
Chapter 1

Cooperative Path-Following Control with Logic-Based Communications: Theory and Practice

*Francisco C. Rego¹ Nguyen T. Hung¹
Colin N. Jones² António M. Pascoal¹ and
A. Pedro Aguiar³*

This chapter introduces an event-driven, logic-based communication system for decentralized control of a network of nonlinear systems (agents) with the objective of driving their outputs along predefined paths at desired speeds, while holding a desired formation pattern compatible with the paths. An extended cooperative path following framework is adopted where communications among agents take place at discrete time instants, instead of continuously. The communication system takes into account explicitly the topology of the communications network, the fact that communications are discrete, and the cost of exchanging information among agents. The theoretical framework adopted allows for the consideration of communication losses and bounded delays. Conditions are derived under which the resulting multi-agent closed-loop system is input-to-state stable, that is, stable and with guaranteed levels of performance in the presence of bounded external disturbances and measurement noise. The set-up derived is used to solve the problem of cooperative path-following control of multiple underactuated autonomous marine vehicles. The results of experimental field tests with a group of marine vehicles are presented and discussed.

1.1 Introduction

The advent of miniaturized sensors and actuators and the availability of powerful embedded computer systems have paved the way for the development of a new breed of autonomous vehicles capable of working in cooperation towards the execution of challenging scientific and commercial missions at sea. At the core of the systems for multiple vehicle operations are those in charge of cooperative navigation and control.

¹Laboratory of Robotics and Engineering Systems (LARSyS), ISR/IST, University of Lisbon, Portugal

²Automatic Control Laboratory, École Polytechnique Fédérale de Lausanne, Switzerland

³Department of Electrical and Computer Engineering, Faculty of Engineering, University of Porto, Portugal

Among the latter, cooperative path following control (CPF) systems are pervasive in a number of applications that include automated geotechnical surveying, marine habitat mapping, and deep sea surveying, to name but a few, see for example [1] and the references therein.

In its essence, the CPF problem can be briefly described as follows: given N autonomous vehicles and different spatial paths assigned to them, derive control laws to drive and maintain the vehicles on their paths with desired speed profiles, while adopting a specified, possibly time-varying formation. In the literature, [1, 2] offer a theoretical overview of the subject and introduce techniques to solve the CPF problem. Different solutions to the CPF and similar problems can be found in [3, 4, 5, 6, 7, 8, 9]. Our key objective in this chapter is to solve the CPF problem under severe communication constraints, when only limited amounts of data can be exchanged among the vehicles per unit of time. Bandwidth limitations of this kind are particularly stringent in underwater applications since communication takes place over low bandwidth, short range communication channels that exhibit intermittent failures, multi-path effects, and delays.

A common strategy to solve the CPF problem consists of decoupling it in two subproblems: i) a path-following (PF) problem, where the goal is to derive closed loop control laws to drive each vehicle to and follow its assigned path while tracking a path-dependent speed profile and ii) a multiple vehicle coordination problem, where the objective is to adjust the speed of each vehicle based on information exchanged with its neighbours, so as to achieve the desired formation pattern. The PF problem has been extensively addressed in the literature, see for example [10] and [11]. The coordination problem, however, requires work to overcome the problems that are naturally imposed by the communications network, both from a theoretical and practical standpoint. Some of the issues that stand in the way of CPF system implementation due to communication constraints have been addressed in the literature using graph theory to model the communications network and Lyapunov-based techniques to cope with intermittent communication failures and switching topologies; see for example [1, 7] and [2].

This chapter extends the CPF framework discussed in [1] to take into account the fact that communication among vehicles occurs at discrete instants of time, instead of continuously. In this respect, the results of this chapter go further than those in [1], where communication failures and switching topologies were considered, but communications take place continuously. The goal of this chapter is also to reduce the frequency of information exchange among vehicles. To this end, we borrow some key ideas introduced in [12] and [13], where the authors tackled the problem of distributed control by resorting to local controllers that use the state of the corresponding local system, together with state estimates of the neighbouring systems it communicates with. The communication strategy adopted in the above references assumes that each system has an internal estimator of its own state, synchronized with that of its neighbours. Therefore, since each agent has access to its own state and its estimates as perceived by its neighbours, this setup allows each system to determine the estimation error of its neighbours by taking the difference between the actual and the estimated state. The communication logic consists of only transmitting

information when the estimation error exceeds a certain threshold. With this method, communication occurs asynchronously at discrete instants of time.

The above strategy avoids communicating at a fixed rate and does so by triggering communications only when certain conditions are met. The same basic principle is inherent to different, yet related work on event-triggered control and communications. The general idea of event-triggering, where a task such as broadcasting a variable or applying a control input only when a certain condition is satisfied has been the subject of intensive work, as reported in [14, 15] and [16] for multiple systems. In the context of control of multiple agents, it is important that each agent be able to compute its own triggering condition, see e.g. [12, 13] and [17]. The work reported in [18, 19] and [20] uses the principle of self-triggered communications to reach consensus of single or double integrators. The application of event-triggered communications to the problem of consensus of linear or nonlinear multi-agent systems is analyzed in [21, 22] and [23].

We present the theoretical framework behind CPF control with logic-based communications. We apply this concept to marine vehicles, and prove stability of the devised control laws even in the presence of packet losses and delays. The present chapter is strongly motivated by and extends previous work by some of the authors in [24]. In particular, we provide full proofs of stability of the cooperative path-following algorithms that consider the interconnection between the coordination controller and the path-following controller. Furthermore, in the stability proofs we consider explicitly packet losses and delays in the communications and provide an upper bound for the delays that guarantee closed-loop system stability.

To show the efficacy of the CPF algorithm with logic-based communications that we propose, we implemented and tested the algorithm with three Medusas, a class of autonomous marine vehicles (AMVs) that have been employed extensively in many European union projects such as MORPH [25], CADDY [26], and WiMUST [27].

1.1.1 Chapter structure

The chapter is organized as follows: In Section 1.2 we propose a general control architecture for CPF with logic-based communications. In Section 1.3 the control objectives are formulated rigorously in terms of a CPF problem and conditions are derived under which the CPF Control System (CPFCS) proposed solves this problem, even when communications are asynchronous. In Section 1.4 we give a specific example of a CPFCS design which solves the CPF problem for a general class of AMVs. Finally, in Section 1.5 we describe the results of field tests with three Medusa AMVs aimed at assessing the efficacy of the CPFCS architecture for multiple vehicle operations.

1.1.2 Notation

This section summarizes the notation used throughout the chapter. The symbol $\|\cdot\|$ represents the 2-norm of a vector of real numbers. The notation $|\cdot|$ represents the cardinality of a set. I_M represents an $M \times M$ identity matrix, and $\mathbf{1}$ represents a $N \times 1$ vector with ones in every entry. When clear from the context, the

superscript of a variable, e.g. X^i , refers to the node index of that variable, where $i \in \mathcal{N} := \{1, \dots, N\}$. In this context, when $i \in \mathcal{N} := \{1, \dots, N\}$, the operator $\text{col}(\cdot)$ as $\text{col}(X^i) := [X^{1^T}, \dots, X^{N^T}]^T$, and the operator $\text{diag}(X^i)$ yields a block diagonal matrix whose diagonal elements are X^1, \dots, X^N . Given $a, b \in \mathbb{R}$ we use the notation $a \oplus b := \max(a, b)$. Given the interval $I \subset [0, \infty)$, $\|u\|_I$ denotes the essential supremum norm of a signal $u : [0, \infty) \rightarrow \mathbb{R}^n$, that is, $\|u\|_I := \text{ess sup}_{t \in I} \|u(t)\|$. For a piecewise continuous signal $v : \mathbb{R} \rightarrow \mathbb{R}^m$, with $m \geq 1$ we will use the notation $v(t^+) := \lim_{s \rightarrow t^+} v(s)$ and $v(t^-) := \lim_{s \rightarrow t^-} v(s)$. We denote the class of continuous differentiable functions up to some order n , of dimension m , as C_m^n . A continuous function $\alpha : [0, \infty) \rightarrow [0, \infty)$ is said to belong to class \mathcal{K} , or $\alpha \in \mathcal{K}$ if it is strictly increasing and $\alpha(0) = 0$. If in addition $\lim_{r \rightarrow \infty} \alpha(r) = \infty$, then α is said to be of class \mathcal{K}_∞ . A continuous function $\beta : [0, \infty) \times [0, \infty) \rightarrow [0, \infty)$ is said to belong to class \mathcal{KL} , or $\beta \in \mathcal{KL}$, if, for each fixed t , the mapping $\beta(r; t)$ belongs to class \mathcal{K} with respect to r and, for each fixed r , the mapping $\beta(r; t)$ is decreasing with respect to t and $\beta(r; t) \rightarrow 0$ as $t \rightarrow \infty$. For a function $f : \mathbb{R} \rightarrow \mathbb{R}^n$ we define the notation $f^{[l]}(x) := \left[f(x), \frac{\partial f}{\partial x}(x), \dots, \frac{\partial^l f}{\partial x^l}(x) \right]$.

We consider a very general setup for networked control, consisting of: i) a set of nodes (agents) \mathcal{N} , with cardinality $N := |\mathcal{N}|$ and ii) a directed communication network between nodes $(\mathcal{N}, \mathcal{A})$, where $\mathcal{A} \subseteq \mathcal{N} \times \mathcal{N}$ is the set of node pairs describing the connections among the nodes (each standing for a data link), i.e. node i can transmit data to node j if and only if $(i, j) \in \mathcal{A}$. Let the in-neighbour set, \mathcal{N}^i , be the index set of the agents that transmit data to agent i , i.e., $\mathcal{N}^i := \{j : (j, i) \in \mathcal{A}\}$. Conversely, the out-neighbour set of agent i , defined as $\{j : (i, j) \in \mathcal{A}\}$, is the set of agents which receive data from agent i . We also define the network maximum degree as $d^* := \max_{i \in \mathcal{N}} (|\mathcal{N}^i|)$. For the general setup proposed in Section 1.3 we consider that the network is directed. However, in the following sections that deal with the particular controller design for AMVs, for practical reasons we consider that the network is undirected, i.e. $(j, i) \in \mathcal{A}$ implies that $(i, j) \in \mathcal{A}$. In the case of an undirected network the in-neighbours of an agent are the same as the out-neighbours of that agent and therefore we will simply refer to them as neighbours.

We define a path as an ordered sequence of nodes such that any pair of consecutive nodes in the sequence is an edge of the graph, i.e., a directed path of length n from node i to node j is a sequence of edges $(i_1, i_2), (i_2, i_3), \dots, (i_{n-1}, i_n) \in \mathcal{A}$, where $i_1 = i$, $i_n = j$. A node of a network is globally reachable if it can be reached from any other node by traversing a path. A network is connected if every node is globally reachable. The adjacency matrix of a graph, denoted A , is a square matrix with rows and columns indexed by the nodes such that the i, j -entry of A is 1 if $j \in \mathcal{N}^i$ and zero otherwise. The degree matrix D of a graph $(\mathcal{N}, \mathcal{A})$ is a diagonal matrix where the i -entry equals $|\mathcal{N}^i|$, the cardinality of the set of neighbours of i , \mathcal{N}^i . The Laplacian of a graph is defined as $L := D - A$. Thus, every row sum of L equals zero, that is, $L\mathbf{1} = \mathbf{0}$. It is well known that if $(\mathcal{N}, \mathcal{A})$ is connected, L has a simple eigenvalue at zero with an associated eigenvector $\mathbf{1}$ and the remaining eigenvalues are all positive, see e.g. [28].

1.2 Cooperative path-following control system architecture

We propose a CPF control architecture for a group of N decoupled agents, $\Sigma_i; i \in \mathcal{N}$ modeled by general systems of the form

$$\Sigma_i : \quad \dot{x}_i = F_i(x_i, u_i, w_i), \quad (1.1a)$$

$$y_i = H_i(x_i, u_i, v_i), \quad (1.1b)$$

$$z_i = J_i(x_i), \quad (1.1c)$$

where $x_i \in \mathbb{R}^{n_i}$ denotes the state of agent i , $u_i \in \mathbb{R}^{m_i}$ its control input, $z_i \in \mathbb{R}^{q_i}$ is the output to be controlled, $y_i \in \mathbb{R}^{p_i}$ is the measured noisy output available for control, w_i is an input disturbance, and v_i is the measurement noise. Loosely speaking, in the absence of disturbances and measurement noise, the output z_i is a variable that we require to reach and follow a desired feasible spatial path $z_{d_i} : \mathbb{R} \rightarrow \mathbb{R}^{q_i}$ parametrized by $\gamma_i \in \mathbb{R}$. For example, in the case of a robot moving in 2D, z_i is the position of the center of mass of the vehicle which we require to converge to and follow a desired geometric path z_{d_i} with a given speed assignment along the path. In the above, F_i , H_i and J_i are appropriately defined functions.

We consider an architecture for cooperative path-following control that consists of a control system for each agent which communicates with its set of neighbours. The innovation consists of introducing a communication system which considers asynchronous communications among agents. The objective of the resulting CPFCS is to drive, in the presence of disturbances and measurement noise, the output of each agent z_i to converge to and remain inside a tube centered around the desired path $z_{d_i}(\gamma_i)$, while ensuring that its rate of progression \dot{y}_i also converges to and remains inside a tube centered around the desired speed profile $v_r(\gamma_i)$. Notice that $v_r(\gamma_i)$, together with the path parametrization function, define the desired speed profile along the path in Cartesian coordinates. Additionally, the CPFCS must also guarantee that the path variables $\gamma_i; i \in \mathcal{N}$, are synchronized, that is, all the coordination errors $\gamma_i - \gamma_j; i, j \in \mathcal{N}$ converge to and remain inside a ball around the origin, thus ensuring approximate consensus of the path variables. The path variables γ_i are often called parameterizing variables, path-following variables or, given their role in the coordination of agents, coordination states. By enforcing coordination of the path-following variables, a proper choice of the latter will enforce desired geometric formations along the paths, see [1] for a discussion of this and related issues. For proper formation control, upon achieving consensus on the coordination states we wish the time derivatives of γ_i to be equal for all $i \in \mathcal{N}$. This is achieved by guaranteeing that $v_r(\cdot)$ is the same for all the agents. Notice that the speed $v_r(\gamma_i)$ is not the actual vehicle speed in Cartesian coordinates, for it expresses the desired rate at which parameter γ_i changes. The architecture for a general CPFCS proposed in this chapter is shown in Figure 1.1. The architecture consists of three interconnected subsystems for each agent:

Path-following controller Σ_i^{pf} - This dynamical system admits as inputs the agent's measured output y_i and its output is the agent's input u_i , computed so as to make it follow the path at the assigned speed, asymptotically. In preparation

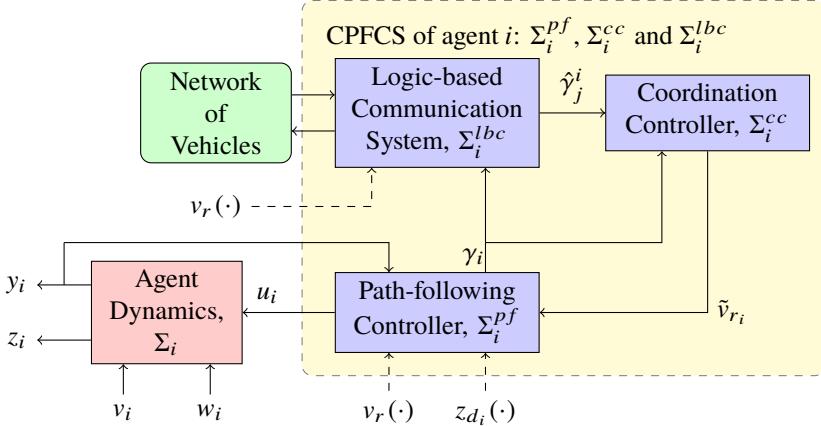


Figure 1.1: Cooperative path-following control system (CPFCS) architecture with logic-based communication system.

for its connection with the coordination controller, this system produces also a path variable γ_i . Furthermore, it accepts corrective speed action from the coordination controller via the signal \tilde{v}_{r_i} . This corrective action is aimed at synchronizing the vehicles along their paths. Notice that the dynamics of the parameterizing variable γ_i are defined internally at this stage and play the role of an extra design knob to tune the performance of the path-following control law, as will become clear later. This system has knowledge of the desired spatial path $z_{d_i}(\cdot)$ and speed profile $v_r(\cdot)$. The manner by which this system depends on the desired path and speed profile will become clear in the next section.

Coordination controller Σ_i^{cc} - This is a distributed control law whose inputs are the path variable γ_i , and estimates $\hat{\gamma}_j^i$ of the coordination states γ_j ; $j \in \mathcal{N}^i$. Its output is the speed correction signal \tilde{v}_{r_i} , which is used to synchronize agent i with its neighbours.

Logic-based communication system Σ_i^{lbc} - This dynamical system serves as an interface between agent i and its neighbours. To this end, it receives and sends appropriate information to the latter asynchronously and computes estimates of the path variables $\hat{\gamma}_j$. The information communicated to the neighbours and the communication triggering condition depend only on local information, namely on the path variable γ_i . Its output is an estimate $\hat{\gamma}_j^i$ of the coordination states of the neighbouring agents γ_j ; $j \in \mathcal{N}^i$. This communication system computes its estimates based on the received information from the neighbours. This system has knowledge of the desired speed profile $v_r(\cdot)$. Again, the manner by which this system depends on the desired speed profile will become clear in the next section.

In order to assess the conditions required for CPFCS design, its objectives are rigorously defined in the form of a CPF problem with logic-based communications in

the next section. Our work builds upon the results obtained in [1], where the CPF problem was defined and conditions were derived under which the underlying path-following and coordination controllers solve the CPF problem. In the present work, in Section 1.3, we generalize the framework developed in [1] to solve the CPF problem, considering bounded estimation errors of the path variables. We also derive a method aimed at guaranteeing that the estimation error is bounded, even when asynchronous communications take place among the agents.

1.3 Problem statement

The purpose of the path-following and coordination controllers is to solve the CPF problem, assuming that each agent estimates the coordination state of its in-neighbours with bounded estimation error. In this section we review the concepts of path-following and coordination controllers introduced in [1] and consider the case where the in-neighbours' path-variables are not transmitted continuously but are instead estimated with information given through a logic-based communication system. We apply a small gain theorem to derive conditions under which the interconnection of the path-following controller and the coordination controller yields an input to state stable (ISS) system.

1.3.1 Path-following problem

A path following controller can be considered to have two assignments, the geometric task and the dynamic task, see for example [11]. The geometric task consists of driving the agent output z_i to a desired path z_{d_i} parametrized by a continuous scalar variable γ_i which, as mentioned before, we call the path-following variable, path variable, parameterizing variable, or coordination state. The dynamic task consists of forcing the path-following variables γ_i to a certain dynamic behaviour. More specifically, the dynamic task considered in this section is a speed assignment where we require the parameterizing variables to have a desired speed $v_r(\gamma_i)$. In summary, the geometric task consists of driving to a value close to zero the path-following errors defined as

$$e_i := z_i - z_{d_i}(\gamma_i),$$

and the dynamic task consists of driving to close to zero the speed errors defined as

$$e_{\dot{\gamma}_i} := \dot{\gamma}_i - v_r(\gamma_i) - \tilde{v}_{r_i},$$

where, as explained before, \tilde{v}_{r_i} is the speed correction signal. For the sake of clarity and rigor, in what follows we give a formal definition of the output path-following problem. This is instrumental in understanding the conditions that a path-following controller must satisfy in order to perform successfully the geometric and dynamic tasks.

To formulate the problem of path following, we must take into consideration pairs of spatial paths and speed profiles along them. Allowable pairs (z_{d_i}, v_r) of a path $z_{d_i}(\cdot)$ and a speed profile $v_r(\cdot)$, must be compatible with the vehicle dynamics, in the

sense that a pair is allowable if there is an input such that the vehicle will follow its assigned path at a desired speed. Moreover, the controller design methods may require certain assumptions on the paths and speed profiles. To formalize these constraints, we consider that there exists a class of admissible mission profiles $\mathcal{M}_i \subseteq C_{n_i}^0 \times C_1^0$ for each agent $i \in \mathcal{N}$. We say that a mission is admissible if $(z_{d_i}, v_r) \in \mathcal{M}_i$. The class \mathcal{M}_i depends on the type of systems Σ_i to be controlled and the control design methods, and should contain any mission specification that might be required of the system. For the sake of generality we purposefully do not define the set \mathcal{M}_i at this point, since different controller design methods and different types of agents have in general different classes of admissible missions. As an example, in the controller design considered in the next section the class of admissible missions is defined in Section 1.4.5.

Definition 1 (Path-following controller): Consider a speed profile v_r and a path z_{d_i} such that $(z_{d_i}, v_r) \in \mathcal{M}_i$. We define a path-following controller as a system Σ_i^{pf} specified by appropriately defined functions F_i^{pf} , H_i^{pf} and H_i^γ , where H_i^γ is continuous, and integers $n_v, n_z \in \mathbb{N}$ as follows:

$$\Sigma_i^{pf} : \dot{x}_i^{pf} = F_i^{pf} \left(x_i^{pf}, y_i, \tilde{v}_{r_i}, v_r^{[n_v]}(\gamma_i), z_{d_i}^{[n_z]}(\gamma_i) \right), \quad (1.2a)$$

$$u_i = H_i^{pf} \left(x_i^{pf}, y_i, \tilde{v}_{r_i}, v_r^{[n_v]}(\gamma_i), z_{d_i}^{[n_z]}(\gamma_i) \right), \quad (1.2b)$$

$$\gamma_i = H_i^\gamma \left(x_i^{pf} \right), \quad (1.2c)$$

where the notation for $v_r^{[n_v]}(\cdot)$ and $z_{d_i}^{[n_z]}(\cdot)$ is defined in the notation section, Section 1.1.2.

Remark 1: In the path-following controller of Section 1.4.2 we have $n_v = 1$ and $n_z = 2$. However, one might have different values of n_v and n_z for other types of path-following controllers, as seen in [11].

Definition 2 (Error signal): Consider an agent Σ_i with dynamics (1.1), a controller Σ_i^{pf} with dynamics (1.2), and a prescribed speed profile v_r and a path z_{d_i} such that $(z_{d_i}, v_r) \in \mathcal{M}_i$. We define an error signal

$$\mathbf{e}_i := F_i^e \left(x_i, x_i^{pf}, \tilde{v}_{r_i}, v_r^{[n_v]}(\gamma_i), z_{d_i}^{[n_z]}(\gamma_i) \right),$$

where F_i^e is an appropriately defined function such that the path-following and speed errors satisfy the condition

$$\|e_i\| \oplus \|e_{\dot{\gamma}_i}\| \leq \sigma^e (\|\mathbf{e}_i\|), \quad (1.3)$$

where $\sigma^e \in \mathcal{K}_\infty$.

Remark 2: A possible construction of the above error signal is given by $\mathbf{e}_i := [e_i^T, e_{\dot{\gamma}_i}]^T$. However, the definition of the path-following problem may require other definitions of the error signal such as the one in Lemma 1, Section 1.4.2.

Definition 3 (Output path-following problem): Consider an agent Σ_i with dynamics (1.1) and a controller Σ_i^{pf} with dynamics (1.2). The controller Σ_i^{pf} is said to solve

the output path-following problem if, for every prescribed speed profile v_r and path z_{d_i} such that $(z_{d_i}, v_r) \in \mathcal{M}_i$, the error signal \mathbf{e}_i of Definition 5 is input-to-state stable (ISS) with respect to w_i , v_i , and \tilde{v}_{r_i} , that is, there exist functions $\sigma_w^e, \sigma_v^e, \sigma_{\tilde{v}_r}^e \in \mathcal{K}_\infty$, and $\beta^e \in \mathcal{KL}$ such that

$$\begin{aligned} \|\mathbf{e}_i(t)\| &\leq \beta^e(\|\mathbf{e}_i(0)\|, t) \\ &\oplus \sigma_w^e(\|w_i\|_{[0,t]}) \oplus \sigma_v^e(\|v_i\|_{[0,t]}) \oplus \sigma_{\tilde{v}_r}^e(\|\tilde{v}_{r_i}\|_{[0,t]}). \end{aligned} \quad (1.4)$$

Recalling that the geometric task amounts to imposing a (small) bound on the path-following errors e_i and the dynamic task consists of forcing a bound on the speed errors $e_{\dot{\gamma}_i}$, it follows from condition (1.3) that both tasks are satisfied if the signal \mathbf{e}_i can be bounded. The ISS condition (1.4) implies that, regardless of the initial condition, asymptotically (as t goes to infinity) \mathbf{e}_i goes to a neighbourhood of $\mathbf{0}$ whose size depends only on the magnitudes of the process noise w_i , measurement noise v_i , and speed correction \tilde{v}_{r_i} .

1.3.2 Coordination control problem

Besides meeting the requirements of path-following, that is, making each agent follow a desired path z_{d_i} (albeit with a small asymptotic error) at some required speed v_r , we also require coordination of the entire group of agents so as to achieve a desired formation pattern compatible with the paths adopted. Ideally, any two agents $i, j \in \mathcal{N}$ are coordinated, are synchronized, or have reached agreement, if and only if $\gamma_i - \gamma_j = 0$. As explained before, the choice of the path parameters γ_i is crucial to define the type of formation that the vehicles will adopt. Since nullifying the coordination errors $\gamma_i - \gamma_j$ is a very strict requirement, we require instead that the coordination errors became bounded by a small number after some settling time. In preparation for a rigorous analysis of the conditions required for the coordination controller to coordinate the agents, we now define the coordination control problem formally.

We start by recalling that the coordination controller acts on the path-following controller Σ_i^{pf} , through the speed correction signal \tilde{v}_{r_i} , with the objective of synchronizing the agents along their paths, and admits as inputs the path-following variable γ_i . Finally, in what concerns the interaction with the communication system, notice that the latter is responsible for providing the coordination controller with the values of the path-following variables of the in-neighbours γ_j ; $j \in \mathcal{N}^i$. Since we aim to save the number of communications among agents, the actual values of the path-following variables of the in-neighbours are not available continuously, and the output of the communication system consists of estimates of the path-following variables $\hat{\gamma}_j^i$; $j \in \mathcal{N}^i$. We define the estimation error as $\tilde{\gamma}_j^i := \hat{\gamma}_j^i - \gamma_j$.

Definition 4 (Coordination controller): *The class of coordination controllers that we consider is described by control laws Σ_i^{cc} ; $i \in \mathcal{N}$ of the form*

$$\Sigma_i^{cc} : \tilde{v}_{r_i} = H_i^{cc}(\gamma_i, \hat{\gamma}_j^i; j \in \mathcal{N}^i), \quad (1.5a)$$

where H_i^{cc} is an appropriately defined function.

Definition 5 (Coordination error signal): Consider a set of N coordination controllers $\Sigma_i^{cc}; i \in \mathcal{N}$ given by Definition 4. We define a coordination error signal $\xi := F^\xi(\gamma_i; i \in \mathcal{N})$, where F^ξ is an appropriately defined function that verifies the conditions

$$\max_{i \in \mathcal{N}; j \in \mathcal{N}^i} (\|\gamma_i - \gamma_j\|) \leq \sigma_\xi^\xi(\|\xi\|), \quad (1.6a)$$

$$\max_{i \in \mathcal{N}} (\|\tilde{v}_{r_i}\|) \leq \sigma_\xi^{\tilde{v}_r}(\|\xi\|) \oplus \max_{i \in \mathcal{N}; j \in \mathcal{N}^i} \left(\sigma_\gamma^{\tilde{v}_r}(\|\tilde{\gamma}_j^i\|) \right), \quad (1.6b)$$

where $\sigma_\xi^\xi, \sigma_\xi^{\tilde{v}_r}, \sigma_\gamma^{\tilde{v}_r} \in \mathcal{K}_\infty$.

Remark 3: Note a possible construction of the coordination error signal ξ is

$$\xi := \text{col}(\gamma_i - \gamma_j; i \in \mathcal{N}, j \in \mathcal{N}^i).$$

However, the definition of the coordination problem may require other definitions of the coordination error signal such as the one in Lemma 2, Section 1.4.3.

Definition 6 (Coordination problem): Consider a set of N agents $\Sigma_i; i \in \mathcal{N}$ with dynamics (1.1) equipped with path-following controllers $\Sigma_i^{pf}; i \in \mathcal{N}$ of Definition 1 and coordination controllers $\Sigma_i^{cc}; i \in \mathcal{N}$ of Definition 4, together with the error signals $e_i; i \in \mathcal{N}$ and the coordination error signal ξ of Definitions 2 and 5, respectively. We say that the coordination controllers $\Sigma_i^{cc}; i \in \mathcal{N}$ solve the coordination problem if, for every prescribed speed profile v_r and paths $z_{d_i}; i \in \mathcal{N}$ satisfying $(z_{d_i}, v_r) \in \mathcal{M}_i; i \in \mathcal{N}$, there exist functions $\beta^\xi \in \mathcal{KL}$ and $\sigma_e^\xi, \sigma_\gamma^\xi \in \mathcal{K}_\infty$ such that the evolution of the coordination error signal ξ satisfies the inequality

$$\|\xi(t)\| \leq \beta^\xi(\|\xi(0)\|, t) \oplus \max_{i \in \mathcal{N}; j \in \mathcal{N}^i} \left(\sigma_\gamma^\xi(\|\tilde{\gamma}_j^i\|_{[0,t]}) \oplus \sigma_e^\xi(\|e_i\|_{[0,t]}) \right). \quad (1.7)$$

In plain terms, our approach to coordination amounts to imposing an upper bound on the coordination errors $\|\gamma_i - \gamma_j\|; \forall i \in \mathcal{N}; j \in \mathcal{N}^i$. From the properties (1.6), the coordination errors can be bounded if a bound can be found for the error signal ξ . The ISS condition (1.7) implies that if the error signal e_i , and the estimation errors $\tilde{\gamma}_j^i$ are bounded, then, after a sufficiently large time we can compute a bound on ξ , which depends on the bounds on the error signals and the estimation errors. The impact of these bounds on the controller performance, given by $\sigma_e^\xi(\|e_i\|_{[0,t]})$, and $\sigma_\gamma^\xi(\|\tilde{\gamma}_j^i\|_{[0,t]})$, are expected to be small when compared to the precision required for coordination control. This was found to be the case in the experiments of Section 1.5.

1.3.3 Cooperative path-following

At this point, it is important to note that the coordination control problem and the path-following problem are not independent because the dynamics of ξ and e are interconnected. This interconnection can be clearly seen in equations (1.4) and (1.7) of Definitions 3 and 6, respectively. Therefore, to ensure that the interconnection of the path-following and coordination controllers solves the geometric and dynamic tasks

and coordination is achieved, it is required to solve the two problems simultaneously, taking explicitly into account the connection among them. The CPF problem defined next, if solved, guarantees that the objectives specified in terms of the dynamic and geometric tasks and the coordination problem are achieved simultaneously in the presence of bounded process and measurement noise.

Definition 7 (CPF problem): Consider a set of N agents $\Sigma_i; i \in \mathcal{N}$ with dynamics (1.1), equipped with path-following controllers $\Sigma_i^{pf}; i \in \mathcal{N}$ and coordination controllers $\Sigma_i^{cc}; i \in \mathcal{N}$ defined by (1.2) and (1.5), respectively, and the error signals $e_i; i \in \mathcal{N}$ and coordination error signal ξ of Definitions 2 and 5, respectively. We say that the path-following and coordination controllers, $\Sigma_i^{pf}; i \in \mathcal{N}$ and $\Sigma_i^{cc}; i \in \mathcal{N}$, solve the CPF problem if, for every prescribed speed profile v_r and paths $z_{d_i}; i \in \mathcal{N}$ satisfying $(z_{d_i}, v_r) \in \mathcal{M}_i; i \in \mathcal{N}$, $\bar{e} := [\text{col}(e_i)^T, \xi^T]^T$ is ISS with respect to w_i, v_i , and $\tilde{\gamma}_j^i$, that is, if there exist functions $\sigma_w^{\bar{e}}, \sigma_v^{\bar{e}}, \sigma_{\gamma}^{\bar{e}} \in \mathcal{KL}_{\infty}$, and $\beta^{\bar{e}} \in \mathcal{KL}$ such that

$$\begin{aligned} \|\bar{e}(t)\| &\leq \beta^{\bar{e}}(\|\bar{e}(0)\|, t) \\ &\oplus \max_{i \in \mathcal{N}; j \in \mathcal{N}^i} (\sigma_w^{\bar{e}}(\|w_i\|_{[0,t]}) \oplus \sigma_v^{\bar{e}}(\|v_i\|_{[0,t]}) \oplus \sigma_{\gamma}^{\bar{e}}(\|\tilde{\gamma}_j^i\|_{[0,t]})). \end{aligned} \quad (1.8)$$

The geometric and dynamic tasks are satisfied and coordination is achieved if $e_i, e_{\dot{\gamma}_i}$ and $\gamma_i - \gamma_j; j \in \mathcal{N}^i$ are bounded. The conditions (1.3) and (1.6) guarantee that if the error signal \bar{e} is bounded then $e_i, e_{\dot{\gamma}_i}$ and $\gamma_i - \gamma_j; j \in \mathcal{N}_i$ are bounded. Finally, the ISS condition (1.8) guarantees that if v_i and w_i are bounded, then after a sufficiently large time we can compute a bound on \bar{e} which depends only on v_i, w_i and the estimation error $\tilde{\gamma}_j^i; j \in \mathcal{N}^i$. Therefore, if the CPF problem is solved, for bounded measurement and process noise (v_i and w_i , respectively), the path-following and coordination controllers satisfy the geometric and dynamic tasks and achieve coordination.

The following theorem gives conditions under which a set of path-following and coordination controllers solves the CPF problem, taking into account the interconnection of the path-following and coordination controllers which solve the path-following and coordination problems, respectively.

Theorem 1 (Cooperative path-following): Consider a set of N agents $\Sigma_i; i \in \mathcal{N}$ with dynamics (1.1), equipped with path-following controllers $\Sigma_i^{pf}; i \in \mathcal{N}$ and coordination controllers $\Sigma_i^{cc}; i \in \mathcal{N}$ defined by (1.2) and (1.5), respectively. Assume that the path-following and coordination controllers solve the path-following and coordination problems, i.e. inequalities (1.3)-(1.4), (1.6)-(1.7) hold. Suppose further that $\tilde{\gamma}_j^i := \hat{\gamma}_j^i - \gamma_j$ are bounded for all $i \in \mathcal{N}$ and $j \in \mathcal{N}^i$ and

$$\sigma_{\tilde{v}_r}^e \circ \sigma_{\xi}^{\tilde{v}_r} \circ \sigma_e^{\xi}(r) < r, \forall r > 0. \quad (1.9)$$

Then, the path-following and coordination controllers, Σ_i^{pf} and Σ_i^{cc} , solve the CPF problem.

Proof. From (1.4) and (1.7) we conclude that the path-following and coordination controllers can be viewed as two interconnected ISS systems with outputs e and

$\tilde{v}_r := \text{col}(\tilde{v}_{r_i})$. A straightforward application of the small-gain theorem in [29] and [30] implies that if (1.9) holds, then the whole interconnected system is ISS. We can then conclude that (1.8) holds because $\tilde{\gamma}_j^i$ for all $i \in \mathcal{N}$ and $j \in \mathcal{N}^i$ is bounded. \square

This theorem states that the set of path-following and coordination controllers solves the CPF problem, that is, the interconnection of the path-following controller and the coordination controller is ISS if the estimation errors are bounded, all the path-following controllers and coordination controllers solve the path-following and coordination problems, and inequality (1.9) holds. In the particular example of Section 1.4, inequality (1.9) is trivially satisfied since $\sigma_{\tilde{v}_r}^e \equiv 0$. For an example of CPF control design where the interconnection must be explicitly considered see [1].

1.3.4 Logic-based communication system

In many applications that require the use of multiple marine vehicles working in cooperation, underwater communications have tight bandwidth limitations. For this reason, the latter must be addressed explicitly in the design of cooperative controls systems, such as those in charge of CPF, by reducing the rate at which data are transmitted among agents. To this end, in what follows we assume that instead of having each agent send continuously the path-following variables γ_i ; $i \in \mathcal{N}$ to its out-neighbours, messages are exchanged through a communication system at discrete instants of time asynchronously, using some logic that defines a communication triggering condition (CTC). Since the triggering condition that we consider is based on the perceived estimates in the out-neighbours of the local generalized path-variable γ_i , computed used only transmitted information, we consider the estimators to be an integral part of the communication system, henceforth called a logic-based communication system.

To guarantee that the coordination control error is smaller than some upper bound, we need to ensure that the communication system produces estimates of the in-neighbours' path variables $\hat{\gamma}_j^i$; $j \in \mathcal{N}^i$ with a bounded estimation error $\tilde{\gamma}_j^i := \hat{\gamma}_j - \gamma_j$. To meet this goal, one of the key contributions of this chapter is the introduction, in the general set-up for cooperative path following, of a logic based communication system capable of keeping the estimation errors $\tilde{\gamma}_i^j$ bounded.

Inspired by the setup proposed in [13], the communication system of each agent i is composed by a bank of estimators of the in-neighbours' path-following variables $\hat{\gamma}_j^i$; $j \in \mathcal{N}^i$ and a communication logic block. The estimators run in open-loop most of the time but are sometimes reset (not necessarily periodically) to correct their states when measurements are received through the network.

The communication logic is responsible for assessing for each agent $i \in \mathcal{N}$, how well each of the out-neighbour agents can predict the coordination state γ_i through $\hat{\gamma}_i^j$; $i \in \mathcal{N}^j$, formally defined as the estimate of γ_i obtained by agent j , which agent i can compute using the data transmitted to agent j . Using that information, the communication logic is responsible decide when it should communicate the actual measured value to its out-neighbours so as to guarantee that $\|\tilde{\gamma}_i^j(t)\| \leq \epsilon$, for all j such that $i \in \mathcal{N}^j$, and all $t \geq 0$, for some positive scalar $\epsilon > 0$.

In this section, we describe a general logic-based communications system using the principle of synchronized estimators found in [20].

Definition 8 (Logic-based communication system): Consider a reference speed $v_r(\cdot)$. Let $t_k^{ij}, k \geq 0$ indicate the instants of data transmission between agent i and a generic out-neighbour j , which occur when a CTC is satisfied. The logic-based communication system at agent i , Σ_i^{lbc} , is formally defined as the composition of the subsystems $\Sigma_{ij}^{lbc}; j \in \mathcal{N}^j$ and $\Sigma_{li}^{lbc}; l \in \mathcal{N}^i$, where Σ_{ij}^{lbc} is defined by the functions F_{ij}^{lbc} , H_{ij}^{lbc} and J_{ij}^{lbc} as

$$\Sigma_{ij}^{lbc} : \begin{cases} x_{ij}^{lbc}(t^+) = J_{ij}^{lbc}(x_{ij}^{lbc}, x_{li}^{lbc}, \gamma_i; l \in \mathcal{N}^i), & \text{for } t = t_k^{ij}, k \geq 0, \\ \dot{x}_i^{lbc} = F_i^{lbc}(x_i^{lbc}, v_r(\tilde{\gamma}_i^j)), & \text{otherwise,} \end{cases} \quad (1.10a)$$

$$\tilde{\gamma}_i^j = H_{ij}^{lbc}(x_{ij}^{lbc}), \quad (1.10b)$$

with H_{ij}^{lbc} continuous and J_{ij}^{lbc} , satisfying, for any admissible values of x_{ij}^{lbc} , x_{li}^{lbc} , and γ_i , the following relation:

$$H_{ij}^{lbc}(J_{ij}^{lbc}(x_{ij}^{lbc}, x_{li}^{lbc}, \gamma_i; l \in \mathcal{N}^i)) = \gamma_i.$$

Moreover, the communication system of agent i sends a message containing $x_{li}^{lbc}; l \in \mathcal{N}^i$, and γ_i , or a function of these data to its out-neighbours ($j \in \mathcal{N}$ such that $i \in \mathcal{N}^j$) at times t_k^{ij} when the following CTC holds:

$$\|\tilde{\gamma}_j^i\| = \epsilon. \quad (1.11)$$

In the definition above, information is sent from agent i at instants t_k^{ij} and x_{ij}^{lbc} is updated instantaneously. Accordingly, at instants $t^{ji}; j \in \mathcal{N}^i$ information is received by agent i from its in-neighbour j . While no data are sent or received, the communication system only uses internal information to compute the path variables estimates.

Notice that, Σ_i^{lbc} is composed by the subsystems $\Sigma_{ij}^{lbc}; j \in \mathcal{N}^j$ and by synchronized copies of the subsystems $\Sigma_{ji}^{lbc}; i \in \mathcal{N}^i$ contained in the the out-neighbour agents j . By ensuring that at transmission times t_k^{ij} , on agent i and j the state x_{ij}^{lbc} is updated at the same time with the same rule, we guarantee that the estimates are kept synchronized. The purpose of this setup is that it allows each agent to compute the estimation error of its own path variable $\tilde{\gamma}_i^j$, which allows for the computation of the triggering condition 1.11.

We can now show that the logic-based communication systems of Definition 8. yields bounded estimation errors.

Theorem 2: Given the set of logic-based communication systems $\Sigma_i^{lbc}; i \in \mathcal{N}$ of Definition 8, then for every $i \in \mathcal{N}$

$$\|\tilde{\gamma}_j^i(t)\| \leq \epsilon, \quad \forall j \in \mathcal{N}^i, t \geq 0 \quad (1.12)$$

Proof. From Definition 8 one has that for any transmission time t_k^{ij} , $\|\tilde{\gamma}_i^j(t_{k+1}^{ij})\| = \epsilon$ and $\|\tilde{\gamma}_i^j(t_k^{ij+})\| = 0$. Moreover, since during the interval (t_k^{ij}, t_{k+1}^{ij}) x_{ij}^{lbc} and x_i^{pf} are

differentiable and because H_{ij}^{lbc} and H_i^γ are continuous, $\tilde{\gamma}_j^i$ is continuous in that interval. From the continuity of $\tilde{\gamma}_j^i$ and the fact that $\|\tilde{\gamma}_i^j(t)\| \neq \epsilon$ for $t \in (t_k^{ij}, t_{k+1}^{ij})$, we observe that (1.12) is satisfied for $t \in (t_k^{ij}, t_{k+1}^{ij})$. Applying the same reasoning for any transmission time t_k^{ij} , we show that (1.12) is satisfied for $t \geq 0$. \square

The communication system introduced in Definition 8 represents a general architecture for non-delayed communications without packet losses. To introduce robustness with respect to packet losses we will require that the communication systems send a reply whenever they receive a message to the sender of that same message. Therefore, in the remainder of this chapter we assume that the communication network is undirected, that is, if $i \in \mathcal{N}^j$, then $j \in \mathcal{N}^i$.

It should also be noted that in Definition 8 we allow for the estimates of the path-following variable of agent i , γ_i , on its out-neighbours, $\hat{\gamma}_i^j$; $i \in \mathcal{N}^j$, to be different from each other, that is, it is admissible that for some $j \neq l$ such that $i \in \mathcal{N}^j \cup \mathcal{N}^l$, $\hat{\gamma}_i^j \neq \hat{\gamma}_i^l$. However, in the communication system of Section 1.4 all estimates of the path-following variable of an agent are equal, that is for every two agents j and l such that $i \in \mathcal{N}^j \cup \mathcal{N}^l$, we have $\hat{\gamma}_i^j = \hat{\gamma}_i^l$ (except at certain moments in the case of communication losses). An example of communication system where the values of $\hat{\gamma}_i^j$; $i \in \mathcal{N}^j$ might be different from each other can be seen in [24].

1.4 Controller Design: CPF for multiple AMVs

In this section we describe how the general framework described in the previous section can be applied to the design of a CPF system for multiple vehicles. The design exercise is done for a group of AMVs maneuvering in 2D.

1.4.1 Vehicle model

We start by describing the mathematical model of a class of AMVs (agents) used for motion control design. We write the kinematic equations of motion of a vehicle moving in the horizontal plane by using a global inertial coordinate frame $\{\mathcal{U}\}$ and a body-fixed coordinate frame $\{\mathcal{B}\}$ with the origin at the vehicle's center of mass, yielding

$$\dot{x} = u_w \cos(\psi) - v_w \sin(\psi) + v_{c_x}, \quad (1.13a)$$

$$\dot{y} = u_w \sin(\psi) + v_w \cos(\psi) + v_{c_y}, \quad (1.13b)$$

$$\dot{\psi} = r, \quad (1.13c)$$

where u_w and v_w are body-fixed frame components of the vehicle's velocity with respect to the water, v_{c_x} and v_{c_y} are inertial frame components of the water current velocity, assumed to be constant, x and y are the inertial Cartesian coordinates of the vehicle's center of mass, and ψ defines its orientation (heading angle). The kinematic equations (1.13b)-(1.13c) can be written in compact form by defining $\mathbf{p} := [x, y]^T$,

$\mathbf{v}_{c_{xy}} := [v_{c_x}, v_{c_y}]^T$, and $\mathbf{v} := [u_w, v_w]^T$, leading to

$$\begin{aligned}\dot{\mathbf{p}} &= R(\psi)\mathbf{v} + \mathbf{v}_{c_{xy}} \\ \dot{\psi} &= r,\end{aligned}\tag{1.14}$$

where $R(\psi)$ is the orthonormal transformation matrix from $\{\mathcal{B}\}$ to $\{\mathcal{U}\}$, given by

$$R(\psi) := \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix}.\tag{1.15}$$

To simplify notation we denote $R(\psi)$ simply as R . In what follows, since we will be working mostly with body fixed coordinates, it will be useful to express the water current in body fixed coordinates as $\mathbf{v}_c := R^T \mathbf{v}_{c_{xy}}$. However, it should be noted that in this case \mathbf{v}_c is time varying, which is not the case for $\mathbf{v}_{c_{xy}}$.

Before proceeding into the description of the path-following controller, we first consider the inner-loop dynamic task, which consists of making the vehicle's surge velocity u_w and heading rate r track desired references, given by u_d and r_d , respectively, to be defined later. In what follows, we assume that the vehicle is equipped with inner loop controllers that satisfy the following stability assumption:

Assumption 1: Let $\tilde{u} := u_w - u_d$ and $\tilde{r} := r - r_d$ be the speed and heading rate tracking errors respectively, and v_w the sway velocity. We assume there exist positive constants $\epsilon_{\tilde{r}}, \epsilon_{\tilde{u}}, \epsilon_v$ such that for all $t \geq 0$

$$\|\tilde{r}(t)\| \leq \epsilon_{\tilde{r}}, \quad \|\tilde{u}(t)\| \leq \epsilon_{\tilde{u}}, \quad \|v_w(t)\| \leq \epsilon_v.$$

The constants $\epsilon_{\tilde{r}}, \epsilon_{\tilde{u}}$ and ϵ_v correspond to bounds on the tracking errors due to the dynamics of the vehicle, disturbances, or unforeseen behaviour due to plant modeling uncertainty. We also assume that each vehicle contains an observer of the water current velocity \mathbf{v}_c which satisfies the following assumption:

Assumption 2: Let $\tilde{\mathbf{v}}_c := \mathbf{v}_c - \hat{\mathbf{v}}_c$ be the estimation error of the water current velocity estimator. There exist a positive constant $\epsilon_{\tilde{v}_c}$ such that for all $t \geq 0$

$$\|\tilde{\mathbf{v}}_c(t)\| \leq \epsilon_{\tilde{v}_c}.$$

For an example of a water current estimator and its technical details, see [24]. Given the above mentioned assumptions, the kinematics of the vehicle can be described as

$$\begin{aligned}\dot{\mathbf{p}} &= R \left(\begin{bmatrix} u_d + \tilde{u} \\ v_w \end{bmatrix} + \hat{\mathbf{v}}_c + \tilde{\mathbf{v}}_c \right), \\ \dot{\psi} &= r_d + \tilde{r}\end{aligned}$$

This design corresponds to the general form of an agent Σ given in (1.1) where the state x , measured output y , control input u , process disturbance w , and measurement noise v are explicitly expressed as

$$x := \begin{bmatrix} v_c \\ p \\ \psi \end{bmatrix}, \quad y := \begin{bmatrix} \hat{v}_c \\ p \\ \psi \end{bmatrix}, \quad u := \begin{bmatrix} u_d \\ r_d \end{bmatrix}, \quad w := \begin{bmatrix} \tilde{u} \\ \tilde{r} \\ v_w \end{bmatrix}, \quad v := \tilde{\mathbf{v}}_c.$$

From Assumption 1, we have that $\|w\|_{[0,t]} \leq \epsilon_w := \sqrt{\epsilon_u^2 + \epsilon_r^2 + \epsilon_v^2}$ for $t \geq 0$, and from Assumption 2 we obtain $\|v\|_{[0,t]} \leq \epsilon_v := \epsilon_{v_c}$.

1.4.2 Path-following controller

To solve the problem of driving a vehicle along a desired path, the key idea exploited is to make the vehicle approach a virtual target that moves along the path. Let $z_d(\gamma)$ be the position of the target, and $v_r(\gamma)$ the desired rate of the progression of the path parameter γ . We decompose the motion-control problem into the inner-loop dynamic task mentioned in Assumption 1, and an outer-loop kinematic task, which consists of assigning references for surge speed and heading rate references $u := [u_d, r_d]^T$ and for the second derivative of the path parameter $\ddot{\gamma}$, in order to solve the path-following problem.

Define the position error e_B as the difference between the positions of the vehicle and the virtual target expressed in the body frame $\{B\}$, that is, $e_B := R^T(p - z_d)$. The dynamics of e_B are described by

$$\dot{e}_B = -S(r)e_B + \begin{bmatrix} u_d + \tilde{u} \\ v_w \end{bmatrix} + v_c - R^T \frac{\partial z_d}{\partial \gamma} \dot{\gamma}, \quad S(r) = \begin{bmatrix} 0 & -r \\ r & 0 \end{bmatrix}, \quad (1.16)$$

where we used the fact that $\dot{R} = RS(r)$.

Following the methods adopted in [31], to make the control input $u := [u_d, r_d]^T$ appear in the position error dynamics we introduce a constant design vector $\delta := [\delta, 0]^T$, $\delta < 0$. From (1.16), defining $e_\delta := e_B - \delta$ simple computations show that the position error dynamics are given by

$$\dot{e}_\delta = -S(r)e_\delta + \Delta u - R^T \frac{\partial z_d}{\partial \gamma} \dot{\gamma} - \begin{bmatrix} 0 \\ \delta \tilde{r} \end{bmatrix} + \begin{bmatrix} \tilde{u} \\ v_w \end{bmatrix} + \hat{v}_c + \tilde{v}_c, \quad (1.17)$$

where

$$\Delta := \begin{bmatrix} 1 & 0 \\ 0 & -\delta \end{bmatrix}$$

In what follows we assume that we explicitly control the virtual target speed error defined as

$$e_{\dot{\gamma}} := \dot{\gamma} - (v_r + \tilde{v}_r), \quad (1.18)$$

thereby introducing an additional control variable. We can observe that this design corresponds to defining the objective output as $z := p + R\delta$, the path following controller Σ^{PF} with state $x^{PF} = [\gamma, e_{\dot{\gamma}}]^T$, and the path following error as $e := z - z_d(\gamma) = Re_\delta$.

Lemma 1 (Path following controller): *Let $z_d \in C_2^1$ be a desired path and $v_r \in C_1^0$ a desired speed assignment. Consider the vehicle model described by (1.13), in closed-loop with the output feedback control law composed by inner-loop controllers*

that satisfy Assumption 1, a water current estimator which satisfies Assumption 2, and, the outer-loop controller given by

$$\dot{e}_\gamma = -k_\gamma e_\gamma + \frac{1}{c_\gamma} e_\delta^T R^T \frac{\partial z_d}{\partial \gamma}(\gamma), \quad (1.19)$$

$$u = \Delta^{-1} \left(-K_k e_\delta - \hat{v}_c + R^T \frac{\partial z_d}{\partial \gamma}(\gamma) (v_r + \tilde{v}_r) \right), \quad (1.20)$$

where $K_k := \text{diag}(k_x, k_y)$, and the positive scalar design parameters k_x , k_y , k_γ and c_γ . Then, the error vector

$$\mathbf{e} := \begin{bmatrix} e_\delta \\ \sqrt{c_\gamma} e_\gamma \end{bmatrix} = \begin{bmatrix} R^T e \\ \sqrt{c_\gamma} e_\gamma \end{bmatrix}$$

is ISS with respect to w and v , that is, there exist functions $\sigma_w^e, \sigma_v^e \in \mathcal{KL}_\infty$, $\beta^e \in \mathcal{KL}$ such that

$$\|\mathbf{e}\| \leq \beta^e (\|\mathbf{e}(0)\|, t) \oplus \sigma_w^e (\|w\|_{[0,t]}) \oplus \sigma_v^e (\|v\|_{[0,t]}). \quad (1.21)$$

Therefore, the control laws (1.19) and (1.20) solve the path-following problem of Definition 3.

Proof. See the Appendix. □

With this strategy, the evolution of the position of the virtual target z_d depends on the position error e_δ in that if the vehicle is ahead/behind the desired position, the virtual target moves faster/slower towards the position of the vehicle. That is, unlike trajectory tracking, where the virtual target moves along the path at the speed reference v_r , in path-following the target dynamics actively aid in the convergence of the vehicle towards the path.

1.4.3 Coordination controller

Consider now the coordination control problem with a communication topology defined by a graph $(\mathcal{N}, \mathcal{A})$. We assume that the graph is undirected, that is, the communication links are bidirectional, in that if $i \in \mathcal{N}^j$, then $j \in \mathcal{N}^i$.

Remark 4: The assumption of an undirected graph is important in this chapter for the case where packet losses exist and we require that the vehicles send an acknowledgment message to their in-neighbours acknowledging that a data message was received. If we assume that there are no packet losses, this assumption can be lifted.

Using a Lyapunov-based design and backstepping techniques, we propose a decentralized feedback law for \tilde{v}_{ri} as a function of the information obtained from the neighbouring agents. Following [32], let Q be an $(N-1) \times N$ matrix with orthonormal rows that are each orthogonal to $\mathbf{1}$, that is,

$$Q\mathbf{1} = \mathbf{0}, \quad QQ^T = I_{N-1}, \quad Q^T Q = \Pi := I_N - \frac{1}{N}\mathbf{1}\mathbf{1}^T. \quad (1.22)$$

We introduce the coordination or synchronization error vector

$$\zeta := Q\gamma, \quad (1.23)$$

where $\gamma := \text{col}(\gamma_i)$. From (1.18), the dynamics of the coordination subsystem can be written in vector form as

$$\dot{\gamma} = \bar{v}_r + \tilde{v}_r + \bar{e}_\gamma \quad (1.24)$$

where $\bar{v}_r := \text{col}(v_r(\gamma_i))$, $\bar{e}_\gamma := \text{col}(e_{\gamma_i})$, and $\tilde{v}_r := \text{col}(\tilde{v}_{r_i})$. Consider the control Lyapunov function $V := \frac{1}{2}\xi^T\xi$. Computing its time-derivative yields

$$\dot{V} = \xi^T Q(\bar{v}_r + \tilde{v}_r + \bar{e}_\gamma). \quad (1.25)$$

To make ξ ISS with respect to input \bar{e}_γ , a natural choice would be $\tilde{v}_r = -kLQ^T\xi = -kL\Pi\gamma = -kL\gamma$, where L is the Laplacian of the graph $(\mathcal{N}, \mathcal{A})$ and k is a positive scalar, or equivalently, $\tilde{v}_{r_i} = -k \sum_{j \in \mathcal{N}^i} (\gamma_i - \gamma_j)$ (the so-called neighbouring rule). To reduce the communication rate using a logic based dynamical system, we will lift the assumption that each agent receives information from its neighbourhood continuously. We assume instead that it relies on the estimates $\hat{\gamma}_j^i$; $j \in \mathcal{N}^i$. Therefore, defining the estimation error $\tilde{\gamma}_j^i := \hat{\gamma}_j^i - \gamma_j$, the coordination control law becomes

$$\tilde{v}_{r_i} = -k \sum_{j \in \mathcal{N}^i} (\gamma_i - \hat{\gamma}_j^i) = -k \sum_{j \in \mathcal{N}^i} (\gamma_i - \gamma_j) + k \sum_{j \in \mathcal{N}^i} \tilde{\gamma}_j^i \quad (1.26)$$

or, in vector form, $\tilde{v}_r = -kLQ^T\xi + k\bar{\gamma}$, where $\bar{\gamma} := \text{col}(\sum_{j \in \mathcal{N}^i} \tilde{\gamma}_j^i)$. The time derivative of V becomes

$$\dot{V} = -k\xi^T Q L Q^T \xi + \xi^T Q (v_r + \bar{e}_\gamma + k\bar{\gamma}). \quad (1.27)$$

In this case, the term $-k\xi^T Q L Q^T \xi$ is negative definite provided that the graph that models the constraints imposed by the network topology among the agents is connected, see e.g. [28]. The following result is obtained.

Lemma 2 (Coordination): *Consider a set of N agents Σ_i ; $i \in \mathcal{N}$ with dynamics (1.1), a set of controllers Σ_i^{pf} ; $i \in \mathcal{N}$ with dynamics (1.2), and the error signal of Definition 2, and assume that v_r is globally Lipschitz continuous with a Lipschitz constant smaller than or equal to l . If $(\mathcal{N}, \mathcal{A})$ is connected, then the coordination control of problem of Definition 6 is solved with the coordination control law (1.26), that is, ξ is ISS with respect to the inputs e_i and $\tilde{\gamma}_j^i$ for $k > l/\sigma_2$, where σ_2 is the second lowest singular of L . Stated equivalently, there exist functions $\beta^\xi \in \mathcal{KL}$, $\sigma_\gamma^\xi, \sigma_e^\xi \in \mathcal{K}_\infty$ such that the evolution of the coordination error signal ξ satisfies the inequality*

$$\|\xi(t)\| \leq \beta^\xi(\|\xi(0)\|, t) \oplus \max_{i \in \mathcal{N}; j \in \mathcal{N}^i} \left(\sigma_\gamma^\xi(\|\tilde{\gamma}_j^i\|_{[0,t]}) \oplus \sigma_e^\xi(\|e_i\|_{[0,t]}) \right). \quad (1.28)$$

Proof. See the Appendix. □

1.4.4 Logic-based communication system

In this subsection we present the logic-based communication system. We will start with the case where the communication links are ideal, that is, there are no delays or packet losses. We then move on to the case where there are bounded communication delays. Finally, we will describe a communication system that is robust to limited packet losses.

1.4.4.1 Ideal communication links

The logic-based communication system structure for a node i , with its neighbours j_1 to j_{d_i} , with $d_i := |\mathcal{N}^i|$ belonging to \mathcal{N}^i , is illustrated in Figure 1.2. The communication system is composed by observers of the path-following variables of the neighbours j_1 to j_{d_i} , $\hat{\gamma}_{j_1}^i$ to $\hat{\gamma}_{j_{d_i}}^i$, and an observer of the own path-following variable, $\hat{\gamma}_i^i$. The observers of the path-following variables of the neighbours $\hat{\gamma}_j^i$ are reset when a message is received by the respective neighbour. Equivalently, the observer of the local path-following variable, $\hat{\gamma}_i^i$ only uses data that are sent to the neighbours. The observer of the local path-following variable, $\hat{\gamma}_i^i$ is used for the computation of the communication triggering condition in the following manner. The emitter compares the observed path-following variable $\hat{\gamma}_i^i$ to the real γ_i and if the norm of the difference reaches the value ϵ a data message is sent to the neighbours containing the present γ_i . Let t_k^i , $k > 0$ denote the instants of time at which agent i transmits information

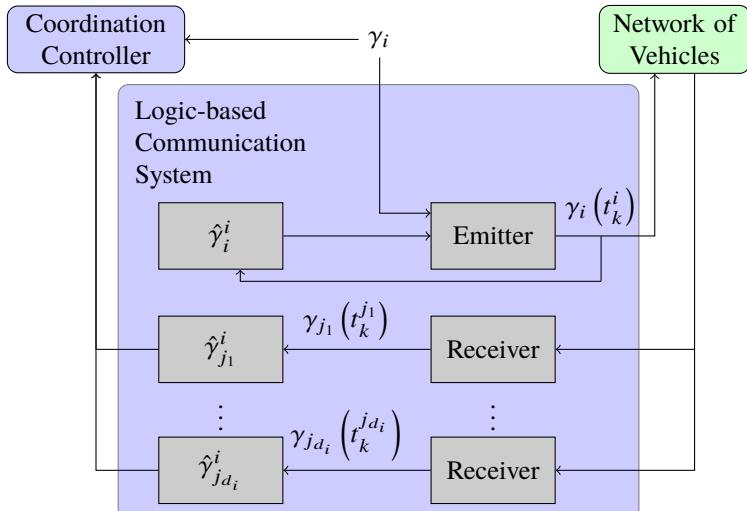


Figure 1.2: Logic-based communication system for ideal communication links.

to the neighbours. Following the procedure described in Section 1.3, we propose for each agent i the following logic-based communication system:

$$\begin{cases} \dot{\hat{\gamma}}_i^j(t) = v_r(\hat{\gamma}_i^j), & \text{for } t_k^i < t < t_{k+1}^i, \\ \hat{\gamma}_i^j(t_k^{i+}) = \gamma_i(t_k^i), & \text{for } t = t_k^i. \end{cases}$$

Since the communication links are ideal, $\hat{\gamma}_i^i = \hat{\gamma}_i^j$ for all $j \in \mathcal{N}^i$ and all $t \in \mathbb{R}^+$.

We note that if we define $T_r(\gamma) := \int_0^\gamma \frac{1}{v_r(\sigma)} d\sigma$ and its inverse as $\Gamma_r(t)$, that is, $T_r(\Gamma_r(t)) := t$, we have for $t_k^i < t < t_{k+1}^i$ that $\hat{\gamma}_i(t) = \Gamma_r(t - t_k^i + T_r(\gamma_i(t_k^i)))$. Therefore, the estimator can also be expressed as

$$\hat{\gamma}_i^j(t) = \Gamma_r(t - t_r^i(t) + T_r(\gamma_i(t_r^i(t)))), \quad (1.29)$$

where $t_r^i(t) := \max_{k \in \mathbb{N}, t_k^i < t} t_k^i$ is the last time agent i sent a data message. When v_r is constant, as in the field tests of Section 1.5, the above expression simplifies to $\hat{\gamma}_i(t) = \gamma_i(t_r^i(t)) + (t - t_r^i(t)) v_r$. The choice of the estimator (1.29) is motivated by its simplicity and the fact that if, for some $t_0 > 0$, $\bar{e}_\gamma(t) = \mathbf{0}$ and $\xi(t) = \mathbf{0}$ for $t > t_0$, if $\hat{\gamma}_i^j(t_\gamma) = \gamma_i(t_\gamma)$ then $\hat{\gamma}_i^j(t) = \gamma_i(t) = 0$ for $t > t_0$. However, it is possible to choose more complex reproductions of the corresponding dynamic models, see [33] and [24].

To bound the estimation errors we introduce the communication triggering condition (CTC) $\|\tilde{\gamma}_i^j\| \geq \epsilon$ where $\epsilon > 0$, with $\tilde{\gamma}_i^j := \hat{\gamma}_i^j - \gamma_i$ (see Definition 8). In this setup, agent i transmits to j a data message with γ_i at time t_k^i when the CTC is satisfied. Note that the post reset value of $\tilde{\gamma}_i^j$ is $\tilde{\gamma}_i^j(t_k^i) = 0$. Consequently, at any time, $\|\tilde{\gamma}_i^j\| \leq \epsilon$.

1.4.4.2 Delayed information

We now consider the case where the communication channels have bounded, time-varying and non-homogeneous delays. Consider the following situation: agent i sends data at time t_k^i , and agent j receives it at time $t_k^i + \tau_k^{ij}$. We assume that

$$\tau_k^{ij} \leq \bar{\tau}, \quad \forall i, j, k$$

where the constant $\bar{\tau} > 0$ is known a priori. We also assume that all the vehicles run synchronized clocks. This is a realistic assumption in view of the recent availability of affordable chip atomic clocks [34]. The main idea is to keep the estimators always synchronized, therefore both the emitter and the receiver only update the estimators at some time, with the same information, at $t_k^i + \bar{\tau}$.

Suppose that at time t_k^i agent i transmits a message that contains the following data: $\{t_k^i, \gamma_i\}$. Then, the estimators $\hat{\gamma}_i$ in agent i and its neighbours \mathcal{N}^i cannot be immediately updated. This is because we must guarantee that the value of the state estimate $\hat{\gamma}_i$ can be computed in all agents in \mathcal{N}^i . To this end, both estimates can only be updated at time $t = t_k^i + \bar{\tau}$. Upon receiving t_k^i , the coordination state estimate $\hat{\gamma}_i^j$ running in agent j should be updated at time $t = t_k^i + \bar{\tau}$ to

$$\hat{\gamma}_i^j((t_k^i + \bar{\tau})^+) = \Gamma_r(\bar{\tau} - t_k^i + T_r(\gamma_i(t_k^i)))$$

In the case of a constant v_r , this simplifies to $\hat{\gamma}_i^j((t_k^i + \bar{\tau})^+) = \gamma_i(t_k^i) + \bar{\tau} v_r$. With the above procedure, we guarantee that the estimators are always synchronized. The estimator can thus be described as in (1.29) where $t_r^i(t)$ is redefined as $t_r^i(t) :=$

$\max_{k \in \mathbb{N}, t_k^i + \bar{\tau} < t} t_k^i$. Therefore, one can consider the ideal case mentioned previously as a particular instance of the delayed communication case with $\bar{\tau} = 0$.

Notice that in general $\tilde{\gamma}_i^j(t_k^i + \bar{\tau})$ will not be zero because $e_{\tilde{\gamma}_i}$ and ξ may not be zero in the interval $[t_k^i, t_k^i + \bar{\tau}]$. Thus, we can only guarantee that this technique is valid if $\bar{\tau}$ so small that $\tilde{\gamma}_i^j$ satisfies $\tilde{\gamma}_i^j((t_k^i + \bar{\tau})^+) < \epsilon$. The stability guarantees for the delayed case will be stated in Theorem 3.

1.4.4.3 Communication losses

We now address the case when the data messages are not always received, i.e. there are packet losses. To make the communication system robust to limited communication losses we require each agent to send an acknowledgment message upon receiving a data message. The agent which sent the data message only updates his estimators $2\bar{\tau}$ time units after the data message has been sent in case the acknowledgment messages from all the agents was received, otherwise another data message with the same content is sent. This process is repeated until the agent receives the acknowledgment messages from all the neighbours. This guarantees that the receiving agents receive a data message at some point after the CTC is satisfied.

Consider the case where at time t_k^i agent i transmits to agent j a data message, which contains the following data: $\{t_k^i, \gamma_i\}$. Upon receiving the data message and sending an acknowledgment message, the coordination state estimate $\hat{\gamma}_i^j$ running in agent j should be updated at time $t = t_k^{ij} + 2\alpha_k^{ij}\bar{\tau}$, where α_k^{ij} is the number of messages containing $\{t_k^i, \gamma_i\}$ sent by agent i before agent j receives it, to

$$\hat{\gamma}_i^j((t_k^i + 2\alpha_k^{ij}\bar{\tau})^+) = \Gamma_r(2\alpha_k^{ij}\bar{\tau} - t_k^i + T_r(\gamma_i(t_k^i))).$$

Equivalently we define as α_k^{ii} the number of messages containing $\{t_k^i, \gamma_i\}$ that agent i sends after t_k^i , before at least one acknowledgment from all of its neighbours is received. The estimator can be represented formally as

$$\hat{\gamma}_i^j(t) = \Gamma_r(t - t_r^{ij}(t) + T_r(\gamma_i(t_r^{ij}(t)))), \quad (1.30)$$

where $t_r^{ij}(t) := \max_{k \in \mathbb{N}, t_k^i + 2\alpha_k^{ij}\bar{\tau} < t} t_k^i$. In what follows we assume that for a finite number N_{\max} of consecutive data messages sent containing the same data, at least one acknowledgment message from each of the neighbours is received. That is, we assume that there exists an integer N_{\max} such that for any $i \in \mathcal{N}$ and $j \in \mathcal{N}^i$ and $k \geq 0$, $\alpha_k^{ij} \leq \alpha_k^{ii} \leq N_{\max}$.

Note that if one agent receives a message while another misses it, then, unlike the case without communication losses, there is a brief period when the estimators are desynchronized, that is $\hat{\gamma}_i^j \neq \hat{\gamma}_i^l$ for $j \neq l$. However, the synchronization is recovered after the acknowledgment messages of all the neighbour agents are received. As in the case without communication losses, if $\bar{\tau} := 2\bar{\tau}N_{\max}$ is sufficiently small so as to guarantee that $\tilde{\gamma}_i^j$ satisfies $\tilde{\gamma}_i^j((t_k^i + 2\alpha_k^{ij}\bar{\tau})^+) < \epsilon$, it follows that the estimation error is bounded. We can now show that the proposed logic-based communication system

in the presence of communication losses and delays, yields bounded estimation error if certain conditions apply.

Theorem 3: Consider a set of N agents Σ_i ; $i \in \mathcal{N}$ with dynamics (1.1), equipped with CPFCS structures consisting of path-following controllers Σ_i^{pf} ; $i \in \mathcal{N}$ and co-ordination controllers Σ_i^{cc} ; $i \in \mathcal{N}$ defined by, (1.2) and (1.5) respectively, that solve the CPF problem of Definition 7, together with the proposed logic-based communication system of this section in the presence of delayed information and communication losses. Further assume that $\sigma_{\xi}^{\tilde{v}_r}(s) = \alpha_{\xi}^{\tilde{v}_r}s$, $\sigma_{\gamma}^{\tilde{v}_r}(s) = \alpha_{\gamma}^{\tilde{v}_r}s$, $\sigma^e(s) = \alpha^e s$ and $\sigma_{\gamma}^{\bar{e}}(s) = \alpha_{\gamma}^{\bar{e}}s$, for appropriately defined positive constants $\alpha_{\xi}^{\tilde{v}_r}$, $\alpha_{\gamma}^{\tilde{v}_r}$, α^e and $\alpha_{\gamma}^{\bar{e}}$. Then, for sufficiently small values of $\tilde{\tau} := 2\bar{\tau}N_{max}$, $\|\bar{e}(0)\|$, $\max_{i \in \mathcal{N}} \|v_i\|_{[0,\infty)}$, and $\max_{i \in \mathcal{N}} \|w_i\|_{[0,\infty)}$, the CPFCS structures solve the CPF problem.

In particular, we can compute continuous function $\bar{\alpha}^\epsilon : [0, c) \rightarrow [1, \infty)$ and $\sigma^\epsilon : [0, c) \rightarrow [0, \infty)$ for some positive constant c with $\bar{\alpha}^\epsilon(0) = 1$ and $\sigma^\epsilon(0) = 0$, such that if

$$\epsilon > \sigma^\epsilon(\tilde{\tau}) \left(\beta^{\bar{e}}(\|\bar{e}(0)\|, 0) \oplus \max_{i \in \mathcal{N}} (\sigma_w^{\bar{e}}(\|w_i\|_{[0,\infty)}) \oplus \sigma_v^{\bar{e}}(\|v_i\|_{[0,\infty)})) \right) \quad (1.31)$$

holds then, for all $t \geq 0$,

$$\max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \|\tilde{\gamma}_i^j(t)\| \leq \bar{\alpha}^\epsilon(\tilde{\tau})\epsilon. \quad (1.32)$$

Proof. See the Appendix. \square

Remark 5: In the ideal communications case where $N_{max} = \bar{\tau} = 0$ we have that the proposed estimators of the communication system follow the structure in (1.10) with $x_{ij}^{lbc} := \hat{\gamma}_j$ and Theorem 2 guarantees that the estimation errors are bounded, and consequently from Theorem 1, if (1.9) is satisfied, the closed-loop system solves the CPF problem.

Theorem 3 goes further than the ideal communications case considered in Section 1.3, since it also addresses the case where there are packet losses and delays.

Theorem 3 provides stability guarantees in the case of delays and packet losses. The following corollaries of Theorem 3 provide stability guarantees for the case of ideal communication links.

Corollary 1 (Ideal Communication): The CPFCS structure considered in Theorem 3 solve the CPF Problem for ideal communication links, i.e. when there are no delays or communication losses.

Proof. Since the ideal communication case can be viewed as a particular case of that with delayed communication for which $\bar{\tau} = 0$, Theorem 3 provides a proof of stability of the overall system for the case with ideal communication links. Moreover, it follows from Theorem 3 that the boundedness assumptions on $\|\bar{e}(0)\|$, $\max_{i \in \mathcal{N}} \|v_i\|_{[0,\infty)}$, and $\max_{i \in \mathcal{N}} \|w_i\|_{[0,\infty)}$ are not required anymore since the condition (1.31) is always satisfied because $\sigma^\epsilon(\tilde{\tau})(0) = 0$. \square

1.4.5 Stability of the overall-closed loop system

Applying the previous results we can now show that the CPFCS structure consisting of the path-following controllers described in Section 1.4.2, the coordination controllers described in Section 1.4.3, and logic-based communication systems described in Section 1.4.4 solves, for the class of AMVs described in Section 1.4.1, the CPF problem globally in the case of ideal communication links and locally, i.e. for sufficiently small values of $\|\bar{e}(0)\|$, $\max_{i \in \mathcal{N}} \|v_i\|_{[0,\infty)}$ and $\max_{i \in \mathcal{N}} \|w_i\|_{[0,\infty)}$, the case of delayed communications. The class of admissible missions \mathcal{M}_i is defined as follows: we say that $(z_{d_i}, v_r) \in \mathcal{M}_i$ if $z_d \in C_2^1$ and $v_r \in C_1^0$ is globally Lipschitz continuous with a Lipschitz constant lower or equal to a non-negative scalar l .

Theorem 4: Consider the overall closed-loop system consisting of N AMVs with the dynamics given by (1.13). Assume the AMVs are equipped with inner loop controllers satisfying Assumption 1 and CPFCS structures consisting of path-following controllers of the form (1.19)-(1.20), with $k_x, k_y, k_\gamma, c_\gamma > 0$, and coordination controllers of the form (1.26), with $k > l/\sigma_2$, together with the proposed logic-based communication systems. For the case of ideal communication links or, in the presence of delayed information and/or communication losses, for sufficiently small values of $\bar{\tau}$, $\|\bar{e}(0)\|$, $\max_{i \in \mathcal{N}} \|v_i\|_{[0,\infty)}$, and $\max_{i \in \mathcal{N}} \|w_i\|_{[0,\infty)}$, the CPFCS structures solve the CPF problem, for every prescribed speed profile v_r and paths z_{d_i} ; $i \in \mathcal{N}$ satisfying $(z_{d_i}, v_r) \in \mathcal{M}_i$; $i \in \mathcal{N}$.

Proof. Lemma 1 implies that the set of Σ_i^{pf} ; $i \in \mathcal{N}$ solves the path-following problem with $\sigma_{\bar{v}_r}^e \equiv 0$. Therefore, given Lemma 2 and since (1.9) holds, Theorem 1 implies that the CPFCSSs solve the CPF problem. The conclusion of the theorem follows from a straightforward application of Theorem 3 and its Corollary 1. \square

1.5 Field tests with AMVs

This section describes the results of field trials with AMVs that illustrate the efficacy of the proposed cooperative path following algorithms with logic-based communications derived previously.

1.5.1 Test Set-up

The field tests were performed with 3 Medusa-class AMVs built at IST. We henceforth refer to the vehicles as Black, Red, and Yellow Medusas, corresponding to their colors. See Figure 1.3 for a close up and an aerial view of the vehicles while maneuvering in coordination at sea. For simplicity of systems implementation, in the tests the vehicles operated at the surface, acting as proxies for autonomous underwater vehicles. Each Medusa vehicle was equipped with a navigation system using GPS-RTK that allowed for the computation of its position. In the set-up adopted, the vehicles communicated over Wi-Fi. The software modules were implemented using the Robot Operating System (ROS), in Python, and ran on an EPIC computer board (model NANO-PV-D5251). All Medusa-class vehicles are equipped with two thrusters on starboard and portside, to generate longitudinal forces and torque about the vertical axis. Motor

drivers allow the onboard computer to assign an RPM control signal to each thruster, which will in turn generate a desired thrust force. The Medusas are also equipped with low-level control systems that the CPF algorithms build upon. We now describe the surge speed and heading rate control loops.

Surge speed: The surge speed controller is responsible for making the vehicle track a desired surge speed command. Estimates of the state to be controlled can be obtained from a navigation filter which uses GPS data by projecting the velocity with respect to the ground on the longitudinal axis of the vehicle, and subtracting the projection of the water current vector estimate on the longitudinal axis of the vehicle. A Proportional-Integral (PI) control law computes the commanded common mode (CM) for the two longitudinal thrusters, where the common mode is defined as the mean of the commands for the two thrusters, as a percentage of the maximum RPMs, defined as $CM(\%) := \frac{\text{Left RPMs}(\%) + \text{Right RPMs}(\%)}{2}$, so as to make the vehicle move at the desired surge speed.

Heading rate: The heading rate controller is responsible for making the vehicle track a desired heading rate command. Estimates of the heading rate are performed using a filter with data from an Attitude and Heading Reference System (AHRS). A PI control law computes the commanded differential mode (DM) for the two longitudinal thrusters, where the common mode is defined as the mean of the commands for the two thrusters, as a percentage of the maximum RPMs, defined as $DM(\%) := \frac{\text{Left RPMs}(\%) - \text{Right RPMs}(\%)}{2}$, to make the vehicle rotate at the desired heading rate.

For further details on the technology used in the Medusa vehicles, we refer the reader to [35]. The communication topology between the vehicles used during the



a - The Medusa-class of vehicles



b - The Red, Yellow and Black Medusas performing a cooperative maneuver at sea.

Figure 1.3: The Medusa AMVs and an aerial snapshot during the tests.

field tests is one where the red vehicle communicates with the black and the yellow vehicles, but the yellow and black vehicles do not communicate among themselves. Furthermore, in order to emulate the conditions encountered underwater, communications occur asynchronously, as defined by the logic-based communication system. The type of mission performed in all the reported trials was the following:

- Nominal path for the RED vehicle: A lawnmowing path with 30m length for straight line segments and 12m radius for circumference segments

- Formation: Alongside alignment, with 5m of separation and the red vehicle at the center following the nominal path.
- Normalized speed along the nominal path: $v_r = 0.5\text{m/s}$

In each test, a saturation of the RPM control signal in one of the vehicles (the Medusa-black vehicle) was temporarily enforced in order to test the resilience of the proposed algorithms to transient events that may have a negative impact on coordination. In the following plots, when displaying the dimensionless path-following variable, the latter corresponds to the arc-length in meters along the nominal path, i.e. the path taken by the red vehicle.

In the field tests, the vehicles performed coordinated path-following with logic-based communications with values of $\epsilon = 0.2\text{m}$, $\epsilon = 0.6\text{m}$, and $\epsilon = 1.4\text{m}$. The current was negligible and thus the current estimator used, akin to that proposed in [24] but using a simple quasi-steady relationship between command mode thruster activity and surge speed and assuming negligible sideslip, had almost no effect on the system. The implementation of the general architecture proposed for cooperative path following relied on the inner loop controllers installed on board the vehicles, together with the systems in charge of path following, coordination, and logic based communications. Adopting the theoretical framework for integrated system design, tuning of all system parameters was done in simulation using a dynamic model of the Medusa vehicles. The reader will find in [36] a description of a model of the Medusa vehicles, together with details of the Matlab-based simulation used for mission simulation and parameter tuning. The following values for the most relevant parameters were used on all the vehicles during the trials:

- Path-following controller: $k_x = k_y = 0.35$, $k_{\dot{\gamma}} = 1$, $c_{\dot{\gamma}} = 0.5$, $\delta = 0.2\text{m}$.
- Coordination controller: $k = 1$.

In what follows, due to space limitations, we focus our attention on the mechanism adopted for logic-based communications.

1.5.2 Results

This section contains the results of the field tests described before. For all the performed tests in this section, the saturation of the propeller rotation in the black vehicle, occurs between 400 and 550 seconds from the beginning of the mission, which corresponds to the period from the beginning of the third circular segment to the middle of the fourth straight line segment, see Fig. 1.4.

1.5.2.1 Test with $\epsilon = 0.2$

Figures 1.4 and 1.5 show the vehicle paths and the evolution of the path-following variables for $\epsilon = 0.2$. The path-following variable is in meters.

One can observe from Figures 1.4 and 1.5 that coordination is maintained throughout the test, with the vehicles running at a lower speed during the enforced saturation. This is because the coordination scheme will temporarily force the red and yellow vehicles to slow down in an attempt to remain in formation with the black vehicle. Once the saturation is removed, all vehicles speed up to coordinate at the

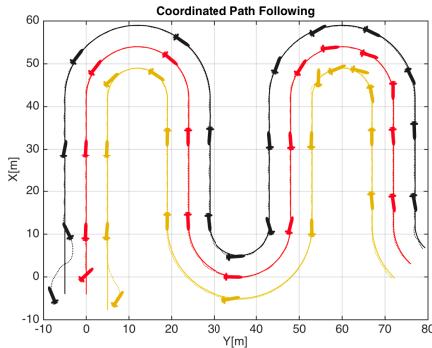


Figure 1.4 Vehicle paths for logic-based communications with $\epsilon = 0.2$.

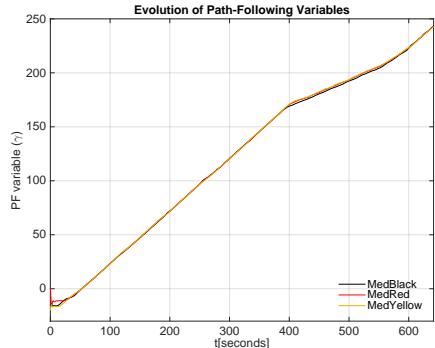


Figure 1.5 Path-following variables for logic-based communications with $\epsilon = 0.2$.

same normalized, desired speed. The communication events among the vehicles and the estimation error of the vehicles are plotted in Figure 1.6. From this figure one

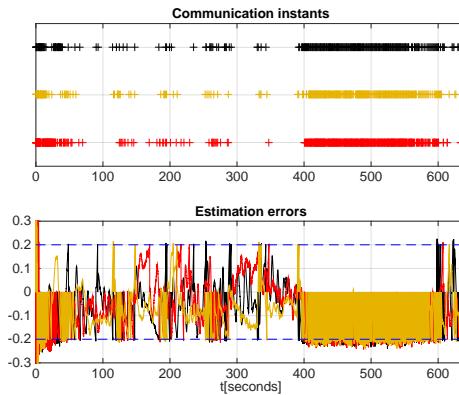


Figure 1.6: Communication events and estimation error on the red vehicle for $\epsilon = 0.2$. Crosses mark the times at which communications take place.

can observe large periods without communications after the initial period when the vehicles are still converging to the paths from their initial positions and before the enforced saturation. However, the overall number of communications is still high. During the enforced saturation, the vehicles move at a constant speed which is lower than their reference. Therefore, between communications the estimation error norm increases at a constant rate and the vehicles communicate at a fixed rate. In this case, the period between communications is approximately 1 second. As expected, since we do not have packet losses in this test, the estimation error of all the vehicles is almost always below 0.2, and only passes slightly 0.2 due to the sampling rate of the computer, set at 0.2 seconds.

1.5.2.2 Test with $\epsilon = 0.6$

For $\epsilon = 0.6$, the path following variables of the three vehicles are show in Figure 1.8. Figure 1.7 contains the communication instants and the estimation errors of the path-following variables.

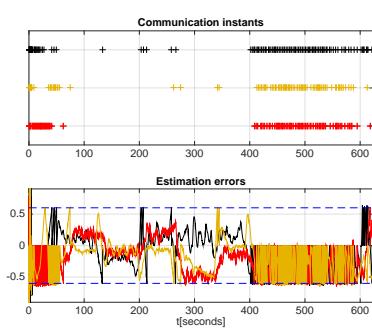


Figure 1.7 Communication events and estimation errors for $\epsilon = 0.6$.

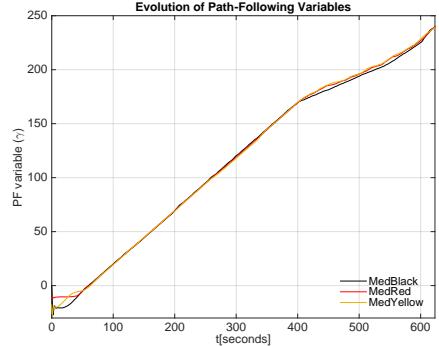


Figure 1.8 Path-following variables for logic-based communications with $\epsilon = 0.6$.

From Figure 1.8 it is visible a slightly worse performance during the saturation than for the case with $\epsilon = 0.2$. From Figure 1.7 one can observe that after the initial convergence and before the enforced saturation, the communications are sporadic and occur only at the instants when the vehicles enter or leave the circular segments. This is because, in order to maintain coordination on the circular segments the outer vehicle must move faster than the nominal speed and the inner vehicle more slowly, also with respect to the nominal normalized speed. As a consequence, the outer and inner vehicles are required to respectively accelerate and decelerate instantaneously to follow their desired paths.

1.5.2.3 Test with $\epsilon = 1.4$

For $\epsilon = 1.4$, the vehicle path following variables are shown in Figure 1.10. The communication events among the vehicles are shown in Figure 1.9.

Throughout most of the mission the vehicles are well coordinated. However, it is apparent that during the enforced saturation we have a much worse performance than in the previous cases, with the black vehicle further behind the other two vehicles. In this case we obtain a very low number of exchanged messages after the vehicles reach coordination and before the saturation. In total, after coordination and before saturation we can count only two communications from the black vehicle and one from the yellow. This illustrates clearly the tradeoff involved in performance versus communication rates.

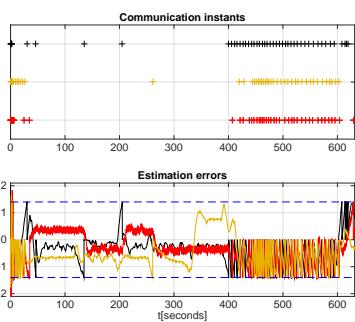


Figure 1.9 Communication events and estimation errors for $\epsilon = 1.4$.

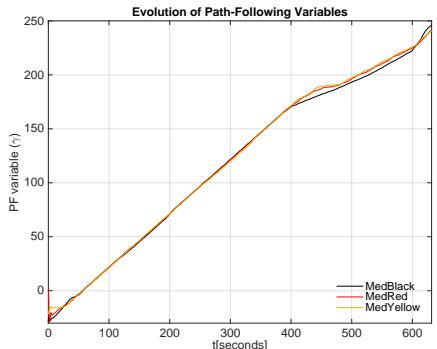


Figure 1.10 Path-following variables for logic-based communications with $\epsilon = 1.4$.

1.6 Conclusions

A distributed control system for multiple vehicles was introduced that makes use of logic-based communications with low bandwidth requirements to solve robustly the CPF problem, i.e. to steer and maintain the vehicles at pre-defined paths at a desired common normalized speed profile, while holding a given formation pattern. The control system takes into account the topology of the communication network between vehicles and the fact that communication takes place at discrete instants of time, instead of continuously. The communication logic also considers packet losses and delays in the communication network. Using the mathematical tools of graph theory and Lyapunov stability, conditions were derived under which the overall closed loop system is input-to-state stable. To show how the theoretical framework adopted can be applied in practice, tests were done with a group of three autonomous marine robots. The tests showed that the framework holds considerable promise for real system implementation. Future work will address the implementation of proposed cooperative path following strategy for a larger number of vehicles exchanging messages over hybrid acoustic/optical modems.

Acknowledgements

We thank the DSOR team, composed by Jorge Ribeiro, Henrique Silva, Manuel Rufino, Pedro Góis, Miguel Ribeiro, João Botelho and Luís Sebastião for their efforts in the development of the MEDUSA vehicles and their support and collaboration on the planning and execution of our sea trials.

This work was supported by the H2020 EU Marine Robotics Research Infrastructure Network (Project ID 731103), the MarineUAS project under the Marie Curie Skłodowska grant agreement No 642153, and the Portuguese FCT Project UID/EEA/5009/2013. The fist author benefited from grant SFRH/BD/51929/2012 of the Foundation for Science and Technology (FCT), Portugal and grants BL23/2018-IST-ID and BL24/2018-IST-ID of the FCT-CMU + Atlântico project.

Appendix

Proof of Lemma 1. The proof exploits the use of the Lyapunov function defined as $V := \frac{1}{2}\|\mathbf{e}\|^2 = \frac{1}{2}(\|e_\delta\|^2 + c_{\dot{\gamma}} e_{\dot{\gamma}}^2)$.

Equations (1.17) and (1.20) yield

$$\dot{\mathbf{e}} = -S(r)e_\delta - K_k e_\delta - R^T \frac{\partial z_d}{\partial \gamma} e_{\dot{\gamma}} - \begin{bmatrix} 0 \\ \delta \tilde{r} \end{bmatrix} + \begin{bmatrix} \tilde{u} \\ v_w \end{bmatrix} + \tilde{\mathbf{v}}_c.$$

Because $x^T S(r)x = 0; \forall x \in \mathbb{R}^2$, the time derivative of V is given by

$$\begin{aligned} \dot{V} &= e_\delta^T \left(-K_k e_\delta - \begin{bmatrix} 0 \\ \delta \tilde{r} \end{bmatrix} + \begin{bmatrix} \tilde{u} \\ v_w \end{bmatrix} + \tilde{\mathbf{v}}_c \right) - k_{\dot{\gamma}} c_{\dot{\gamma}} e_{\dot{\gamma}}^2 \\ &= -e_\delta^T K_k e_\delta - c_{\dot{\gamma}} k_{\dot{\gamma}} e_{\dot{\gamma}}^2 + e_\delta^T \left(\begin{bmatrix} \tilde{u} \\ v_w \end{bmatrix} - \delta \begin{bmatrix} 0 \\ \tilde{r} \end{bmatrix} + \tilde{\mathbf{v}}_c \right) \end{aligned}$$

Defining $K_e := \text{diag}\{k_x, k_y, k_{\dot{\gamma}}\}$ we obtain

$$\dot{V} \leq -\mathbf{e}^T K_e \mathbf{e} + \|\mathbf{e}\| ((1 + \delta)\|w\| + \|v\|) \quad (\text{A.33})$$

The rest of the proof involves dominating all positive terms of (A.33) by $-\mathbf{e}^T K_e \mathbf{e} \leq -\min(k_x, k_y, k_{\dot{\gamma}})\|\mathbf{e}\|^2$. To do this, notice from (A.33) that, defining a positive scalar θ in the interval $(0, 1)$,

$$\dot{V} \leq -(1 - \theta) \min(k_x, k_y, k_{\dot{\gamma}})\|\mathbf{e}\|^2 + \|\mathbf{e}\| \left((1 + \delta)\|w\| + \|v\| - \theta \min(k_x, k_y, k_{\dot{\gamma}})\|\mathbf{e}\| \right)$$

and therefore, defining

$$\sigma_w^e(x) := \frac{2(1 + \delta)x}{\theta \min(k_x, k_y, k_{\dot{\gamma}})}, \quad \sigma_v^e(x) := \frac{2x}{\theta \min(k_x, k_y, k_{\dot{\gamma}})},$$

one obtains

$$\dot{V} \leq -(1 - \theta) \min(k_x, k_y, k_{\dot{\gamma}})\|\mathbf{e}\|^2, \forall \|\mathbf{e}\| \geq \sigma_w^e(\|w\|) \oplus \sigma_v^e(\|v\|),$$

from which (1.21) follows with $\beta^e(x, t) := x e^{-(1 - \theta) \min(k_x, k_y, k_{\dot{\gamma}})t}$. \square

Proof of Lemma 2. Defining $\bar{\gamma} := \frac{1}{N} \mathbf{1}^T \gamma$, γ can be decomposed as $\gamma = \mathbf{1}\bar{\gamma} + Q^T \xi$. Using this fact and recalling that $v_r(\gamma)$ is assumed to be globally Lipschitz we can conclude that

$$\xi^T Q \bar{v}_r = \xi^T Q \Pi (\mathbf{1} v_r(\bar{\gamma}) - (\mathbf{1} v_r(\bar{\gamma}) - \bar{v}_r)) = \xi^T Q \Pi (\bar{v}_r - \mathbf{1} v_r(\bar{\gamma})) \leq l \|\xi\|^2.$$

From the definition of $\bar{e}_{\dot{\gamma}}$ and Definition 2 one has that

$$\|\bar{e}_{\dot{\gamma}}\| \leq \sqrt{N} \max_{i \in \mathcal{N}} \sigma^e(\|\mathbf{e}_i\|) \quad (\text{A.34})$$

We now use the fact that, since $(\mathcal{N}, \mathcal{A})$ is connected, \mathbf{L} has a simple eigenvalue at zero with an associated eigenvector $\mathbf{1}$ and the remaining eigenvalues are all positive,

see e.g. [28]. Applying the Courant-Fisher theorem in [37] and noting that $Q\mathbf{1} = \mathbf{0}$, it follows that

$$\xi^T Q L Q^T \xi \geq \sigma_2 \|\xi\|^2, \forall \xi \in \mathbb{R}^{N-1} \quad (\text{A.35})$$

From (1.25), (A.34), (A.35), and the fact that $\|\bar{\gamma}\| \leq d^* \sqrt{N} \max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \|\tilde{\gamma}_j^i\|$, we can bound the time derivative of V to obtain

$$\begin{aligned} \dot{V} &\leq -k\xi^T Q L Q^T \xi + l\|\xi\|^2 + \xi^T Q \bar{e}_\gamma + k\xi^T Q \bar{\gamma} \\ &\leq -k\sigma_2 \|\xi\|^2 + l\|\xi\|^2 + \sqrt{N} \|\xi\| \max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \left(\sigma^e(\|\boldsymbol{e}_i\|) + kd^* \|\tilde{\gamma}_j^i\| \right). \end{aligned}$$

As a consequence,

$$\dot{V} < -(1-\theta)(k\sigma_2 - l)\|\xi\|^2$$

for

$$\|\xi\| \geq \max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \left(\sigma_\gamma^\xi \left(\|\tilde{\gamma}_j^i\|_{[0,t]} \right) \oplus \sigma_e^\xi \left(\|\boldsymbol{e}_i\|_{[0,t]} \right) \right),$$

and $1 > \theta > 0$, where $\sigma_e^\xi(s) := \frac{2\sqrt{N}}{(k\sigma_2 - l)\theta} \sigma^e(s)$, $\sigma_\gamma^\xi(s) := \frac{2\sqrt{N}kd^*}{(k\sigma_2 - l)\theta} s$. The conclusions of the lemma follow with $\beta^\xi(x, t) := xe^{-(1-\theta)(k\sigma_2 - l)t}$ \square

The following sequence of lemmas is instrumental in the proof of Theorem 3.

Lemma 3: Consider the CPFCS strictures of Theorem 3 in the presence of delayed information and packet losses. If, for some $\bar{t} > 0$, for any time $t_k^i \leq \bar{t}$ the post-reset estimation errors $\tilde{\gamma}_i^j$; $j \in \mathcal{N}^i$ satisfy

$$\left\| \tilde{\gamma}_i^j \left((t_k^i + 2\alpha_k^{ij} \bar{\tau})^+ \right) \right\| < \epsilon, \forall j \in \mathcal{N}^i, \quad (\text{A.36})$$

then, for sufficiently small $\bar{\tau}$,

$$\max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \left\| \tilde{\gamma}_i^j \right\|_{[0, \bar{t} + \bar{\tau}]} \leq \alpha^\epsilon(\bar{\tau})\epsilon + \alpha^{\bar{e}}(\bar{\tau})\|\bar{e}\|_{[0, \bar{t} + \bar{\tau}]} \quad (\text{A.37})$$

for continuous functions $\alpha^\epsilon : [0, b] \rightarrow [1, \infty)$ and $\alpha^{\bar{e}} : [0, b] \rightarrow [0, \infty)$ and for some positive constant b with $\alpha^\epsilon(0) = 1$ and $\alpha^{\bar{e}}(0) = 0$.

Proof. If (A.36) is satisfied for any time $t_k^i < \bar{t}$, then using the same arguments as in the proof of Theorem 2, one obtains that for any time $t_k^i < \bar{t}$

$$\left\| \tilde{\gamma}_i^j(t_k^i) \right\|_{(t_k^i + 2\alpha_k^{ij} \bar{\tau}, t_{k+1}^i]} \leq \epsilon, \forall j \in \mathcal{N}^i. \quad (\text{A.38})$$

Recall also that we assumed that $2\alpha_k^{ij} \bar{\tau} \leq \bar{\tau}$ for all $i \in \mathcal{N}$, $j \in \mathcal{N}^i$ and $k \geq 0$. From the additional assumptions that $\sigma_\xi^{\bar{v}_r}(s) = \alpha_\xi^{\bar{v}_r}s$, $\sigma_\gamma^{\bar{v}_r}(s) = \alpha_\gamma^{\bar{v}_r}s$ and $\sigma^e(s) = \alpha^e s$, for

appropriately defined positive constants $\alpha_{\xi}^{\tilde{v}_r}$, $\alpha_{\gamma}^{\tilde{v}_r}$ and α^e we have that

$$\begin{aligned}\max_{i \in \mathcal{N}} \|\tilde{v}_{r_i}\|_{[0, \bar{t} + \tilde{\tau}]} &\leq \alpha_{\xi}^{\tilde{v}_r} \|\xi\|_{[0, \bar{t} + \tilde{\tau}]} + \alpha_{\gamma}^{\tilde{v}_r} \max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \|\tilde{\gamma}_i^j\|_{[0, \bar{t} + \tilde{\tau}]} \\ &\leq \alpha_{\xi}^{\tilde{v}_r} \|\bar{e}\|_{[0, \bar{t} + \tilde{\tau}]} + \alpha_{\gamma}^{\tilde{v}_r} \max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \|\tilde{\gamma}_i^j\|_{[0, \bar{t} + \tilde{\tau}]}, \\ \max_{i \in \mathcal{N}} \|e_{\dot{\gamma}_i}\|_{[0, \bar{t} + \tilde{\tau}]} &\leq \alpha^e \max_{i \in \mathcal{N}} \leq \|e_i\|_{[0, \bar{t} + \tilde{\tau}]} \leq \|\bar{e}\|_{[0, \bar{t} + \tilde{\tau}]}.\end{aligned}$$

From the definition of $\tilde{\gamma}_i^j$ and the fact that $\dot{\gamma}_i = v_r(\gamma_i) + \tilde{v}_{r_i} + e_{\dot{\gamma}_i}$ and $\dot{\hat{\gamma}}_i^j = v_r(\hat{\gamma}_i^j)$, when the derivative is defined, we have that for $t \in [t_k^i, t_k^i + 2\alpha_k^{ij}\tilde{\tau}]$

$$\dot{\hat{\gamma}}_i^j = v_r(\hat{\gamma}_i^j) - v_r(\gamma_i) - \tilde{v}_{r_i} - e_{\dot{\gamma}_i},$$

and, therefore,

$$\|\dot{\hat{\gamma}}_i^j\| \leq l \|\tilde{\gamma}_i^j\| + (\alpha_{\xi}^{\tilde{v}_r} + \alpha^e) \|\bar{e}\|_{[0, \bar{t} + 2\tilde{\tau}]} + \alpha_{\gamma}^{\tilde{v}_r} \max_{o \in \mathcal{N}, p \in \mathcal{N}^o} \|\tilde{\gamma}_o^p\|_{[0, \bar{t} + \tilde{\tau}]}.$$

Using the comparison lemma in [38] (Lemma 3.4) it follows that for $t \in [t_k^i, t_k^i + 2\alpha_k^{ij}\tilde{\tau}]$, $\|\tilde{\gamma}_i^j(t)\| < \Gamma(t)$, where $\Gamma : [t_k^i, \infty) \rightarrow [\epsilon, \infty)$ is defined by $\Gamma(t_k^i) = \epsilon$ and

$$\dot{\Gamma} = l\Gamma + (\alpha_{\xi}^{\tilde{v}_r} + \alpha^e) \|\bar{e}\|_{[0, \bar{t} + \tilde{\tau}]} + \alpha_{\gamma}^{\tilde{v}_r} \max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \|\tilde{\gamma}_i^j\|_{[0, \bar{t} + \tilde{\tau}]}.$$

As a consequence,

$$\begin{aligned}\|\tilde{\gamma}_i^j\|_{[t_k^i, t_k^i + 2\alpha_k^{ij}\tilde{\tau}]} &\leq \Gamma(t_k^i + \tilde{\tau}) \\ &\leq \epsilon e^{l\tilde{\tau}} + z(l, \tilde{\tau}) \left((\alpha_{\xi}^{\tilde{v}_r} + \alpha^e) \|\bar{e}\|_{[0, \bar{t} + \tilde{\tau}]} + \alpha_{\gamma}^{\tilde{v}_r} \max_{o \in \mathcal{N}, p \in \mathcal{N}^o} \|\tilde{\gamma}_o^p\|_{[0, \bar{t} + \tilde{\tau}]} \right),\end{aligned}$$

where

$$z(l, t) := \begin{cases} \frac{e^{lt}-1}{l}, & \text{if } l > 0 \\ t, & \text{if } l = 0 \end{cases}.$$

From (A.38),

$$\max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \|\tilde{\gamma}_i^j\|_{[0, \bar{t} + \tilde{\tau}]} \leq \max_{i \in \mathcal{N}, j \in \mathcal{N}^i, t_k^i \leq \bar{t}} \|\tilde{\gamma}_i^j\|_{[t_k^i, t_k^i + 2\alpha_k^{ij}\tilde{\tau}]}.$$

Thus, if $\tilde{\tau}$ is small enough so that $\alpha_{\gamma}^{\tilde{v}_r} z(l, \tilde{\tau}) < 1$, we have

$$\begin{aligned}\max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \|\tilde{\gamma}_i^j\|_{[0, \bar{t} + \tilde{\tau}]} &\leq \epsilon e^{l\tilde{\tau}} \\ &\quad + z(l, \tilde{\tau}) \left((\alpha_{\xi}^{\tilde{v}_r} + \alpha^e) \|\bar{e}\|_{[0, \bar{t} + \tilde{\tau}]} + \alpha_{\gamma}^{\tilde{v}_r} \max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \|\tilde{\gamma}_i^j\|_{[0, \bar{t} + 2\tilde{\tau}]} \right).\end{aligned}$$

Finally,

$$\max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \|\tilde{\gamma}_i^j\|_{[0, \bar{t} + \tilde{\tau})} \leq \epsilon \frac{e^{l\tilde{\tau}}}{1 - \alpha_\gamma^{\tilde{v}_r} z(l, \tilde{\tau})} + \frac{z(l, \tilde{\tau}) (\alpha_\xi^{\tilde{v}_r} + \alpha^e)}{1 - \alpha_\gamma^{\tilde{v}_r} z(l, \tilde{\tau})} \|\bar{e}\|_{[0, \bar{t} + \tilde{\tau})},$$

and (A.37) holds with

$$\alpha^\epsilon(x) := \frac{e^{lx}}{1 - \alpha_\gamma^{\tilde{v}_r} z(l, x)}, \quad \alpha^{\bar{e}}(x) := \frac{z(l, x)}{1 - \alpha_\gamma^{\tilde{v}_r} z(l, x)} (\alpha_\xi^{\tilde{v}_r} + \alpha^e),$$

for all $x \geq 0$. \square

Lemma 4: Consider the CPFCS structures of Theorem 3 in the presence of delayed information and packet losses. If, for some $\bar{t} > 0$, for any time $t_k^i \leq \bar{t}$ the post-reset values of the estimation errors $\tilde{\gamma}_i^j$; $j \in \mathcal{N}^i$ satisfy

$$\left\| \tilde{\gamma}_i^j \left((t_k^i + 2\alpha_k^{ij} \tilde{\tau})^+ \right) \right\| < \epsilon, \forall j \in \mathcal{N}^i,$$

then, for sufficiently small $\tilde{\tau}$, we can compute continuous functions $\tilde{\alpha}^\epsilon : [0, c) \rightarrow [1, \infty)$ and $\tilde{\alpha}^{\bar{e}} : [0, c) \rightarrow [0, \infty)$, for some positive constant c with $\tilde{\alpha}^\epsilon(0) = 1$ and $\tilde{\alpha}^{\bar{e}}(0) = 0$, such that for all $0 \leq t < \bar{t} + \tilde{\tau}$

$$\max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \left\| \tilde{\gamma}_i^j(t) \right\| \leq \tilde{\alpha}^{\bar{e}}(\tilde{\tau}) \check{e} + \tilde{\alpha}^\epsilon(\tilde{\tau}) \epsilon, \quad (\text{A.39})$$

where

$$\check{e} := \beta^{\bar{e}}(\|\bar{e}(0)\|, 0) \oplus \max_{i \in \mathcal{N}} \left(\sigma_w^{\bar{e}}(\|w_i\|_{[0, \infty)}) \oplus \sigma_v^{\bar{e}}(\|v_i\|_{[0, \infty)}) \right).$$

Proof. From Definition 7 and the assumption that $\sigma_\gamma^{\bar{e}}(s) = \alpha_\gamma^{\bar{e}} s$, for an appropriately defined positive constant $\alpha_\gamma^{\bar{e}}$ we obtain

$$\|\bar{e}\|_{[0, \bar{t} + \tilde{\tau})} \leq \check{e} + \alpha_\gamma^{\bar{e}} \max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \left\| \tilde{\gamma}_i^j \right\|_{[0, \bar{t} + \tilde{\tau})}.$$

If $\tilde{\tau}$ is sufficiently small to satisfy the conditions of Lemma 3 and $\alpha_\gamma^{\bar{e}} \alpha^{\bar{e}}(\tilde{\tau}) < 1$, then (A.39) is satisfied for $0 \leq t < \bar{t} + \tilde{\tau}$ with

$$\tilde{\alpha}^\epsilon(x) := \frac{\alpha^\epsilon(x)}{1 - \alpha_\gamma^{\bar{e}} \alpha^{\bar{e}}(x)}, \quad \tilde{\alpha}^{\bar{e}}(\tilde{\tau}) := \frac{\alpha^{\bar{e}}(x)}{1 - \alpha_\gamma^{\bar{e}} \alpha^{\bar{e}}(x)},$$

for all $x \geq 0$. \square

Lemma 5: Consider the CPFCS structures of Theorem 3 in the presence of delayed information and packet losses. For sufficiently small values of $\tilde{\tau}$, $\|\bar{e}(0)\|$, $\max_{i \in \mathcal{N}} \|v_i\|_{[0, \infty)}$, and $\max_{i \in \mathcal{N}} \|w_i\|_{[0, \infty)}$ such that (1.31) holds, if $\max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \left\| \tilde{\gamma}_i^j(t) \right\|$ satisfies (A.39) for $0 \leq t < \bar{t} + \tilde{\tau}$, for some $\bar{t} > 0$ then, for any time $t_k^i \leq \bar{t}$, the post-reset value of the estimation errors $\tilde{\gamma}_i^j$; $j \in \mathcal{N}^i$ satisfy

$$\left\| \tilde{\gamma}_i^j \left((t_k^i + 2\alpha_k^{ij} \tilde{\tau})^+ \right) \right\| < \epsilon, \forall j \in \mathcal{N}^i, \quad (\text{A.40})$$

and (1.32) holds for all $0 \leq t \leq \bar{t} + \tilde{\tau}$.

Proof. As in Lemma 4, for simplicity of notation we denote

$$\check{e} := \beta^{\bar{e}}(\|\bar{e}(0)\|, 0) \oplus \max_{i \in \mathcal{N}} (\sigma_w^{\bar{e}}(\|w_i\|_{[0, \infty)}) \oplus \sigma_v^{\bar{e}}(\|v_i\|_{[0, \infty)})).$$

The post reset estimation error can be expressed as

$$\begin{aligned} \tilde{\gamma}_i^j \left((t_k^i + 2\alpha_k^{ij} \bar{\tau})^+ \right) &= \hat{\gamma}_i^j \left((t_k^i + 2\alpha_k^{ij} \bar{\tau})^+ \right) - \gamma_i(t_k^i + 2\alpha_k^{ij} \bar{\tau}) \\ &= \Gamma_r(2\alpha_k^{ij} \bar{\tau} - t_k^i + T_r(\gamma_i(t_k^i))) - \gamma_i(t_k^i + 2\bar{\tau}). \end{aligned}$$

Defining, on the interval $t \in [t_k^i, t_k^i + 2\alpha_k^{ij} \bar{\tau}]$ the function

$$\eta(t) := \Gamma_r(t - t_k^i + T_r(\gamma_i(t_k^i))) - \gamma_i(t),$$

we have $\eta(t_k^i) = 0$ and $\eta(t_k^i + 2\alpha_k^{ij} \bar{\tau}) = \tilde{\gamma}_i^j \left((t_k^i + 2\alpha_k^{ij} \bar{\tau})^+ \right)$. From the derivative of the inverse function we obtain

$$\dot{\eta} = v_r(\Gamma_r(t - t_k^i + T_r(\gamma_i(t_k^i)))) - \dot{\gamma}_i = v_r(\gamma_i + \eta) - v_r(\gamma_i) - \tilde{v}_{r_i} - e_{\dot{\gamma}_i}.$$

Taking norms and using the results from Lemmas 4, 3 and their proofs we obtain

$$\begin{aligned} \|\dot{\eta}\| &\leq l\|\eta\| + (\alpha_{\xi}^{\tilde{v}_r} + \alpha^e)\|\bar{e}\|_{[0, \bar{\tau}]} + \alpha_{\gamma}^{\tilde{v}_r} \max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \|\tilde{\gamma}_i^j\|_{[0, \bar{\tau}]} \\ &\leq l\|\eta\| + (\alpha_{\xi}^{\tilde{v}_r} + \alpha^e)\check{e} + (\alpha_{\gamma}^{\tilde{v}_r} + \alpha_{\gamma}^{\bar{e}}(\alpha_{\xi}^{\tilde{v}_r} + \alpha^e)) \max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \|\tilde{\gamma}_i^j\|_{[0, \bar{\tau}]} \\ &\leq l\|\eta\| + (\alpha_{\gamma}^{\tilde{v}_r} \tilde{\alpha}^{\bar{e}}(\tilde{\tau}) + (\alpha_{\xi}^{\tilde{v}_r} + \alpha^e)(1 + \alpha_{\gamma}^{\bar{e}} \tilde{\alpha}^{\bar{e}}(\tilde{\tau})))\check{e} \\ &\quad + (\alpha_{\gamma}^{\tilde{v}_r} + \alpha_{\gamma}^{\bar{e}}(\alpha_{\xi}^{\tilde{v}_r} + \alpha^e))\tilde{\alpha}^{\epsilon}(\tilde{\tau})\epsilon, \end{aligned}$$

Following the the proof of Lemma 3, it can be seen that

$$\begin{aligned} \|\tilde{\gamma}_i^j \left((t_k^i + 2\alpha_k^{ij} \bar{\tau})^+ \right)\| &= \|\eta(t_k^i + 2\alpha_k^{ij} \bar{\tau})\| \\ &\leq z(l, \tilde{\tau})(\alpha_{\gamma}^{\tilde{v}_r} \tilde{\alpha}^{\bar{e}}(\tilde{\tau}) + (\alpha_{\xi}^{\tilde{v}_r} + \alpha^e)(1 + \alpha_{\gamma}^{\bar{e}} \tilde{\alpha}^{\bar{e}}(\tilde{\tau})))\check{e} \\ &\quad + z(l, \tilde{\tau})(\alpha_{\gamma}^{\tilde{v}_r} + \alpha_{\gamma}^{\bar{e}}(\alpha_{\xi}^{\tilde{v}_r} + \alpha^e))\tilde{\alpha}^{\epsilon}(\tilde{\tau})\epsilon \end{aligned}$$

Finally, $\|\tilde{\gamma}_i^j \left((t_k^i + 2\alpha_k^{ij} \bar{\tau})^+ \right)\| < \epsilon$ if $\tilde{\tau}$ is sufficiently small such that

$$(\alpha_{\gamma}^{\tilde{v}_r} + \alpha_{\gamma}^{\bar{e}}(\alpha_{\xi}^{\tilde{v}_r} + \alpha^e))\tilde{\alpha}^{\epsilon}(\tilde{\tau})z(l, \tilde{\tau}) < 1,$$

and for sufficiently small values of \check{e} such that $\epsilon > \tilde{\sigma}^{\epsilon}(\tilde{\tau})\check{e}$, with

$$\tilde{\sigma}^{\epsilon}(x) := \frac{z(l, x)(\alpha_{\gamma}^{\tilde{v}_r} \tilde{\alpha}^{\bar{e}}(x) + (\alpha_{\xi}^{\tilde{v}_r} + \alpha^e)(1 + \alpha_{\gamma}^{\bar{e}} \tilde{\alpha}^{\bar{e}}(x)))}{1 - (\alpha_{\gamma}^{\tilde{v}_r} + \alpha_{\gamma}^{\bar{e}}(\alpha_{\xi}^{\tilde{v}_r} + \alpha^e))\tilde{\alpha}^{\epsilon}(x)z(l, x)}.$$

Because $\log(1 + x) \leq x$ for $x \geq 0$, if we define

$$\sigma^\epsilon(x) := \frac{1}{\log\left(1 + \frac{1}{\tilde{\sigma}^\epsilon(x)}\right)},$$

it follows by continuity that $\sigma^\epsilon(0) := \lim_{x \rightarrow 0^+} \sigma^\epsilon(x) = 0$ and $\tilde{\sigma}^\epsilon(x) \leq \sigma^\epsilon(x)$. By requiring that $\epsilon > \sigma^\epsilon(\tilde{\tau})\bar{\epsilon}$, (1.32) holds with

$$\bar{\alpha}^\epsilon(x) := \left(\tilde{\alpha}^{\bar{\epsilon}}(\tilde{\tau}) \log\left(1 + \frac{1}{\sigma^\epsilon(\tilde{\tau})}\right) + \tilde{\alpha}^\epsilon(\tilde{\tau}) \right) \epsilon.$$

Since $\bar{\alpha}^\epsilon(0)$ is not defined, we define it formally by continuity as $\bar{\alpha}^\epsilon(0) = \lim_{x \rightarrow 0^+} \bar{\alpha}^\epsilon(x) = 1$, where we used the fact that $\lim_{x \rightarrow 0^+} \tilde{\alpha}^{\bar{\epsilon}}(x) \log\left(1 + \frac{1}{\sigma^\epsilon(x)}\right) = 0$. \square

Proof of Theorem 3. Suppose that for some time t_k^i , for all previous data messages sent from any agent $j \in \mathcal{N}$ at times $t_o^j \leq t_k^i$, the post-reset values of the estimation errors satisfy $\left\| \tilde{\gamma}_j^p \left((t_o^j + 2\alpha_o^{jp} \tilde{\tau})^+ \right) \right\| < \epsilon$ for all $p \in \mathcal{N}^j$. This holds true if t_k^i is the time when the first data message is sent. If $\tilde{\tau}$, $\|\bar{\epsilon}(0)\|$, $\max_{i \in \mathcal{N}} \|v_i\|_{[0, \infty)}$, and $\max_{i \in \mathcal{N}} \|w_i\|_{[0, \infty)}$, are sufficiently small so as to satisfy the conditions of Lemmas 4 and 5, we have

$$\left\| \tilde{\gamma}_i^j \left((t_k^i + 2\alpha_k^{ij} \tilde{\tau})^+ \right) \right\| \leq \tilde{\epsilon} < \epsilon, \quad \forall j \in \mathcal{N}^i \quad (\text{A.41})$$

and we can apply Lemmas 4 and 5 for the next received data message.

From Lemma 4, when $\dot{\tilde{\gamma}}_i^j(t)$ is defined, $\max_{i \in \mathcal{N}, j \in \mathcal{N}^i} \left\| \dot{\tilde{\gamma}}_i^j(t) \right\| \leq v_{\dot{\tilde{\gamma}}}$, where $v_{\dot{\tilde{\gamma}}}$ and $\tilde{\epsilon}$ in (A.41) depend on $\|\bar{\epsilon}(0)\|$, $\max_{i \in \mathcal{N}} \|v_i\|_{[0, \infty)}$, $\max_{i \in \mathcal{N}} \|w_i\|_{[0, \infty)}$, and ϵ but not on time. Therefore, the triggering condition $\left\| \dot{\tilde{\gamma}}_i^j(t) \right\| = \epsilon$ can only occur N times within $\frac{\epsilon - \tilde{\epsilon}}{v_{\dot{\tilde{\gamma}}}}$ time units. Repeating the same reasoning every transmission time t_k^i ; $i \in \mathcal{N}$ and $k \geq 0$, the theorem holds by recursion. \square

References

- [1] Ghabcheloo R, Aguiar AP, Pascoal A, et al. Coordinated Path-Following in the Presence of Communication Losses and Time Delays. SIAM J Control Optim. 2009 February;48(1):234–265.
- [2] Ihle IAF, Arcak M, Fossen TI. Passivity-based designs for synchronized path-following. Automatica. 2006;43:1508–1518.
- [3] Giulietti F, Pollini L, Innocenti M. Autonomous formation flight. IEEE Control Systems. 2000 dec;20(6):34 – 44.
- [4] Stilwell DJ, Bishop BE. Platoons of underwater vehicles. IEEE Control Systems. 2000 dec;20(6):45 – 52.
- [5] Ogren P, Egerstedt M, Hu X. A control Lyapunov function approach to multiagent coordination. IEEE Transactions on Robotics and Automation. 2002 October;18(5):847 – 851.

- [6] Jadbabaie A, Lin J, Morse AS. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*. 2003 June;48(6):988 – 1001.
- [7] Moreau L. Stability of multiagent systems with time-dependent communication links. *IEEE Transactions on Automatic Control*. 2005 February;50(2):169 – 182.
- [8] Ma CQ, Zhang JF. Necessary and sufficient conditions for consensusability of linear multi-agent systems. *IEEE Transactions on Automatic Control*. 2010;55(5):1263–1268.
- [9] Dong W. On consensus algorithms of multiple uncertain mechanical systems with a reference trajectory. *Automatica*. 2011;47(9):2023–2028.
- [10] Soetanto D, Lapierre L, Pascoal A. Adaptive, non-singular path-following control of dynamic wheeled robots. vol. 2 of 42nd IEEE Conference on Decision and Control; 2003. p. 1765 – 1770.
- [11] Skjetne R, Fossen TI, Kokotović PV. Robust output maneuvering for a class of nonlinear systems. *Automatica*. 2004;40(3):373 – 383.
- [12] Yook JK, Tilbury DM, Soparkar NR. Trading computation for bandwidth: Reducing communication in distributed control systems using state estimators. *IEEE Transactions on Control Systems Technology*. 2002 jul;10(4):503–518.
- [13] Xu Y, Hespanha JP. Communication Logic Design and Analysis for Networked Control Systems. In: Menini L, Zaccarian L, Abdallah CT, editors. *Current Trends in Nonlinear Systems and Control. Systems & Control: Foundations & Applications*. Birkhäuser Boston; 2006. p. 495–514.
- [14] Tabuada P. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control*. 2007;52(9):1680–1685.
- [15] Aström KJ. Event based control. Analysis and design of nonlinear control systems. 2008;3:127–147.
- [16] Wang X, Sun Y, Hovakimyan N. Asynchronous task execution in networked control systems using decentralized event-triggering. *Systems & Control Letters*. 2012;61(9):936–944.
- [17] Battistelli G, Benavoli A, Chisci L. Data-driven communication for state estimation with sensor networks. *Automatica*. 2012;48(5):926–935.
- [18] Dimarogonas DV, Johansson KH. Event-triggered control for multi-agent systems. 48th IEEE Conference on Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. IEEE; 2009. p. 7131–7136.
- [19] Dimarogonas DV, Frazzoli E, Johansson KH. Distributed event-triggered control for multi-agent systems. *IEEE Transactions on Automatic Control*. 2012;57(5):1291–1297.
- [20] Seyboth GS, Dimarogonas DV, Johansson KH. Event-based broadcasting for multi-agent average consensus. *Automatica*. 2013;49(1):245–252.
- [21] Garcia E, Cao Y, Casbeer DW. Cooperative control with general linear dynamics and limited communication: Centralized and decentralized event-triggered control strategies. American Control Conference (ACC). IEEE; 2014. p. 159–164.

- [22] Viel C, Bertrand S, Piet-Lahanier H, et al. New state estimator for decentralized event-triggered consensus for multi-agent systems. IFAC-PapersOnLine. 2016;49(5):365–370.
- [23] Almeida J, Silvestre C, Pascoal A. Synchronization of Multiagent Systems Using Event-Triggered and Self-Triggered Broadcasts. IEEE Transactions on Automatic Control. 2017 September;62(9):4741–4746.
- [24] Vanni F, Aguiar AP, Pascoal AM. Cooperative path-following of underactuated autonomous marine vehicles with logic-based communication. NGCUV’08-IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles; 2008. p. 1–6.
- [25] Kalwa J, Pascoal A, Ridao P, et al. The European R&D-Project MORPH: Marine robotic systems of self-organizing, logically linked physical nodes. IFAC Proceedings Volumes. 2012;45(5):349–354.
- [26] Mišković N, Bibuli M, Birk A, et al. CADDY-Cognitive Autonomous Diving Buddy: Two Years of Underwater Human-Robot Interaction. Marine Technology Society Journal. 2016;50(4):54–66.
- [27] Abreu P, Antonelli G, Arrichiello F, et al. Widely scalable mobile underwater sonar technology: An overview of the H2020 WiMUST project. Marine Technology Society Journal. 2016;50(4):42–53.
- [28] Godsil C, Royle GF. Algebraic graph theory. vol. 207. Springer Science & Business Media; 2013.
- [29] Jiang ZP, Teel AR, Praly L. Small-gain theorem for ISS systems and applications. Mathematics of Control, Signals, and Systems (MCSS). 1994;7:95–120.
- [30] Isidori A. Nonlinear control systems II. Springer; 1999.
- [31] Aguiar AP, Hespanha JP. Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty. IEEE Transactions on Automatic Control. 2007;52(8):1362–1379.
- [32] Kaminer I, Pascoal A, Xargay E, et al. Time-Critical Cooperative Control of Autonomous Air Vehicles. Butterworth-Heinemann; 2017. .
- [33] Vanni FV. Coordinated motion control of multiple autonomous underwater vehicles. M.Sc. Thesis, Instituto Superior Técnico; 2007.
- [34] Kebkal K, Kebkal O, Glushko E, et al. Underwater acoustic modems with integrated atomic clocks for one-way traveltimes underwater vehicle positioning. Underwater Acoustics Conference and Exhibition (UACE); 2017. .
- [35] Abreu PC, Botelho J, Góis P, et al. The MEDUSA class of autonomous marine vehicles and their role in EU projects. OCEANS 2016-Shanghai. IEEE; 2016. p. 1–10.
- [36] Ribeiro J. Motion Control of Single and Multiple Autonomous Marine Vehicles. M.Sc. Thesis, Instituto Superior Técnico; 2011.
- [37] Horn RA, Johnson CR. Matrix analysis. Cambridge university press; 1990.
- [38] Khalil HK. Nonlinear Systems. Prentice-Hall, New Jersey; 1996.