}$ with $r_{\mathrm{x}}, r_{\mathrm{y}}, r_{\mathrm{z}}, \omega_1 \neq \omega_2 \neq 0$ guarantees observability of the target's state, see Theorem 2 in [Hung and Pascoal, 2020b].

Result 2: two trackers-single target

Consider the scenario where two trackers are used to localize a target. In this case, the target's state is completely observable at an arbitrarily time t_0 if and only if the columns of matrix $\mathcal{O}(t)$, defined by

$$\mathcal{O}(t) \triangleq [(\mathbf{p}^{[2]}(t) - \mathbf{p}^{[1]}(t))^{\mathrm{T}} \quad (t - t_0)(\mathbf{p}^{[2]}(t) - \mathbf{p}^{[1]}(t))^{\mathrm{T}}], \tag{6.12}$$

are linearly independent on an interval $[t_0, t]$ for some $t > t_0$, see Theorem 5 in [Hung and Pascoal, 2020b]. In the 2-D case, it was shown in [Hung et al., 2020a] that the optimal trajectories for the tracker that maximize the range-information available for estimating the target's state are the ones satisfying

$$(\mathbf{p}_k^{[1]} - \mathbf{q}_k) \perp (\mathbf{p}_k^{[2]} - \mathbf{q}_k) \tag{6.13}$$

for all range measurement time instants k, i.e. the relative position vectors from each tracker to the target are orthogonal, see Proposition 2 in [Hung et al., 2020a]. Note that this tracker-target geometry is optimal for target state estimation for the cases where the target's velocity vector is known, see [Crasta et al., 2018, Hung et al., 2020a].

Result 3: N trackers-single target

If N trackers ($N \geq 3$) are used to localize the target, it was shown in in [Hung et al., 2020a], Proposition 3, for the 2-D case that the optimal tracker trajectories for target state estimation are the ones for which the trackers' positions are distributed uniformly around the target and the angle made by the relative vectors from two adjacent trackers to the target is $2\pi/N$. This result is illustrated in Fig.6.1 for the case of three trackers. In the next section, we will use the aforementioned knowledge to plan optimal trajectories for the trackers and design a cooperative distributed control and estimation strategy to simultaneously localize and pursue the target.

6.4 Target localization and pursuit with one tracker

We start by consider a scenario where only one tracker is used to localize and pursue a target. In this case, for the sake of clarity we drop the superscript $[i]$ in the relevant variables in this section. In what follows we derive a desired trajectory for the tracker,

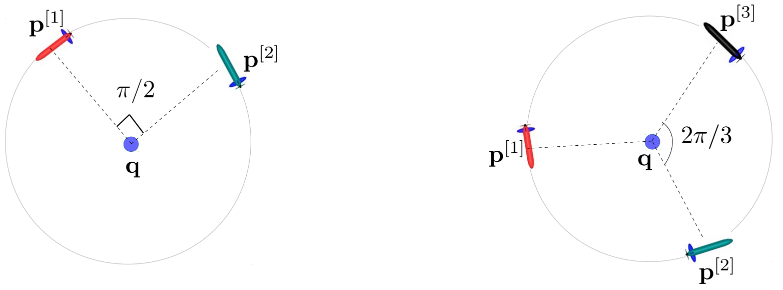


Figure 6.1: Examples of optimal relative tracker-target geometries that maximize the information available to estimate the target's state. (Left: $N = 2$, Right: $N = 3$). Positions: $\mathbf{q}$ (target), $\mathbf{p}^{[i]}$ (trackers).

together with the corresponding trajectory tracking controllers.

6.4.1 A trajectory tracking strategy for target pursuit

For the purpose of illustrating the underlying idea of the proposed method, we assume that the target's position $\mathbf{q}(t)$ and its velocity $\boldsymbol{v}(t)$ are known. This assumption will be relaxed in the next subsection. Let $\mathcal{P} : \gamma \rightarrow \mathbf{r}(\gamma) \in \mathbb{R}^3$ be a path defined in an inertial frame, given by

$$\mathbf{r}(\gamma) = [r_\mathrm{x}\cos(\gamma+\gamma_0), r_\mathrm{y}\sin(\gamma+\gamma_0), r_\mathrm{z}\sin(\frac{\gamma}{c}+\gamma_0)]^\mathrm{T}, \tag{6.14}$$

where γ is a path parameterizing variable (e.g. path length), and $r_\mathrm{x}, r_\mathrm{y}, r_\mathrm{z} \in \mathbb{R}_+, c \in \mathbb{R}\setminus\{0,1\}$ and $\gamma_0 \in \mathbb{R}$ are constant parameters. For the 2D-case, $\mathbf{r}(\gamma) \in \mathbb{R}^2$ is defined as

$$\mathbf{r}(\gamma) = [r_\mathrm{x}\cos(\gamma+\gamma_0), r_\mathrm{y}\sin(\gamma+\gamma_0)]^\mathrm{T}. \tag{6.15}$$

Borrowing the concept of moving path following (MPF) proposed in [Oliveira et al., 2016, Reis et al., 2019] we let $\mathbf{p}_\mathrm{d} : \mathbb{R}\times\mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^3$ be a desired S-T curve that consists of the composition of motions along the target and along a path around the target, defined as

$$\mathbf{p}_\mathrm{d}(t) = \mathbf{r}(\gamma(t)) + \mathbf{q}(t), \tag{6.16}$$

where the spatial part $\mathbf{r}(\gamma)$ is described by (6.14) and (6.15) for the 3-D and 2-D cases, respectively. The underlying idea behind the method proposed for target pursuit is that if the tracker can be controlled s.t. $\mathbf{p} \rightarrow \mathbf{p}_\mathrm{d}$, then it will converge to a vicinity of the target defined by $\mathbf{r}$, i.e. $\|\mathbf{p}-\mathbf{q}\| \rightarrow \|\mathbf{r}(\gamma)\| \overset{(6.14)}{\leq} \sqrt{r_\mathrm{x}^2+r_\mathrm{y}^2+r_\mathrm{z}^2}$ for all γ. For reasons that

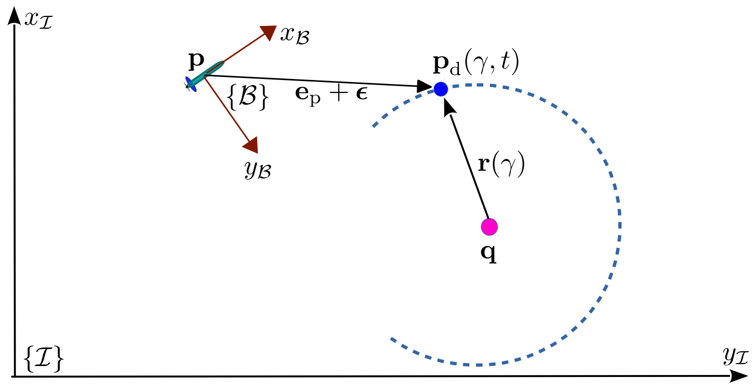


Figure 6.2: Illustration of the proposed methodology in 2-D.

will become clear latter, γ is allowed to be a function of time, with dynamics to be chosen appropriately to shape the approach of the tracker to the desired S-T curve. Furthermore, if the dynamics of γ can be chosen s.t. $\dot{\gamma} \to \omega_{\rm d}$, where

$$\omega_{\rm d} = \bar{\omega}, \tag{6.17}$$

with a constant $\bar{\omega} \neq 0$, then the tracker will encircle the target with angular rate $\bar{\omega}$. For the 2-D case, the S-T curve described by (6.16) that the tracker must track takes the form of the optimal trajectories for range-based target localization, while for 3-D case, it ensures the observability of the target's state, see Section 6.3.2.

Inspired from the work in [Aguiar and Hespanha, 2007] we now derive two tracking controllers for the tracker in order have it pursue and encircle the target. To this end, define

$$\begin{aligned}\mathbf{e}_{\rm p} =& R^{\rm T}(\boldsymbol{\eta})(\mathbf{p}-\mathbf{p}_{\rm d}) - \boldsymbol{\epsilon}\\ \overset{(6.16)}{=}& R^{\rm T}(\boldsymbol{\eta})(\mathbf{p}-\mathbf{r}(\gamma)-\mathbf{q}) - \boldsymbol{\epsilon}\end{aligned} \tag{6.18}$$

as the position error between the tracker and the desired trajectory expressed in the tracker's body frame $\{\mathcal{B}\}$, where $\boldsymbol{\epsilon}$ is an arbitrarily small non-zero vector, see the illustration of this error for the case of 2D in Fig.6.2. The idea behind the use of vector $\boldsymbol{\epsilon}$ was originally introduced in [Aguiar and Hespanha, 2007] and will become clear next. By definition, if $\mathbf{e}_{\rm p}$ converges to zero then the tracker converges to the vicinity of the target centered at the target with radius $\|\mathbf{r}\| + \|\boldsymbol{\epsilon}\|$. Clearly, by proper choice of $r_{\rm x}, r_{\rm y}, r_{\rm z}$, and $\boldsymbol{\epsilon}$ s.t. $\|\mathbf{r}(\gamma)\| + \|\boldsymbol{\epsilon}\| \leq r_c$ for all γ the pursuit task described by (6.10) will be fulfilled. Define also

$$e_\gamma = \dot{\gamma} - \omega_{\rm d} \tag{6.19}$$

as the speed tracking error for the temporal evolution of γ. Let $\mathbf{e} = [\mathbf{e}_\mathrm{p}^\mathrm{T}, e_\gamma]^\mathrm{T}$ be the complete tracking error vector. Our main objective is to derive tracking control laws for $\mathbf{u}$ defined in (6.4) to drive $\mathbf{e}$ to zero. Taking the time derivative of (6.18) and using (6.1),(6.2),(6.3), (6.4) yields the dynamics of $\mathbf{e}_\mathrm{p}$, given as

$$\begin{aligned} \dot{\mathbf{e}}_\mathrm{p} &= -S(\boldsymbol{\omega})\mathbf{e}_\mathrm{p} - S(\boldsymbol{\omega})\boldsymbol{\epsilon} + \mathbf{v} - R^\mathrm{T}(\boldsymbol{\eta})\left(\mathbf{r}'(\gamma)\dot{\gamma} + \boldsymbol{v}\right) \\ &= -S(\boldsymbol{\omega})\mathbf{e}_\mathrm{p} + \Delta\mathbf{u} - R^\mathrm{T}(\boldsymbol{\eta})\left(\mathbf{r}'(\gamma)\dot{\gamma} + \boldsymbol{v}\right) \end{aligned} \tag{6.20}$$

where $\mathbf{r}'(\gamma) \triangleq \frac{\partial \mathbf{r}(\gamma)}{\partial \gamma}$, and

$$\Delta = \begin{bmatrix} 1 & 0 & \text{-}\epsilon_3 & \epsilon_2 \\ 0 & \epsilon_3 & 0 & \text{-}\epsilon_1 \\ 0 & \text{-}\epsilon_2 & \epsilon_1 & 0 \end{bmatrix}, \Delta = \begin{bmatrix} 1 & \epsilon_2 \\ 0 & \text{-}\epsilon_1 \end{bmatrix} \tag{6.21}$$

for the 3-D and the 2-D case, respectively where ϵ_i denotes the i^th component of vector $\boldsymbol{\epsilon}$. From (6.19) the dynamics of the error e_γ are described by

$$\dot{e}_\gamma = \ddot{\gamma}. \tag{6.22}$$

To drive the tracking error $\mathbf{e}$ to zero, the following controllers can be used:

Tracking Controller - Type I

$$\begin{aligned} \mathbf{u} &= \boldsymbol{k}(\mathbf{x}) \\ \dot{\gamma} &= \omega_\mathrm{d}. \end{aligned} \tag{6.23}$$

Tracking Controller - Type II

$$\begin{aligned} \mathbf{u} &= \boldsymbol{k}(\mathbf{x}) \\ \ddot{\gamma} &= -k_\gamma e_\gamma + \mathbf{e}_\mathrm{p}^\mathrm{T} R^\mathrm{T}(\boldsymbol{\eta})\mathbf{r}'(\gamma), \end{aligned} \tag{6.24}$$

with $\boldsymbol{k}(\mathbf{x})$ given by

$$\boldsymbol{k}(\mathbf{x}) = \bar{\Delta}\left(R^\mathrm{T}(\boldsymbol{\eta})\left(\mathbf{r}'(\gamma)\omega_\mathrm{d} + \boldsymbol{v}\right) - K_\mathrm{p}\mathbf{e}_\mathrm{p}\right), \tag{6.25}$$

where $\bar{\Delta} = \Delta^\mathrm{T}(\Delta\Delta^\mathrm{T})^{-1}$, K_p is a positive definite matrix with appropriate dimension, $k_\gamma > 0$, and ω_d is given by (6.17). With the proposed controllers, we obtain the following result.

Lemma 6.1. *Consider the tracking error system described by* (6.20) *and* (6.22)*. Suppose* *that the target's state* $\mathbf{x}$ *(including both the position* $\mathbf{q}$ *and velocity* $\boldsymbol{v}$*) is completely known. Then, both tracking controllers given by* (6.23) *and* (6.24) *guarantee that the tracking error* $\mathbf{e}$ *converges to zero exponentially fast.*

Proof: See section 6.10.1 .

Remark 6.4. *The main difference between Controller-Type I given by* (6.23) *and Controller-Type II given by* (6.24) *is the strategy to control the evolution of the path parameter* γ*. With the first the derivative of* γ *is fixed, whereas with the second the dynamics of* γ *depends on feedback terms from the tracking error. The first is clearly simpler to implement, yet thanks to the feedback, the second has the potential to speed up the convergence of the tracking error to zero.*

Remark 6.5. *In* (6.18)*, imposing* $\boldsymbol{\epsilon}$ *as a non-zero vector is made to ensure that the matrix* $\bar{\Delta}$ *non-singular, thus ensuring that the proposed controllers given by* (6.23) *and* (6.24) *are well-defined.*

6.4.2 Unknown target pursuit

In the previous section, we showed that if the target's state is fully known, then both control laws (6.23) and (6.24) steer the tracker to a vicinity of the target and encircle it with a desired angular speed $\omega_{\mathrm{d}} = \bar{\omega}$. The convergence rate was shown to be exponential. However, in the context of target localization and pursuit, the target's state is unknown, and must be estimated on-line. To address this problem, we suppose that the tracker is equipped with a filter, which is capable of estimating the target's state (both the target's position and velocity vectors). This can be done using the target model given in (6.7) and the range measurement model (6.9), as explained later. Let $\hat{\mathbf{x}} = [\hat{\mathbf{q}}, \hat{\boldsymbol{v}}]$ be the estimate of the target's state $\mathbf{x}$, where $\hat{\mathbf{q}}$ denotes an estimate of the true target position vector $\mathbf{q}$, while $\hat{\boldsymbol{v}}$ denotes the estimate of the true target velocity vector $\boldsymbol{v}$. Define also

$$\hat{\mathbf{e}}_{\mathrm{p}} = R^{\mathrm{T}}(\boldsymbol{\eta})\,(\mathbf{p} - \mathbf{r}(\gamma) - \hat{\mathbf{q}}) - \boldsymbol{\epsilon} \tag{6.26}$$

which is similar to (6.18) but the estimated target position ($\hat{\mathbf{q}}$) is used, instead of the true target position $\mathbf{q}$. The following control laws are proposed for unknown target pursuit.

Tracking Controller - Type I - Unknown Target

$$\begin{aligned}\mathbf{u} &= \boldsymbol{k}(\hat{\mathbf{x}}) \\ \dot{\gamma} &= \omega_{\mathrm{d}}.\end{aligned} \tag{6.27}$$

Tracking Controller - Type II - Unknown Target

$$\begin{aligned}\mathbf{u} &= \boldsymbol{k}(\hat{\mathbf{x}}) \\ \ddot{\gamma} &= -k_\gamma e_\gamma + \hat{\mathbf{e}}_{\mathrm{p}}^{\mathrm{T}} R^{\mathrm{T}}(\boldsymbol{\eta})\mathbf{r}'(\gamma),\end{aligned} \tag{6.28}$$

where e_γ is given by (6.19), ω_{d} is given by (6.17), and

$$\boldsymbol{k}(\hat{\mathbf{x}}) = \bar{\Delta}\left(R^{\mathrm{T}}(\boldsymbol{\eta})\left(\mathbf{r}'(\gamma)\omega_{\mathrm{d}} + \hat{\boldsymbol{v}}\right) - K_{\mathrm{p}}\hat{\mathbf{e}}_{\mathrm{p}}\right). \tag{6.29}$$

where $\hat{\mathbf{e}}_{\mathrm{p}}$ is given by (6.26). Let $\tilde{\mathbf{x}}$ be the estimation error of the target's state, defined as

$$\tilde{\mathbf{x}} = \hat{\mathbf{x}} - \mathbf{x}. \tag{6.30}$$

The following theorem states the main result on single tracker target pursuit.

Theorem 6.1. *Consider the tracking error system described by* (6.20) *and* (6.22)*. Then, the controllers given by* (6.27) *and* (6.28) *guarantee that the tracking error system is input-to state stable (ISS) with respect to the state* $\mathbf{e}$ *and the input* $\tilde{\mathbf{x}}$*, i.e. there exist* $\beta \in \mathcal{KL}$ *and* $\alpha \in \mathcal{K}$ *functions that satisfy*

$$\|\mathbf{e}\| \leq \beta(\|\mathbf{e}(0)\|, t) + \alpha(\sup_{t\geq 0}\|\tilde{\mathbf{x}}(t)\|) \tag{6.31}$$

for any initial condition $\mathbf{e}(0)$*.*

Proof: See section 6.10.2.

The theorem implies that the tracking controllers (6.27) and (6.28) are robust against the estimation error $\tilde{\mathbf{x}}$, i.e. the tracking error $\mathbf{e}$ is bounded for any bounded estimation error $\tilde{\mathbf{x}}$. In addition, inequality (6.31) implies that asymptotically, the tracking error $\mathbf{e}$ depends only on the estimation error $\tilde{\mathbf{x}}$ and, if $\tilde{\mathbf{x}}$ converges to zero, then $\mathbf{e}$ converges to zero as well. Clearly, the above analyses imply that the tracking error depends only on the performance of the filter that estimates the target's state.

In the context of the present chapter, for the sake of simplicity we adopt an EKF in information form (also called the information filter in the literature) to estimate the target's state. Note that the information form and the standard form of EKF are equivalent. The latter propagates the covariance matrix of the estimated state, whereas the former propagates the inverse of the covariance matrix. We use the information form because it is more convenient to decentralize the filter in the case of multiple trackers, to be studied later, where a distributed EKF is required to address the constraint on the inter-tracker communication network [Mutambara, 1998].
The EKF in information form is described as follows. Let $Y_k \triangleq \{y_m\}_{m=0}^{k}$ denote the set of ranges from the tracker to the target acquired up to time k. Let also $p(\mathbf{x}_k|Y_{k-1}) \sim \mathcal{N}(\hat{\mathbf{x}}_{k|k-1}, P_{k|k-1})$ and $p(\mathbf{x}_k|Y_k) \sim \mathcal{N}(\hat{\mathbf{x}}_{k|k}, P_{k|k})$ be the prior and posterior densities of the target's state, estimated at time k, respectively. Denote by $\mathbf{z}$ and Ω the information vector and the information matrix, respectively, defined as

$$\begin{array}{ll} \mathbf{z}_{k|k-1} \triangleq P_{k|k-1}^{-1}\hat{\mathbf{x}}_{k|k-1} & \mathbf{z}_{k|k} \triangleq P_{k|k}^{-1}\hat{\mathbf{x}}_{k|k} \\ \Omega_{k|k-1} \triangleq P_{k|k-1}^{-1} & \Omega_{k|k} \triangleq \ P_{k|k}^{-1}. \end{array} \tag{6.32}$$

With the above definition, the prior and posterior densities of the target can be computed recursively using Algorithm 6.1 below. In the algorithm, W, V are arbitrarily positive definite matrix of appropriate dimensions [Battistelli and Chisci, 2016]. A typical choice for such matrices is to take W as an estimate of the inverse covariance of the process disturbance $\mathbf{w}_k$, and each V as an estimate of the inverse variance of the range measurement noise η_k. With the range measurement model (6.9), $V = 1/\sigma$. With the EKF, the estimation error $\tilde{\mathbf{x}}$ is guaranteed to be bounded provided that the estimate of the target's state is initialized sufficiently close to the true target's state, and the process and measurement noises are sufficiently small [Reif et al., 1999].

6.5 SLAP with multiple trackers

In this section, we consider scenarios where multiple trackers are used in a cooperative manner to localize and pursue the target. The use of multiple trackers potentially enhances both performance and robustness of the target localization and pursuit system. Intuitively, more trackers means that more range measurements can be acquired on a given interval and, therefore, more information is available to estimate the target's state. Multiple trackers also implies that redundant trackers are always available in cases any

Algorithm 6.1 EKF for a single tracker

```
procedure INITIALIZATION
    At $k = 0$, initialize $\hat{\mathbf{x}}_{1|0}, P_{1|0}$
    $\Omega_{1|0} = P_{1|0}^{-1}, \quad \mathbf{z}_{1|0} = P_{1|0}^{-1}\hat{\mathbf{x}}_{1|0}$
return $\hat{\mathbf{x}}_{1|0}, \Omega_{1|0}, \mathbf{z}_{1|0}$
At each discrete time $k$, repeat the following procedures:
procedure MEASUREMENT-UPDATE
    if obtain a new range then
        $C_k = \frac{\partial d_k}{\partial \mathbf{x}}(\hat{\mathbf{x}}_{k|k-1})$
        $\tilde{y}_k = y_k - d_k(\hat{\mathbf{x}}_{k|k-1}) + C_k\hat{\mathbf{x}}_{k|k-1}$
        Compute the innovation terms
            $$\tilde{\mathbf{z}}_k = (C_k)^\top V\tilde{y}_k, \quad \tilde{\Omega}_k = (C_k)^\top V C_k. \tag{6.33}$$
    else set $\tilde{\mathbf{z}}_k = \mathbf{0}, \tilde{\Omega}_k = 0$.
    Correction
        $$\mathbf{z}_{k|k} = \mathbf{z}_{k|k-1} + \tilde{\mathbf{z}}_k, \qquad \Omega_{k|k} = \Omega_{k|k-1} + \tilde{\Omega}_k \tag{6.34}$$
  return $\mathbf{z}_{k|k}, \Omega_{k|k}, \hat{\mathbf{x}}_{k|k} = \Omega_{k|k}^{-1}\mathbf{z}_{k|k}$
procedure PREDICTION
        $$\begin{aligned} \hat{\mathbf{x}}_{k+1|k} &= F\hat{\mathbf{x}}_{k|k} \\ \Omega_{k+1|k} &= W - WF(\Omega_{k|k} + F^\top WF)^{-1}F^\top W \end{aligned} \tag{6.35}$$
  return $\hat{\mathbf{x}}_{k+1|k}, \Omega_{k+1|k}, \mathbf{z}_{k+1|k} = \Omega_{k+1|k}\hat{\mathbf{x}}_{k+1|k}$
```

trackers in the group fail their mission or must leave it for any reason. However, both from a theoretical and practical standpoint, the use of multiple trackers raises new challenges in terms of the design of the corresponding control and target motion estimation systems. The main challenge comes from the fact that communications among the trackers are constrained by the inter-tracker communication network, where trackers can only exchange information with their neighbors rather than with all trackers in the network. This requires that the design of control and estimation systems be addressed in a distributed manner.

6.5.1 Distributed estimation and control architecture

At the control level, the main objective is to design a distributed control system to drive the trackers along desired trajectories that i) ensure the observability of the target's motion and ii) maintain an optimal geometrical formation relative to the target with the objective of acquiring maximal range-information to estimate the target's state; see the characterization of such trajectories in Sections 6.3.2, and illustrations of optimal geometrical formations in Fig.6.1 for different numbers of trackers. Notice that there may exist multiple trajectories that satisfy conditions i) and ii). However, for the sake of simplicity, we inherit the methodology for the case of single tracker-single target presented in the previous section. That is, for each tracker i, a desired S-T curve, defined as

$$\mathbf{p}_{\mathrm{d}}^{[i]}(\gamma^{[i]},t) = \mathbf{r}(\gamma^{[i]}) + \mathbf{q}(t), \tag{6.36}$$

is assigned for it to track, where $\mathbf{q}(t)$ is the target's trajectory and $\mathbf{r}^{[i]}(\cdot)$ is a corresponding spatial path encircling the target. This path is described by (6.14) and (6.15) for 3-D and 2-D cases, respectively, and parameterized by the variable $\gamma^{[i]}$ and the set of constant parameters $\{r_{\mathrm{x}}^{[i]}, r_{\mathrm{y}}^{[i]}, r_{\mathrm{z}}^{[i]}, \gamma_0^{[i]}\}$. It was explained in the previous section that if trackers can be controlled such that they converge to their corresponding S-T curves, then the pursuit task stated in Problem 6.1 will be fulfilled. In addition, with this design, if the paths $\mathbf{r}^{[i]}(\cdot); i \in \mathcal{V}$ are parameterized appropriately such that $\gamma_0^{[1]} - \gamma_0^{[2]} = \pi/2$ for $N = 2$ and $\gamma_0^{[i]} - \gamma_0^{[j]} = 2\pi/N$ for $N \geq 3$, where i, j are any two adjacent trackers, then the S-T curves will yield the desired geometrical formations relative to the target when $\gamma^{[i]} = \gamma^{[j]}$ for all $i, j \in \mathcal{V}$. In this context, $\gamma^{[i]}; i \in \mathcal{V}$ are called coordination states and the problem of coordinating the S-T curves to make the coordination states become equal (reach consensus) is called a coordination/consensus problem. This strategy borrows from the concepts of cooperative path following explained in [Ghabcheloo et al., 2009, Hung et al., 2020b, Rego et al., 2019a]. See also the diagrams that illustrate the key concepts behind the design methodology in Fig.6.3.

With the new requirement on reaching an optimal geometrical formation relative to the target, the cooperative target pursuit problem stated in Problem 6.1, see (6.10), is re-stated as follows.

Problem 6.2 (Cooperative control for target pursuit). *Consider the problem of cooperative target localization and pursuit stated in Problem 6.1. Supposed that each tracker is assigned a trajectory given by* (6.36) *to track. Find control laws for the trackers' inputs* $\mathbf{u}^{[i]}$ *and for the evolution of the path parameters* $\dot{\gamma}^{[i]}$ *(or* $\ddot{\gamma}^{[i]}$*) to fulfill the following tasks:*

i Tracking:

$$\lim_{t\to\infty} \mathbf{p}_{\mathrm{d}}^{[i]}(\gamma^{[i]}(t),t) - \mathbf{p}^{[i]}(t) = 0 \tag{6.37}$$

for all $i \in \mathcal{V}$.

ii Coordination (consensus):

$$\lim_{t\to\infty} \gamma^{[i]}(t) - \gamma^{[j]}(t) = 0, \tag{6.38a}$$

$$\lim_{t\to\infty} \dot{\gamma}^{[i]}(t) = \bar{\omega} \tag{6.38b}$$

for all $i, j \in \mathcal{V}$, where $\bar{\omega} \neq 0$ is the common desired nominal angular speed assigned for the path parameters $\gamma^{[i]}$ for all $i \in \mathcal{V}$.

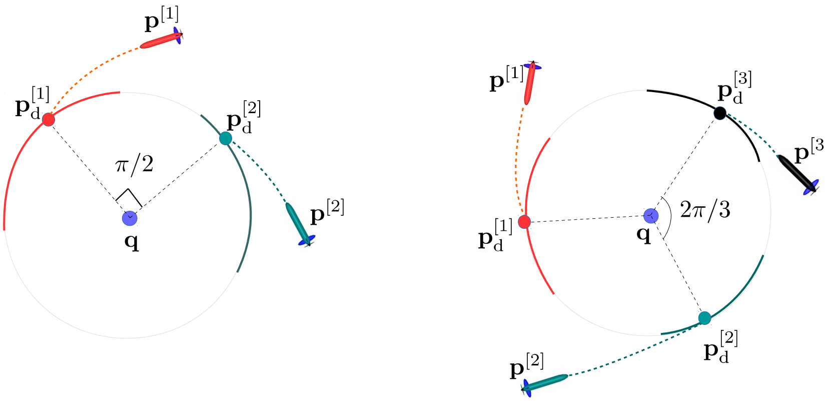

$\mathbf{p}^{[1]}$
$\mathbf{p}_{\mathrm{d}}^{[1]}$
$\pi/2$
$\mathbf{p}_{\mathrm{d}}^{[2]}$
$\mathbf{q}$
$\mathbf{p}^{[2]}$
$\mathbf{p}^{[1]}$
$\mathbf{p}_{\mathrm{d}}^{[3]}$
$\mathbf{p}^{[3}$
$2\pi/3$
$\mathbf{p}_{\mathrm{d}}^{[1]}$
$\mathbf{q}$
$\mathbf{p}_{\mathrm{d}}^{[2]}$
$\mathbf{p}^{[2]}$

Figure 6.3: Design methodology for cooperative target pursuit. (Left: $N = 2$, Right: $N = 3$). Positions: $\mathbf{q}$ (target), $\mathbf{p}^{[i]}$ (trackers), and $\mathbf{p}_{\mathrm{d}}^{[i]}$ (desired S-T curve).

In order to solve the problem of cooperative target localization and pursuit while fulfilling the coordination requirement stated in (6.38), we propose a distributed estimation and control (DEC) system for each tracker $i; i \in \mathcal{V}$, as depicted in Fig.6.4. The underlying idea behind this architecture is briefly described as follows.

i) Cooperative Estimation: the main goal of this block is to solve the cooperative localization task given by (6.11) in a distributed manner. To this end, we adopt the distributed EKF (DEKF) methodology proposed in [Battistelli and Chisci, 2016]. In the DEKF, the trackers exchange the latest PDFs about the target's state (after being corrected with the innovation information obtained by local range measurements) with their neighbors and apply a fusion (consensus) algorithm. As a result of this cooperative process, the estimates of the target's state reach consensus and are potentially more accurate than that obtained in the case each tracker estimates the target's state by itself (non-cooperative strategy).

ii) Cooperative Control: the main goal of this block is to solve the coordination task given by (6.38) by using a distributed coordination/consensus control law that computes correction speeds $v_c^{[i]}; i \in \mathcal{V}$ about the nominal speed $\bar{\omega}$ to coordinate the path parameters and make them reach consensus. The control law for $v_c^{[i]}$ uses local information ($\gamma^{[i]}$) (and possibly $\dot{\gamma}^{[i]}$) and information from its in-neighboring trackers ($\gamma^{[j]}, \dot{\gamma}^{[j]}; j \in \mathcal{N}_{\text{in}}^{[i]}$). For this reason, the trackers are required to communicate and exchange the corresponding path variables $\gamma^{[i]}$ (and possibly $\dot{\gamma}^{[i]}$) with theirs neighbors.

iii) Tracking Controller: this controller aims to make the trackers converge to and follow their assigned trajectorys given by (6.36), i.e. solve the tracking task given by (6.37). To track the trajectories, the tracking controllers developed in previous section will be used with a sightly modification that takes into account the cooperative information provided by the cooperative layer.

The design of the complete DEC system is described next.

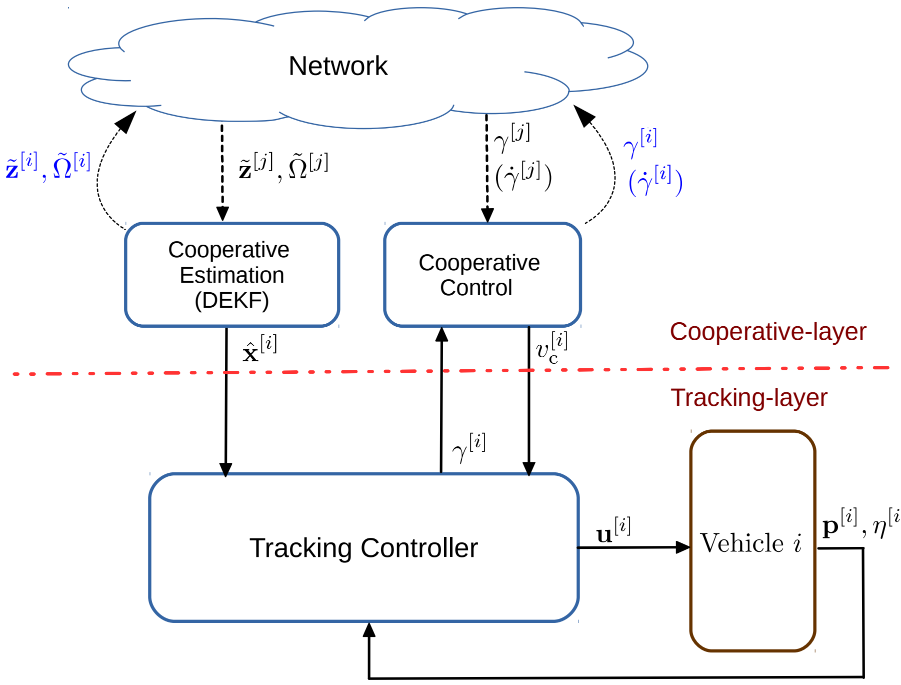


Figure 6.4: The DEC system as seen by tracker i. In the figure, $j \in \mathcal{N}_{\text{in}}^{[i]}$.

6.5.2 Cooperative target estimation

In this section, we propose a DEKF that solves the cooperative estimation problem defined by (6.11). The design of the DEKF is inspired from the work in [Battistelli and Chisci, 2016] and is described next. Let $Y_k^{[i]} \triangleq \{y_m^{[i]}\}_{m=0}^k$ denote the set of ranges to the target measured by tracker $i; i \in \mathcal{V}$ and $\mathcal{I}_k^{[j]}; j \in \mathcal{N}_{\text{in}}^{[i]}$ the collective information (PDFs of the target's state estimated by its in-neighboring trackers) that the tracker has received up to time k. For each tracker i let also

$$
\begin{aligned}
&p^{[i]}(\mathbf{x}_k|Y_{k-1}^{[i]}, \mathcal{I}_{k-1}^{[j]}) \sim \mathcal{N}(\hat{\mathbf{x}}_{k|k-1}^{[i]}, P_{k|k-1}^{[i]}) : \text{the local PDF of target before correction,}\\
&p^{[i]}(\mathbf{x}_k|Y_{k}^{[i]}, \mathcal{I}_{k-1}^{[j]}) \sim \mathcal{N}(\tilde{\mathbf{x}}_{k}^{[i]}, \tilde{P}_{k}^{[i]}) : \text{the local PDF of target after correction}\\
&p^{[i]}(\mathbf{x}_k|Y_{k}^{[i]}, \mathcal{I}_{k}^{[j]}) \sim \mathcal{N}(\hat{\mathbf{x}}_{k|k}^{[i]}, P_{k|k}^{[i]}) : \text{the local PDF of target after fusion}
\end{aligned}
$$

The relation of the above PDFs is illustrated in Fig.6.5 while their parameters (means and covariances) are computed by a DEKF proposed in Algorithm 6.2. Note that in the algorithm, variables $\mathbf{z}^{[i]}$ and $\Omega^{[i]}$ (res. $\tilde{\mathbf{z}}^{[i]}$ and $\tilde{\Omega}^{[i]}$) denote the corresponding information vector and information matrix, respectively, and have the relation with means and covariances through (6.32). Compared with Algorithm 6.1, there are two key procedures, namely, "Communication" and "Fusion", that must be implemented prior to the "Prediction" procedure. The "Communication" procedure is used to transmit the latest local

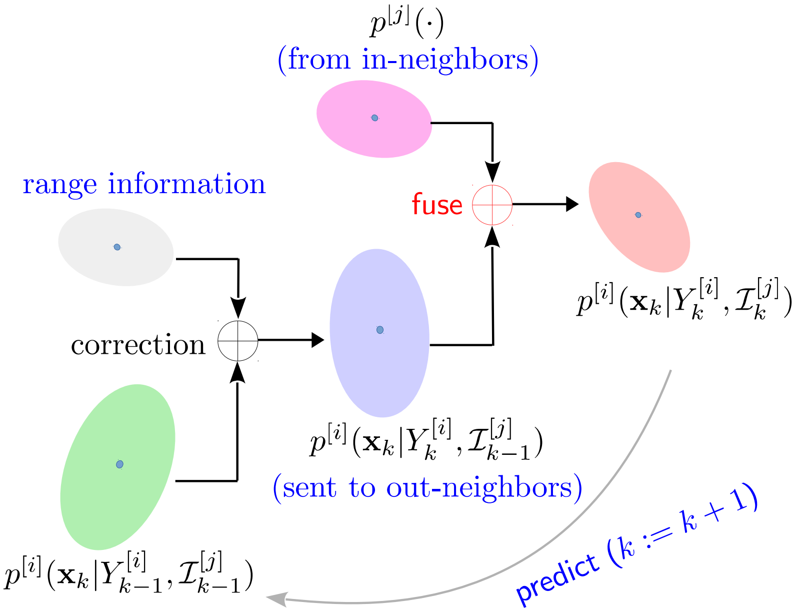

$p^{[j]}(\cdot)$
(from in-neighbors)
range information
fuse
correction
$p^{[i]}(\mathbf{x}_k|Y_k^{[i]}, \mathcal{I}_k^{[j]})$
$p^{[i]}(\mathbf{x}_k|Y_k^{[i]}, \mathcal{I}_{k-1}^{[j]})$
(sent to out-neighbors)
predict $(k := k+1)$
$p^{[i]}(\mathbf{x}_k|Y_{k-1}^{[i]}, \mathcal{I}_{k-1}^{[j]})$

Figure 6.5: The DEKF mechanism seen by Tracker i. In the figure, $j \in \mathcal{N}_{\text{in}}^{[i]}$.

PDF about the target estimated by each tracker to its out-neighbors. The "Fusion" procedure is implemented to fuse, at the level of each tracker, the local corrected PDF about the target's state with the PDFs received from the neighbors. This is done to improve the accuracy of the target 's state estimate and also to ensure that consensus is reached on the estimates of the target's state. This procedure can be implemented using (6.42). Note that in these equations, $\pi^{[i,j]}; i \in \mathcal{V}, j \in \mathcal{N}_{\text{in}}^{[i]} \cup \{i\}$ are weighting parameters that must be chosen such that $\pi^{[i,j]} > 0$ and $\sum_{j \in \mathcal{N}_{\text{in}}^{[i]} \cup \{i\}} \pi^{[i,j]} = 1$ for all $i \in \mathcal{V}$. The prediction step is done similarly to the case of single tracker, computed by (6.43). Similar to Algorithm 6.1, W is chosen as inverse of covariance matrix of the process noise while $V^{[i]}$ is normally chosen as inverse of covariance matrix of the range measurement noise. That is, $V^{[i]} = 1/\sigma$ for all $i \in \mathcal{V}$.

Remark 6.6. *Algorithm 6.1 is a special case of Algorithm 6.2 with $N = 1$.*

Remark 6.7. *For the sake of simplicity and in order to save on communications, at each sampling period Algorithm 6.2 requires that the trackers exchange information with their neighbors only one time in order to update the "Fusion" step. This is equivalent to setting $L = 1$ in [Battistelli and Chisci, 2016], where L is the number of interactions among the trackers during each sampling interval $[k, k+1]$ used to fuse the information about the estimated target's state. The larger the number L, the faster will the convergence of the DEKF be; however, more communications will be required among the trackers [Battistelli and Chisci, 2016].*

6.5.3 Cooperative target pursuit

We now propose controllers for each tracker i to solve the problem of cooperative control for target pursuit stated in Problem 6.2. Inspired by the work in the field of consensus of multi agent systems [Ren and Beard, 2008], at the cooperative control level, a distributed control law for the correction speed $v_{\text{c}}^{[i]}$ in a form of consensus protocol is given as

$$v_{\text{c}}^{[i]} = -k_c \sum_{j \in \mathcal{N}_{\text{in}}^{[i]}} (\gamma^{[i]} - \gamma^{[j]}), \tag{6.44}$$

where $k_c > 0$ is a coupling gain. With this correction speed, the total desired speed that for $\gamma^{[i]}$ to track is given by

$$\omega_{\mathrm{d}}^{[i]} = \bar{\omega} + v_{\mathrm{c}}^{[i]}, \tag{6.45}$$

Algorithm 6.2 Distributed EKF for tracker i

```
procedure Initialization
    At $k = 0$, initialize $\hat{\mathbf{x}}_{1|0}^{[i]}, P_{1|0}^{[i]}$
    $\Omega_{1|0}^{[i]} = [P_{1|0}^{[i]}]^{-1}, \quad \mathbf{z}_{1|0}^{[i]} = [P_{1|0}^{[i]}]^{-1}\hat{\mathbf{x}}_{1|0}^{[i]}$
return $\hat{\mathbf{x}}_{1|0}^{[i]}, \Omega_{1|0}^{[i]}, \mathbf{z}_{1|0}^{[i]}$
At each discrete time $k$, repeat the following procedures:
procedure Local correction
    if obtain a new range then
        $C_k^{[i]} = \frac{\partial d_k^{[i]}}{\partial \mathbf{x}}(\hat{\mathbf{x}}_{k|k-1}^{[i]})$
        $\tilde{y}_k^{[i]} = y_k^{[i]} - d_k^{[i]}(\hat{\mathbf{x}}_{k|k-1}^{[i]}) + C_k^{[i]}\hat{\mathbf{x}}_{k|k-1}^{[i]}$
            $$\begin{aligned} \tilde{\mathbf{z}}_k^{[i]} &= \mathbf{z}_{k|k-1}^{[i]} + (C_k^{[i]})^{\mathrm{T}} V^{[i]} \tilde{y}_k^{[i]} \\ \tilde{\Omega}_k^{[i]} &= \Omega_{k|k-1}^{[i]} + (C_k^{[i]})^{\mathrm{T}} V^{[i]} C_k^{[i]} \end{aligned} \tag{6.40}$$
    else set $\tilde{\mathbf{z}}_k^{[i]} = \mathbf{z}_{k|k-1}^{[i]}, \tilde{\Omega}_k^{[i]} = \Omega_{k|k-1}^{[i]}$.
procedure Communication
    Transmit to its out-neighbors a message $\mathcal{M}_{\mathrm{e}}^{[i]}(k)$, defined as
            $$\mathcal{M}_{\mathrm{e}}^{[i]}(k) \triangleq \{\tilde{\mathbf{z}}_k^{[i]}, \tilde{\Omega}_k^{[i]}\} \tag{6.41}$$
procedure Fusion (Consensus Estimation)
            $$\begin{aligned} \mathbf{z}_{k|k}^{[i]} &= \pi^{[i,i]}\tilde{\mathbf{z}}_k^{[i]} + \sum_{j \in \mathcal{N}_{\mathrm{in}}^{[i]}} \pi^{[i,j]}\tilde{\mathbf{z}}_k^{[j]} \\ \Omega_{k|k}^{[i]} &= \pi^{[i,i]}\tilde{\Omega}_k^{[i]} + \sum_{j \in \mathcal{N}_{\mathrm{in}}^{[i]}} \pi^{[i,j]}\tilde{\Omega}_k^{[j]} \end{aligned} \tag{6.42}$$
    return $\mathbf{z}_{k|k}^{[i]}, \Omega_{k|k}^{[i]}, \hat{\mathbf{x}}_{k|k}^{[i]} = [\Omega_{k|k}^{[i]}]^{-1}\mathbf{z}_{k|k}^{[i]}$
procedure Prediction
            $$\begin{aligned} \hat{\mathbf{x}}_{k+1|k}^{[i]} &= F\hat{\mathbf{x}}_{k|k}^{[i]} \\ \Omega_{k+1|k}^{[i]} &= W - WF(\Omega_{k|k}^{[i]} + F^{\mathrm{T}}WF)^{-1}F^{\mathrm{T}}W \end{aligned} \tag{6.43}$$
    return $\hat{\mathbf{x}}_{k+1|k}^{[i]}, \Omega_{k+1|k}^{[i]}, \mathbf{z}_{k+1|k}^{[i]} = \Omega_{k+1|k}^{[i]}\hat{\mathbf{x}}_{k+1|k}^{[i]}$
```

where as explained before, $\bar{\omega}$ is the nominal desired speed for all $\gamma^{[i]}; i \in \mathcal{V}$. Similarly to (6.19), define also

$$e_{\gamma}^{[i]} = \dot{\gamma}^{[i]} - \omega_{\mathrm{d}}^{[i]} \tag{6.46}$$

as the speed tracking error for the evolution of $\gamma^{[i]}$ for each tracker $i; i \in \mathcal{V}$. Similar to (6.26), let

$$\hat{\mathbf{e}}_{\mathrm{p}}^{[i]} = R^{\mathrm{T}}(\boldsymbol{\eta}^{[i]}) \left(\mathbf{p}^{[i]} - \mathbf{r}(\gamma^{[i]}) - \hat{\mathbf{q}}^{[i]}\right) - \boldsymbol{\epsilon}. \tag{6.47}$$

To track the trajectory we adopt the tracking controllers given by (6.27) and (6.28) for each tracker $i; i \in \mathcal{V}$ as follows.

Tracking Controller - Type I

$$\mathbf{u}^{[i]} = \boldsymbol{k}(\hat{\mathbf{x}}^{[i]}) \tag{6.48a}$$

$$\dot{\gamma}^{[i]} = \omega_{\mathrm{d}}^{[i]}. \tag{6.48b}$$

Tracking Controller - Type II

$$\mathbf{u}^{[i]} = \boldsymbol{k}(\hat{\mathbf{x}}^{[i]}) \tag{6.49a}$$

$$\ddot{\gamma}^{[i]} = -k_{\gamma} e_{\gamma}^{[i]} + (\hat{\mathbf{e}}_{\mathrm{p}}^{[i]})^{\mathrm{T}} R^{\mathrm{T}}(\boldsymbol{\eta}^{[i]}) \mathbf{r}'(\gamma^{[i]}) + \dot{v}_{\mathrm{c}}^{[i]}, \tag{6.49b}$$

where in (6.48) and (6.49)

$$\boldsymbol{k}(\hat{\mathbf{x}}^{[i]}) = \bar{\Delta} \left(R^{\mathrm{T}}(\boldsymbol{\eta}^{[i]}) \left(\mathbf{r}'(\gamma^{[i]}) \omega_{\mathrm{d}}^{[i]} + \hat{\boldsymbol{v}}^{[i]}\right) - K_{\mathrm{p}} \hat{\mathbf{e}}_{\mathrm{p}}^{[i]}\right) \tag{6.50}$$

and in (6.49)

$$\dot{v}_{c}^{[i]} = -k_{c} \sum_{j \in \mathcal{N}_{\mathrm{in}}^{[i]}} (\dot{\gamma}^{[i]} - \dot{\gamma}^{[j]}), \tag{6.51}$$

with $\omega_{\mathrm{d}}^{[i]}, e_{\gamma}^{[i]}$ and $\hat{\mathbf{e}}_{\mathrm{p}}^{[i]}$ are given by (6.45)–(6.47).

Remark 6.8. *Both types of controllers require the trackers to exchange the coordination state $\gamma^{[i]}; i \in \mathcal{V}$ to compute the correction speed $v_{\mathrm{c}}^{[i]}$. However, Tracking Controller-Type II in* (6.49) *also requires the trackers to exchange $\dot{\gamma}^{[i]}; i \in \mathcal{V}$ (in order to compute $\dot{v}_{\mathrm{c}}^{[i]}$) .*

6.5.4 Stability analysis of the complete DEC system

We now show that the proposed DEC system designed in Sections 6.5.2 and 6.5.3 solves the cooperative target localization and pursuit problem while verifying a robust stability property. To this end, we analyze the convergence of estimation, coordination, and pursuit errors of the complete DEC system, defined as follows.

Estimation error:

Let

$$\tilde{\mathbf{x}}^{[i]} \triangleq \hat{\mathbf{x}}^{[i]} - \mathbf{x}. \tag{6.52}$$

be the estimation error of the target's state estimated by tracker $i; i \in \mathcal{V}$. Collectively, the total estimation error of the complete DEC system is defined as

$$\tilde{\mathbf{x}} \triangleq \operatorname{col}\left(\tilde{\mathbf{x}}^{[1]}, ..., \tilde{\mathbf{x}}^{[N]}\right) \in \mathbb{R}^{nN}. \tag{6.53}$$

Coordination error:

Recall that with the proposed methodology, the path parameters $\gamma^{[i]}$ represent the coordination states of the trackers. Let

$$\xi_i = \gamma^{[i]} - \sum_{j=1}^{N} r_j \gamma^{[j]} \tag{6.54}$$

be the coordination error among the trackers, where r_i is the i^{th} component of the left eigenvector of the Laplacian matrix of the underlying communication graph. Recall from Lemma 2.1 that this eigenvector is denoted by $\mathbf{r} \in \mathbb{R}^N$. Note also that if the graph is balanced, then $\mathbf{r} = \mathbf{1}/N$. In this case, ξ_i measures the disagreement between the coordination state $\gamma^{[i]}$ and the average of all coordination states. Let also $\boldsymbol{\xi} \triangleq [\xi_1^{\mathrm{T}}, ..., \xi_N^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^N$ be the coordination error vector that captures the disagreement among the coordination states. From (6.54), $\boldsymbol{\xi}$ can be rewritten as

$$\boldsymbol{\xi} = W\boldsymbol{\gamma}, \tag{6.55}$$

where $\boldsymbol{\gamma} \triangleq [\gamma^{[1]}, ..., \gamma^{[N]}]^{\mathrm{T}} \in \mathbb{R}^N$ and

$$W \triangleq I_N - \mathbf{1}\mathbf{r}^{\mathrm{T}}. \tag{6.56}$$

With the above definition, it is clear that all coordination states are synchronized, that is, $\gamma^{[1]} = \gamma^{[2]} = ... = \gamma^{[N]}$ if and only if $\boldsymbol{\xi} = \mathbf{0}$. Therefore, to analyze consensus of the coordination states, we analyze the convergence of the coordination error $\boldsymbol{\xi}$ to zero. The dynamics of $\boldsymbol{\xi}$ are given by

$$\begin{aligned}\dot{\boldsymbol{\xi}} &= W\dot{\boldsymbol{\gamma}} \overset{(6.44)-(6.46)}{=} W(\mathbf{1}\bar{\omega} + -k_c L\boldsymbol{\gamma} + \mathbf{e}_\gamma)\\ &= -k_c L\boldsymbol{\xi} + W\mathbf{e}_\gamma\\ &\triangleq \mathbf{f}_1(\boldsymbol{\xi}, \mathbf{e}),\end{aligned} \tag{6.57}$$

where $\mathbf{e}_\gamma \triangleq [e_\gamma^{[1]}, ..., e_\gamma^{[N]}]^{\mathrm{T}} \in \mathbb{R}^N$. Above we used Lemma 2.1 for the last equality, that is, because $\mathbf{r}^{\mathrm{T}}\mathbf{1} = 1$ and $\mathbf{r}^{\mathrm{T}}L = 0$; therefore, $W\mathbf{1} = \mathbf{0}$ and $WL = LW$.

Pursuit error:

Similarly to (6.18), let

$$\mathbf{e}_{\mathrm{p}}^{[i]} = R^{\mathrm{T}}(\boldsymbol{\eta}^{[i]})(\mathbf{p}^{[i]} - \mathbf{r}(\gamma^{[i]}) - \mathbf{q}) - \boldsymbol{\epsilon} \tag{6.58}$$

be the position error between tracker i and the vicinity of the target. From (6.47) and (6.58) it follows that

$$\hat{\mathbf{e}}_{\mathrm{p}}^{[i]} = \mathbf{e}_{\mathrm{p}}^{[i]} - R^{\mathrm{T}}(\boldsymbol{\eta}^{[i]})\tilde{\mathbf{q}}^{[i]}, \tag{6.59}$$

where $\tilde{\mathbf{q}}^{[i]} \triangleq \hat{\mathbf{q}}^{[i]} - \mathbf{q}^{[i]}$ is the the estimation error of the the target's position, estimated by tracker i. The dynamics of the error can be obtained similarly to (6.20) as

$$\dot{\mathbf{e}}_{\mathrm{p}}^{[i]} = -S(\boldsymbol{\omega}^{[i]})\mathbf{e}_{\mathrm{p}}^{[i]} + \Delta\mathbf{u}^{[i]} - R^{\mathrm{T}}(\boldsymbol{\eta}^{[i]})\left(\mathbf{r}'(\gamma^{[i]})\dot{\gamma}^{[i]} + \boldsymbol{v}\right)$$

Note that the control law for $\mathbf{u}^{[i]}$ is identical for two types of tracking controllers (6.48) and (6.49). Thus, making $\mathbf{u}^{[i]} = \boldsymbol{k}(\hat{\mathbf{x}}^{[i]})$, where $\boldsymbol{k}(\hat{\mathbf{x}}^{[i]})$ given by (6.50) and using the relation in (6.46) in the above equation, we obtain

$$\begin{aligned}\dot{\mathbf{e}}_{\mathrm{p}}^{[i]} = & - S(\boldsymbol{\omega}^{[i]})\mathbf{e}_{\mathrm{p}}^{[i]} + R^{\mathrm{T}}(\boldsymbol{\eta}^{[i]})\left(\tilde{\boldsymbol{v}}^{[i]} - \mathbf{r}'(\gamma^{[i]})e_{\gamma}^{[i]}\right) \\ & - K_{\mathrm{p}}\left(\mathbf{e}_{\mathrm{p}}^{[i]} - R^{\mathrm{T}}(\boldsymbol{\eta}^{[i]})\tilde{\mathbf{q}}^{[i]}\right),\end{aligned} \tag{6.60}$$

where $\tilde{\boldsymbol{v}}^{[i]} \triangleq \hat{\boldsymbol{v}}^{[i]} - \boldsymbol{v}^{[i]}$ is the estimation error of the target's velocity, estimated by tracker i.

We now derive the dynamics for $e_{\gamma}^{[i]}$ defined by (6.46). For the Tracking-Controller Type I given by (6.48), $e_{\gamma}^{[i]} = 0$, hence $\dot{e}_{\gamma}^{[i]}(t) = 0$ for all t and $i \in \mathcal{V}$. For the Tracking-Controller Type II, the dynamics of $e_{\gamma}^{[i]}$ are given by

$$\dot{e}_{\gamma}^{[i]} \overset{(6.45),(6.46)}{=} \ddot{\gamma}^{[i]} - \dot{v}_{\mathrm{c}}^{[i]}$$

Substituting $\ddot{\gamma}^{[i]}$ in (6.49b) in the above equation yields

$$\dot{e}_{\gamma}^{[i]} = -k_{\gamma}e_{\gamma}^{[i]} + \hat{\mathbf{e}}_{\mathrm{p}}^{\mathrm{T}}R^{\mathrm{T}}(\boldsymbol{\eta})\mathbf{r}'(\gamma). \tag{6.61}$$

Let also $\mathbf{e}^{[i]} = [(\mathbf{e}_{\mathrm{p}}^{[i]})^{\mathrm{T}}, e_{\gamma}^{[i]}]^{\mathrm{T}}$ be the tracking error vector of tracker i. Collectively, the pursuit error of the complete DEC system is defined as

$$\mathbf{e} \triangleq \mathrm{col}(\mathbf{e}^{[1]}, ..., \mathbf{e}^{[N]}). \tag{6.62}$$

From (6.60) and (6.61), the dynamics of the total pursuit error can be rewritten in the general form

$$\dot{\mathbf{e}} = \mathbf{f}_2(\mathbf{e}, \tilde{\mathbf{x}}, \boldsymbol{\xi}). \tag{6.63}$$

We now study the stability of the complete DEC system by analyzing the inter-connected system consisting of the systems (6.57) and (6.63), see Fig.6.6. First, we analyze the stability of each sub-system as follows

Lemma 6.2 (stability of the coordination system)**.** *Consider the coordination error system described by* (6.57)*. Assume further that the communication digraph is strongly connected. Then, the coordination error system is ISS with respect to the state* $\boldsymbol{\xi}$ *and the input* $\mathbf{e}$*.*

Proof: See section 6.10.3.

Lemma 6.3 (stability of the pursuit error system)**.** *Consider the complete pursuit error system described by* (6.63)*. Then, the complete tracking error system is ISS with respect to the state* $\mathbf{e}$ *and the input* $\tilde{\mathbf{x}}$*, where* $\tilde{\mathbf{x}}$ *is the total estimation error of the target state given by* (6.53) *.*

Proof. See section 6.10.4.

Remark 6.9. *Lemma 6.3 implies that with tracking controllers given in* (6.48) *and* (6.49)*, the tracking error system is independent of the correction speed* $v_{\mathrm{c}}^{[i]}$*.*

Theorem 6.2. *Consider the closed-loop complete DEC system composed by the coordination error system* (6.57) *and the pursuit error system* (6.63)*. Let assumptions in Lemma 6.2 and Lemma 6.3 hold. Then, the complete DEC system is ISS respect to the state* $\boldsymbol{\mu} \triangleq [\boldsymbol{\xi}^{\mathrm{T}}, \mathbf{e}^{\mathrm{T}}]^{\mathrm{T}}$ *and the input* $\tilde{\mathbf{x}}$*, where* $\tilde{\mathbf{x}}$ *is the total estimation error of the target state given by* (6.53)*.*

Theorem 6.2 follows immediately from Lemma 6.2 and Lemma 6.3 and the stability of cascaded ISS systems [Sontag, 2008]. The ISS property of the close-looped DEC system implies that the pursuit and coordination error ($\boldsymbol{\mu}$) depend only on the estimation error of the target' state ($\tilde{\mathbf{x}}$) generated by the DEKF and is robust against this error in the sense that as long as $\tilde{\mathbf{x}}$ is bounded, then $\boldsymbol{\mu}$ is bounded. Furthermore, if $\tilde{\mathbf{x}}$ converges zero then $\boldsymbol{\mu}$ converges to zero as well. According to [Battistelli and Chisci, 2016], provided that the digraph induced by the tracker's network is strongly connected and the initial estimates of the target's state are sufficiently close to the true target, then $\tilde{\mathbf{x}}$ is guaranteed to be bounded and asymptotically converge to a ball whose size depends on the size of the covariances of range measurement and process noises. Note that this requirement on the connectivity of network is satisfied with Lemma 6.2, whereas the requirement on the initial estimates is standard with EKF.

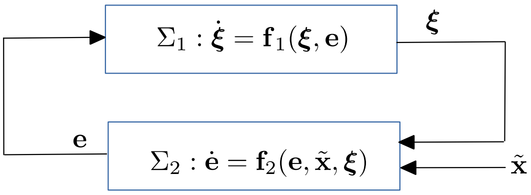

$\Sigma_1 : \dot{\boldsymbol{\xi}} = \mathbf{f}_1(\boldsymbol{\xi}, \mathbf{e})$
$\boldsymbol{\xi}$
$\mathbf{e}$
$\Sigma_2 : \dot{\mathbf{e}} = \mathbf{f}_2(\mathbf{e}, \tilde{\mathbf{x}}, \boldsymbol{\xi})$
$\tilde{\mathbf{x}}$

Figure 6.6: The closed-loop DEC system.

6.6 Simulation examples

In this section, we will present simulation results that illustrate the performance of the DEC in pursuing and localizing a target. We consider two scenarios: the first with one tracker ($N = 1$) and the second with three trackers ($N = 3$). With $N = 3$, the communication topology adopted is depicted in Fig.6.7, which shows the indexes of the trackers and the directional communication links among them (represented by arrows). The simulation parameters for the 2-D and 3-D cases are given in Table 6.1 and Table 6.2, respectively. In all simulations the sampling interval for range measurements is 2 seconds and the Tracking Controller-Type II was used.

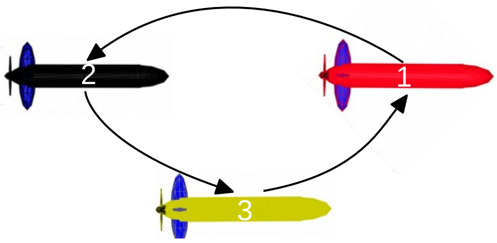
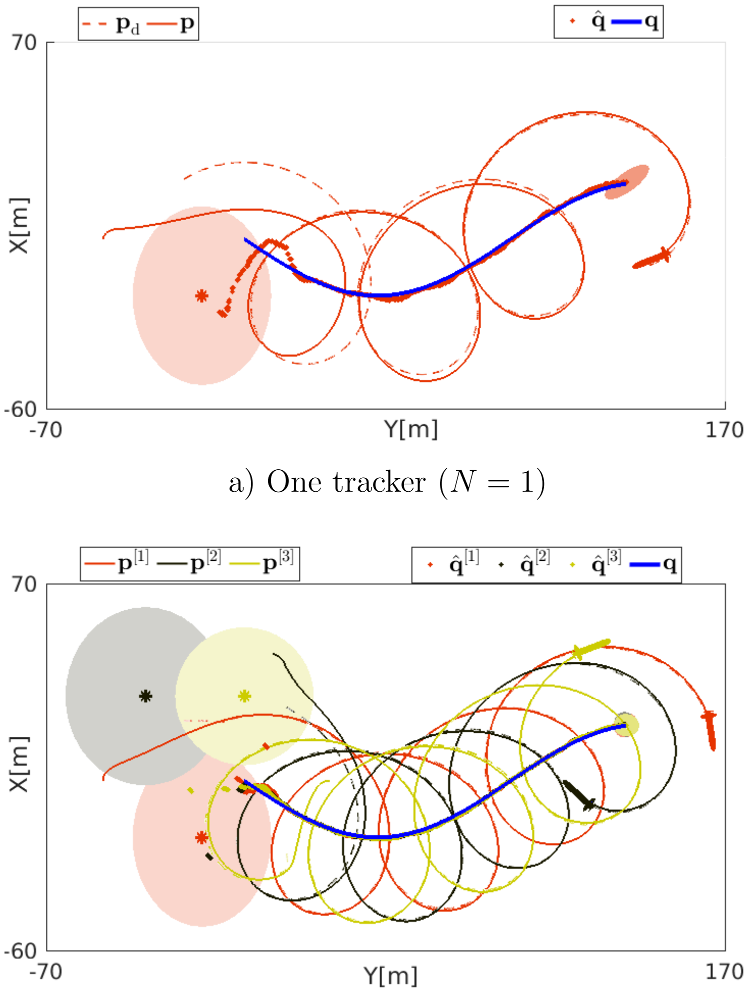


Figure 6.7: The communication network of three trackers. Arrows indicate directions of the information flow, thus inducing a directed graph.

6.6.1 2-D cases

The performance of the DEC strategy for localizing and pursuing the target in 2-D using one tracker and three trackers are plotted in Fig.6.8 and Fig.6.9. Fig.6.8 (a) shows that in both scenarios the trackers converge to and stay in the vicinity of the target, and encircle the latter. The figure also indicates that the target's positions estimated by the trackers converge to a small neighborhood of the true target's position. This can be verified by observing Fig.6.9 where it is evident that the pursuit errors and the localization errors converge nearly to zero, implying that both the pursuit and localization tasks are fulfilled.

Fig.6.9 also indicates that with three trackers, the convergence of the pursuit and the localization errors is faster than that obtained using only Tracker 1. Fig.6.8(b) also shows that the uncertainty regions of the target' position estimated by the three trackers (computed from the covariance matrices $P^{[i]}$ and represented by the ellipsoids) converge to almost the same size, which is smaller than the size of the uncertainty region of the target' position estimated using only Tracker 1.

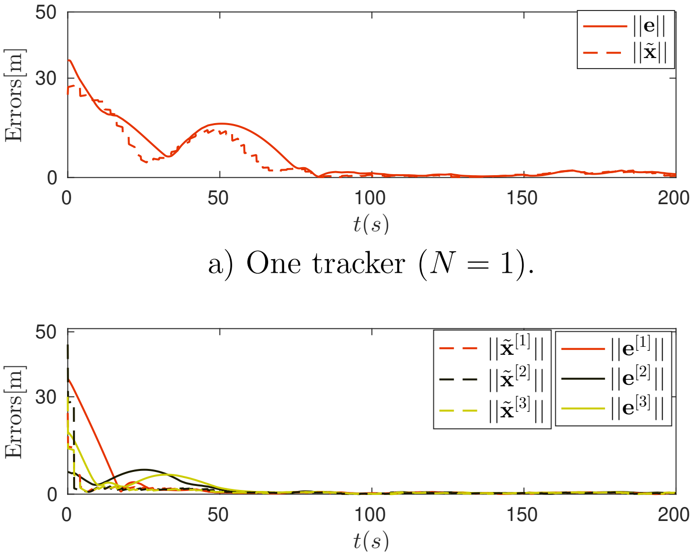


a) One tracker ($N = 1$)

b) Three trackers ($N = 3$).

Figure 6.8: 2-D cases: The filled ellipsoids represent the uncertainties (covariances) of the target's position estimated by the trackers at the beginning and end of the simulation. Trajectories. $\mathbf{p}^{[i]}$: trackers, $\mathbf{p}_\mathrm{d}^{[i]}$: desired path, $\mathbf{q}$: target, $\hat{\mathbf{q}}^{[i]}$: estimated target.

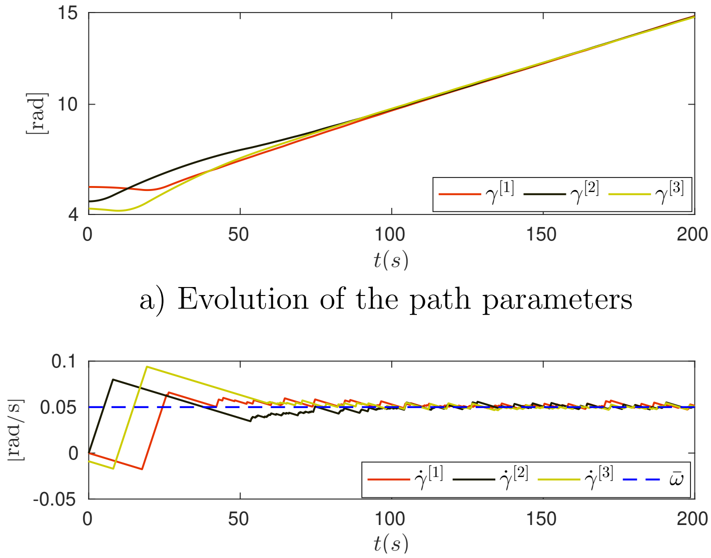
Fig.6.10 shows the performance of the coordination system. It is visible that all the path parameters reach consensus and evolve with the common desired speed $\bar{\omega}$. This implies that the trackers converge to and maintain in the desired formation that allows them to acquire maximal range information to estimate the target's state.



a) One tracker ($N = 1$).

b) There trackers ($N = 3$)

Figure 6.9: 2-D cases: Pursuit errors ($\|\mathbf{e}^{[i]}\|$) and localization errors ($\|\tilde{\mathbf{x}}^{[i]}\|$), $i = 1, 2, 3$.



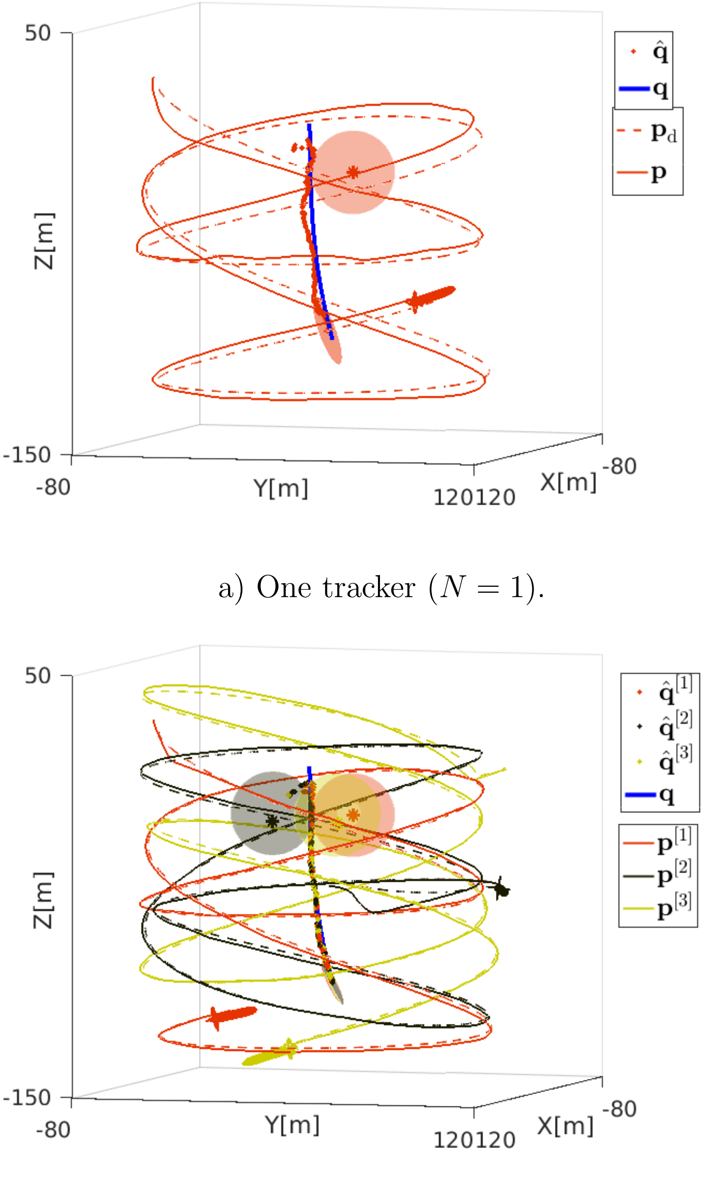
a) Evolution of the path parameters

b) Speeds of the path parameters

Figure 6.10: 2-D cases. Coordination performance in case with three trackers ($N = 3$).

6.6.2 3-D cases

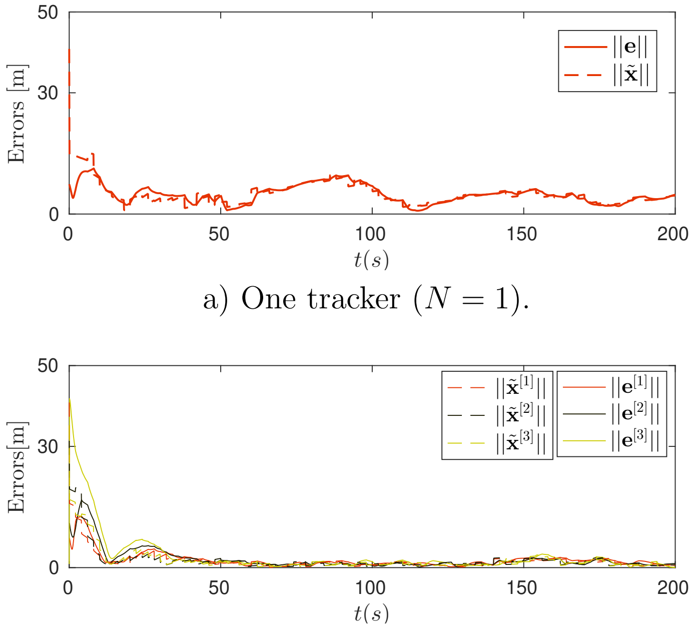
The performance of the DEC strategy for 3-D cases is illustrated in Fig.6.11 and Fig.6.12. The figures show that the pursuit and localization errors are bounded and converge nearly to zero in both scenarios. However, Fig.6.12 indicates that the errors converge faster, and asymptotically are smaller with three trackers. Fig.6.13 shows the performance of the coordination system in the case of there trackers. It is visible that all the path parameters reach consensus and evolve with the common desired speed $\bar{\omega}$. This implies that the coordination task is fulfilled as well.



a) One tracker ($N = 1$).

b) Three trackers ($N = 3$).

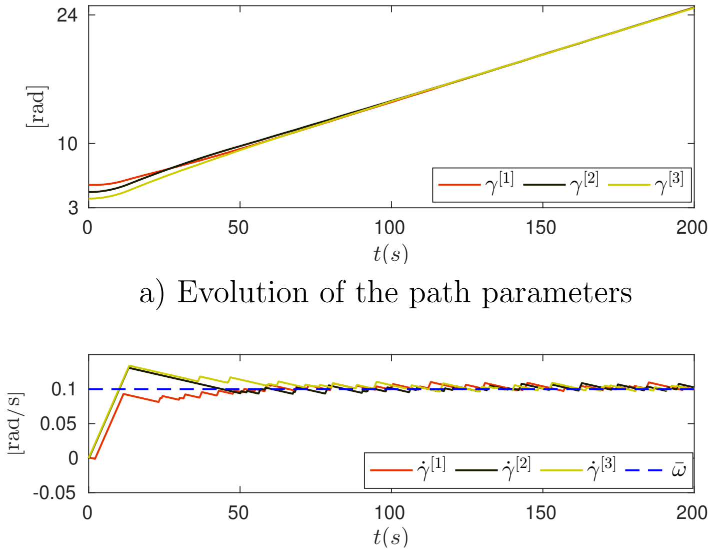
Figure 6.11: 3-D cases: The filled ellipsoids represent the uncertainties (covariances) of the target's position estimated by the trackers at the beginning and the end of the simulation. Trajectories. $\mathbf{p}^{[i]}$: trackers, $\mathbf{q}$: target, $\hat{\mathbf{q}}^{[i]}$: estimated target.



a) One tracker ($N = 1$).

b) Three trackers ($N = 3$).

Figure 6.12: 3-D cases: Pursuit errors ($\|\mathbf{e}^{[i]}\|$) and localization errors ($\|\tilde{\mathbf{x}}^{[i]}\|$), $i = 1, 2, 3$.



a) Evolution of the path parameters

b) Speeds of the path parameters

Figure 6.13: 3-D cases. Coordination performance in case with three trackers ($N = 3$).

6.7 DEC with event-triggered communications

Recall that in order to implement the DEC strategy proposed in the previous section, the trackers are required to exchange the following types of information.

i) The first is associated with the DEKF described in Algorithm 6.2 that includes messages $\mathcal{M}_{\mathrm{e}}^{[i]}(k)$ given by (6.41) exchange periodically at every $t = kT_s$.

ii) The second is associated with the cooperative controller that requires the exchange of messages $\mathcal{M}_{\mathrm{c}}^{[i]}(t) \triangleq \{\gamma^{[i]}(t)\}$. Notice that if the Tracking Controller Type-2 is used then $\mathcal{M}_{\mathrm{c}}^{[i]}$ also includes $\dot{\gamma}^{[i]}$ (see (6.51)). For clarity of exposition, in the set-up exposed before we assume that these messages are exchanged continuously in time.

In practice, however, continuous communications are virtually impossible via wireless communication networks. A standard way is to transmit $\mathcal{M}_{\mathrm{c}}^{[i]}$ periodically and set the communication period as small as possible in order to achieve a performance comparable to that obtained using continuous communications. To make it simpler for a practical implementation, both messages $\mathcal{M}_{\mathrm{e}}^{[i]}$ and $\mathcal{M}_{\mathrm{c}}^{[i]}$ can be included in one package and transmitted at the same time, at every T_s. In this section, we propose a far more viable approach than continuous or periodic communications, that exploits the techniques of event-triggered communication (ETC) in [Battistelli et al., 2019,Hung et al., 2019]. Using this setup, the trackers exchange information when found necessary, without noticeable degradation of the performance achieved with the DEC developed in the previous section.

6.7.1 DEKF with event-triggered communications

We start by designing an ETC mechanism to decide when a generic tracker i must transmit the latest density about the target (stored in message $\mathcal{M}_{\mathrm{e}}^{[i]}$, see (6.41)) to its out-neighbors. Recall that in order to implement the DEKF each tracker i must access periodically to the latest local densities of the target computed by its in-neighbors (see Fusion step in Algorithm 6.2). The underlying idea behind the ETC mechanism is that if the tracker i can predict/estimate these densities well then there is no need for its in-neighbors to transmit the densities periodically. To this end, at each tracker i we define

$$\bar{p}^{[j]}(\mathbf{x}_k|\bar{\mathbf{x}}_{k-1}^{[j]}, \bar{P}_{k-1}^{[j]}) \sim \mathcal{N}(\bar{\mathbf{x}}_k^{[j]}, \bar{P}_k^{[j]}), \tag{6.64}$$

as the estimate of $p^{[j]}$ - the density of the target computed by tracker $j, j \in \mathcal{N}_{\mathrm{in}}^{[i]}$. The density $\bar{p}^{[j]}$ can be computed by propagating from the latest density about target received

from tracker j through the target model. Formally, let $\{k_l^{[j]}\}_{l\in\mathbb{N}}$ (to be determined by the ETC mechanism) denote the sequence of discrete time instants at which tracker j broadcasts target density $p^{[j]}$ (stored in message $\mathcal{M}_{\mathrm{e}}^{[j]}$). The density $\bar{p}^{[j]}$ is computed as follows.
At $k = k_{l+}^{[j]}$:

$$\bar{\mathbf{x}}_k^{[j]} = \tilde{\Omega}_k^{[j]}\tilde{\mathbf{z}}_k^{[j]}, \qquad \bar{P}_k^{[j]} = [\tilde{\Omega}_k^{[j]}]^{-1}, \tag{6.65}$$

where the pair $(\tilde{\mathbf{z}}_k^{[i]}, \tilde{\Omega}_k^{[i]})$ parameterizes $p^{[j]}$ - the latest density of the target computed by the tracker j (see how to compute this pair in (6.40)). In intervals $[k_l^{[j]}, k_{l+1}^{[j]}), l \in \mathbb{N}$, the density $\bar{p}^{[j]}$ propagates in an open loop manner through target's model, given by

$$\begin{aligned}\bar{\mathbf{x}}_{k+1}^{[j]} &= F\bar{\mathbf{x}}_k^{[j]}\\ \bar{P}_{k+1}^{[j]} &= F\bar{P}_k^{[j]}F^{\mathrm{T}} + Q.\end{aligned} \tag{6.66}$$

The above equations imply that whenever tracker i received the latest information about the target from tracker $j, j \in \mathcal{N}_{\text{in}}^{[i]}$, the density $\bar{p}^{[j]}$ will be updated with this latest information using (6.65). See an illustration of the evolution of the *estimated density* $\bar{p}^{[j]}$ and the *correct density* $p^{[j]}$ in Fig.6.14.

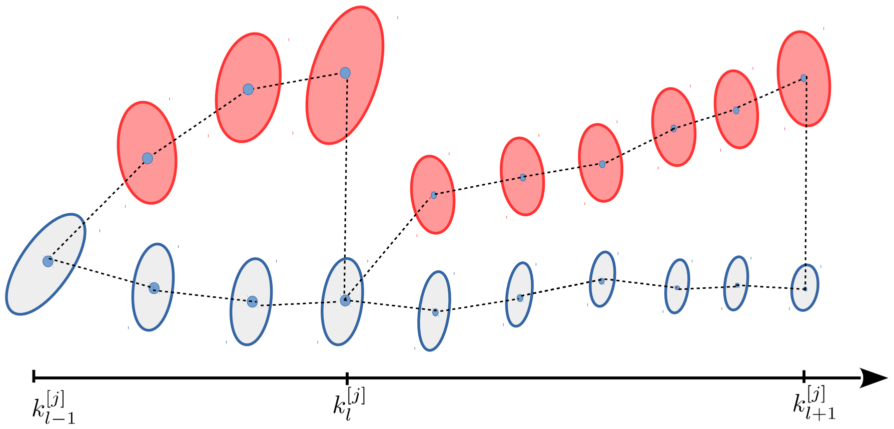


Figure 6.14: Illustration of the *estimated density* $\bar{p}^{[j]}$(red) and the latest *correct density* $p^{[j]}$ (gray). The *estimated density* is corrected (reset to the *correct density*) at every $k_l^{[j]}, l \in \mathbb{N}$.

In order to monitor "how well" the tracker i predict/estimate the density computed by the tracker j the tracker j builds run density *identical* to $\bar{p}^{[j]}$ that was built at tracker i with (6.65) and (6.66). Thus, the discrepancy between the estimated density $\bar{p}^{[j]}$ and the correct density $p^{[j]}$ at tracker i can be monitored by the tracker j as well. To quantify this discrepancy we adopt a measure, call the Kullback-Leibler Divergence (KLD), defined as

$$\begin{aligned}\mathcal{KLD}_k^{[j]}(p^{[j]}||\,\bar{p}^{[j]}) =& \frac{1}{2}\text{trace}\left([\bar{P}_k^{[j]}]^{-1}\tilde{P}_k^{[j]} - I_n\right) + ... \\ & \frac{1}{2}\left\|\bar{\mathbf{x}}_k^{[j]} - \tilde{\mathbf{x}}_k^{[j]}\right\|_{[\bar{P}_k^{[j]}]^{-1}} + \frac{1}{2}\log\left(\frac{\det(\bar{P}_k^{[j]})}{\det(\tilde{P}_k^{[j]})}\right),\end{aligned} \tag{6.67}$$

where $\tilde{P}_k^{[j]} = [\tilde{\Omega}_k^{[j]}]^{-1}$ and $\tilde{\mathbf{x}}_k^{[j]} = [\tilde{\Omega}_k^{[j]}]^{-1}\tilde{\mathbf{z}}_k^{[j]}$; and n is the dimension of $\bar{\mathbf{x}}_k^{[j]}$ which is identical to the dimension of the target's state [Battistelli et al., 2019]. To save communications, the tracker j keeps checking $\mathcal{KLD}_k^{[j]}$ and only transmits $p^{[j]}$ whenever this divergence exceeds a given value. Formally, the tracker j will transmit message $\mathcal{M}_\text{e}^{[j]}$ that stores $p^{[j]}$ whenever $\mathcal{KLD}_k^{[j]} \geq g^{[j]}(k)$ where $g^{[j]} : k \to \mathbb{R}_{\geq 0}$ is a positive threshold function that is designed to bound the difference between the two densities. Formally, the sequence $\{k_l^{[j]}\}_{l\in\mathbb{N}}$ is specified by

$$k_{l+1}^{[j]} = \inf\{k > k_l^{[j]} : \mathcal{KLD}_k^{[j]} \geq g^{[j]}(k)\} \tag{6.68}$$

for all $l \in \mathbb{N}$ and $i = 1, ..., N$. In summary, with the ETC mechanism presented above, the DEKF with event-triggered communication for a generic tracker i can be implemented with Algorithm 6.3. Note that the fusion law in this algorithm is computed with the estimated densities rather than the true densities as in Algorithm 6.2.

6.7.2 Cooperative pursuit with event-triggered communications

In this section, we propose an ETC mechanism to decide when a generic tracker i will transmit message $\mathcal{M}_\text{c}^{[i]}$ for the coordination control purposes. The ETC mechanism described in this section is in line with the work in [Hung et al., 2019,Hung et al., 2020b] and is presented next. Recall that to compute the correction speed $v_\text{c}^{[i]}$ given in (6.44), each tracker i needs to assess the variables $\gamma^{[j]}; j \in \mathcal{N}_\text{in}^{[i]}$ continuously. In order to overcome this unpractical requirement, tracker i will estimate these variables and use their estimates in (6.44) rather than the true ones. Let $\hat{\gamma}_i^{[j]}$ be the estimate of $\gamma^{[j]}$, estimated by tracker i. In the ETC mechanism, the cooperative control law for $v_\text{c}^{[i]}$ is given by

$$v_\text{c}^{[i]} = -k_c \sum_{j\in\mathcal{N}_\text{in}^{[i]}} (\gamma^{[i]} - \hat{\gamma}_i^{[j]}), \tag{6.70}$$

Algorithm 6.3 DEKF-ETC for tracker i

```
⋮
Similar to that of Algorithm 6.2
⋮
procedure Communication
    if $\mathcal{KLD}_k^{[i]}(p^{[i]}||\,\bar{p}^{[i]}) \geq g^{[i]}(k)$ then
        Broadcast message $\mathcal{M}_{\mathrm{e}}^{[i]}(k)$ given by (6.41).
        Update $\bar{\mathbf{x}}_k^{[i]}, \bar{P}_k^{[i]}$ using (6.65)
    End
procedure Fusion (Consensus)
    $\begin{aligned} \mathbf{z}_{k|k}^{[i]} &= \pi^{[i,i]}\tilde{\mathbf{z}}_k^{[i]} + \sum_{j\in\mathcal{N}_{\mathrm{in}}^{[i]}} \pi^{[i,j]}\bar{\mathbf{z}}_k^{[j]} \\ \Omega_{k|k}^{[i]} &= \pi^{[i,i]}\tilde{\Omega}_k^{[i]} + \sum_{j\in\mathcal{N}_{\mathrm{in}}^{[i]}} \pi^{[i,j]}\bar{\Omega}_k^{[j]} \end{aligned} \qquad (6.69)$
    return $\mathbf{z}_{k|k}^{[i]}, \Omega_{k|k}^{[i]}, \hat{\mathbf{x}}_{k|k}^{[i]} = [\Omega_{k|k}^{[i]}]^{-1}\mathbf{z}_{k|k}^{[i]}$
⋮
Similar to that of Algorithm 6.2
⋮
procedure Propagate density $\bar{p}^{[i]}$
    Compute $\{\bar{\mathbf{x}}_{k+1}^{[i]}, \bar{P}_{k+1}^{[i]}\}$ using (6.66)
return $\bar{\mathbf{x}}_{k+1}^{[i]}, \bar{P}_{k+1}^{[i]}$
```

where $k_c > 0$. To analyze the ETC mechanism, (6.70) can be rewritten as

$$v_{\mathrm{c}}^{[i]} = -k_c \sum_{j\in\mathcal{N}_{\mathrm{in}}^{[i]}} (\gamma^{[i]} - \gamma^{[j]} + \tilde{\gamma}^{[j]}), \tag{6.71}$$

where

$$\tilde{\gamma}^{[j]} \triangleq \gamma^{[j]} - \hat{\gamma}_i^{[j]}; \quad j \in \mathcal{N}_{\mathrm{in}}^{[i]} \tag{6.72}$$

is the estimation error of $\gamma^{[j]}$ computed at tracker i. It can be seen that compared with the control law for continuous communications in (6.44), $v_{\mathrm{c}}^{[i]}$ in (6.70) has contributions from the above error. The underlying idea in the proposed ETC mechanism is that if this error can be enforced to be bounded then, as we will show later, the coordination error $\boldsymbol{\xi}$ between the trackers will also be bounded. To make $\tilde{\gamma}^{[j]}; j \in \mathcal{N}_{\mathrm{in}}^{[i]}, i \in \mathcal{V}$ bounded, we define at each tracker j the variable $\hat{\gamma}^{[j]}; j \in \mathcal{V}$ as a "replica" of $\hat{\gamma}_i^{[j]}; i \in \mathcal{N}_{\mathrm{out}}^{[i]}$ and keep

them synchronized. To this end, their models are proposed as follows.
Let $\{t_n^{[i]}\}; n \in \mathbb{N}$ be the sequence of time instants at which tracker i sends its current value of $\gamma^{[i]}(t_n^{[i]}); n \in \mathbb{N}$ to its out-neighbors. Note that this sequence will be specified by the ETC mechanism. For $\mathcal{T}_n^{[i]} \triangleq [t_n^{[i]}, t_{n+1}^{[i]})$:

$$\dot{\hat{\gamma}}^{[i]}(t) = \bar{\omega}, \tag{6.73a}$$

$$\hat{\gamma}^{[i]}(t_n^{[i]}) = \gamma^{[i]}(t_n^{[i]}). \tag{6.73b}$$

for all $i \in \mathcal{N}$. Similarly, let $\{t_n^{[ji]}\}; n \in \mathbb{N}$ be the sequence of time instants at which tracker $j; j \in \mathcal{N}_{\text{out}}^{[i]}$ receives $\gamma^{[i]}(t_n^{[i]})$.
For $\mathcal{T}_n^{[ji]} \triangleq [t_n^{[ji]}, t_{n+1}^{[ji]})$:

$$\dot{\hat{\gamma}}_j^{[i]}(t) = \bar{\omega}, \tag{6.74a}$$

$$\hat{\gamma}_j^{[i]}(t_n^{[ji]}) = \gamma^{[i]}(t_n^{[i]}); i \in \mathcal{V}. \tag{6.74b}$$

With the above mechanism and provided there are no communication delays, i.e. $t_n^{[i]} = t_n^{[ji]}$ for all n and i, then from (6.73) and (6.74) we conclude that $\hat{\gamma}^{[i]}(t) = \hat{\gamma}_j^{[i]}(t)$ for all t, i.e. $\hat{\gamma}^{[i]}$ is a copy $\hat{\gamma}_j^{[i]}$. Thus, from (6.72), the estimation error can be expressed as $\tilde{\gamma}^{[i]} = \gamma^{[i]} - \hat{\gamma}^{[i]}$ for all $i \in \mathcal{V}$. To ensure that the estimation error $\gamma^{[i]}; i \in \mathcal{V}$ bounded, we allow tracker i to transmit $\gamma^{[i]}$ whenever this error hits a designed bounded threshold that, in general, can be parameterized by a function of time that we call $h^{[i]}(t)$. Formally, The sequence $\{t_n^{[i]}\}$ is specified by

$$t_{n+1}^{[i]} = \inf\{t > t_n^{[i]} : |\tilde{\gamma}^{[i]}(t)| \geq h^{[i]}(t)\}. \tag{6.75}$$

With the ETC mechanism, whenever the tracker makes a transmission $\hat{\gamma}^{[i]}$ will be immediately reset to $\gamma^{[i]}$ (see (6.73b)), it is guaranteed that

$$|\tilde{\gamma}^{[i]}(t)| \leq h^{[i]}(t) \tag{6.76}$$

for all t and $i \in \mathcal{V}$. Later, we will show that thank to (6.76) the coordination error among the trackers will always be bounded and the bound depends explicitly on $h^{[i]}; i \in \mathcal{V}$- the user designed functions to trade off performance of trackers' coordination and the cost of communications. In summary, the proposed ETC mechanism for the coordination purposes can be implemented using Algorithm 6.4.

Algorithm 6.4 Coordination with ETC mechanism for tracker i

```
At every time $t$, agent $i$ implements the following procedure:
procedure COORDINATION AND COMMUNICATION
    if $|\tilde{\gamma}^{[i]}(t)| \geq h^{[i]}(t)$ then
        Broadcast $\gamma^{[i]}(t)$;
        Reset $\hat{\gamma}^{[i]}$ using (6.73b);
    if Receive $\gamma^{[j]}$ from agent $j \in \mathcal{N}_{\text{in}}^{[i]}$ then
        Reset $\hat{\gamma}_j^{[i]}$ using (6.74b);
    Run the estimators (6.73) and (6.74);
    Compute $v_{\text{c}}^{[i]}$ using (6.70);
return $v_{\text{c}}^{[i]}$
```

6.7.3 Stability analysis of the complete DEC System with the event-triggered communication mechanisms

In order to draw a conclusion for the stability of the complete DEC system under the ETC mechanism, similar to Section 6.5.4 we examine the stability of sub-systems.

Coordination error system:

Firstly, we study the stability of the coordination system that is presented next. From (6.45) and (6.46) we obtain

$$\begin{aligned}\dot{\gamma}^{[i]} &= \bar{\omega} + e_{\gamma}^{[i]} + v_{\text{c}}^{[i]}\\ &\overset{(6.71)}{=} \bar{\omega} + e_{\gamma}^{[i]} - k_c \sum_{j \in \mathcal{N}_{\text{in}}^{[i]}} (\gamma^{[i]} - \gamma^{[j]} + \tilde{\gamma}^{[j]}).\end{aligned}$$

Recall that $\boldsymbol{\gamma} = [\gamma^{[1]}, ..., \gamma^{[N]}]^{\text{T}}$, thus

$$\dot{\boldsymbol{\gamma}} = \mathbf{1}\omega + \mathbf{e}_{\gamma} - k_c L \boldsymbol{\gamma} - k_c \mathcal{A} \tilde{\boldsymbol{\gamma}}, \tag{6.77}$$

where $\tilde{\boldsymbol{\gamma}} \triangleq [\tilde{\gamma}^{[1]}, ..., \tilde{\gamma}^{[N]}]^{\text{T}}$ and $\mathcal{A}$ is the adjacency matrix of the digraph. Now we consider again the coordination error vector $\boldsymbol{\xi}$ defined by (6.55). It's dynamics in the closed-loop system with ETC mechanism are given by

$$\begin{aligned}\dot{\boldsymbol{\xi}} &= W\dot{\boldsymbol{\gamma}} \overset{(6.77)}{=} W(\mathbf{1}\bar{\omega} + -k_c L \boldsymbol{\gamma} + \mathbf{e}_{\gamma} - k_c \mathcal{A} \tilde{\boldsymbol{\gamma}})\\ &= -k_c L \boldsymbol{\xi} + W\mathbf{e}_{\gamma} - k_c W \mathcal{A} \tilde{\boldsymbol{\gamma}}.\end{aligned} \tag{6.78}$$

With the ETC mechanism described above, we obtain the following result.

Lemma 6.4 (stability of coordination system)**.** *Consider the coordination error system under the ETC mechanism described by* (6.78)*. Assume further that the underlying communication graph is strongly connected. Then, the coordination error system is ISS with respect to the state $\boldsymbol{\xi}$ and the inputs $\mathbf{e}$ and $\mathbf{h} \triangleq [h^{[1]}, ..., h^{[N]}]^{\mathrm{T}}$.*

Proof: See section 6.10.5.

Remark 6.10. *Lemma 6.2 is a special case of Lemma 6.4 when triggering threshold functions $h^{[i]}(t) = 0$ for all t and $i \in \mathcal{V}$. In this case, $\mathbf{h} = \mathbf{0}$; as a consequence, the resulting coordination error system is ISS respect to only the input $\mathbf{e}$. From a communication standpoint, since the triggering thresholds are zero, communications are triggered continuously.*

Pursuit error system:

Notice that with the ETC mechanism, the tracking controllers given by (6.48) and (6.49) use the correction speed $v_{\mathrm{c}}^{[i]}$ in (6.70), instead of (6.44). Therefore, for Tracking Controller-Type II given by (6.49), the $\dot{v}_{c}^{[i]}$ in (6.51) is replaced by

$$\dot{v}_c^{[i]} = -k_c \sum_{j \in \mathcal{N}_{\mathrm{in}}^{[i]}} (\dot{\gamma}^{[i]} - \bar{\omega}), \tag{6.79}$$

which is obtained from (6.70) and (6.74a). As showed in Section 6.5.4 that the pursuit error system is independent with the correction speed $v_{\mathrm{c}}^{[i]}$. Thus, the complete pursuit error system given by (6.63) holds under the ETC mechanism, which leads to the following result.

Lemma 6.5 (Pursuit error system with ETC mechanism)**.** *Consider the complete pursuit error system* (6.63) *resulted by applying the tracking controllers Type I or Type II given by* (6.48) *and* (6.49)*, respectively with $v_{\mathrm{c}}^{[i]}$ given by* (6.70) *and $\dot{v}_{\mathrm{c}}^{[i]}$ is given by* (6.79)*. Then, the complete pursuit error system is ISS with respect to the state $\mathbf{e}$ and the inputs $\tilde{\mathbf{x}}$ and $\boldsymbol{\zeta}$, where $\tilde{\mathbf{x}}$ is the total estimation error of the target's state given by* (6.53)*.*

Proof: The proof Lemma 6.5 is similar to that of Lemma 6.3.
As a consequence, we obtain the following result for the stability of the whole DEC system.

Theorem 6.3. *Consider the closed-loop complete DEC system composed by the coordination error system and the pursuit error system stated in Lemma 6.4 and Lemma 6.5. Then, the complete DEC system is ISS respect to the state* $\boldsymbol{\mu} \triangleq [\boldsymbol{\xi}^{\mathrm{T}}, \mathbf{e}^{\mathrm{T}}]^{\mathrm{T}}$ *and the inputs* $\tilde{\mathbf{x}}$ *and* $\mathbf{h}$*, where* $\tilde{\mathbf{x}}$ *is the total estimation error of the target state given by* (6.53)*.*

Theorem 6.3 follows immediately from Lemma 6.4 and Lemma 6.5 and the stability of cascaded ISS systems [Sontag, 2008]. It can be seen that Theorem 6.2 is a special case of Theorem 6.3 where the $\mathbf{h} = \mathbf{0}$; which, from a communication standpoint, it triggers trackers communicate with their neighbors continuously to exchange messages $\mathcal{M}_{\mathrm{c}}^{[i]}$.

6.8 Simulation examples with ETC mechanisms

In this section we present simulation results illustrating the performance of the ETC mechanisms proposed in the previous section. In this simulation, three trackers are used to localize and pursue a target, in both the 2D and 3D cases, where the communications among them are driven by the ETC mechanisms. The trackers' communication network is set as in Fig.6.7, while all simulation parameters are identical to that used in Section 6.6, i.e. adopted from Table 6.1 and Table 6.2. Recall that in Section 6.6 communications among the trackers were done in a continuous manner for cooperative pursuit task and in a periodic manner for cooperative localization task. The threshold functions for the ETC mechanism described in Algorithm 6.3 is chosen as

$$g^{[i]}(k) = 15e^{-0.05k} + 3, \tag{6.80}$$

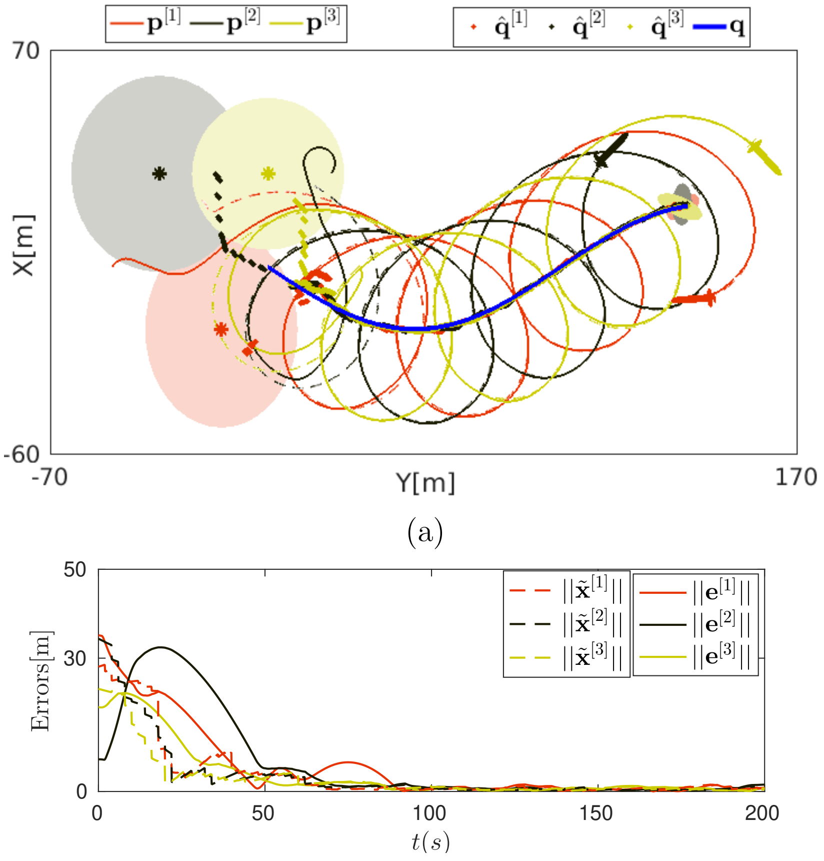
while the threshold function for the ETC mechanism described in Algorithm 6.4 is set as

$$h^{[i]}(t) = 2e^{-0.05t} + 0.1 \tag{6.81}$$

for all $i = 1, 2, 3$.

6.8.1 2-D example

Trackers and target's trajectories in the 2-D example are depicted in Fig.6.15. It is clearly shown in Fig.6.15(a) that the estimates of target's position converge to the target's trajectory, and all trackers converge to and encircle the target. This is in agreement with the results plotted in Fig.6.15(b), where it shows that all estimation and pursuit errors converge to neighborhoods of zero.



(a)

(b)

Figure 6.15: 2-D example with ETC mechanism: (a) The filled ellipsoids represent the uncertainties (covariances) of the target's position estimated by the trackers at the beginning and end of the simulation. Trajectories. $\mathbf{p}^{[i]}$: trackers, $\mathbf{p}_{\mathrm{d}}^{[i]}$: desired path, $\mathbf{q}$: target, $\hat{\mathbf{q}}^{[i]}$: estimated target. (b) Pursuit errors ($\mathbf{e}^{[i]}$) and localization errors ($\tilde{\mathbf{x}}^{[i]}$).

Fig.6.16 shows that the path parameters are synchronized and evolve nearly at the same speed about the desired speed $\bar{\omega}$. These imply that the trackers are coordinated along the S-T curves.

Fig.6.17 and Fig.6.18 illustrate communications between the trackers. Fig.6.17(a) shows time instants at which the trackers broadcast messages $\mathcal{M}_{\mathrm{c}}^{[i]} = \{\gamma^{[i]}\}$, which are associated with the cooperative pursuit task. It indicates that there are only few communications between the trackers which, according to Fig.6.17(b), they only happen when each estimation error $\tilde{\gamma}^{[i]}$ hits threshold function $h^{[i]}; i = 1, 2, 3$. For communications involved in the cooperative localization task, Fig.6.18 shows time instant at which the trackers broadcast messages $\mathcal{M}_{\mathrm{e}}^{[i]}$, which contain the local densities of the target estimated by each tracker. The figure shows that the trackers only transmit the messages to their neighbors whenever the Kullback-Leibler Divergence ($\mathcal{KLD}^{[i]}$) hit threshold functions $g^{[i]}; i = 1, 2.3$.

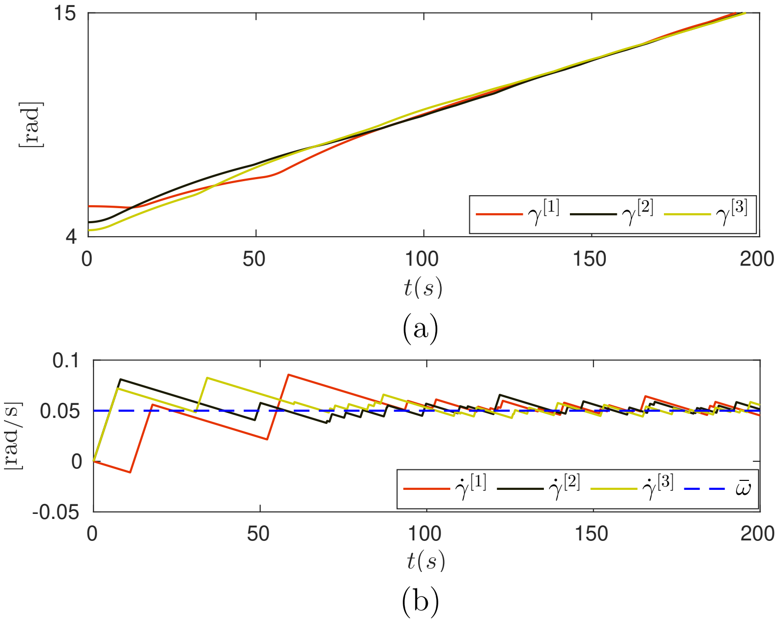
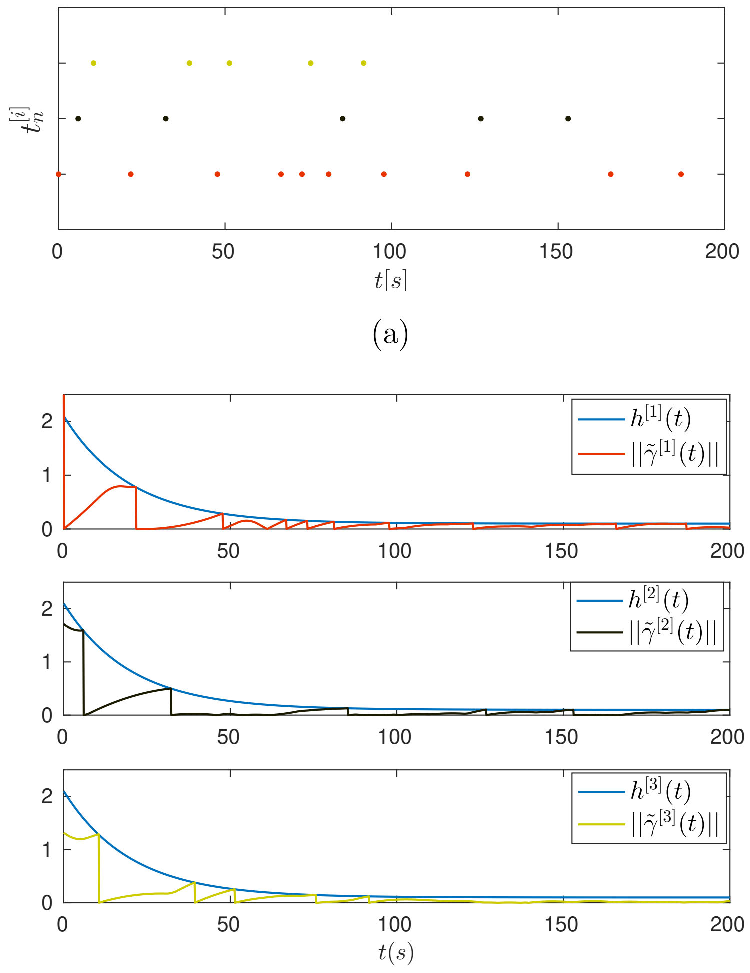


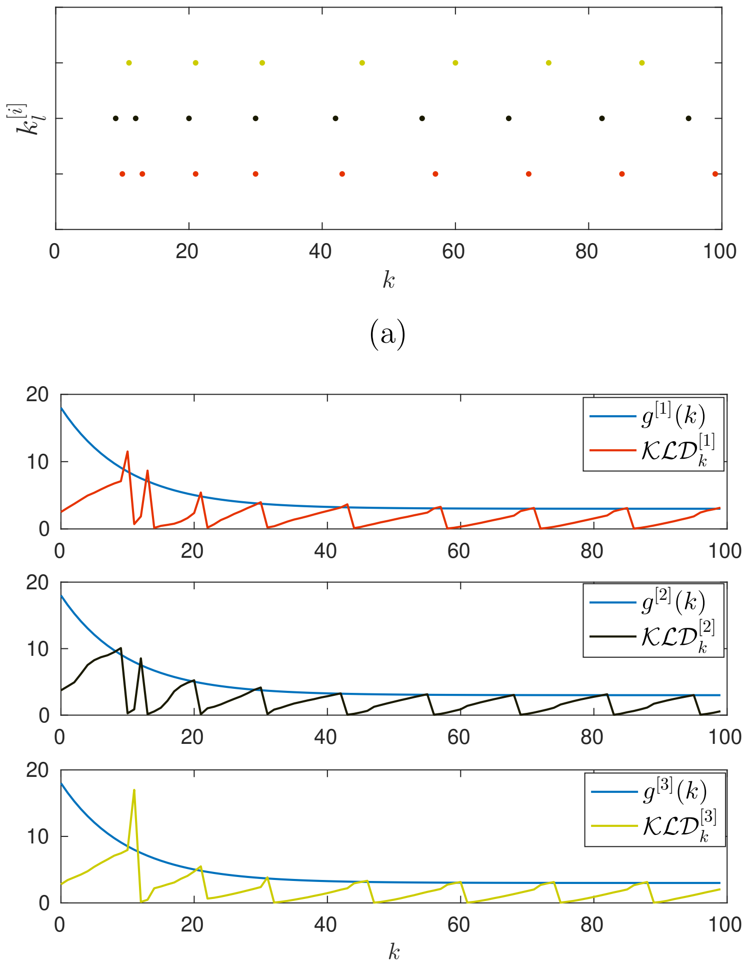
Figure 6.16: 2-D example with ETC mechanism. (a) Evolution of the coordination state $\gamma^{[i]}$. (b) Speeds of the coordination state $\dot{\gamma}^{[i]}$, $i = 1, 2, 3$.



(a)

(b)

Figure 6.17: 2-D example with ETC mechanism: Communications for cooperative tracking S-T curves. (a) Sequences of time instants at which the trackers broadcast messages $\mathcal{M}_{\rm c}^{[i]} = \{\gamma^{[i]}\}, i = 1, 2, 3$. (b) Evolutions of the estimation errors $\tilde{\gamma}^{[i]}$ and the threshold functions $h^{[i]}$.

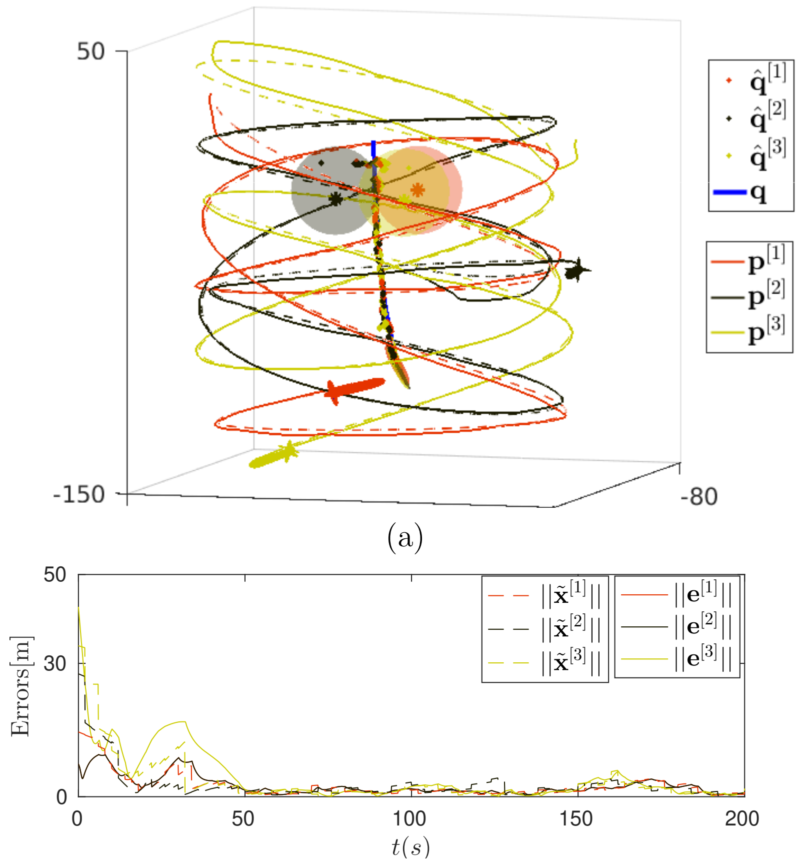


(a)

(b)

Figure 6.18: 2-D example with ETC mechanism: Communications for cooperative localization task (DEKF-ETC). (a) Sequences of discrete time instants at which each the tracker broadcasts $\mathcal{M}_{\mathrm{e}}^{[i]}$ (the local density about the target). (b) Evolutions of the Kullback-Leibler Divergence ($\mathcal{KLD}^{[i]}$), and the threshold functions $g^{[i]}$. Recall that ranges are taken at every k and $t = kT_s$ with $T_s = 2s$ in this simulation.

6.8.2 3-D example



(a)

(b)

Figure 6.19: 3-D example with ETC mechanism: (a) The filled ellipsoids represent the uncertainties (covariances) of the target's position estimated by the trackers at the beginning and end of the simulation. Trajectories. $\mathbf{p}^{[i]}$: trackers, $\mathbf{p}_{\mathrm{d}}^{[i]}$: desired path, $\mathbf{q}$: target, $\hat{\mathbf{q}}^{[i]}$: estimated target. (b) Pursuit errors ($\mathbf{e}^{[i]}$) and localization errors ($\tilde{\mathbf{x}}^{[i]}$).

The performance of the ETC mechanisms for the 3-D example is shown in Fig.6.19 - Fig.6.22. Fig.6.19 shows that all trackers converge to the desired S-T curves and estimate relatively well the trajectory of the target. Fig.6.20 indicates that the coordination states reach consensus and evolve almost the same speeds that are around the common desired speed ω. This implies that the trackers are coordinated along the desired S-T curves and fulfill the cooperative pursuit task. Fig.6.21 shows that in order to achieve the cooperative pursuit task the trackers only transmit their coordination states when found necessary, i.e. when the estimation errors ($\tilde{\gamma}^{[i]}$) exceed designed threshold functions ($h^{[i]}$). Similarly, Fig.6.22 indicates that the trackers only transmit their local densities about the target when the Kullback-Leibler Divergence $((\mathcal{KLD}^{[i]}))$, and the threshold functions $(g^{[i]})$. Compared with the simulation in Section 6.6, it is apparent that the ETC mechanisms reduce the number of communications among the trackers while they still guarantee an adequate performance of target localization and pursuit system.

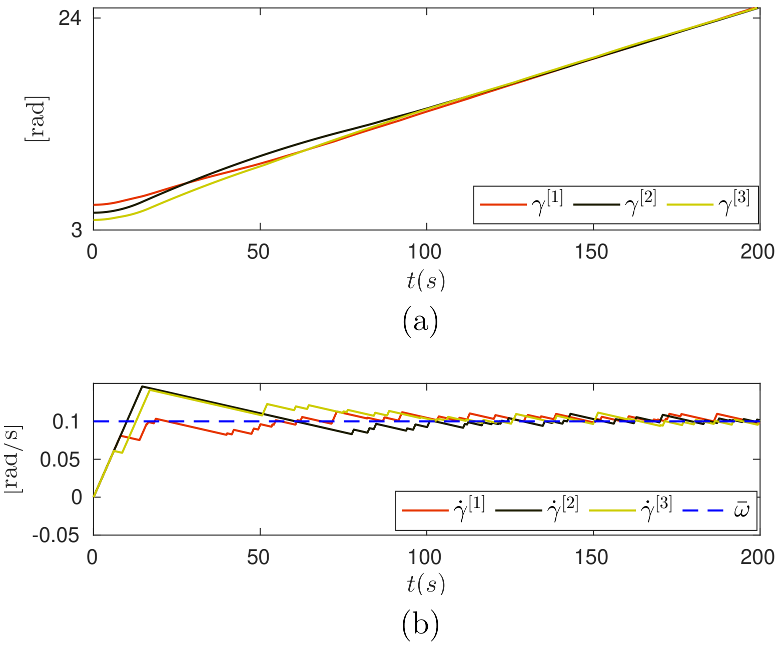
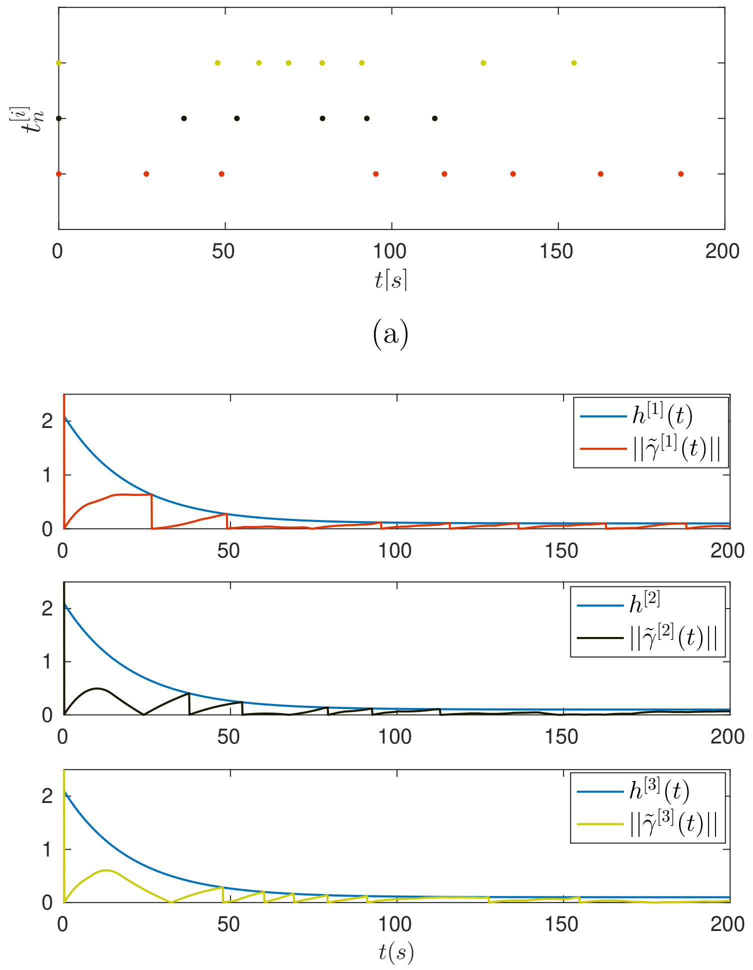


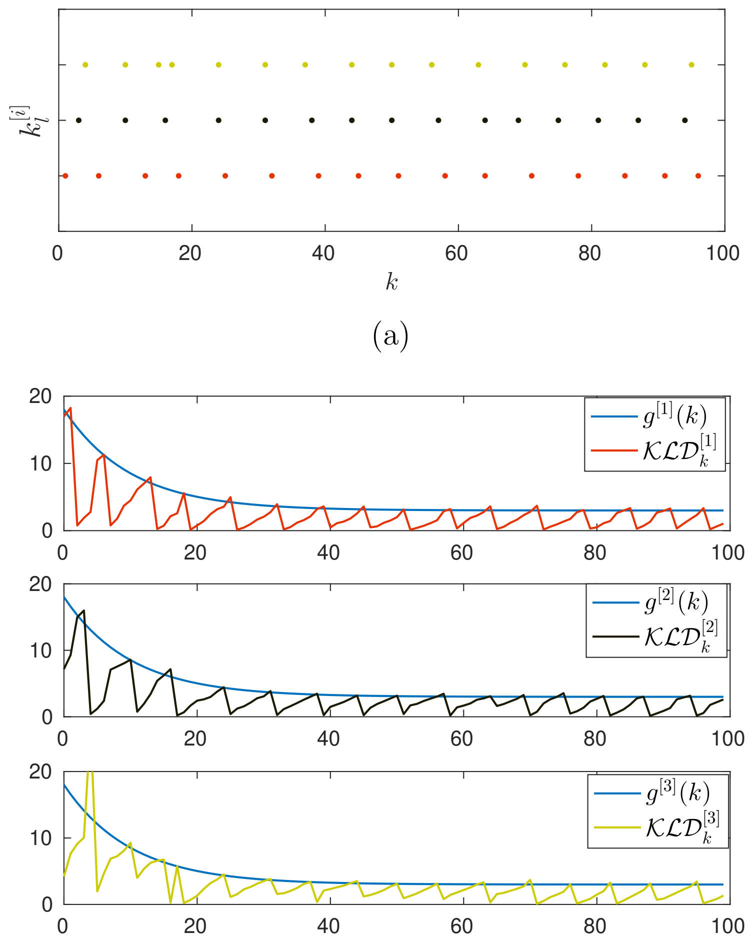
Figure 6.20: 3-D example with ETC mechanism. (a) Evolution of the coordination states $\gamma^{[i]}$. (b) Speeds of the coordination states $\dot{\gamma}^{[i]}$, $i = 1, 2, 3$.



(a)

(b)

Figure 6.21: 3-D example with ETC mechanism: Communications for cooperative tracking S-T curves. (a) Sequences of time instants at which the trackers broadcast $\gamma^{[i]}, i = 1, 2, 3$. (b) Evolutions of the estimation errors $\tilde{\gamma}^{[i]}$ and the threshold functions $h^{[i]}$.



(a)

(b)

Figure 6.22: 3-D example with ETC mechanism: Communications for cooperative localization task (DEKF-ETC). (a) Sequences of discrete time instants at which the trackers broadcast $\mathcal{M}_{\mathrm{e}}^{[i]}$ (the local density about the target), $i = 1, 2, 3$. (b) Evolutions of the Kullback-Leibler Divergence ($\mathcal{KLD}^{[i]}$), and the threshold functions $g^{[i]}$. Recall that ranges are taken at every k and $t = kT_s$ with $T_s = 2s$ in this simulation.

6.9 Conclusions

We proposed an integrated motion planning, control, and estimation framework to solve the problem of range-based simultaneous target localization and pursuit using one or multiple autonomous trackers. At the motion planning level, optimal tracker-target geometrical formations for target localization purposes were derived that led naturally to the concept of S-T curves with a hybrid temporal-spatial parameterization that serve as references for the desired motion of the trackers. This was followed by the derivation of two robust tracking controllers for S-T curve tracking. In the case of multiple trackers, an efficient distributed cooperative estimation and control strategy was proposed to deal with the constraints imposed by the inter-tracker communication network topology. The results of extensive simulations showed the robustness and efficacy of the proposed method. Stability analysis showed that the performance of the complete DEC system depends only on the convergence the EKF (DEKF for the case of multiple trackers) which, in practice, requires proper initialization of the estimated target's state and covariances. For the purpose of reducing communications among the vehicles, we then proposed ETC mechanisms for both cooperative estimation and cooperative control parts. With the ETC mechanisms, the vehicles only need to communicate with their neighbor when they found necessary, making the proposed distributed control and estimation strategy more efficient and useful for practical application. Future work aims at extending the proposed method to the case of multiple targets, with the main focus on multiple tracker motion planning.

6.10 Proofs

6.10.1 Proof of Lemma 6.1

We first prove the result for the controller given by (6.23). Clearly, with (6.23), $e_\gamma(t) = 0$ for all t. Consider the Lyapunov function candidate $V_1 = \frac{1}{2}\|\mathbf{e}_\mathrm{p}\|^2$, yielding $\dot{V}_1 = \mathbf{e}_\mathrm{p}\dot{\mathbf{e}}_\mathrm{p}$. Substituting (6.23) and (6.20) in $\dot{V}_1$, and noting that $\mathbf{e}_\mathrm{p}^\mathrm{T} S(\boldsymbol{\omega})\mathbf{e}_\mathrm{p} = 0$ for all $\mathbf{e}_\mathrm{p}$ and $\boldsymbol{\omega}$, we obtain $\dot{V}_1 = -\mathbf{e}_\mathrm{p}^\mathrm{T} K_\mathrm{p}\mathbf{e}_\mathrm{p} \leq -\lambda_2(K_\mathrm{p})\|\mathbf{e}_\mathrm{p}\|^2$ for all $\mathbf{e}_\mathrm{p}$. Thus, we conclude that $\mathbf{e}_\mathrm{p}$ converges to zero exponentially fast.

We now prove the theorem for the controller given by (6.24). Consider the Lyapunov function candidate KF

$$V_2(\mathbf{e}) = \|\mathbf{e}\|^2 = \frac{1}{2}\|\mathbf{e}_\mathrm{p}\|^2 + \frac{1}{2}e_\gamma^2. \tag{6.82}$$

Its time derivative is given by $\dot{V}_2 = \mathbf{e}_\mathrm{p}^\mathrm{T}\dot{\mathbf{e}}_\mathrm{p} + e_\gamma \dot{e}_\gamma$. Substituting (6.24) and (6.20) in $\dot{V}_2$ yields $\dot{V}_2 = -\mathbf{e}_\mathrm{p}^\mathrm{T} K_\mathrm{p} \mathbf{e}_\mathrm{p} - k_\gamma e_\gamma^2$. Let

$$K = \operatorname{diag}(K_\mathrm{p}, k_z). \tag{6.83}$$

Then, $\dot{V}_2 = -\mathbf{e}^\mathrm{T} K \mathbf{e} \leq -\lambda_{\min}(K)\|\mathbf{e}\|^2 < 0$ for all $\mathbf{e} \neq \mathbf{0}$. We conclude that the origin of $\mathbf{e}$ is GES. ■

6.10.2 Proof of Theorem 6.1

We only prove the theorem for the controller (6.28). The proof for the controller (6.27) is easier and can be done similarly. For the sake of clarity, let $\tilde{\mathbf{q}}$ and $\tilde{\boldsymbol{v}}$ be the target's position and velocity estimation errors, i.e. $\tilde{\mathbf{x}} = \operatorname{col}(\tilde{\mathbf{q}}, \tilde{\boldsymbol{v}})$. From (6.30), it follows that

$$\tilde{\mathbf{q}} = \hat{\mathbf{q}} - \mathbf{q}, \quad \tilde{\boldsymbol{v}} = \hat{\boldsymbol{v}} - \boldsymbol{v}. \tag{6.84}$$

From (6.18) and (6.26), we obtain

$$\hat{\mathbf{e}}_\mathrm{p} = \mathbf{e}_\mathrm{p} - R^\mathrm{T}(\boldsymbol{\eta})(\hat{\mathbf{q}} - \mathbf{q}) \overset{(6.84)}{=} \mathbf{e}_\mathrm{p} - R^\mathrm{T}(\boldsymbol{\eta})\tilde{\mathbf{q}}. \tag{6.85}$$

Further, substituting (6.28) in (6.20) and (6.22) yields

$$\begin{aligned} \dot{\mathbf{e}}_\mathrm{p} &= -\, S(\boldsymbol{\omega})\mathbf{e}_\mathrm{p} + R^\mathrm{T}(\boldsymbol{\eta})\left(\tilde{\boldsymbol{v}} - \mathbf{r}'(\gamma)e_\gamma\right) - K_\mathrm{p}\hat{\mathbf{e}}_\mathrm{p} \\ \dot{e}_\gamma &= -\, k_\gamma e_\gamma + \hat{\mathbf{e}}_\mathrm{p}^\mathrm{T} R^\mathrm{T}(\boldsymbol{\eta})\mathbf{r}'(\gamma). \end{aligned} \tag{6.86}$$

Consider again the Lyapunov function candidate V_2 given by (6.82). Taking the time derivative of V_2, and substituting (6.84)-(6.86) in $\dot{V}_2$, we obtain

$$\begin{aligned} \dot{V}_2 &= \mathbf{e}_\mathrm{p}^\mathrm{T}\dot{\mathbf{e}}_\mathrm{p} + e_\gamma \dot{e}_\gamma \\ &= -\mathbf{e}^\mathrm{T} K \mathbf{e} + \mathbf{e}_\mathrm{p}^\mathrm{T} R^\mathrm{T}(\boldsymbol{\eta})\tilde{\boldsymbol{v}} + \mathbf{e}_\mathrm{p}^\mathrm{T} K_\mathrm{p} R^\mathrm{T}(\boldsymbol{\eta})\tilde{\mathbf{q}} - e_\gamma \tilde{\mathbf{q}}^\mathrm{T}\mathbf{r}'(\gamma) \\ &= -\mathbf{e}^\mathrm{T} K \mathbf{e} + \mathbf{e}^\mathrm{T}\boldsymbol{\Phi} + \mathbf{e}_\mathrm{p}^\mathrm{T} K_\mathrm{p} R^\mathrm{T}(\boldsymbol{\eta})\tilde{\mathbf{q}}, \end{aligned} \tag{6.87}$$

where $\boldsymbol{\Phi} \triangleq \mathrm{col}\left(R^{\mathrm{T}}(\boldsymbol{\eta})\tilde{\boldsymbol{v}}, -\tilde{\mathbf{q}}^{\mathrm{T}}\mathbf{r}'(\gamma)\right)$. Furthermore,

$$\begin{aligned} \|\boldsymbol{\Phi}\|^2 &= \|R^{\mathrm{T}}(\boldsymbol{\eta})\tilde{\boldsymbol{v}}\|^2 + \left\|\tilde{\mathbf{q}}^{\mathrm{T}}\mathbf{r}'(\gamma)\right\|^2 \\ &\leq \|\tilde{\boldsymbol{v}}\|^2 + \|\mathbf{r}'(\gamma)\|^2\|\tilde{\mathbf{q}}\|^2 \\ &\leq \mu\|\tilde{\mathbf{x}}\|^2, \end{aligned} \tag{6.88}$$

where $\mu \triangleq \max_{\gamma}\{1, \|\mathbf{r}'(\gamma)\|^2\}$. Note that in (6.88) we used the fact that multiplying a vector on the left by a rotation matrix does not change the length of the vector. Furthermore, $\mathbf{e}_{\mathrm{p}}^{\mathrm{T}}K_{\mathrm{p}}R^{\mathrm{T}}(\boldsymbol{\eta})\tilde{\mathbf{q}} \leq \|\mathbf{e}\|\|K\|\|\tilde{\mathbf{x}}\|$. Substituting (6.88) in (6.87) yields

$$\begin{aligned} \dot{V}_2 \leq &-\lambda_{\min}(K)\|\mathbf{e}\|^2 + (\sqrt{\mu} + \|K\|)\|\mathbf{e}\|\|\tilde{\mathbf{x}}\| \\ \leq &-(1-\theta)\lambda_{\min}(K)\|\mathbf{e}\|^2, \quad \forall\|\mathbf{e}\| \geq (\sqrt{\mu} + \|K\|)\|\tilde{\mathbf{x}}\|/\theta \end{aligned} \tag{6.89}$$

with any $\theta \in (0,1)$. Invoking Theorem 4.19 in [Khalil, 2002], we conclude that the tracking error system is ISS respect to the state $\mathbf{e}$ and the input $\tilde{\mathbf{x}}$. This implies that there exist $\beta \in \mathcal{KL}$ and $\alpha \in \mathcal{K}$ functions such that (6.31) is satisfied. ■

6.10.3 Proof of Lemma 6.2

Consider a Lyapunov function candidate, defined as

$$V_{\mathrm{c}} = \boldsymbol{\xi}^{\mathrm{T}}R\boldsymbol{\xi}/2, \tag{6.90}$$

where $R \succ 0$ is a diagonal matrix defined in Lemma 2.1. The time derivative of the Lyapunov function is given by

$$\dot{V}_{\mathrm{c}} = \boldsymbol{\xi}^{\mathrm{T}}R\dot{\boldsymbol{\xi}} \overset{(6.57)}{=} -k_{\mathrm{c}}\boldsymbol{\xi}^{\mathrm{T}}RL\boldsymbol{\xi} + \boldsymbol{\xi}^{\mathrm{T}}RW\mathbf{e}_{\gamma}.$$

Because $\boldsymbol{\xi}$, defined in (6.54), is orthogonal to $\mathbf{1}$, using Definition 2.1, it follows that

$$\dot{V}_{\mathrm{c}} \leq -k_{\mathrm{c}}r_{\min}\|\boldsymbol{\xi}\|^2 + \|\boldsymbol{\xi}\|\|RW\|\|\mathbf{e}_{\gamma}\| \tag{6.91}$$

where $r_{\min} = \min_{i=1}^{N} r_i$. Because $\|\mathbf{e}_\gamma\| \leq \|\mathbf{e}\|$, we obtain

$$\begin{aligned} \dot{V}_{\mathrm{c}} \leq & -k_{\mathrm{c}} r_{\min} \|\boldsymbol{\xi}\|^2 + \|RW\| \|\mathbf{e}\| \\ \leq & -(1-\theta) k_{\mathrm{c}} r_{\min} \|\boldsymbol{\xi}\|^2 \qquad \forall \|\boldsymbol{\xi}\| \geq \frac{\|RW\|}{\theta k_{\mathrm{c}} r_{\min}} \|\mathbf{e}\| \end{aligned} \tag{6.92}$$

with any $\theta \in (0,1)$. Invoking Theorem 4.19 in [Khalil, 2002], we conclude that the coordination error system is ISS respect to the state $\boldsymbol{\xi}$ and the input $\mathbf{e}$. ■

6.10.4 Proof of Lemma 6.3

We only prove the result for the controller (6.49). The proof for the controller (6.48) is easier and can be done similarly. To this end, consider the Lyapunov function candidate V_{e}, defined as

$$V_{\mathrm{e}}(\mathbf{e}) = \frac{1}{2} \sum_{i=1}^{N} \|\mathbf{e}^{[i]}\|^2 = \frac{1}{2} \sum_{i=1}^{N} \|\mathbf{e}_{\mathrm{p}}^{[i]}\|^2 + (e_\gamma^{[i]})^2 \tag{6.93}$$

Note that with this definition

$$V_{\mathrm{e}}(\mathbf{e}) = \sum_{i=1}^{N} V_2(\mathbf{e}^{[i]}),$$

where V_2 is given by (6.82). Using a computation similar to that in the proof of Theorem 6.1, we obtain

$$\dot{V}_2(\mathbf{e}^{[i]}) \leq -\lambda_{\min}(K) \|\mathbf{e}^{[i]}\|^2 + (\sqrt{\mu} + \|K\|) \|\mathbf{e}^{[i]}\| \|\tilde{\mathbf{x}}^{[i]}\| \tag{6.94}$$

which is inherited from (6.89). Therefore,

$$\begin{aligned} \dot{V}_{\mathrm{e}} \leq \sum_{i=1}^{N} \dot{V}_2(\mathbf{e}^{[i]}) \leq & -\lambda_{\min}(K) \|\mathbf{e}\|^2 + (\sqrt{\mu} + \|K\|) \|\mathbf{e}\| \|\tilde{\mathbf{x}}\|, \\ \leq & -\lambda_{\min}(K)(1-\theta) \|\mathbf{e}\|^2, \quad \forall \|\mathbf{e}\| \geq \frac{\sqrt{\mu} + \|K\|}{\theta \lambda_{\min}(K)} \|\tilde{\mathbf{x}}\| \end{aligned} \tag{6.95}$$

with $\mathbf{e} \triangleq [\mathbf{e}^{[1]}, ..., \mathbf{e}^{[N]}]^{\mathrm{T}}$ and $\tilde{\mathbf{x}} = [\tilde{\mathbf{x}}^{[1]}, ..., \tilde{\mathbf{x}}^{[N]}]^{\mathrm{T}}$ and $\theta \in (0,1)$. Invoking Theorem 4.19 in [Khalil, 2002], we conclude that the complete DEC system is ISS respect to the state $\mathbf{e}$ and the input $\tilde{\mathbf{x}}$. ■

6.10.5 Proof of Lemma 6.4

The proof of this lemma is similar to the proof of Lemma 6.2. Consider the Lyapunov function candidate V_c given by (6.90). Taking its time derivative with $\dot{\boldsymbol{\xi}}$ given by (6.78) yields

$$\dot{V}_{\mathrm{c}} \leq -k_{\mathrm{c}} r_{\min} \|\boldsymbol{\xi}\|^2 + \|\boldsymbol{\xi}\| \left(\|RW\| \|\mathbf{e}_\gamma\| + k_c \|RW\mathcal{A}\| \|\tilde{\boldsymbol{\gamma}}\| \right) \tag{6.96}$$

Because of (6.76), $\|\boldsymbol{\gamma}\| \leq \|\mathbf{h}\|$, thus

$$\dot{V}_{\mathrm{c}} \leq -k_{\mathrm{c}} r_{\min} \|\boldsymbol{\xi}\|^2 + \|\boldsymbol{\xi}\| \left(\|RW\| \|\mathbf{e}_\gamma\| + k_c \|RW\mathcal{A}\| \|\mathbf{h}\| \right).$$

It can be seen that the above equation extends (6.91), thus, by proceeding similarly to that in the proof of Lemma 6.2, we conclude that the coordination system is ISS respect to the state $\boldsymbol{\xi}$ and the inputs $\mathbf{e}$ and $\mathbf{h}$. ■

6.10.6 Simulation parameters

Table 6.1: Parameters for simulations in 2-D

	Parameters
Target's trajectory	$\mathbf{q} = [20\sin(0.01t + \pi), 0.3t]^{\mathrm{T}}$ m
Range measurement	$\sigma = 0.5$m
S-T curve	$\bar{\omega} = 0.05$ rad/s $r_{\mathrm{x}}^{[i]} = r_{\mathrm{y}}^{[i]} = 30\text{m} \quad \forall i = 1, ..., 3$
Tracking Controller	$\boldsymbol{\epsilon} = [\text{-}0.5, 0]^{\mathrm{T}}, K_{\mathrm{p}} = \mathrm{diag}([0.4, 0.2]),\ k_\gamma = 500$
Coordination Controller	$k_c = 0.2$
DEKF	$V^{[i]} = 10 \quad \forall i = 1, ..., 3$ $Q = 10^{-3}\mathrm{diag}(1, 1, 0.1, 0.1)$ $\pi^i = \pi^j = 0.5 \quad \forall i = 1, ..., 3$ and $j \in \mathcal{N}_{\mathrm{in}}^{[i]}$ $T_{\mathrm{s}} = 2$s

Table 6.2: Parameters for simulations in 3-D

	Parameters
Targets trajectory	$\mathbf{q} = [30\sin(0.01t + \pi), 0.1t, \text{-}0.5t]^{\mathrm{T}}$ m
Range measurement	$\sigma = 0.5$m
S-T curve	$\bar{\omega} = 0.1$ rad/s $r_{\mathrm{x}}^{[i]} = r_{\mathrm{y}}^{[i]} = 80\mathrm{m}, r_{\mathrm{z}}^{[i]} = 50\mathrm{m} \quad \forall i = 1, ..., 3$
Tracking Controller	$\boldsymbol{\epsilon} = [\text{-}1, \text{-}1, 0]^{\mathrm{T}}, K_p = \mathrm{diag}([0.2, 0.2, 0.2]),$ $k_\gamma = 200$
Coordination Controller	$k_c = 0.2$
DEKF	$V^{[i]} = \sigma = 1 \quad \forall i = 1, ..., 3$ $Q = 10^{-3}\mathrm{diag}(1, 1, 1, 0.1, 0.1, 0.1)$ $\pi^i = \pi^j = 0.5 \quad \forall i = 1, ..., 3$ and $j \in \mathcal{N}_{\mathrm{in}}^{[i]}$ $T_s = 2$s

7

Conclusions

Contents

In this thesis, we addressed several problems in the field of cooperative control and estimation of multi agent systems under communication constraints, motivated by practical problems involved in the use of networked multiple autonomous vehicles/robots for scientific and engineering applications. The solutions/algorithms proposed in the thesis are distributed, whereby each agent (e.g. vehicle) only needs to exchange information with its neighbors, a subset of agents in the agents' communication network. Especially, the thesis addressed stringent communication constraints by developing event-triggered communication mechanisms for which, in order to achieve cooperative control and estimation tasks, the agents only communicate with their neighbors when found necessary, making the solutions proposed very efficient for practical implementation. This final chapter summarizes the contributions of the thesis and suggests a number of open problems that warrant future research work.

7.1 Summary

In Chapter 2 we proposed a distributed control algorithm with an event-triggered communication mechanism to solve the problem of consensus/synchronization of homogeneous nonlinear multi agent system. The ETC mechanism was shown to be a useful tool to reduce the number of messages exchanged among the agents and to trade off the frequency of communications among agents against the level of performance achieved in MAS consensus/synchronization.

Chapter 3 addressed the problem of cooperative path following of multiple constrained autonomous vehicles. In order to tackle this problem we proposed a control strategy which builds on tools from model predictive control, consensus/synchronization of networked multi agent systems, and the event-triggered communication mechanism proposed in Chapter 2. We showed that the solution proposed is not only capable of handling the vehicles' input constraints but also of reducing communications among the vehicles, making it efficient for practical implementation.

The second part of the thesis, presented from Chapter 4 to Chapter 6, addressed the problem of range-based simultaneous multiple target localization and pursuit. A qualitative method was proposed in Chapter 4 to address the observability problem of a target for the cases of single and multiple trackers. In this chapter, we proposed a systematic approach to derive a set of conditions on the trackers' motion (i.e. either on their inputs or trajectories) under which the target is observable. Based on those conditions derived, useful instructions for trackers' motion planning were given as well.

Chapter 5 proposed a quantitative approach to the solution of the range-based SLAP problem, which builds upon a tool from estimation theory, called the Bayesian Fisher Information Matrix, as a means to quantify the range-information acquired about the targets' states. By maximizing the FIM a set of optimal trajectories that the trackers must track to maximize the range-information was derived, giving intuitive and useful guidelines for planning the motions of the trackers.

Chapter 6 described a cooperative distributed estimation and control (DEC) strategy to solve the range-based SLAP problem in the case of a single target. At motion planning level, the solution proposed inherits the results in Chapter 4 and 5 to plan desired trajectories for the trackers to track. At control level, we derived robust cooperative and tracking controllers that were proved to have the capability of driving the trackers to the desired trajectories and also position themselves in an optimal relative geometry for the purpose of estimating the target's state. At the estimation level, we developed a distributed extended Kalman filter (DEKF) for the cooperative estimation of the target's state. We then proposed event-triggered communication mechanisms for the DEC strategy and showed that it is not only capable of reducing communications among the vehicles, but also guaranteeing an adequate performance achieved with continuous communications.

In what follows, we discuss a number of open problems related with the topics presented in this thesis that could serve as possible research directions.

7.2 Suggestions for future research

7.2.1 ETC mechanism for synchronization of a broader class of MAS

In Chapter 2 we developed an event-triggered communication mechanism for the state consensus/synchronization problem of homogeneous nonlinear MAS where all agents in the MAS network have identical dynamics, described by model (2.2). In order to achieve synchronization on the agents' trajectories we also assumed that the agents' states are measurable and can be exchanged among the agents, to be included in the control protocol. However, in many situations, the agents' dynamics are different and/or one may be interested in synchronizing the agents' outputs which are functions of the agents' states, rather than synchronizing the full agents' states. Such scenarios lead to a more challenging consensus/synchronization problem, called output synchronization of (heterogeneous) multi agent systems, see for example [Li et al., 2010, Scardovi and Sepulchre, 2009, Wieland et al., 2011, Isidori et al., 2014] and the references therein. To be more precise, let us extend the agents model in (2.2) to a more general form, given as

$$\begin{aligned} \dot{\mathbf{x}}_i &= \mathbf{f}_i(\mathbf{x}_i, \mathbf{u}_i), \\ \mathbf{y}_i &= \mathbf{h}_i(\mathbf{x}_i) \end{aligned} \tag{7.1}$$

for all $i = 1, ..., N$, where $\mathbf{x}_i \in \mathbb{R}^{n_i}, \mathbf{u}_i \in \mathbb{R}^{m_i}$ and $\mathbf{y}_i \in \mathbb{R}^{p_i}$ respectively represent the state, input, and output of each agent i and $\mathbf{f}_i \in \mathbb{R}^{n_i}$, $\mathbf{h}_i \in \mathbb{R}^{p_i}$ are (nonlinear) functions. Supposing that communications among the agents are supported by an inter-agent communication network, the output synchronization problem is defined as that of finding a distributed control protocol for $\mathbf{u}_i$ such that

$$\lim_{t\to\infty} (\mathbf{y}_i(t) - \mathbf{y}_j(t)) = \mathbf{0} \quad \forall i, j = 1, ..., N. \tag{7.2}$$

Notice that the agents' model considered in Chapter 2, given by (2.2), is a special case of (7.1) with $\mathbf{f}_i(\mathbf{x}_i, \mathbf{u}_i) = A\mathbf{x}_i + \mathbf{f}(\mathbf{x}_i, t) + B\mathbf{u}_i$ and $\mathbf{h}_i(\mathbf{x}_i) = \mathbf{x}_i$ for all $i = 1, ..., N$ where the dimensions of $A, B, \mathbf{f}$ are identical for all agents.

We believe that developing new event-triggered communication mechanisms for the output synchronization problem of the general MAS (7.1) to reduce communications among the agents is an interesting and extremely challenging topic for future research. The ETC mechanism proposed in the thesis may be still useful and can be exploited for future work because it builds on the intuitive idea that *"if an agent can predict well the neighbors' information (e.g. neighbors' states) that is necessary to compute a cooperative control protocol then it is not required to access that information continuously"*, thus saving communications among the agents. Exploiting this idea and combining it with existing results with continuous communications for output synchronization of linear heterogeneous MAS ([Wieland et al., 2011]) and nonlinear heterogeneous MAS([Isidori et al., 2014, Zhu et al., 2016, Chowdhury and Khalil, 2021] could be a good starting point towards tackling the general output synchronization problem stated above.

Other challenging problems deserving further investigation in the context of event-triggered communications arise in the context of MAS networks exhibiting time varying topology and temporary communication losses.

7.2.2 Distributed solution for range-based cooperative SLAP of multiple targets

In Chapter 5 we proposed a centralized planning, control, and estimation strategy to solve the problem of range-based SLAP of *multiple targets*. This strategy builds upon the concept of model predictive control, the implementation of which can be summarized in the following steps: i) first, all trackers transmit their local information (i.e. their states and ranges to targets) to a central node, ii) this node then estimates the densities of the targets' states using a centralized filter (e.g. a centralized EKF), iii) afterwards, based on the trackers' states received and the estimated targets' densities, the central node searches for optimal trackers' inputs (e.g. their linear speeds and course rates) in their admissible input spaces with the objective of maximizing the range-information (embodied in a constructed Bayesian FIM) for the estimation of the targets' states iv) in the final step, the central node transmits these optimal inputs to the trackers as motion control references. These steps are repeated at every sampling interval, thus featuring the concept of model predictive control. Note that the third step requires the central node to solve a nonlinear optimization problem with the main objective of maximizing the determinant of the Bayesian FIM, subject to trackers' model and input constraints.

Conceptually, the centralized planing, control, and estimation strategy proposed in Chapter 5 can be considered as an "optimal" approach to the range-based SLAP problem since it drives the trackers in an "optimal" way to acquire maximal range information for the estimation of targets' stare. However, from a practical standpoint, implementing it in a centralized manner may be inefficient and not scalable. A possible direction to continue this research is to decentralize this strategy, making it fully distributed so that the trackers only need to cooperate with their neighbors to estimate target's state and to move, instead of depending on the central node. However, any decentralization strategy must be taken at two levels i) the first involves decentralization of the targets' state estimation task and ii) the second is associated with decentralization of planning and control of the trackers. We believe that the first was already solved with the distributed EKF proposed in Chapter 6, and thus the main challenge comes at the planing and control level. Decentralizing the control and planing task will certainly lead to an interesting but challenging problem in the context of distributed model predictive control (DMPC), or in a broader field, namely, distributed optimization. Recent work and surveys on fast distributed MPC [Van Parys, 2018, Negenborn and Maestre, 2014] and distributed optimization [YAN, 2019] may provide useful hints to tackle this problem.

Recall that in Chapter 6 we proposed an efficient cooperative distributed control, estimation strategy to the cooperative range-based SLAP problem with *a single target*. The solution built on preliminary results regarding the types of optimal trajectories (obtained in Chapter 5) that the trackers must track to acquire maximal range-information for the estimation of the target's state. It seems plausible to extend this idea to the case of SLAP of *multiple targets*. However, a challenge arises from the requirement to plan optimal trajectories for the trackers to ensure that the range information acquired along the planned trajectories is maximal for multiple target estimation purposes. This certainly deserves further research effort.

Bibliography

[YAN, 2019] (2019). A survey of distributed optimization. *Annual Reviews in Control*, 47:278 – 305.

[Abreu et al., 2016a] Abreu, P., Antonelli, G., Arrichiello, F., Caffaz, A., Caiti, A., Casalino, G., Volpi, N. C., De Jong, I. B., De Palma, D., Duarte, H., et al. (2016a). Widely scalable mobile underwater sonar technology: An overview of the h2020 wimust project. *Marine Technology Society Journal*, 50(4):42–53.

[Abreu et al., 2016b] Abreu, P., Morishita, H., Pascoal, A., Ribeiro, J., and Silva, H. (2016b). Marine vehicles with streamers for geotechnical surveys: Modeling, positioning, and control. *IFAC-PapersOnLine*, 49(23):458 – 464. 10th IFAC Conference on Control Applications in Marine SystemsCAMS 2016.

[Abreu et al., 2016c] Abreu, P. C., Botelho, J., Góis, P., Pascoal, A., Ribeiro, J., Ribeiro, M., Rufino, M., Sebastião, L., and Silva, H. (2016c). The medusa class of autonomous marine vehicles and their role in eu projects. In *OCEANS 2016*, pages 1–10. IEEE.

[Aguiar and Hespanha, 2007] Aguiar, A. P. and Hespanha, J. P. (2007). Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty. *IEEE Transactions on Automatic Control*, 52(8):1362–1379.

[Aguiar and Pascoal, 2007] Aguiar, A. P. and Pascoal, A. M. (2007). Coordinated path-following control for nonlinear systems with logic-based communication. In *46th Decision and Control*, pages 1473–1479. IEEE.

[Alessandretti and Aguiar, 2017] Alessandretti, A. and Aguiar, A. P. (2017). A distributed model predictive control scheme for coordinated output regulation. In *IFAC-PapersOnLine*, volume 50, pages 8692 – 8697.

[Almeida et al., 2012] Almeida, J., Silvestre, C., and Pascoal, A. (2012). Cooperative control of multiple surface vessels with discrete-time periodic communications. *International Journal of Robust and Nonlinear Control*, 22(4):398–419.

[Almeida et al., 2017] Almeida, J., Silvestre, C., and Pascoal, A. (2017). Synchronization of multiagent systems using event-triggered and self-triggered broadcasts. *IEEE Transactions on Automatic Control*, 62(9):4741–4746.

[Andersson, 2013] Andersson, J. (2013). *A General-Purpose Software Framework for Dynamic Optimization*. PhD thesis, Arenberg Doctoral School, KU Leuven.

[Andersson et al., 2018] Andersson, J. A. E., Gillis, J., Horn, G., Rawlings, J. B., and Diehl, M. (In Press, 2018). CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*.

[Arrichiello et al., 2013] Arrichiello, F., Antonelli, G., Aguiar, A., and Pascoal, A. (2013). An observability metric for underwater vehicle localization using range measurements. *Sensors*, 13(12):16191–16215.

[Batista et al., 2011] Batista, P., Silvestre, C., and Oliveira, P. (2011). Single range aided navigation and source localization- observability and filter design. *Systems and Control Letters*, 60(8):665 – 673.

[Battistelli and Chisci, 2016] Battistelli, G. and Chisci, L. (2016). Stability of consensus extended kalman filter for distributed state estimation. *Automatica*, 68:169 – 178.

[Battistelli et al., 2019] Battistelli, G., Chisci, L., Gao, L., and Selvi, D. (2019). Event-triggered distributed bayes filter. In *2019 18th European Control Conference (ECC)*, pages 2731–2736.

[Bayat et al., 2016] Bayat, M., Crasta, N., Aguiar, A. P., and Pascoal, A. M. (2016). Range-based underwater vehicle localization in the presence of unknown ocean currents: Theory and experiments. *IEEE Transactions on Control Systems Technology*, 24(1):122–139.

[Bibuli et al., 2009] Bibuli, M., Bruzzone, G., Caccia, M., and Lapierre, L. (2009). Path-following algorithms and experiments for an unmanned surface vehicle. *Journal of Field Robotics*, 26(8).

[Bishop et al., 2010] Bishop, A. N., Fidan, B., Anderson, B. D., Dogancay, K., and Pathirana, P. N. (2010). Optimality analysis of sensor-target localization geometries. *Automatica*, 46(3):479 – 492.

[Boyd et al., 2004] Boyd, S., Boyd, S. P., and Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press.

[Bullo, 2018] Bullo, F. (2018). *Lectures on Network Systems*. CreateSpace Independent Publishing Platform.

[Cao et al., 2017] Cao, K.-C., Jiang, B., and Yue, D. (2017). Cooperative path following control of multiple nonholonomic mobile robots. *ISA transactions*, 71:161–169.

[Chaudhary et al., 2016] Chaudhary, G., Sinha, A., Tripathy, T., and Borkar, A. (2016). Conditions for target tracking with range-only information. *Robotics and Autonomous Systems*, 75:176 – 186.

[Chen, 1984] Chen, C. (1984). Linear system theory and design.

[Chowdhury and Khalil, 2021] Chowdhury, D. and Khalil, H. K. (2021). Practical synchronization in networks of nonlinear heterogeneous agents with application to power systems. *IEEE Transactions on Automatic Control*, 66(1):184–198.

[Cichella et al., 2015] Cichella, V., Kaminer, I., Dobrokhodov, V., Xargay, E., Choe, R., Hovakimyan, N., Aguiar, A. P., and Pascoal, A. M. (2015). Cooperative path following of multiple multirotors over time-varying networks. *IEEE Transactions on Automation Science and Engineering*, 12(3):945–957.

[Clark et al., 2013] Clark, C. M., Forney, C., Manii, E., Shinzaki, D., Gage, C., Farris, M., Lowe, C. G., and Moline, M. (2013). Tracking and following a tagged leopard shark with an autonomous underwater vehicle. *Journal of Field Robotics*, 30(3):309–322.

[Crasta et al., 2018] Crasta, N., Moreno-Salinas, D., Pascoal, A., and Aranda, J. (2018). Multiple autonomous surface vehicle motion planning for cooperative range-based underwater target localization. *Annual Reviews in Control*, 46:326 – 342.

[de la Pena and Christofides, 2008] de la Pena, D. M. and Christofides, P. D. (2008). Lyapunov-based model predictive control of nonlinear systems subject to data losses. *IEEE Transactions on Automatic Control*, 53(9):2076–2089.

[Degroot, 1974] Degroot, M. H. (1974). Reaching a consensus. *Journal of the American Statistical Association*, 69(345):118–121.

[Diestel, 2005] Diestel, R. (2005). *Graph theory. 2005*, volume 101.

[Dimarogonas et al., 2012] Dimarogonas, D. V., Frazzoli, E., and Johansson, K. H. (2012). Distributed event-triggered control for multi-agent systems. *IEEE Transactions on Automatic Control*, 57(5):1291–1297.

[Dong et al., 2018] Dong, Y., Zha, Q., Zhang, H., Kou, G., Fujita, H., Chiclana, F., and Herrera-Viedma, E. (2018). Consensus reaching in social network group decision making: Research paradigms and challenges. *Knowledge-Based Systems*, 162:3 – 13. Special Issue on intelligent decision-making and consensus under uncertainty in inconsistent and dynamic environments.

[Dunbar and Murray, 2006] Dunbar, W. B. and Murray, R. M. (2006). Distributed receding horizon control for multi-vehicle formation stabilization. *Automatica*, 42(4):549 – 558.

[Fan et al., 2015] Fan, Y., Liu, L., Feng, G., and Wang, Y. (2015). Self-triggered consensus for multi-agent systems with zeno-free triggers. *IEEE Transactions on Automatic Control*, 60(10):2779–2784.

[Farina et al., 2012] Farina, M., Ferrari-Trecate, G., and Scattolini, R. (2012). Distributed moving horizon estimation for nonlinear constrained systems. *International Journal of Robust and Nonlinear Control*, 22(2):123–143.

[Fossen, 2011] Fossen, T. I. (2011). *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons.

[Garcia et al., 2014] Garcia, E., Cao, Y., and Casbeer, D. W. (2014). Decentralized event-triggered consensus with general linear dynamics. *Automatica*, 50(10):2633–2640.

[Garcia et al., 2017] Garcia, E., Cao, Y., and Casbeer, D. W. (2017). Periodic event-triggered synchronization of linear multi-agent systems with communication delays. *IEEE Transactions on Automatic Control*, 62(1):366–371.

[Garcia et al., 2013] Garcia, E., Cao, Y., Yu, H., Antsaklis, P., and Casbeer, D. (2013). Decentralised event-triggered cooperative control with limited communication. *International Journal of Control*, 86(9):1479–1488.

[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.

[Grant and Boyd, 2014] Grant, M. and Boyd, S. (2014). CVX: Matlab software for disciplined convex programming, version 2.1.

[Hermann and Krener, 1977] Hermann, R. and Krener, A. (1977). Nonlinear controllability and observability. *IEEE Transactions on automatic control*, 22(5):728–740.

[Horn and Johnson, 2012] Horn, R. A. and Johnson, C. R. (2012). *Matrix Analysis*. Cambridge University Press, 2rd edition.

[Hung et al., 2020a] Hung, N. T., Crasta, N., Moreno-Salinas, D., Pascoal, A. M., and Johansen, T. A. (2020a). Range-based target localization and pursuit with autonomous vehicles: An approach using posterior crlb and model predictive control. *Robotics and Autonomous Systems*, 132:103608.

[Hung and Pascoal, 2018] Hung, N. T. and Pascoal, A. M. (2018). Cooperative path following of autonomous vehicles with model predictive control and event triggered communications. *IFAC-PapersOnLine*, 51(20):562 – 567. 6th IFAC Conference on Nonlinear Model Predictive Control NMPC 2018.

[Hung and Pascoal, 2020a] Hung, N. T. and Pascoal, A. M. (2020a). Consensus/synchronisation of networked nonlinear multiple agent systems with event-triggered communications. *International Journal of Control*, 0(0):1–10.

[Hung and Pascoal, 2020b] Hung, N. T. and Pascoal, A. M. (2020b). Range-based navigation and target localization-observability analysis and guidelines for motion planning. *IFAC-PapersOnLine*, -(-):–. 21st IFAC World Congress.

[Hung et al., 2020b] Hung, N. T., Pascoal, A. M., and Johansen, T. A. (2020b). Cooperative path following of constrained autonomous vehicles with model predictive control and event triggered communications. *International Journal of Robust and Nonlinear Control*.

[Hung et al., 2018] Hung, N. T., Rego, F., Crasta, N., and Pascoal, A. (2018). Input-constrained path following for autonomous marine vehicles with a global region of attraction. *IFAC-PapersOnLine*, 51(29):348 – 353. 11th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles CAMS 2018.

[Hung et al., 2021] Hung, N. T., Rego, F., and Pascoal, A. M. (2021). Distributed control and estimation for range-based target localization and pursuit. *IEEE Transaction on Control System and Technology*, Accepted conditionally with minor revision.

[Hung et al., 2019] Hung, N. T., Rego, F. C., and Pascoal, A. M. (June 2019). Event-triggered communications for the synchronization of nonlinear multi agent systems on weight-balanced digraphs. 2019 18th European Control Conference (ECC), pages 2713–2718. IEEE.

[Indiveri et al., 2016] Indiveri, G., De Palma, D., and Parlangeli, G. (2016). Single range localization in 3-d: Observability and robustness issues. *IEEE Transactions on Control Systems Technology*, 24(5):1853–1860.

[Indiveri et al., 2012] Indiveri, G., Pedone, P., , and Cuccovillo, M. (2012). Fixed target 3d localization based on range data only: a recursive least squares approach. *IFAC Proceedings Volumes*, 45(5):140 – 145. 3rd IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles.

[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.

[Jain et al., 2018] Jain, R. P., Aguiar, A. P., and de Sousa, J. B. (2018). Cooperative path following of robotic vehicles using an event-based control and communication strategy. *IEEE Robotics and Automation Letters*, 3(3):1941–1948.

[Jauffret et al., 2017] Jauffret, C., Pérez, A., and Pillon, D. (2017). Observability: Range-only versus bearings-only target motion analysis when the observer maneuvers smoothly. *IEEE Transactions on Aerospace and Electronic Systems*, 53(6):2814–2832.

[Jiang et al., 2001] Jiang, Z.-P., Lefeber, E., and Nijmeijer, H. (2001). Saturated stabilization and tracking of a nonholonomic mobile robot. *Systems & Control Letters*, 42(5):327–332.

[Kaminer et al., 2017] Kaminer, I., Pascoal, A., Xargay, E., Hovakimyan, N., Cichella, V., and Dobrokhodov, V. (2017). *Time-Critical Cooperative Control of Autonomous Air Vehicles*. Butterworth-Heinemann.

[Kebkal et al., 2017] Kebkal, K., Kebkal, O., Glushko, E., Kebkal, V., Sebastiao, L., Pascoal, A., Gomes, J., Ribeiro, J., SIlva, H., Ribeiro, M., et al. (2017). Underwater acoustic modems with integrated atomic clocks for one-way travel-time underwater vehicle positioning. 2017 Underwater Acoustics Conference and Exhibition.

[Khalil, 2002] Khalil, H. K. (2002). Nonlinear systems, 3rd. *New Jewsey, Prentice Hall*, 9(4.2).

[Klausen et al., 2015] Klausen, K., Fossen, T. I., Johansen, T. A., and Aguiar, A. P. (2015). Cooperative path-following for multirotor uavs with a suspended payload. Control Applications (CCA), 2015 IEEE Conference on, pages 1354–1360. IEEE.

[Klemas, 2015] Klemas, V. V. (2015). Coastal and environmental remote sensing from unmanned aerial vehicles: An overview. *Journal of Coastal Research*, 31(5):1260–1267.

[Lapierre et al., 2006] Lapierre, L., Soetanto, D., and Pascoal, A. (2006). Nonsingular path following control of a unicycle in the presence of parametric modelling uncertainties. *International Journal of Robust and Nonlinear Control*, 16(10):485–503.

[Li et al., 2016] Li, H., Chen, G., Huang, T., Zhu, W., and Xiao, L. (2016). Event-triggered consensus in nonlinear multi-agent systems with nonlinear dynamics and directed network topology. *Neurocomputing*, 185:105–112.

[Li and Duan, 2015] Li, Z. and Duan, Z. (2015). *Cooperative Control of Multi-Agent Systems*. CRC Press.

[Li et al., 2010] Li, Z., Duan, Z., Chen, G., and Huang, L. (2010). Consensus of multi-agent systems and synchronization of complex networks: A unified viewpoint. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 57(1):213–224.

[Li et al., 2012] Li, Z., Liu, X., Fu, M., and Xie, L. (2012). Global H_∞ consensus of multi-agent systems with lipschitz non-linear dynamics. *IET Control Theory & Applications*, 6(13):2041–2048.

[Liuzza et al., 2016] Liuzza, D., Dimarogonas, D. V., Di Bernardo, M., and Johansson, K. H. (2016). Distributed model based event-triggered control for synchronization of multi-agent systems. *Automatica*, 73:1–7.

[Loría and Panteley, 2005] Loría, A. and Panteley, E. (2005). 2 cascaded nonlinear time-varying systems: Analysis and design. In *Advanced topics in control systems theory*, pages 23–64. Springer.

[Mandić et al., 2016] Mandić, F., Rendulić, I., Mivsković, N., and Dula, N. (2016). Underwater object tracking using sonar and usbl measurements. *Journal of Sensors*, 2016.

[Manuel and Bishop, 2014] Manuel, I. L. and Bishop, A. N. (2014). Distributed Monte Carlo information fusion and distributed particle filtering. *IFAC Proceedings Volumes*, 47(3):8681 – 8688. 19th IFAC World Congress.

[Martinez and Bullo, 2006] Martinez, S. and Bullo, F. (2006). Optimal sensor placement and motion coordination for target tracking. *Automatica*, 42(4):661 – 668.

[Martinez et al., 2007] Martinez, S., Cortes, J., and Bullo, F. (2007). Motion coordination with distributed information. *IEEE Control Systems Magazine*, 27(4):75–88.

[Masmitja et al., 2017] Masmitja, I., Bouvet, P. J., Gomariz, S., Aguzzi, J., and del Rio, J. (2017). Underwater mobile target tracking with particle filter using an autonomous vehicle. In *OCEANS 2017 - Aberdeen*, pages 1–5.

[Masmitja et al., 2019] Masmitja, I., Gomariz, S., Del-Rio, J., Kieft, B., O'Reilly, T., Bouvet, P., and Aguzzi, J. (2019). Range-only single-beacon tracking of underwater targets from an autonomous vehicle: From theory to practice. *IEEE Access*, 7:86946–86963.

[Masmitja et al., 2018] Masmitja, I., Gomariz, S., Del-Rio, J., Kieft, B., O'Reilly, T., Bouvet, P.-J., and Aguzzi, J. (2018). Optimal path shape for range-only underwater target localization using a wave glider. *The International Journal of Robotics Research*, 37(12):1447–1462.

[Meng and Chen, 2013] Meng, X. and Chen, T. (2013). Event based agreement protocols for multi-agent networks. *Automatica*, 49(7):2125–2132.

[Moreno-Salinas et al., 2016] Moreno-Salinas, D., Pascoal, A., and Aranda, J. (2016). Optimal sensor placement for acoustic underwater target positioning with range-only measurements. *IEEE Journal of Oceanic Engineering*, 41(3):620–643.

[Moreno-Salinas et al., 2013] Moreno-Salinas, D., Pascoal, A. M., and Aranda, J. (2013). Optimal sensor placement for multiple target positioning with range-only measurements in two-dimensional scenarios. *Sensors*, 13(8):10674–10710.

[Muller et al., 2011] Muller, M. A., Reble, M., and Allgöwer, F. (2011). A general distributed mpc framework for cooperative control. *IFAC Proceedings Volumes*, 44(1):7987 – 7992. 18th IFAC World Congress.

[Mutambara, 1998] Mutambara, A. G. (1998). *Decentralized Estimation and Control for Multisensor Systems*. CRC Press.

[Negenborn and Maestre, 2014] Negenborn, R. R. and Maestre, J. M. (2014). Distributed model predictive control: An overview and roadmap of future research opportunities. *IEEE Control Systems Magazine*, 34(4):87–97.

[Nguyen et al., 2018] Nguyen, T., Qiu, Z., Cao, M., Nguyen, T. H., and Xie, L. (2018). An integrated localization-navigation scheme for distance-based docking of UAVs. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5245–5250.

[Nowzari and Cortés, 2016] Nowzari, C. and Cortés, J. (2016). Distributed event-triggered coordination for average consensus on weight-balanced digraphs. *Automatica*, 68:237–244.

[Nowzari et al., 2019] Nowzari, C., Garcia, E., and Cortés, J. (2019). Event-triggered communication and control of networked systems for multi-agent consensus. *Automatica*, 105:1 – 27.

[Okubo, 1986] Okubo, A. (1986). Dynamical aspects of animal grouping: Swarms, schools, flocks, and herds. *Advances in Biophysics*, 22:1 – 94.

[Olfati-Saber et al., 2007] Olfati-Saber, R., Fax, J. A., and Murray, R. M. (2007). Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233.

[Oliveira et al., 2016] Oliveira, T., Aguiar, A. P., and Encarnação, P. (2016). Moving path following for unmanned aerial vehicles with applications to single and multiple target tracking problems. *IEEE Transactions on Robotics*, 32(5):1062–1078.

[Palma et al., 2017] Palma, D. D., Arrichiello, F., Parlangeli, G., and Indiveri, G. (2017). Underwater localization using single beacon measurements: Observability analysis for a double integrator system. *Ocean Engineering*, 142:650 – 665.

[Pedro Abreu and Silva, 2016] Pedro Abreu, Hélio Morishita, A. P.-J. R. and Silva, H. (2016). Marine vehicles with streamers for geotechnical surveys: Modeling, positioning, and control. In *IFAC-PapersOnLine*, volume 49, pages 458 – 464.

[Philippe Lacomme, 2001] Philippe Lacomme, Jean-Philippe Hardange, J.-C. M. E. N. (2001). Air and spaceborne radar systems: An introduction. William Andrew Publishing, Norwich, NY.

[Pillon et al., 2016] Pillon, D., Perez-Pignol, A., and Jauffret, C. (2016). Observability: range-only vs. bearings-only target motion analysis for a leg-by-leg observer's trajectory. *IEEE Transactions on Aerospace and Electronic Systems*, 52(4):1667–1678.

[Popescu et al., 2004] Popescu, O., Rose, C., and Popescu, D. C. (2004). Maximizing the determinant for a special class of block-partitioned matrices. *Mathematical Problems in Engineering*, 2004(1):49–61.

[Rego et al., 2019a] Rego, F. C., Hung, N. T., Jones, C. N., Pascoal, A. M., and Aguiar, A. P. (2019a). Cooperative path-following control with logic-based communications: Theory and practice. Navigation and Control of Autonomous Marine Vehicles, chapter 8. IET.

[Rego et al., 2019b] Rego, F. F., Pascoal, A. M., Aguiar, A. P., and Jones, C. N. (2019b). Distributed state estimation for discrete-time linear time invariant systems: A survey. *Annual Reviews in Control*, 48:36 – 56.

[Reif et al., 1999] Reif, K., Gunther, S., Yaz, E., and Unbehauen, R. (1999). Stochastic stability of the discrete-time extended Kalman filter. *IEEE Transactions on Automatic Control*, 44(4):714–728.

[Reis et al., 2019] Reis, M. F., Jain, R. P., Aguiar, A. P., and de Sousa, J. B. (2019). Robust cooperative moving path following control for marine robotic vehicles. *Frontiers in Robotics and AI*, 6:121.

[Ren and Beard, 2008] Ren, W. and Beard, R. W. (2008). *Distributed consensus in multi-vehicle cooperative control*. Springer.

[Ristic et al., 2004] Ristic, B., Arulampalam, S., and Gordon, N. (2004). *Beyond the Kalman filter*. Artech House.

[Ristic et al., 2002] Ristic, B., Arulampalam, S., and McCarthy, J. (2002). Target motion analysis using range-only measurements: algorithms, performance and application to isar data. *Signal Processing*, 82(2):273 – 296.

[Rucco et al., 2015] Rucco, A., Aguiar, A. P., Fontes, F. A., Pereira, F. L., and de Sousa, J. B. (2015). A model predictive control-based architecture for cooperative path-following of multiple unmanned aerial vehicles. In *Developments in Model-Based Optimization and Control*, pages 141–160. Springer.

[Scardovi and Sepulchre, 2009] Scardovi, L. and Sepulchre, R. (2009). Synchronization in networks of identical linear systems. *Automatica*, 45(11):2557 – 2562.

[Seeley and Buhrman, 1999] Seeley, T. D. and Buhrman, S. C. (1999). Group decision making in swarms of honey bees. *Behavioral Ecology and Sociobiology*, 45(1):19–31.

[Seyboth et al., 2013] Seyboth, G. S., Dimarogonas, D. V., and Johansson, K. H. (2013). Event-based broadcasting for multi-agent average consensus. *Automatica*, 49(1):245–252.

[Singh and Hahn, 2004] Singh, A. K. and Hahn, J. (2004). Optimal sensor location for nonlinear dynamic systems via empirical gramians. *IFAC Proceedings Volumes*, 37(9):965–970. 7th IFAC Symposium on Dynamics and Control of Process Systems 2004 (DYCOPS -7), Cambridge, USA, 5-7 July, 2004.

[Song, 1999] Song, T. L. (1999). Observability of target tracking with range-only measurements. *IEEE Journal of Oceanic Engineering*, 24(3):383–387.

[Sontag, 2008] Sontag, E. D. (2008). Input to state stability: Basic concepts and results. In *Nonlinear and optimal control theory*, pages 163–220. Springer.

[Su et al., 2016] Su, H., Wang, Z., Song, Z., and Chen, X. (2016). Event-triggered consensus of non-linear multi-agent systems with sampling data and time delay. *IET Control Theory & Applications*, 11(11):1715–1725.

[Tabuada, 2007] Tabuada, P. (2007). Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control*, 52(9):1680–1685.

[Tichavsky et al., 1998] Tichavsky, P., Muravchik, C. H., and Nehorai, A. (1998). Posterior cramer-rao bounds for discrete-time nonlinear filtering. *IEEE Transactions on Signal Processing*, 46(5):1386–1396.

[Van Parys, 2018] Van Parys, R. (2018). *Fast and Distributed Model Predictive Control-Tailored Solutions for Mechatronic Systems*. PhD thesis.

[Van Trees, 2004] Van Trees, H. L. (2004). *Detection, estimation, and modulation theory, part I: detection, estimation, and linear modulation theory*. John Wiley & Sons.

[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(5):1068 – 1074.

[Xu and Hespanha, 2004] Xu, Y. and Hespanha, J. P. (2004). Optimal communication logics in networked control systems. In *Decision and Control 43rd*, volume 4, pages 3527–3532.

[Yu et al., 2015] Yu, S., Li, X., Chen, H., and Allgöwer, F. (2015). Nonlinear model predictive control for path following problems. *International Journal of Robust and Nonlinear Control*, 25(8):1168–1182.

[Yu et al., 2010] Yu, W., Chen, G., Cao, M., and Kurths, J. (2010). Second-order consensus for multiagent systems with directed topologies and nonlinear dynamics. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 40(3):881–891.

[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.

[Zhu et al., 2014] Zhu, W., Jiang, Z.-P., and Feng, G. (2014). Event-based consensus of multi-agent systems with general linear models. *Automatica*, 50(2):552–558.

[Zolich et al., 2017] Zolich, A., Johansen, T. A., Alfredsen, J. A., Kuttenkeuler, J., and Erstorp, E. (2017). A formation of unmanned vehicles for tracking of an acoustic fish-tag. In *OCEANS 2017 - Anchorage*, pages 1–6.

A

Background materials

This appendix lists some important results related to stability of nonlinear systems and matrix calculus that were used in the thesis.

A.1 Input-to-state stability

Consider the system.

$$\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}, \mathbf{u}) \tag{A.1}$$

where: $\mathbf{f} : [0, \infty) \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is piecewise continuous in t and locally Lipschitz in $\mathbf{x}$ and $\mathbf{u}$. The input $\mathbf{u}(t)$ is piecewise continuous, bounded function of t for all $t \geq 0$.

Definition A.1 (Definition 4.7 in [Khalil, 2002])**.** *The system is said to be input-to-state (ISS) stable if there exists $\beta \in \mathcal{KL}$ and $\gamma \in \mathcal{K}$ such that for initial condition $\mathbf{x}(t_0)$ and* *any bounded input* $\mathbf{u}(t)$ *the solution exists for all* $t \geq 0$ *satisfies*

$$\|\mathbf{x}(t)\| \leq \beta(\|\mathbf{x}(t_0)\|, t - t_0) + \gamma(\sup_{t_0 \leq \tau \leq t} \|\mathbf{u}(\tau)\|) \tag{A.2}$$

Theorem A.1 (Theorem 4.19 in [Khalil, 2002])**.** *Let* $V : [0, \infty) \times \mathbb{R}^n \to \mathbb{R}$ *be a* C^1 *function such that*

$$\alpha_1(\|\mathbf{x}\|) \leq V(t, \mathbf{x}) \leq \alpha_2(\|\mathbf{x}\|)$$

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial \mathbf{x}}\mathbf{f}(t, \mathbf{x}, \mathbf{u}) \leq -W(\mathbf{x}), \quad \forall \|\mathbf{x}\| \geq \rho(\|\mathbf{u}\|) > 0$$

$\forall(t, \mathbf{x}, \mathbf{u}) \in [0, \infty) \times \mathbb{R}^n \times \mathbb{R}^m$. *Functions* $\alpha_1, \alpha_2 \in \mathcal{K}_\infty, \rho \in \mathcal{K}, W(\mathbf{x}) > 0$ *and* $W(\mathbf{x}) \in C^0$. *Then the system A.1 is ISS with* $\gamma = \alpha^{-1} \circ \alpha_2 \circ \rho$, *and* V *is called as an ISS-Lyapunov function.*

A.2 Stability of cascaded systems

Consider a cascaded systems

$$\begin{aligned} \Sigma_1 : \dot{\mathbf{x}}_1 &= \mathbf{f}_1(\mathbf{x}_1, \mathbf{u}) \\ \Sigma_2 : \dot{\mathbf{x}}_2 &= \mathbf{f}_2(\mathbf{x}_2, \mathbf{x}_1) \end{aligned} \tag{A.3}$$

Theorem A.2 (Stability of cascaded systems [Sontag, 2008])**.** *Consider the cascaded system described by* (A.3)*. Suppose that subsystem* Σ_1 *is ISS respect to the state* $\mathbf{x}_1$ *and the input* $\mathbf{u}$, *and the subsystem* Σ_2 *is ISS respect to the state* $\mathbf{x}_2$ *and the input* $\mathbf{x}_1$. *Then, the cascaded system is ISS respect to the state* $[\mathbf{x}_1, \mathbf{x}_2]^{\mathrm{T}}$ *and the input* $\mathbf{u}$.

A.3 Matrix calculus

Lemma A.1 (Rotation Matrix Differential Equation, Theorem 2.2 in [Fossen, 2011])**.** *Let* $R_A^B \in SO(n)^1(n = 2, 3)$ *be the rotation matrix from frame* $\{A\}$ *to frame* $\{B\}$*. Then,*

$$\dot{R}_A^B = R_A^B S(\boldsymbol{\omega}_{A/B}^A), \tag{A.4}$$

[1] A special orthogonal group in dimension n, defined as $SO(n) = \{R \in \mathbb{R}^{n \times n} : RR^{\mathrm{T}} = R^{\mathrm{T}}R = I_n, \det R = 1\}$

where $S(\boldsymbol{\omega}^A_{A/B})$ *is a Skew-symmetric matrix*[2] *and* $\boldsymbol{\omega}^A_{A/B} \in \mathbb{R}^n$ *is the angular velocity vector of* $\{A\}$ *respect to* $\{B\}$*, expressed in* $\{A\}$*.*

[2] A square matrix S is called a Skew-symmetric matrix iff $S^{\mathrm{T}} = -S$